SEMAPRO 2015

The Ninth International Conference on Advances in Semantic Processing

ISBN: 978-1-61208-420-6

July 19 - 24, 2015

Nice, France

SEMAPRO 2015 Editors

Alexey Cheptsov, High Performance Computing Center Stuttgart (HLRS), Germany

SEMAPRO 2015

Forward

The Ninth International Conference on Advances in Semantic Processing (SEMAPRO 2015), held between July 19-24, 2015 in Nice, France, considered the complexity of understanding and processing information. Semantic processing considers contextual dependencies and adds to the individually acquired knowledge emergent properties and understanding. Hardware and software support and platforms were developed for semantically enhanced information retrieval and interpretation. Searching for video, voice and speech [VVS] raises additional problems to specialized engines with respect to text search. Contextual searching and special patterns-based techniques are current solutions.

With the progress on ontology, web services, semantic social media, semantic web, deep web search /deep semantic web/, semantic deep web, semantic networking and semantic reasoning, SEMAPRO 2015 constituted the stage for the state-of-the-art for the most recent advances.

The conference had the following tracks:

- Ontology fundamentals
- Semantic applications/platforms/tools
- Semantic Technologies
- Basics on semantics
- Models and ontology-based design of protocols, architectures and services

Similar to previous editions, this event attracted excellent contributions and active participation from all over the world. We were very pleased to receive top quality contributions.

We take here the opportunity to warmly thank all the members of the SEMAPRO 2015 technical program committee, as well as the numerous reviewers. The creation of such a high quality conference program would not have been possible without their involvement. We also kindly thank all the authors that dedicated much of their time and effort to contribute to SEMAPRO 2015. We truly believe that, thanks to all these efforts, the final conference program consisted of top quality contributions.

Also, this event could not have been a reality without the support of many individuals, organizations and sponsors. We also gratefully thank the members of the SEMAPRO 2015 organizing committee for their help in handling the logistics and for their work that made this professional meeting a success.

We hope that SEMAPRO 2015 was a successful international forum for the exchange of ideas and results between academia and industry and to promote further progress in the area of semantic processing. We also hope that Nice, France, provided a pleasant environment during the conference and everyone saved some time to enjoy the charm of the city.

SEMAPRO 2015 Chairs

SEMAPRO Advisory Chairs

Wladyslaw Homenda, Warsaw University of Technology, Poland
Bich-Lien Doan, SUPELEC, France
Alexey Cheptsov, High Performance Computing Center Stuttgart (HLRS), Germany
Sule Yildirim Yayilgan, Gjøvik University College, Norway
Jesper Zedlitz, Christian-Albrechts-Universität Kiel, Germany
Soon Ae Chun, City University of New York, USA
Fabio Grandi, University of Bologna, Italy
David A. Ostrowski, Ford Motor Company, USA
Andrea Perego, European Commission - Joint Research Centre, Ispra, Italy

SEMAPRO Industry/Research Liaison Chairs

Riccardo Albertoni, IMATI-CNR-Genova, Italy
Panos Alexopoulos, iSOCO S.A., Spain
Sofia Athenikos, IPsoft, USA
Isabel Azevedo, ISEP-IPP, Portugal
Sam Chapman, The Floow Limited, UK
Daniele Christen, Parsit Company, Italy
Frithjof Dau, SAP Research Dresden, Germany
Thierry Declerck, DFKI GmbH, Germany
Alessio Gugliotta, Innova SpA, Italy
Shun Hattori, Muroran Institute of Technology, Japan
Tracy Holloway King, eBay Inc., USA
Lyndon J. B. Nixon, STI International, Austria
Zoltán Theisz, evopro Innovation LLC, Hungary
Thorsten Liebig, derivo GmbH - Ulm, Germany
Michael Mohler, Language Computer Corporation in Richardson, USA

SEMAPRO Publicity Chairs

Felix Schiele, Reutlingen University, Germany
Bernd Stadlhofer, University of Applied Sciences, Austria
Ruben Costa, UNINOVA, Portugal
Andreas Emrich, German Research Center for Artificial Intelligence (DFKI), Germany

SEMAPRO 2015

Committee

SEMAPRO Advisory Chairs

Wladyslaw Homenda, Warsaw University of Technology, Poland
Bich-Lien Doan, SUPELEC, France
Alexey Cheptsov, High Performance Computing Center Stuttgart (HLRS), Germany
Sule Yildirim Yayilgan, Gjøvik University College, Norway
Jesper Zedlitz, Christian-Albrechts-Universität Kiel, Germany
Soon Ae Chun, City University of New York, USA
Fabio Grandi, University of Bologna, Italy
David A. Ostrowski, Ford Motor Company, USA
Andrea Perego, European Commission - Joint Research Centre, Ispra, Italy

SEMAPRO Industry/Research Liaison Chairs

Riccardo Albertoni, IMATI-CNR-Genova, Italy
Panos Alexopoulos, iSOCO S.A., Spain
Sofia Athenikos, IPsoft, USA
Isabel Azevedo, ISEP-IPP, Portugal
Sam Chapman, The Floow Limited, UK
Daniele Christen, Parsit Company, Italy
Frithjof Dau, SAP Research Dresden, Germany
Thierry Declerck, DFKI GmbH, Germany
Alessio Gugliotta, Innova SpA, Italy
Shun Hattori, Muroran Institute of Technology, Japan
Tracy Holloway King, eBay Inc., USA
Lyndon J. B. Nixon, STI International, Austria
Zoltán Theisz, evopro Innovation LLC, Hungary
Thorsten Liebig, derivo GmbH - Ulm, Germany
Michael Mohler, Language Computer Corporation in Richardson, USA

SEMAPRO Publicity Chairs

Felix Schiele, Reutlingen University, Germany
Bernd Stadlhofer, University of Applied Sciences, Austria
Ruben Costa, UNINOVA, Portugal
Andreas Emrich, German Research Center for Artificial Intelligence (DFKI), Germany

SEMAPRO 2015 Technical Program Committee

Nasser Alalwan, King Saud University - Riyadh, Saudi Arabia
Riccardo Albertoni, IMATI-CNR-Genova, Italy
José F. Aldana Montes, University of Málaga, Spain
Panos Alexopoulos, iSOCO S.A., Spain
Mario Arrigoni Neri, University of Bergamo, Italy
Sofia Athenikos, IPsoft, USA
Agnese Augello, ICAR - CNR, Italy
Isabelle Augenstein, University of Sheffield, UK
Isabel Azevedo, ISEP-IPP, Portugal
Bruno Bachimont, Universite de Technologie de Compiegne, France
Ebrahim Bagheri, Ryerson University, Canada
Khalid Belhajjame, Université Paris-Dauphine, France
Helmi Ben Hmida, Fraunhofer Institute for Computer Graphics Research IGD, Germany
Jorge Bernardino, ISEC - Polytechnic Institute of Coimbra, Portugal
Christopher Brewster, Aston University - Birmingham, UK
Volha Bryl, University of Mannheim, Germany
Dilletta Romana Cacciagrano, University of Camerino, Italy
Nicoletta Calzolari, CNR-ILC (Istituto di Linguistica Computazionale del CNR), Italy
Delroy Cameron, Wright State University, USA
Ozgu Can, Ege University, Turkey
Tru Hoang Cao, Vietnam National University - HCM & Ho Chi Minh City University of Technology, Vietnam
Rodrigo Capobianco Guido, São Paulo State University, Brazil
Sana Châabane, ISG - Sousse, Tunisia
Sam Chapman, The Floow Limited, UK
Chao Chen, Capital One, USA
Alexey Cheptsov, High Performance Computing Center Stuttgart (HLRS), Germany
Dickson Chiu, University of Hong Kong, Hong Kong
Smitashree Choudhury, UK Open University - Milton Keynes, UK
Sunil Choenni, Ministry of Security and Justice, Netherlands
Daniele Christen, Parsit Company, Italy
Soon Ae Chun, City University of New York, USA
Paolo Ciancarini, Università di Bologna, Italy
Ruben Costa, UNINOVA - Instituto de Desenvolvimento de Novas Tecnologias, Portugal
Frithjof Dau, SAP Research Dresden, Germany
Cláudio de Souza Baptista, Computer Science Department, University of Campina Grande, Brazil
Thierry Declerck, DFKI GmbH, Germany
Gianluca Demartini, University of Fribourg, Switzerland
Chiara Di Francescomarino, Fondazione Bruno Kessler - Trento, Italy
Gayo Diallo, University of Bordeaux, France
Alexiei Dingli, The University of Malta, Malta
Christian Dirschl, Wolters Kluwer, Germany

Bich Lien Doan, SUPELEC, France
Milan Dojčinovski, Czech Technical University in Prague, Czech Republic
Raimund K. Ege, Northern Illinois University, USA
Enrico Francesconi, Institute of Legal Information Theory and Techniques, Italy
Naoki Fukuta, Shizuoka University, Japan
Frieder Ganz, University of Surrey, U.K.
Rosa M. Gil Iranzo, Universitat de LLeida, Spain
Zbigniew Gontar, University of Lodz, Poland
Sotirios K. Goudos, Aristotle University of Thessaloniki, Greece
Fabio Grandi, University of Bologna, Italy
William Grosky, University of Michigan-Dearborn, USA
Tudor Groza, University of Queensland, Australia
Francesco Guerra, University of Modena and Reggio Emilia, Italy
Alessio Gugliotta, Innova SpA, Italy
Armin Haller, CSIRO ICT Centre - Canberra, Australia
Carmem S. Hara, Universidade Federal do Parana, Brazil
Brian Harrington, University of Toronto Scarborough, Canada
Sven Hartmann, Clausthal University of Technology, Germany
Shun Hattori, Muroran Institute of Technology, Japan
Tracy Holloway King, eBay Inc., U.S.A.
Wladyslaw Homenda, Warsaw University of Technology, Poland
Ching-Hsien (Robert) Hsu, Chung Hua University, Taiwan
Thomas Hubauer, Siemens Corporate Technology - Munich, Germany
Sergio Ilarri, University of Zaragoza, Spain
Muhammad Javed, Wayne State University - Detroit, USA
Wassim Jaziri, ISIM Sfax, Tunisia
Achilles Kameas, Hellenic Open University, Greece
Katia Kermanidis, Ionian University - Corfu, Greece
Holger Kett, Fraunhofer Insitute for Industrial Engineering IAO, Germany
Pavel Klinov, University of Ulm, Germany
Mieczyslaw Kokar, Northeastern University, USA
Sefki Kolozali, University of Surrey, UK
Jaroslav Kuchar, Czech Technical University in Prague, Czech Republic
Jose Emilio Labra Gayo, University of Oviedo, Spain
Christoph Lange, University of Bonn / Fraunhofer IAIS, Germany
Kyu-Chul Lee, Chungnam National University - Daejeon, South Korea
Thorsten Liebig, derivo GmbH - Ulm, Germany
Antonio Lieto, University of Turin & ICAR CNR, Italy
Sandra Lovrenčić, University of Zagreb - Varaždin, Croatia
Hongli Luo, Indiana University - Purdue University Fort Wayne, U.S.A.
Rabi N. Mahapatra, Texas A&M University, USA
Eetu Mäkelä, Aalto University, Finland
Maria Maleshkova, The Open University, UK
Erik Mannens, Ghent University, Belgium

Elio Masciari, ICAR-CNR, Italy
Miguel Felix Mata Rivera, Instituto Politecnico Nacional, Mexico
Maristella Matera, Politecnico di Milano, Italy
Dennis McLeod, University of Southern California - Los Angeles, USA
Michele Melchiori, Università degli Studi di Brescia, Italy
Elisabeth Métais, Cedric-CNAM, France
Vasileios Mezaris, Informatics and Telematics Institute (ITI) and Centre for Research and Technology Hellas (CERTH) - Thermi-Thessaloniki, Greece
Michael Mohler, Language Computer Corporation in Richardson, U.S.A.
Shahab Mokarizadeh , Royal Institute of Technology (KTH) - Stockholm, Sweden
Anne Monceaux, Airbus Group Innovations, France
Alessandro Moschitti, Qatar Computing Research Institute, Qatar
Mir Abolfazl Mostafavi, Université Laval - Québec, Canada
Ekawit Nantajeewarawat, Sirindhorn International Institute of Technology / Thammasat University, Thailand
Vlad Nicolicin Georgescu, SP2 Solutions, France
Lyndon J. B. Nixon, STI International, Austria
Csongor Nyulas, Stanford Center for Biomedical Informatics, USA
David A. Ostrowski, Ford Motor Company, USA
Vito Claudio Ostuni, Polytechnic University of Bari, Italy
Peera Pacharintanakul, TOT, Thailand
Andrea Perego, European Commission - Joint Research Centre, Ispra, Italy
Max Petrenko, NTENT / Ontological Semantics Technology Lab - Texas A&M University-Commerce, USA
Livia Predoiu, University of Oxford, UK
Hemant Purohit, Wright State University, USA
Filip Radulovic, Universidad Politécnica de Madrid, Spain
Jaime Ramírez, Universidad Politécnica de Madrid, Spain
Isidro Ramos, Valencia Polytechnic University, Spain
Werner Retschitzegger, Johannes Kepler University Linz, Austria
German Rigau, IXA NLP Group. EHU, Spain
Juergen Rilling, Concordia University, Canada
Tarmo Robal, Tallinn University of Technology, Estonia
Alejandro Rodríguez González, Centre for Biotechnology and Plant Genomics, UPM-INIA, Spain
Marco Ronchetti, Universita' degli Studi di Trento, Italy
Marco Rospocher, Fondazione Bruno Kessler (FBK), Italy
Thomas Roth-Berghofer, University of West London, U.K.
Michele Ruta, Technical University of Bari, Italy
Gunter Saake, University of Magdeburg, Germany
Melike Sah, Trinity College Dublin, Ireland
Satya Sahoo, Case Western Reserve University, USA
Adriano A. Santos, Universidade Federal de Campina Grande, Brazil
Minoru Sasaki, Ibaraki University, Japan
Felix Schiele, Hochschule Reutlingen, Germany

Kinga Schumacher, German Research Center for Artificial Intelligence (DFKI) - Berlin, Germany
Wieland Schwinger, Johannes Kepler University Linz, Austria
Floriano Scioscia, Politecnico di Bari, Italy
Giovanni Semeraro, University of Bari "Aldo Moro", Italy
Kunal Sengupta, Wright State University - Dayton, USA
Luciano Serafini, Fondazione Bruno Kessler, Italy
Md. Sumon Shahriar, Tasmanian ICT Centre/CSIRO, Australia
Sofia Stamou, Ionian University, Greece
Vasco Soares, Instituto de Telecomunicações / Polytechnic Institute of Castelo Branco, Portugal
Ahmet Soylu, University of Oslo, Norway
Mari Carmen Suárez-Figueroa, Universidad Politécnica de Madrid (UPM), Spain
Lars G. Svensson, German National Library, Germany
Cui Tao, Mayo Clinic - Rochester, USA
Saïd Tazi, LAAS-CNRS, Université de Toulouse / Université Toulouse1, France
Zoltán Theisz, evopro Innovation LLC, Hungary
Tina Tian, Manhattan College, U.S.A.
Ioan Toma, University of Innsbruck, Austria
Tania Tudorache, Stanford University, USA
Christina Unger, CITEC - Bielefeld University, Germany
Willem Robert van Hage, SynerScope B.V. / Innovation Lab TU Eindhoven, Netherlands
Luc Vouligny, Hydro-Québec Research Institute, Canada
Holger Wache, University of applied Science and Arts Northwestern Switzerland, Switzerland
Shenghui Wang, OCLC Research, Netherlands
Mari Wigham, Food and Biobased Research, Wageningen UR, Netherlands
Wai Lok Woo, Newcastle University, UK
Honghan Wu, University of Aberdeen, UK
Sule Yildirim Yayilgan, Gjøvik University College, Norway
Fouad Zablith, American University of Beirut, Lebanon
Filip Zavoral, Charles University in Prague, Czech Republic
Yuting Zhao, The University of Aberdeen, UK
Hai-Tao Zheng, Tsinghua University, China
Ingo Zinnikus, German Research Center for Artificial Intelligence (DFKI), Germany
Amal Zouaq, Royal Military College of Canada, Canada

Copyright Information

Table of Contents

Assessing Legislative Alignment – An Ontological Approach

Work in progress

Rosetta Romano
Faculty of Government Business and Law
University of Canberra
Canberra, Australia
rosetta.romano@canberra.edu.au

***Abstract*—An ontology provides the agreed definitions and describes how the terms in a subject area or domain, are related. It is a model that can be read by humans and coded for use by computers. Across the globe, governments are using ontologies in innovative ways to solve long-standing government problems. The problem is that there is no single approach used by government agencies to assess whether their systems are aligned to the legislation. In a social welfare setting, if there is any misalignment between the legislation and the systems, then, it may result in an unintentional disadvantage to those most in need. This paper outlines the research design using a case study to detect and to compare the ontological patterns existing in legislation and an online claim form relating to a family tax benefit in Australia.**

Keywords-Ontology; Ontology alignment; Legislation; Government claim forms; Online claim forms.

I. Introduction

Around the globe, different governments are using ontologies in innovative ways to solve long-standing government problems. An ontology is an artefact that provides a community with the agreed definitions and describes how the terms in a subject area or domain, are related. As a model, it can be read by humans, and coded for computers. Besides being useful as an agreed dictionary, its strength lies in the way that technology can consume it. Ontologies allow sophisticated machine manipulation, aggregation of information, pattern analysis and inferences from vast quantities of data that ordinary humans would not be able to handle [1]. It is for this capability that governments are using ontologies to contribute to the development of e-government. E-government is a way for government to use new technologies to provide people with more convenient access to government information and services, to improve the quality of services and to provide greater opportunities to participate in the democratic institutions and processes [2].

In Australia, e-government is supporting the move away from traditional service delivery. Historically, a single agency would have sole responsibility to deliver all components of a service to consumers. Connected government, and increased partnership with the private sector, now requires the responsibility for the delivery of services to be shared, and new ways of using technology to manage the complexity need to be found.

The literature has reported many different ways that governments are using ontologies to solve long-standing government problems. For example, in Greece, Italy, Denmark and Germany, ontologies have been used to enhance public participation in the development of legislation [3]. In the Netherlands, ontologies have been used to compare legislation across jurisdictions [4], while in Spain, ontologies have been used to improve the retrieval of legal documents for citizens [5], and in the UK, the government has used ontology to model the notification of multiple agencies of a change of circumstance and replace it with a single local authority [6].

This paper reports on research-in-progress to address another long-standing problem for government responding to frequent and complex legislative change. In Australia, ministers must establish audit committees and provide an annual compliance report that the effectiveness of review for monitoring compliance with laws, regulations and associated government policies [7]. This requires processes to ensure that information systems evolve in line with the law. An information system is the application of people, technologies and procedures to solve a business problem and government information systems are used to solve government problems [8].

Many government information systems involve decision making. Decision making is big business for many government agencies [9]. For the recipients of welfare, any misalignment between the legislation and the systems may result in unintentional disadvantage. There is no single approach used by government agencies to assess whether the government information systems and legislation are conceptually aligned. This paper outlines the design of novel research using the ontology patterns existing in legislation to assess the alignment between legislation and government information systems.

There has been work to explore how administrative organisations can use ontologies to manage the complex policy change management [10]. This research-in-progress explores the comparison of ontological patterns existing in different artefacts within a single domain to assure ministers and service consumers that systems and legislation are aligned. The artefacts being compared are the legislation and

ISBN: 978-1-61208-420-6

the claim form for an Australian Government Family Tax Benefit (FTB) payment. These artefacts are key components of the government information systems used to administer the law.

The FTB claim form is completed by service consumers and used as evidence by the service providing agencies that are responsible for the administration of the payment. It is a record of the claimant's application for the payment. The evidence being collected in the form should be aligned to the regulatory requirements [10]. For the FTB payment in Australia, the legislation supports two consecutive legislative processes: (1) assess the eligibility of the applicant, and (2) if the applicant is eligible, then, assess the payment value.

The FTB claim form should therefore be designed to collect the data that is necessary for the government information systems to assess (1) the eligibility in accordance with the '*A New Tax System (Family Assistance) Act 1999*', and (2) to determine the value of the payment in accordance with *A New Tax System (Family Assistance) (Administration) Act 1999*. No more, and no less. If the claim form seeks more data, or less data, then this may indicate misalignment.

The online claim form for a FTB payment is very complex. An applicant seeking the payment must provide no less than 946 data items about themselves, 145 about their partner, and 52 items about each of their children. Perhaps the complex legislation requires all this data. In that case, we would expect that the investigation will determine that alignment exists.

Government information systems are developed by experts who have a deep understanding of the legislation and the government information systems that enact it. Rather than relying on a few key experts perhaps there is a way to model the information so that the knowledge can be shared by government agencies and service consumers alike. Meeting the greater expectation from citizens is made possible with modern information technologies [12]. With a model of the knowledge that is currently hidden behind complex legislation, more opportunities to streamline payments and processes, reduce duplication and enhance the online experience are expected to emerge.

By comparing a conceptual model of the legislation to the conceptual model of the claim form, it should be possible to identify any misalignment between them. A conceptual model is an abstract and simplified description of the reality that is being represented [2].

The conceptual structure in the legislation and the online claim form will each be modelled as ontologies. An ontology is defined as "an explicit specification of a conceptualisation" [13]. An ontology specifies and organises the concepts in a domain [14] in a model as an abstract and simplified view of the domain [15]. It is a shared understanding accomplished by agreeing on an appropriate way to conceptualize the domain, and then to make it explicit in some language [16]. An ontology can be used by humans and formalised for computers.

Like an ontology, legislation provides definitions of terms in a domain and describes the relations between these terms. It is a primary source for government agencies to harvest terms to build an ontology. While it is a rich source, legislation is difficult to understand because: not all terms are defined; the relations between terms are not always clear; and the context can sometimes only be understood by accessing all cross-referenced sections or legislation.

This paper describes the research design that will be used to develop a conceptual model of the legislation and the claim form related to the FTB payment domain. The research in progress will contribute a strategy and method of conducting ontological analysis, and a novel means of using ontology to determine alignment. It will apply an instrumental case study to gain a broader appreciation of how legislation is being translated in the claim form. It is expected that the processes used to detect, extract and analyse the concepts from legislation will be generalizable.

The remainder of the paper is organised as follows: in Section II, the research setting is described. In Section III, the research strategy is outlined. In Section IV, the research limitations and risks are presented. Finally the research contributions are discussed in Section V.

II. Research Setting

In Australia, the government has the powers to pass laws. There are three arms of government: the parliament, the executive and the judiciary (see Figure 1). The *parliament* makes the law; the *executive* operationalises the law, and the the *judiciary* interprets the law. The interpretation of the law is not a focus of this paper. In Figure 1 the research setting, overview and scope are modelled. This research in progress will apply a case study using the FTB payment to demonstrate how the conceptual structure of the legislation drafted by the *parliament*, i.e., the government agency making the law, has been operationalised in the claim form by the executive, i.e., the service delivery agency Department of Human Services, on behalf of the APS policy agency, the Department of Social Services. While the minister for the policy department is responsible for the legislation, (2) the minister for the service delivery department is responsible for delivering the services i.e., to determine the eligibility for the payment, and to assess the value of the payment.

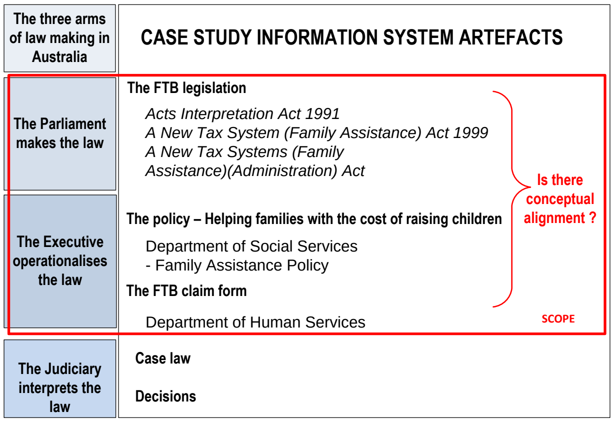

The three arms of law making in Australia
CASE STUDY INFORMATION SYSTEM ARTEFACTS
The Parliament makes the law
The FTB legislation
Acts Interpretation Act 1991
A New Tax System (Family Assistance) Act 1999
A New Tax Systems (Family Assistance)(Administration) Act
Is there conceptual alignment ?
The policy – Helping families with the cost of raising children
The Executive operationalises the law
Department of Social Services
- Family Assistance Policy
The FTB claim form
Department of Human Services
SCOPE
The Judiciary interprets the law
Case law
Decisions

Figure 1. Research setting, overview and scope.

III. Research Strategy

This section outlines the proposed research strategy in three sections. Section A identifies the research methodology to be used. Section B provides an overview of case study methodology, and Section C describes the research methods that will be used to undertake the data collection and analysis.

A. *Research methodology*

This research will use a case study methodology. The case study has two phases the build and the appraisal (see Figure 2). This work-in-progress paper describes the research design for the first build phase only. In the build phase, an ontological investigation will be undertaken to identify the concepts as they exist in the two pieces of legislation and the claim form. Then, in the appraisal phase, the domain ontology developed by the researcher will be reviewed by key informants to appraise its appropriateness to confirm whether it as an objective representation of the FTB domain. The assessors will include representatives from candidate legal, ICT, Business and policy departmental groups.

B. *Case study methodology*

Case study is a research strategy that has been used in both policy and public administration research [17]. The research in progress will apply an instrumental case study that is defined as a study that uses a particular case to gain a broader appreciation of an issue or phenomenon [17]. The research will use the legislation and the claim form relating to the payment of FTB in normal circumstances to gain a broader and deeper appreciation of alignment issues that may exist. The reason for concentrating on the normal circumstances is to constrain the study, and ensure it is completed within a reasonable timeframe.

Regulation includes any laws or other rules that govern the conduct of people or businesses (service consumers) and affect them either directly or indirectly, sometimes in ways that are more apparent than others [11]. For example, it is apparent that the payment of FTB is covered by Division 1 'Family tax benefit', of Part 3 'Payment of family assistance', in the '*A New Tax System (Family Assistance) (Administration) Act, 1999*'. In Australia, the *Acts Interpretation Act, 1901* is a reference for reading any Commonwealth Act. It provides a dictionary to make Commonwealth legislation shorter, less complex and more consistent in operation and should be referred to for common definitions of such terms as person, individual, and Minister.

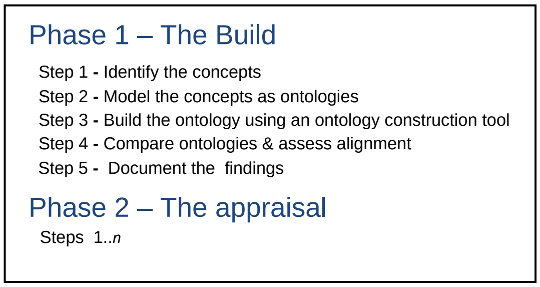

Phase 1 – The Build
Step 1 - Identify the concepts
Step 2 - Model the concepts as ontologies
Step 3 - Build the ontology using an ontology construction tool
Step 4 - Compare ontologies & assess alignment
Step 5 - Document the findings
Phase 2 – The appraisal
Steps 1..*n*

Figure 2. Research setting, overview and scope.

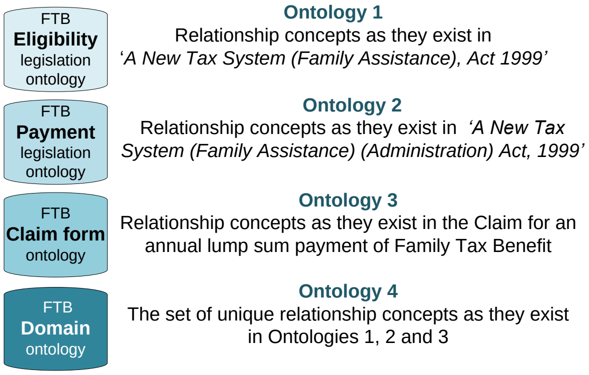

FTB
Eligibility
legislation
ontology
Ontology 1
Relationship concepts as they exist in
'A New Tax System (Family Assistance), Act 1999'
FTB
Payment
legislation
ontology
Ontology 2
Relationship concepts as they exist in 'A New Tax System (Family Assistance) (Administration) Act, 1999'
FTB
Claim form
ontology
Ontology 3
Relationship concepts as they exist in the Claim for an annual lump sum payment of Family Tax Benefit
FTB
Domain
ontology
Ontology 4
The set of unique relationship concepts as they exist in Ontologies 1, 2 and 3

Figure 3. The ontology outputs from Phase 1 – The Build.

A reader must be more attuned, to less-apparent connections existing in legislation. These legislative connections are only possible to identify by tracing all cross-referencing in the legislation. For FTB, the two core pieces of legislation cross-reference another 13 pieces of legislation. These include the *Migration Act, 1958*; *Income Tax Assessment Act, 1997*; *Military Rehabilitation and Compensation Act, 2004*; and the *Social Security Act, 1991*. The connections are not intuitive, but exist nevertheless.

C. *Research method*

The case study uses two different methods to generate the data for the research. The first method requires the researcher to undertake a manual exercise to identify the terms in the two pieces of legislation and the claim form, and to develop three separate models as ontologies as well as a single view, or domain model (see Figure 3). The process used to build the ontologies and any observations from the build phase will presented for appraisal to key informants from Business, ICT, and legal areas of the policy and service delivery departments.

The research will develop a set of assumptions to indicate where misalignment may exist, and if found to exist, will require some correction. Although the researcher will suggest the possible implication of any misalignment identified, it is only by appraisal by key informants that the action to correct the misalignment will be made to the department responsible for the information system.

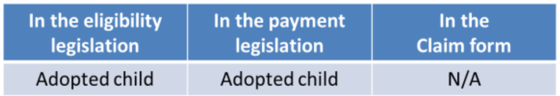
So far, the research in progress has identified the following three patterns indicating possible misalignment, (1) the legislative terms, or synonyms for these terms in Ontologies 1 or 2, are not present in the Claim form Ontology 3, (2) the Claim form ontology 3 introduces terms that are not present in the legislation ontologies 2 or 3, and (3) relations between terms in the Claim form ontology do not maintain the structure used in the legislation ontologies 1 and 2.

Where any of these patterns are identified then a closer investigation will be undertaken. Figure 4 is an example where the legislative term 'Adopted child' is not used in the Claim form. Further investigation shows that the Claim form is using a related term 'adoptive parent'. That is, the Claim form has introduced a term that is not present in the legislation. If a synonym term is being used, it will be necessary to conduct an ontological assessment to confirm whether it is a synonym or another term. A synonymy is a relation between terms in a given language representing the same concept [19]. For example, in the legislation providing for payment of the FTB, two forms of FTB Child 'of' are used. (1) FTB Child of the individual, and (2) FTB Child of the adult. By analyzing the legislation, the individual and the adult are synonyms, therefore, when building the domain ontology only one relationship will be modelled, and one synonym will be recorded. This is an example of how logic based reasoning is being used to understand the differences between similar terms. The power of ontology is that it can return inferences and aggregations provided the information has been declared (coded) in a software-processable format.

These examples demonstrate how a manual process can be used to identify misalignment between the legislation and the Claim form in the FTB domain. Rather than supporting two views, government can agree to harmonize and use only one term in the future. Alternatively it may be agreed that more than one term will be maintained, and this may require the development of further guidance material for service providers and service consumers. Whatever the decision, an agreed and explicit understanding can be captured in an ontology, and this would remove the reliance on a few highly skilled legal interpreters. It is likely that the research in progress will detect ambiguities existing in the legislation, or the claim form. By removing the ambiguities, a closer legislative alignment will be possible.

Another related method is ontology matching. The focus of matching is to discover the differences between two ontology versions. The challenges for ontology matching process, have received recent attention, and include: discovering missing background knowledge, selection, user involvement, explanation of matching results, and alignment management [18]. Once a domain ontology has been created it will be important to undertake matching on a regular basis to manage the continued alignment. This is another requirement for legislative change management that will be considered in future research.

The representation of the knowledge will be captured in Resource Description Framework (RDF), a standardized syntax for encoding RDF statements to make them software processable [20]. All RDF triples can be developed as a distributed graph, and captured as a Uniform Resource Identifier (URI) address for location by other resources. The relationships between the service providers and service subject/predicate/object form statements. Two examples of triples are shown in Figure 5. The triples describe that (1) an applicant must be responsible for a child, and that (2) an applicant must apply for the payment to the Secretary.



In the eligibility legislation	In the payment legislation	In the Claim form
Adopted child	Adopted child	N/A

Figure 4. Adopted child example.

The ontological representations will be entered in an ontology builder to provide a model of the relationships. The output will include a model of the FTB domain and, an ontology for each of the separate pieces of legislation, and the claim form.

In the development of the domain ontology, government is interested in the relationship of the parent and the child. For the FTB, the agencies connected to this payment through the legislation are interested in the way the FTB legislation describes these terms. For service consumers who are parents with children, they too would enjoy a model that described the relationship for FTB, as opposed to any other payment. Government as a whole would also be interested in understanding the 'parent picture' in its entirety. All these views can be accommodated using ontologies.

This section has outlined the proposed research strategy for the first phase of this research. It has described the case study methodology and the methods that will be used to build the research artefacts and to gather the necessary data to conduct the research.

IV. RESEARCH LIMITATIONS AND RISKS

This section will describe the limitations and risks of undertaking this research using the methods outlined in Section III. Two limitations of the research arise because the selection of the legislation is restricted to the FTB payment in normal circumstances. The first limitation is that only some of the legislation relating to the FTB payment will be modelled as an ontology. The second limitation is that only system end-points will be compared i.e., the current legislation and claim form. The changes to the legislation and the form that have occurred since the legislation's commencement date cannot and will not be individually analysed. This point-in time analysis will be useful to identify the alignment issues, but it will not be possible to understand the reasons for the alignment issues.

V. CONTRIBUTION

The manual process used to detect and compare the ontological patterns existing in the legislation and the claim form are expected to be transferable to other information system artefacts used to operationalise the legislation. If the terms and relationships existing in the family tax benefit domain can be modeled as an ontology, then, it should be possible to model all legislation being administered by Government. The contribution is a novel strategy and method of conducting ontological analysis, and a novel means of conducting alignment assessment. The power of

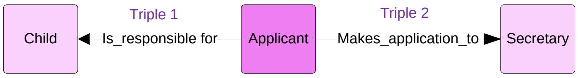


Figure 5. Two examples of relationships captured as RDF triples.

ontologies in e-government would mean that a new use of technologies will result in more convenient access to government information and services, and provide service consumers greater opportunities to participate in the democratic institutions and processes [2].

If a domain ontology exists, and a legislation change occurs, the ontology could be used by government to identify the owners of: systems, processes, activities, guidelines, forms, etc., that may be impacted. Evidence-based assessments would improve the quality of such assessments for policy makers, service providers, and service consumers. Policy makers exploring changes to legislation would be better informed as an evidence-based estimate of a whole of government impact would be possible. Service Delivery departments could schedule programs of work anticipating the changes to the systems and processes to comply with the legislation, and with this knowledge, they will be able to improve their engagement through clear messaging to service consumers.

VI. Conclusion

This research develops a new way to reveal important aspects of the relationship between legislation and its implementation in government information systems, using ontologies. This work-in-progress paper has described the research approach that will be used to assure ministers and service consumers that systems and legislation are aligned. A strategy has been outlined describing the method of conducting ontological analysis as a novel way to use ontology to determine alignment. An instrumental case study has been described using the FTB legislation applying to the payment in normal circumstances. The reasons for the selection of the FTB payment in normal circumstances have been outlined. The limitations and the risks of this research design and the contributions of the proposed research have been discussed. The research offers a new approach using technology for all government agencies to assure their ministers that information systems are aligned to the legislation. This research attempts to develop a method to detect the underlying ontologies existing in legislation that can be used more broadly across all legislation being used in Australian government service delivery. Future work will explore automatic and semi-automatic ways to identify relationships existing in legislation.

References

[1] P. Lambe, Organising Knowledge: Taxonomies, Knowledge and Organisational Effectiveness. Oxford: Chandos Publishing (Oxford) Limited, 2007.

[2] J.E. Stiglitz, and P.R.Orszag, The Role of Government in a Digital Age, Report commissioned by the Computer & Communications Industry Association (CCIA), 2000.

[3] E. N. Loukis, "An Ontology for G2G Collaboration in Public Policy Making, Implementation and Evaluation." Artificial Intelligence and Law, 15.1, 2007, pp. 19-48.

[4] A. Boer, T. van Engers, and R. Winkels. "Using Ontologies for Comparing and Harmonizing Legislation," in Proceedings of the 9th International Conference on Artificial Intelligence and Law. ACM, 2003, pp. 60-69.

[5] A. Gomez-Perez, F. Ortiz-Rodriguez, and B. Villazon-Terrazas, "Legal Ontologies for the Spanish e-Government," in Current Topics in Artificial Intelligence, Springer. Berlin Heidelberg, 2006, pp. 301-310.

[6] L. Cabral, J. Domingue, L. Gutierrez, M. Rowlatt, and R. Davies, "WP 9: Case Study eGovernment D9.5 Change of Circumstance WSMO Descriptions." Change 9 (2005): 5.

[7] The Australian National Audit Office, Public Sector Audit Committees. Independent Assurance and Advice for Accountable Authorities. Better Practice Guide. Commonwealth of Australia. March 2015.

[8] K. Lynch, P. Baltzan, and P. Blakey, Business Driven Information Systems 2nd ed., Australia: McGraw-Hill Education – Europe, 2013.

[9] Australian Government, Administrative Review Council, "Automated assistance in administrative decision making" Report no. 46, to the Attorney-General, November 2004.

[10] Y. Gong, and M. Janssen. Adaptive and Compliant Policy Implementation: Creating Administrative Processes Using Semantic Web Services and Business Rules. In Collaborative, Trusted and Privacy-Aware e/m-Services. Springer Berlin Heidelberg, 2013, pp. 298-310.

[11] The Auditor-General, "Administering Regulation, Better Practice Guide", March 2007.

[12] J. Bourgon, "The Future of Public Service: a Search for a New Balance," Australian Journal of Public Administration, 67, no. 4, 2008, pp. 390-404.

[13] T. R. Gruber, "A Translation Approach to Portable Ontology Specifications," Knowledge Acquisition, 5, no. 2, 1993, pp. 193-220.

[14] R. Mizoguchi, "Tutorial on Ontological Engineering Part 2: Ontology Development, Tools and Languages," New Generation Computing, 22, no. 1, 2004, pp. 61-96.

[15] G. P. Malafsky, and B. Newman. "Organising Knowledge with Ontologies and Taxonomies", [Online]. Available from: http://www.techi2.com/download/Malafsky%20KM%20taxonomy_ontology.pdf. 2015.05.08.

[16] R. Jasper, and M. Uschold. "A Framework for Understanding and Classifying Ontology Applications," in Proceedings of the 12th International Workshop on Knowledge Acquisition, Modelling, and Management KAWVol. 99, 1999.

[17] R. K. Yin, Case Study Research Design and Methods. Beverly Hills. Sage Publications, 1994.

[18] P. Shvaiko, and J. Euzenat. "Ten Challenges for Ontology Matching," in Proceedings of the 7th International conference on ontologies, dataBases, and applications of semantics (ODBASE), 2008, 1163-1181.

[19] ISO 1087-1:2000 Terminilogy work – Vocabulary – Part 1: Theory and Application, ISO, 2000.

[20] L. W. Lacy, Owl: Representing Information Using the Web Ontology Language, Trafford Publishing, Victoria, BC, Canada, 2005.

Storing and Querying Ontologies in Relational Databases: An Empirical Evaluation of Performance of Database-Based Ontology Stores

Raoul Kwuimi
Department of Information and Communication Technology
Vaal University of Technology
Private Bag X021, Andries Potgieter Blvd,
Vanderbijlpark, 1900
Vanderbijlpark, South Africa
e-mail: kwuimi@gmail.com

Jean Vincent Fonou-Dombeu
Department of Software Studies
Vaal University of Technology
Private Bag X021, Andries Potgieter Blvd,
Vanderbijlpark, 1900
Vanderbijlpark, South Africa
e-mail: fonoudombeu@gmail.com

***Abstract*—Mapping ontologies to relational databases is an active research topic in Semantic Web. Therefore, several platforms have been developed to enable the storage and query of ontologies in relational databases. However, only a few studies have empirically measured and compared their performances in terms of speed and scalability. In this paper, two popular database-based ontologies stores, namely, Jena API and Sesame are used to load and query five selected ontologies of different sizes into MySQL relational database. Various metrics including (1) the loading times of ontologies into the relational databases, (2) the response times of SPARQL queries executed on the stored ontologies databases and (3) the sizes of the ontologies databases are used to measure and compare the performance of the two Semantic Web platforms. Experiments show that (1) both platforms are scalable and could successfully parse, load and query ontologies of different formats (OWL/RDF) and sizes into relational databases, (2) Jena API performs faster with small size ontologies, whereas, Sesame is more efficient with bigger size ontologies with regards to loading of ontologies into relational databases, (3) Sesame provides quicker responses to SPARQL queries compared to Jena API and (4) the disk space required to store the resulting ontologies databases in both platforms are proportional to the initial sizes of the ontologies and is higher in Jena API than in Sesame.**

Keywords-Jena API; Sesame; SPARQL; Ontology Storage; Relational Databases .

I. INTRODUCTION

The Semantic Web is an improvement of the current World Wide Web (WWW) in which web contents are represented on the basis of their meaning rather than web links as in the current internet. The meaning of web content are represented with ontology. Ontology as explained in [2] is a knowledge base system that contains a vocabulary of basic terms concerning a particular domain and semantic interconnections between those terms. It is the formal representation of data used on the semantic web. Several languages are used to represent ontologies in Semantic Web; they include Extensible Markup Language (XML), DARPA Agent Markup Language (DAML), Resource Description Framework (RDF), RDF Schema (RDFS) and Web Ontology Language (OWL) [3]. Two of those languages are widely used and recommended by the World Wide Web Consortium (W3C) including RDF/RDFS and OWL languages [1] [2] [4]. Ontology generated in these languages need to be persistently stored and used within Semantic Web applications.

In the Semantic Web domain, 3 techniques are used for ontologies storage, namely, (1) In-memory storage, (2) File or native storage, and (3) database storage. The in-memory storage is efficient only for small size ontologies, i.e., when the ontology has less instances or statements. It provides quick query response times because the ontology is residing in the main memory of the computer. When the ontology is large in size, persistent storage is appropriate as the ontology can no longer be stored in the main memory of the computer. Native storage makes use of files to store ontologies. The database technology has been used for more than 30 years [3]. In Semantic Web database storage is useful in many cases where storage is required on the web [5]. In fact, ontologies used in online systems today are of hundreds of Megabytes to thousands of Gigabytes in size; they need to be stored in relational databases for their efficient and optimal utilization [6] [7] [8].

Several platforms have been developed to enable the persistent storage and query of ontologies in relational databases. Relational databases are mostly used over object and object relational databases because, it provides performance, maturity, availability and reliability [43]; the most commonly used platforms are: AllegroGraph, Jena API, Open Anzo, Oracle Semantic [8], Minerva [18] [42] and Sesame [12]. Oracle semantic and AllegroGraph are currently available only in the form of trial versions [8]. Further, Open Anzo, AllegroGraph and Minerva do not process ontologies written in RDF syntax. Jena API and Sesame support both OWL and RDF ontologies as well as MySQL which is a widely used Relational Database Management System (RDMS) on the web. Further, Sesame and Jena API are both open source platforms and are accessible free of charge with full functions and supports. To date, only a few studies have empirically measured and compared their performances in terms of speed and scalability.

In this study, Jena API and Sesame are used to load and query five selected ontologies of different sizes into MySQL relational databases. Various metrics including (1) the loading times of ontologies into the relational databases, (2) the response times of SPARQL queries executed on the stored ontologies databases and (3) the sizes of the ontologies databases are used to measure and compare the performance of the two Semantic Web platforms. Experiments show that (1) both platforms are scalable and could successfully parse, load and query ontologies of different formats (OWL/RDF) and sizes into relational databases, (2) Jena API performs faster with small size ontologies, whereas, Sesame is more efficient with bigger size ontologies with regards to loading ontologies into relational databases, (3) Sesame provides quicker responses to SPARQL queries when compared to Jena API and (4) the disk space required to store the resulting ontologies databases in both platforms are proportional to the initial sizes of the ontologies and higher in Jena API than in Sesame.

The rest of the paper is organized as follow. Section 2 discusses existing approaches for storing ontology on the Semantic Web. Characteristics of existing platforms for ontologies storage and query are presented in Section 3. Section 4 describes the experimental design of the study in terms of the dataset, performance metrics and tools employed. The last part of Section 4 presents and discusses the experiments and results of the study. Related studies are discussed in Section 5 and a conclusion ends the paper in Section 6.

II. ONTOLOGY STORAGE TECHNIQUES

Ontology storage is based on 3 main models (Figure 1). These include: (1) In-memory storage, (2) Native or File-based storage and (3) Databases-based Storage [9] [11]. In-memory or Memory-based storage uses the central memory of the computer to store ontologies. It is very efficient and fast with small scale ontologies. The drawback of this technique is that as the ontology get larger, it becomes more difficult to manipulate. In fact, ontologies stored using the in-memory storage technique need to be loaded in the memory every time a user wants to run an application that is using it. The native storage technique uses files to store ontologies. Ontologies statements are stored in triple store in the form of (S, P, O) where S is the Subject, P the Predicate and O the Object. The advantages of native storage are that data loading and data query are fast [37]. In order to retrieve data easily and quickly with fewer errors, index algorithms such as the B-tree or B+ [10] [37] are used. Structuring and editing of ontologies are very efficient as well [39]. The main drawback of this technique is that large scale ontologies are difficult to process. Furthermore Native storage needs to implement functionality such as data recovering, query optimization, controlled access and transaction processing in order to improve its data processing and management [37]. In the database-based storage, the ontology is stored in a Relational Database (RDB). Ontology storage in RDB needs to provide at least three of the following technologies: store and scalability, support for reasoning, and SPARQL query facilities [38] [39] [41]. Database-based storage is usually grouped into 2 main types [39], namely, generic and ontology specific (Figure 1). The generic schema [11] uses one table to store all triples or statements in the ontology. The table contains 3 columns, each representing an element of the ontology statement including Subject, Predicate and Object. Every row in the table is an ABox fact [11]. ABox are statements that describe the relationship between instances of the ontology [18]. TBox facts are ontology statements that describe relationship between classes and properties [18]. Many tables are required to store axioms or TBox facts of the ontology. The ontology specific format (Figure 1) creates tables according to the contents of the ontology. It has 3 modes of representation: horizontal, vertical and hybrid [11] [39]. In the horizontal mode also called one-table-per-class mode, every class is represented by a table with 2 columns. The first column represents the instance ID and the second column represents the predicate in which the instance ID belongs to. Properties are stored as values in the second column in the class table. In the vertical representation, also called one-table-per-property mode or decomposition storage model [11], tables are created for all properties of the ontology. Every table contains two columns as in the horizontal model including the Subject and Object columns to record the subjects and objects of ABox and TBox facts of the ontology. The hybrid model combines both vertical and horizontal representations in which tables are created for classes and properties.

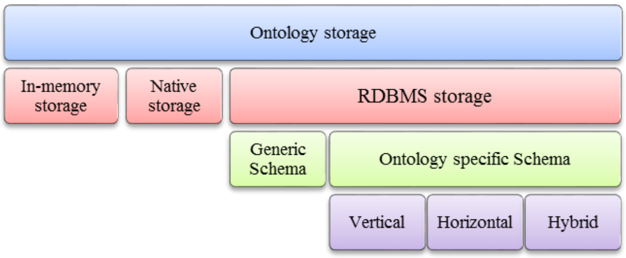


Figure 1. Ontology storage models

As shown in Figure 1 above, unlike in-memory and native storages, only RDBMS storage gives the possibility to elaborate further on the storage technics employed.

III. ONTOLOGY STORAGE AND QUERY PLATFORMS

As mentioned earlier, several platforms have been developed to enable the store and query of ontologies in relational databases. The commonly used platforms are: AllegroGraph, Jena API, Open Anzo, Oracle Semantic [8], Minerva [42] and Sesame [12]. AllegroGaph store ontologies as graphs [8]. It is installed as a server application and requires client applications such as Java, C#, Python, Ruby, Perl or Lips to access it. It supports SPARQL as query language but provides API for direct access to Subject, Predicates and Objects of ontology triples or statements without any use of SPARQL queries. Minerva [18] [42] is a component of the Integrated Ontology Development Toolkit (IODT). It is used as a plugin in Eclipse IDE. It stores OWL ontologies and supports the SPARQL query language. It also supports IBM DB2 and Derby as backend databases. Open Anzo is a Semantic Web platform developed by IBM. It can be used in three different modes: (1) embedded in an application, (2) installed as a server application and accessed remotely by clients or (3) run locally [8]. It supports the SPARQL query language. Further, it supports persistent storage through its Storage Service Layer which interacts with Relational Databases. In order to interact with Open Anzo, the client stack layer uses three different languages, namely, Java, Java Script or dot Net [8]. Open Anzo supports DB2 and Oracle as backend databases. Jena API is integrated into Eclipse IDE as a library and uses a variety of DBMS such as Oracle, PostgreSQL and MySQL [8] [16]. It enables ontologies to be stored in three storage models: in-memory, native or RDB. The query languages supported by Jena API are SPARQL and RDQL. The Oracle Semantic [8] is a Jena Adapter that works with

Oracle databases [8]. It is a plugins that implements Jena Graph and Jena Model interfaces. It also supports the SPARQL query. Sesame is a Software Development Kit (SDK) that was developed in the European IST project On-to-Knowledge [12]. It enables ontologies to be queried or exported. Two languages are used for ontology query in Sesame, namely, SPARQL and SeRQL. The Sesame architecture [12] has one component called the SAIL API which translates an ontology file into its RDB representation as well as enables Sesame to interface 2 DBMS, namely, MySQL and PostgreSQL. A comparative of the characteristics of the abovementioned platforms is provided in Table 1. The columns OWL and RDF show the platforms that support ontologies in these formats. The third column indicates those that are open source or not. Jena API and Sesame are used in this study as they both support RDF and OWL ontologies as well as MySQL RDBMS. Furthermore, Sesame and Jena API are both open source platforms and are accessible free of charge with full functions and support from the Internet.

TABLE I. CHARACTERISTICS OF ONTOLOGY STORAGE PLATFORMS

Ontology	OWL	RDF	Open Source	Availability
Allegrograph	no	yes	no	Commercial/free
Jena API	yes	yes	yes	free
Sesame	yes	yes	yes	free
Open Anzo	no	yes	yes	free
Oracle Semantic	yes	yes	no	Commercial/free
Minerva	no	yes	no	free

TABLE 1 shows a résumé on different platforms' attributes that guided us to select the two platforms used on the experiments.

IV. EXPERIMENTS

A. *Dataset*

The dataset is constituted of five ontologies, namely, Gene Ontology (GO) [20] [21] [24] [26] [27], WordNet [29] [30] [31], OntoDPM [32], Biological Top Level (BioTop) [33] [34] and Central Government ontology (CGOV) [28]; they have all been used intensively in related studies.

The GO ontology describes the biology domain in terms of molecular function, cellular components and biological process. It contains the vocabulary used in the biology field and the relationship between terms [21] [22] [23]. The WordNet ontology is an electronic lexical database for the English language [36]. It contains verbs, nouns, adverbs and adjectives. Written in a machine readable format, online dictionaries access it for public usage [29]. The OntoDPM ontology is a knowledge-based model for e-government monitoring of development projects in developing countries [32]. The BioTop ontology is an ontology of the life sciences domain which focuses on molecular biology [33]. It is used as a top level ontology to link the Open Biomedical Ontologies (OBO). The CGOV is an ontology of the UK central government; it models the structure of the UK central government [28].

TABLE II. CHARACTERISTICS OF ONTOLOGIES IN THE DATASET

Ontologies	Format	Size (Bytes)	No. Classes	No. Properties	No. Individuals
OntoDPM	OWL	38,578	30	19	18
CGOV	RDF	68,551	46	46	-
BioTop	OWL	429,989	389	92	-
WordNet	RDF	100,428,111	-	-	-
GO	OWL	106,912,638	-	-	-

Table 2 provides some metadata on the abovementioned ontologies constituting the dataset in this study in terms of their formats (RDF/OWL), sizes, and number of classes, properties and individuals. Some cells of Table 2 were not filled in due to the fact that the expected values were unavailable. In fact, in order to get the metadata in Table 2, an online ontology documentation tool called parrot is used [25]. Ontologies to be analysed are loaded within Parrot in three different ways including (1) uploading the ontology file, (2) pasting the code of the ontology or (3) providing the http address of the ontology. After loading the ontology and executing Parrot, ontologies characteristics such as the number of classes, properties and individuals are displayed. The loading of large ontologies such as GO and WordNet resulted in errors and therefore no characteristics were retrieved.

B. *Performance Metrics*

Three standard database performance metrics were used to measure and compare the performance of Sesame and Jena API in storing and querying ontologies in relational databases including,

(1) The loading time which is a common performance metric used in RDBMS studies [17] [18] [19]; it represents the time taken by a platform to process, parse and load an ontology into a relational database, (2) The query response time (QRT) [40] which represents the time taken by a platform to display the result of a query and (3) The repository size [19] which is the space disk needed for the storage of the resulting ontologies databases. In this study, the query response time is the average response times of several consecutive executions of the same query.

C. *Computer and Software Environments*

The experiments were carried out on a computer with the following characteristics: 64-bit Genuine Intel 2160 processor, Windows 8 release preview, 4 GB RAM and 160 GB hard drive. Protégé version 4.3 was installed in the computer and used to create the OWL code of OntoDPM ontology. The Apache tomcat server version

6.0 was installed in order to deploy the Sesame server. The Wamp server was installed as well to enable access to MySQL backend DBMS via Sesame and Jena API. Finally, Jena API was configured in the Eclipse IDE version 4.2. The metadata on the ontologies in the dataset such as the numbers of classes, properties, instances, etc. were determined with the online Semantic Web ontology documentation software, named, Parrot [25].

D. *Experimental Results*

1) *Data Loading into RDB*

The ontologies were loaded into MySQL relational databases via Sesame and Jena API, respectively. In Sesame, ontologies were loaded in command line mode [35]. A sample code used to load the ontologies in Jena API is provided below. The code shows part of the Jena application that reads and loads ontologies into MySQL databases.

```
1. ModelMaker maker =
ModelFactory.createModelRDBMaker(conn);
2. Model loader = maker.createDefaultModel();
3. FileInputStream inputStreamfile = null;
4. File file = new File ("c:\\Devel\\gene.owl");
5. inputStreamfile = new FileInputStream(file);
6. InputStreamReader reader = null;
7. reader =new InputStreamReader(inputStreamfile,
"UTF-8");
8. loader.read(readed, null);
9. reader.close();
10. loader.commit();
```

In the above code, line 1 creates a model, namely, *maker* which will be used to create the link between a model and the relational database. Line 2 creates a new model which will be used to store the ontology. Lines 3, 4 and 5 create a *FileInputStream* and file objects and then loads the file into the newly created file object. Line 6 creates the reader and line 7 loads the ontology into the reader. Line 8 reads the file from the reader and loads it into the ontology model. Finally line 9 closes the model and line 10 commits the model into the database.

Table 3 shows the loading times of the 5 ontologies presented in Sub-Section IV.A into MySQL databases with both Sesame and Jena API. It shows that Jena loads smaller ontologies (in the range of Kilobytes) (Table 2) faster than Sesame. But, for bigger ontologies (in the range of Megabytes) Sesame performs better with regards to loading ontologies into MySQL RDBMS.

The reason is the fact that Sesame opens an ontology file (OWL in this case), reads and loads it straight into MySQL database, whereas, Jena needs to first load the ontology into a RDF graph in the main memory before transferring it into MySQL database.

TABLE III. LOADING TIMES OF ONTOLOGIES INTO MYSQL DATABASES

Ontologies	Sesame Time (hh:mm:ss.000)	Jena API Time (hh:mm:ss:000)
OntoDPM	00:02:27.0	00:00:32.325
CGOV	00:05:15.776	00:00:45.318
BioTop	00:11:35.95	00:04:23.858
WordNet	14:27:34.387	17:44:10.365
GO	15:50:51.910	16:20:48.830

The data in Table 3 is represented graphically in Figure 2 in which the blue and red bars represent the loading times of ontologies into MySQL databases with Sesame and Jena API, respectively. The blue bars show that in Sesame the loading time is proportional to the size of the ontology, whereas, the red bars suggest that the loading time in Jena is disproportional to ontology sizes. In fact, the Gene Ontology which is bigger than WordNet (Table 2) took less time to be loaded into MySQL database. Figure 2 also shows that Jena loads small ontologies faster (less than a minute) and is slower in loading big ontologies compared to Sesame.

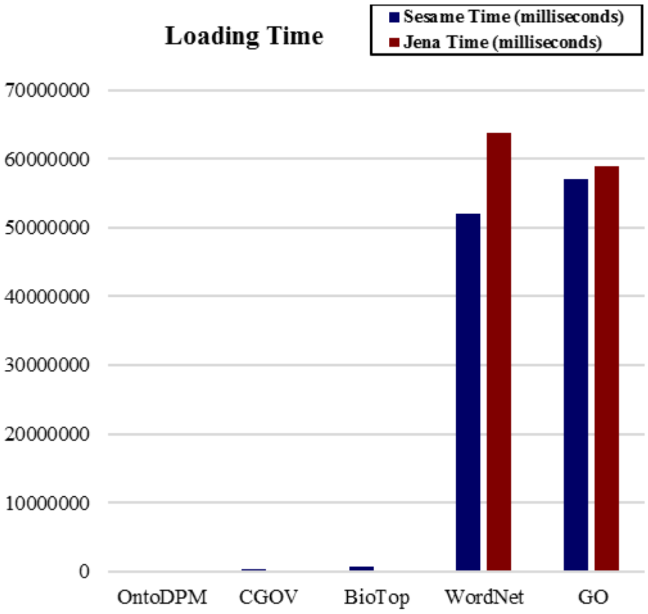


Figure 2. Chart of Loading Times of Ontologies into MySQL Databases

2) *Queries Response Times*

The query response time (QRT) [40] is the average time taken by a query to return a result. A sample SPARQL [13] query that searches for classes and their subclasses in the MySQL ontologies databases is given in the code below.

```
PREFIX rdf:http://www.w3.org/1999/02/22-rdf-syntax-ns#
PREFIX rdfs:<http://www.w3.org/2000/01/rdf-schema#>
SELECT ?subject ?object
WHERE {?subject rdfs:subClassOf ?object} LIMIT 10.
```

The sample SPARQL query above was executed five consecutive times on each MySQL ontology database and the average response/execution times were recorded in Table 4.

TABLE IV. AVERAGE QUERIES RESPONSE TIMES ON ONTOLOGY DATABASES

Ontologies	Average Time in Sesame (ms)	Average Time in Jena API (ms)
OntoDPM	616.2	2393
CGOV	732.4	2321.6
BioTop	824.4	2369.4
WordNet	91.2	2428.2
GO	4135	2424

Table 4 represents the chart in Figure 3. It shows that the average queries response times are generally lower in Sesame (blue bars) than in Jena API (red bars). Further, the average queries response times in Jean API are almost constant on all ontology databases (red bars).

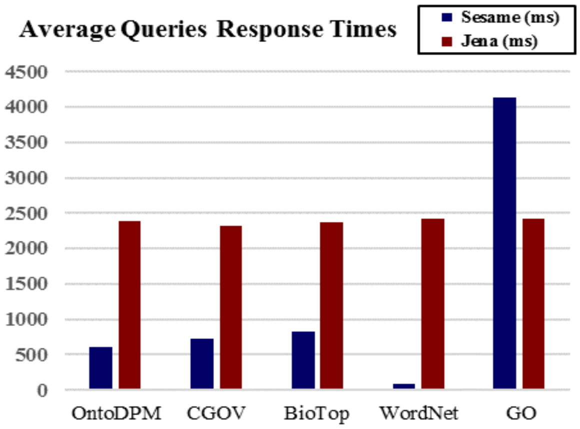


Figure 3. Chart of Average Queries Response Times on Ontologies Databases in Sesame and Jena API

3) *Disk Space for Storing Ontologies Databases*

Figure 4 is a comparison of the disk space used to store the resulting 5 ontologies databases into MySQL RDBMS via Sesame and Jena. The orange bars represent the initial sizes of the ontologies; it can be observed that OntoDPM and CGOV ontologies are very small as described in Table 2. The blue bars show the space required to store the ontologies in the Sesame repository; WordNet and GO required more space due their initial sizes. The red bars show the space required to store the ontologies into MySQL databases via Jena; the spaces used to store the ontology databases for WordNet and GO, are almost double of the space used in Sesame. The ontology databases for OntoDPM, CGOV and BioTop occupied less disk space due to their initial small sizes (Table 2).

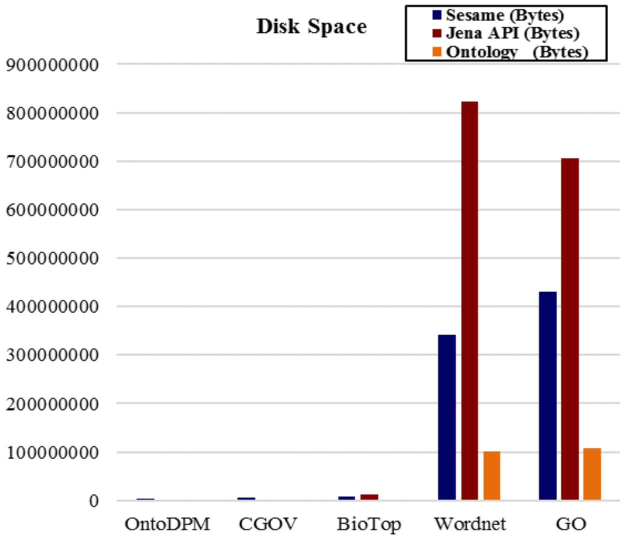


Figure 4. Disk Space Occupied by Ontologies Databases in Sesame and Jena API

As represented above, the graph in Figure 4 clearly shows that the required space is proportional to the ontologies independently of their format.

V. RELATED WORK

Mapping ontology to relational database is an active research topic in Semantic Web. Various techniques for mapping ontology features to that of relational database to enable the persistent storage of ontologies into RDB are presented in [14] [15] [16]. Three ontologies were stored and queried in MySQL databases via Jena API in [17]; the authors drew a similar conclusion as that of this study with regards to the scalability of Jena API. In [44] system properties of Jena Against sesame are provided. The authors describe the main difference between Jena and Sesame in terms of the properties that they are sharing and those which are different. [45] Provides similar analysis as in [44] but both do not provide an empirical analysis of the two platforms in terms of the performance. Several RDF databases solutions are reviewed in [8] and [10]. In [8], an evaluation of selected platforms including Sesame and Jena was carried out. However, not only was the study limited to RDF ontologies, but, the evaluation also was limited to the query response times only. In [10], ontology storage models such as generic and ontology specific schema as well as the functionalities of an RDF middleware and RDF query languages are discussed in detail.

VI. CONCLUSION AND FUTURE WORK

In this research, five ontologies were loaded into MySQL relational databases using two popular Semantic Web platforms, namely, Sesame and Jena API. Three metrics were used to measure and compare the performances of both platforms in terms of speed and scalability. The experiments showed that both platforms are scalable and could successfully parse and load ontologies of different sizes into relational database and that Sesame loads bigger ontologies faster than Jena API into relational databases. Experiments also show that Sesame provides quicker responses to SPARQL queries compared to Jena API.

The future direction of the research would be to extend the study with more platforms so as to provide a more comprehensive performance evaluation of existing Semantic Web platforms for storing and querying ontologies in relational databases.

REFERENCES

[1] A., Lally, J. M. Prager, M. C. McCord, B. K. Boguraev, S., S. Patwardhan, J. Fan, P. Fodor, and J. Chu-Carroll, “Question analysis: how Watson reads a clue”, IBM Journal of Research and Development, Vol. 56, No. 3/4, May-June 2012, pp. 1-14.

[2] A. Gali, C. X. Chen, K. T. Claypool, and R. Uceda-sosa, “From ontology to relational databases”, ER Workshops, LNCS, Vol. 3289, 2004, pp. 278-289.

[3] M. Laclavík, “Ontology and agent based approach for knowledge management”, Doctoral Thesis, Slovak Academy of Sciences, 2005.

[4] X. Lili, S. Lee, and S. Kim, “E-R model based RDF data storage in RDB”, In Proceedings of the 3rd IEEE International Conference on Computer Science and Information Technology (ICCSIT), Chengdu, China, 9-11 July 2010, pp.258-262.

[5] G. De Melo, F. Suchanek, and A. Pease, “Integrating YAGO into the suggested upper merged ontology, tools with artificial intelligence”, In Proceedings of the 20th IEEE International Conference on Tools with Artificial Intelligence. ICTAI'08. Dayton, OH, 3-5 November, 2008, pp.190-193.

[6] A.M. George, B. Richard, F. Christiane, G. Derek, and M. Katherine, “Introduction to WordNet: an on-line lexical database”, International Journal of Lexicography Vol. 3, Issue 4, 1990, pp.235-244.

[7] M. Morsey, J. Lehmann, S. Auer, C. Stadler, and S. Hellmann, “DBpedia and the live extraction of structured data from wikipedia”, Program: electronic library and information systems Vol.46, No. 2, 2012, pp. 157-181.

[8] F. Stegmaier, U. Gröbner, M. Döller, H. Kosch, and G. Baese, “Evaluation of current RDF database solutions”. In Proceedings of the 10th International Workshop on Semantic Multimedia Database Technologies in conjunction with the 4th International Conference on Semantic and Digital Media Technologies (SAMT 2009),Graz, Austria, 2-4 December 2009 ,pp.1-17.

[9] D. HuiJun, W. WenGuo, Y. Jian, “A B-Tree algorithm for partitioned index based on CIDR list”. International Conference of Information Technology, Computer Engineering and Management Sciences, Nanjing, Jiangsu, 24-25 September, 2011, pp. 124-128.

[10] A. Hertel, J. Broekstra, and H. Stuckenschmidt, “RDF Storage and Retrieval Systems”, International Handbooks on Information Systems, 2009, pp.489-508.

[11] F. Dieter, V.H. Frank, K. Michel, and A. Hans, “On-To-Knowledge: ontology-based tools for knowledge management”, In Proceedings of the eBusiness and eWork (EMMSEC 2000) Conference, Madrid, Spain, 18-20 October, 2000, pp. 1-7.

[12] D. Fensel, J. Hendler, H. Lieberman, W. Wahlster, and T. Berners-Lee, “Sesame: an architecture for storing and querying RDF data and schema information, spinning the semantic web: bringing the World Wide Web to its full potential”, MIT Press, 2005, pp. 197-222.

[13] S. Harris, “SPARQL query processing with conventional relational database systems”, In Proceedings of the 6th International Workshop on Scalable Semantic Web Knowledge Base Systems (SSWS), New York City, USA, November 2005, 20-22 pp.235-244.

[14] W. Teswanich, and S. Chittayasothom, “A transformation from RDF documents and schemas to relational databases”, In Proceedings of the IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, Victoria, Canada, 22-24 August 2007, pp.38-41.

[15] S. Ramanujam, A. Gupta, L. Khan, S. Seida, and B. Thuraisingham, “R2D: A bridge between the semantic web and relational visualization tools”, In Proceedings of the 3rd IEEE International Conference on Semantic Computing (ICSC), Berkeley, CA, USA, 14-16 September 2009, pp.303-311.

[16] A. Alamri, “The relational database layout to store ontology knowledge base”, In Proceedings of the International Conference on Information Retrieval & Knowledge Management (CAMP), Kuala Lumpur, Malaysia, 13-15 March, 2012, pp.74-81.

[17] J. V. Fonou-Dombeu, N. Phiri and M. Huisman, “Persistent storage and query of e-government ontologies in relational databases”, In Proceedings of the 3rd International Conference on Electronic Government and the Information System Perspective, Munich, Germany,1-4 September, 2014, pp.118-132.

[18] J. Zhou, L. Ma, O. Liu, L. Zhang, Y. Yu, and Y. Pen, “Minerva : a scalable owl ontology storage and inference system”, In Proceedings of the First Asian conference on The Semantic Web, Beijing, China,3-7 September 2006, pp.426-443.

[19] G. Yuanbo, P. Zhengxiang, and J. Helfin, “LUBM a benchmark for OWL knowledge base systems”, In Proceedings of the 3rd International Conference on Semantic Web (ICSW),Hiroshima, Japan, 7-11 November, 2004, pp.158-182.

[20] K. Taha, “GOseek: A gene ontology search engine using enhanced keywords”, In Proceedings of the 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Osaka, Japan, 3-7 July 2013 pp.1502-1505.

[21] X. Qing-Wei, L. Qiang, L. Qian, L. Qing-Ming, and L. Yi-Xue, “Semantic query and reasoning among gene ontology annotations”, In Proceedings of the 1st International Conference on Bioinformatics and Biomedical Engineering, ICBBE, Wuhan, China, 6-8 July, 2007, pp. 298-301.

[22] M. Gan , and R. Jiang, “Inferring semantic similarity through correlating information contents of gene ontology terms”, In Proceeding of the IEEE International Conference on Bioinformatics and Biomedicine (BIBM) ,Shanghai, China, 18-21 December, 2013, pp.1-5.

[23] C. Chien-Ming ,C. Chia-Sheng, H. Zhen-Li, and P. Tun-Wen, “Discover significant associations of orthologous simple sequence repeat patterns with gene ontology terms”, IEEE International Conference on Bioinformatics and Biomedicine Workshop, Washington, USA, 1-4 November, 2009,pp.209-213.

[24] L. Jing, Ng, M.K.;Y. Liu , “Construction of gene networks with hybrid approach from expression profile and Gene Ontology”, IEEE Transactions on Information Technology in Biomedicine, Vol.14 , Issue: 1, 2009, pp.107-118.

[25] Parrot online tool, Available at http://ontoruleproject.eu/parrot.html [Retrieved: July,2015]

[26] Gene Ontology Consortium, “The Gene Ontology (GO) database and informatics Resource”, Nucleic Acids Research Journal, Jan, Vol. 32, No. 1, 2004, pp. 258-261.

[27] B. Smith, J. Williams, and S. Schulze-Kremer, “The ontology of the Gene Ontology”, In Proceedings of the Annual Symposium of the American Medical Informatics Association, Washington DC, USA, November, 2003, pp.609-613.

[28] CGOV: Central Government Ontology. An Ontology of the UK Central Government, Available at

http://reference.data.gov.uk/def/central-government [Retrieved: May,2014]

[29] G.A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. Miller, "WordNet: An on-line lexical database", International Journal of Lexicography, Vol 3, No.4, 1990, pp.235-244.

[30] M. Van Assem , A. Gangemi, and G. Schreiber, "Conversion of WordNet to a standard RDF/OWL representation", In Proceedings of the Language Resources and Evaluation conference, Geneo, Italy, 24-26 May, 2006, pp.1-6.

[31] T. Pedersen, S. Patwardhan, and J. Michelizzi, "WordNet::similarity - measuring the relatedness of concepts", In Proceedings of the Nineteenth National Conference on Artificial Intelligence (AAAI-04), San Jose, CA,USA, July 25-29, 2004, pp. 1024-1025.

[32] J.V. Fonou-Dombeu, and M. Huisman, "Combining ontology development methodologies and semantic web Platforms for E-government Domain Ontology Development. International Journal of Web & Semantic Technology (IJWesT) Vol.2, No.2, October, 2011, pp.12-25.

[33] S. Schulz, E. Beisswanger, J. Wermter, and U. Hahn, "Towards an upper-level ontology for molecular biology", In Proceedings of the AMIA annual symposium, 2006, pp.694-698.

[34] E. Beisswanger, S. Schulz B., H. Stenzhorn , and U Hahn, "BIOTOP: An upper domain ontology for the life sciences :a description of its current structure, contents, and interfaces to OBO ontologies", Journal of Applied Ontology- Towards a meta ontology for the Biomedical Domain, Vol. 3, No. 4,2008, pp.205-212.

[35] Sesame online documentation, available at http://openrdf.callimachus.net/sesame/2.7/docs/users.docbook ?view [Retrieved: September, 2014]

[36] X. Huang, and C. Zhou, "An OWL-based WordNet lexical ontology", Journal of Zhejiang University Science A, Vol.8, No. 6, May 2007 pp.864-870.

[37] S. Heymans, L. Ma, D. Anicic, Z. Ma, N. Steinmetz, Y. Pan, J. Mei, A. Fokoue, A. Kalyanpur, A. Kershenbaum, E. Schonberg, K. Srinivas, C. Feier, G. Hench, B. Wetzstein, and U. Keller. "Ontology reasoning with large data repositories, Semantic web and beyond computing for human experience", Vol.7, 2008, pp.89-128.

[38] Y. Yuang, G. Li, and X.Wang, "Semantic web rough ontology: definition, model and storage method", In Proceedings of the 6th International Conference on Information Management, Innovation Management and Industrial Engineering (ICIII), Xi'an, China, 23-24 November, 2013, pp.104-107.

[39] Z. Zhou, and X. Yongkang, "A study on ontology storage based on relational database", china, IEEE Conference Anthology, 1 8 Jan, 2013, pp.1-5.

[40] R. Agrawal A. Somani, and Y. Xu, "Storage and querying of e-commerce data", In Proceedings of the 27th International Conference on Very Large Data Bases, Rome, Italy, September 11-14, 2001, pp.149-158.

[41] J. Lu, M. Li , Z. Lei , B. Jean-Sébastien , W. Chen , P. Yue , and Y. Yong, "SOR:A practical System for ontology storage, reasoning and search", In Proceedings of the 33rd international conference on very large databases, Vienna, Austria, September 23-28, 2007, pp.1402-1405.

[42] J. Zhou, L. Ma, Q. Liu, L. Zhang, Y. Yu, and Y. Pan, "Minerva a scalable owl ontology storage and inference system", In Proceedings of the first Asian semantic web conference, Beijing, China, 3-7 September, 2006,.pp 429 443.

[43] I. Astrova, N. Korda, and A. Kalja, "Storing OWL Ontologies in SQL Relational Databases", International Journal of Electrical, Computer, and Systems Engineering, Vol. 1, No 4, 2007.

[44] http://blog.sparna.fr/2012/05/08/rdf-sesame-jena comparaison-des-fonctionnalites/ [Retrieved: July, 2015]

[45] http://db-engines.com/en/system/Jena%3BSesame [Retrieved: July, 2015]

Matching Terminological Heterogeneous Ontologies by Exploiting Partial Alignments

Frederik Christiaan Schadd*, Nico Roos†
Department of Knowledge Engineering, Maastricht University
Maastricht, The Netherlands
Email: *frederik.schadd@maastrichtuniversity.nl, †roos@maastrichtuniversity.nl

Abstract—**Matching ontologies which utilize significantly heterogeneous terminologies is a challenging task for existing matching techniques. These techniques typically exploit lexical resources in order to enrich the ontologies with additional terminology such that more terminological matches can be found. However, they are limited by the availability of an appropriate lexical resource for each matching task. For this scenario, we propose a new technique exploiting partial alignments. We evaluate our technique on a dataset which is characterized by matching problems with significant terminological heterogeneities. Further, we compare our technique with the performance of matching systems utilizing lexical resources to establish whether a partial-alignment-based matcher can perform similarly to a lexical-based matcher. Lastly, we provide a performance indication of a system utilizing both partial alignments and lexical resources.**

Keywords–Ontologies; Semantic Interoperability; Ontology Alignment; Partial Alignments.

I. Introduction

Semantically structured data facilitates many services which are used in a modern society, ranging from agent communication [1] to semantic querying systems [2]. An important criterion for the functionality of such systems is their ability to access multiple sources of semantically structured data. The structure of this data is determined by an ontology, which can be defined using expressive languages, such as the Resource Description Framework Schema (RDFS) or the Web-Ontology-Language (OWL). Information exchange between two sources is possible if both are based on the same ontology.

A common issue is that two sources of semantic data are based on two different ontologies modelling the same domain. The ontologies can differ with regard to their terminology, structure, scope or granularity [3]. In order to transfer data between different ontologies, the data has to be transformed such that it complies with the ontology of the receiving knowledge system. For this to happen, a mapping between the two ontologies is needed. This mapping specifies for every concept in the first ontology whether there is an concept in the second ontology modelling the same information. The process of creating such a mapping is known as ontology mapping.

Ontologies can differ by their applied terminologies. If two ontologies are created by different domain experts then it may happen that the experts prefer different terms in order to refer to the same concepts. This problem can be exacerbated if the two ontologies conform to different design principles, thus varying with regard to their naming formats. Alternatively, the ontologies may simply differ with regard to the used natural language, which can occur in international data-exchange situations. In these scenarios, the two given ontologies have very little overlap with regard to their terminologies, a problem which we refer to as a *terminological gap*. Name-based approaches for ontology mapping are thus unlikely to produce satisfying mapping results in such scenarios.

Terminological gaps between ontologies are typically overcome by exploiting additional resources. Existing techniques exploit lexical resources in order to identify additional names for ontology concepts [4], thus increasing the chance that corresponding concepts are associated with similar names. These techniques however require the presence of an appropriate lexical resource which is modelled with in such detail that alternative labels can be extracted for all concepts of both ontologies. Thus, it is not always the case that a suitable lexical resource is available. However, it might be that there is a different type of resource available for this scenario, being a partial alignment [5]. A partial alignment is an incomplete mapping between the given ontologies stemming from previous matching efforts. An example of such an effort is a domain expert being unable to complete the mapping due to time constraints. The main problem is how a partial alignment can be exploited to aid the matching between ontologies between which exists a terminological gap.

In this paper, we tackle this problem by proposing a profile-based similarity which exploits the correspondences of the given partial alignment. A typical profile similarity creates a virtual document of each concept by gathering the encoded terminology of related concepts and itself. The core intuition is that two concepts are considered similar if their documents are similar. A key component here is that information of related concepts is exploited as well. Our approach is based on enriching the given ontologies by extracting the encoded semantic relations of each correspondence of a partial alignment, also known as *anchors*. We define an extension of a given profile similarity which utilizes the added relations in order to identify additional terminology for each concept. We evaluate our approach on a dataset consisting of matching problems with distinct terminological gaps. Further, we compare our approach to the performance of existing systems utilizing lexical resources. Lastly, we provide a performance indication for systems utilizing both lexical resources and partial alignments.

The remainder of this paper is structured as follows. We discuss relevant work in Section II. We introduce profile similarities and their ability to deal with terminological heterogeneous ontologies in Section III. The proposed approach is detailed in Section IV. The experimental results are presented in Section V. We present the conclusions and future research topics in Section VI.

II. Related Work

Profile similarities have seen a rise in use since their inception. Initially developed for the Falcon-AO system [6], this type of similarity has seen use in ontology mapping systems, such as AML [7] and RiMoM [8]. These systems typically apply the same scope when gathering information for a concept profile, being the parent concepts and children concepts. Some systems, such as YAM++ [9], limit the scope to the information of the concept annotations and labels.

ISBN: 978-1-61208-420-6

There exist some works that aim to extend the scope of exploited profile information in order to improve the effectiveness of the similarity. The deployed profile similarity in the mapping system PRIOR [10] extends the scope of exploited information to the grand-parent concepts and grand-children concepts, providing a larger scope of exploitable context.

The combination of profile similarities with a set of provided anchors has been tackled in [11]. However, [11] has some fundamental differences compared to this paper. In [11], ontology concepts are compared to the given anchors using a selection of similarity metrics, e.g., a string, instance, and lexical metric. Instead of extracting terminology, the concept profiles are created using the results of these similarity calculations. Concepts are matched if they exhibit comparable degrees of similarities towards the provides anchors. Therefore, this technique can only match terminological ontologies if appropriate similarity metrics are applied and both ontologies contain the exploited meta-data for these similarities.

A. *Semantic Enrichment*

One way in which additional information can be exploited is through *semantic enrichment*. *Semantic enrichment* describes any process which takes any ontology O as input and produces as output the enhanced ontology $E(O)$, such that $E(O)$ expresses more semantic information than O. Typically, a semantic enrichment process exploits resources such as stop word lists or lexical resources for this purpose. Semantic enrichment has been applied in ontology mapping systems in a non-profile context. Examples are the addition of synonyms to the concept descriptions by exploiting lexical resources. LogMap [4] is capable of adding information from WordNet or UMLS to the ontologies prior to mapping. YAM++ [9] uses a machine translator to generate English translations of labels prior to mapping. Multilingual ontology mapping has been specifically addressed in [12]. Ontologies are enriched with multilingual labels using a machine translator. A feature vector for each match candidate is constructed using a combination of similarities and aggregation techniques. Match candidates are then classified using a support vector machine.

A noteworthy application of semantic enrichment for a profile similarity is the work by Su et al. [13]. Here, the semantic enrichment process exploits a set of documents. Using a linguistic classifier and optional user input the documents are assigned to the ontology concepts, such that each assignment asserts that the ontology concept is discussed in its associated document. The concept profiles are then created by gathering terminological information from the assigned documents.

III. Profile Similarities and Terminological Gaps

Profile similarities are a robust and effective type of similarity metric and deployed in a range of state-of-the-art ontology matching systems [6][7][8]. They rely on techniques pioneered in the field of information retrieval [14], where the core problem is the retrieval of relevant documents when given an example document or query. Thus, the stored documents need to be compared to the example document or query in order to determine which stored document is the most relevant to the user. A profile similarity adapts the document comparison techniques by constructing a virtual document for each ontology concept, also referred to as the *profile* of that concept, and determines the similarity between two concepts x and y by comparing their respective profiles. The core intuition of this approach is that x and y can be considered similar if their corresponding profiles can also be considered similar.

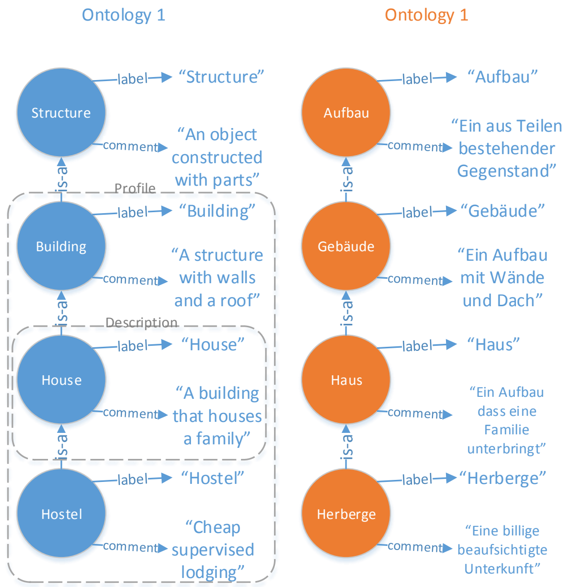

Ontology 1
Ontology 1
Structure
label
"Structure"
comment
"An object constructed with parts"
is-a
Profile
Building
label
"Building"
comment
"A structure with walls and a roof"
is-a
Description
House
label
"House"
comment
"A building that houses a family"
is-a
Hostel
label
"Hostel"
comment
"Cheap supervised lodging"
Aufbau
label
"Aufbau"
comment
"Ein aus Teilen bestehender Gegenstand"
is-a
Gebäude
label
"Gebäude"
comment
"Ein Aufbau mit Wände und Dach"
is-a
Haus
label
"Haus"
comment
"Ein Aufbau dass eine Familie unterbringt"
is-a
Herberge
label
"Herberge"
comment
"Eine billige beaufsichtigte Unterkunft"

Figure 1. Illustration of a terminological gap between two ontologies modelling identical concepts.

As their origin implies, profile similarities are language-based techniques [15]. Language-based techniques interpret their input as an occurrence of some natural language and use appropriate techniques to determine their overlap based on this interpretation. A language-based technique might for instance perform an analysis on the labels of the concept in order to determine their overlap. For instance, given the two concepts *Plane* and *Airplane* a language-based analysis of their labels would result in a high score since the label *Plane* is completely contained within the label *Airplane*. Thus, despite the labels being different, a high similarity score would still be achieved. However, the degree of surmountable label-difference has a limit for language-based techniques. The labels of the concepts *Car* and *Automobile* have very little in common with regard to shared characters, tokens or length. Thus, many language-based techniques are unlikely to result in a high value.

Profile similarities have the advantage that they draw from wide range of information per concept. Thus terminological differences between the labels of two concepts can still be overcome by comparing additional information. This additional information typically includes the comments and annotations of the given concept and the information of semantically related concepts [6][10].

In order for two profiles to be similar, they must contain some shared terminology. For example, the concepts *House* and *Home* can still be matched if their parents contain the word *Building* or if a concept related *Home* contains the word *"House"*. In order for profile similarities to be effective, it is still required that the two given ontologies O_1 and O_2 exhibit some overlap with regard to their terminologies. However, this is not always the case as two ontologies can model the same domain using a completely different terminology. This can be the result of one ontology using synonyms, different naming conventions or the usage of acronyms. Furthermore, two ontologies might even be modelled in a different natural language. For example, one might need to match two biomedical ontologies where one is modelled in English and one in Latin. Thus, it is a real possibility that even if there is some overlap, there can exist corresponding parts of two ontologies exhibit little to no terminological overlap. The terminological gap between two ontologies is illustrated in Figure 1.

Figure 1 displays an example *Ontology 1* next to a series of concepts from a different ontology, *Ontology 2*, modelling the same entities. The terminological gap is illustrated through the fact that all information in *Ontology 2* is modelled in German instead of English. As we can see, comparing the concept *House* with its equivalent concept *Haus* using a typical profile similarity is unlikely to produce a satisfying result since the neither they nor their related concepts contain any overlapping terminology. Therefore, additional measures are necessary in order to ensure the effectiveness of profile similarities when the given ontologies have little to no shared terminology.

IV. Anchor-Based Profile Enrichment

A typical profile similarity is inadequate for ontology matching problems with significant terminological gaps. One way of tackling this issue is through semantic enrichment by exploiting lexical resources such as WordNet [16] or UMLS [17]. Techniques which fall under this category work by looking up each concept in the given resource and adding synonyms, additional descriptions or translations to the concept definition. However, these techniques rely on several assumptions: (1) the availability of an appropriate resource for the given matching problem, (2) the ability to locate appropriate lexical entries given the naming formats of the ontologies, and (3) the ability to disambiguate concept meanings such that no incorrect labels or comments are added to the concept definition. We can see that the performance of such techniques is severely impacted if any of these assumptions fail. If (1) and (2) fail then it is not possible to add additional information to the concept definition, thus causing the ontology concepts to be compared using only their standard profiles. To ensure the ability of identifying correct lexical entries when dealing with ambiguous concepts, one needs to apply a disambiguation technique. State-of-the-art disambiguation systems can achieve an accuracy of roughly 86% [18], meaning that even if a state-of-the-art system is applied there is still a significant proportion of concepts which would be associated with unrepresentative information based on incorrectly designated lexical entries.

If an appropriate lexical resource is not available, other measures are necessary to overcome the terminological gap. These typically are the exploitation of other ontological features, for example the ontology structure. However, it may be the case that instead of a lexical resource a different kind of resource is available to be exploited. For a given mapping problem it is possible that an incomplete alignment, also refereed to as *partial alignment*, is available as additional input. A partial alignment can stem from efforts such as a domain expert attempting to create an alignment, but being unable to complete it due to given circumstances, or from a high-precision system generating such an alignment. The correspondences of the given partial alignment can then be exploited in order to determine the unidentified correspondences.

Our approach aims at adapting profile similarities to be appropriate for matching problems with significant terminological gaps through the exploitation of partial alignments. It is based on the insight that an ontology will consistently use its own terminology. For instance, if an ontology uses the term *Paper* to refer to scientific articles, it is unlikely to use the equivalent term *Article* in the descriptions of other concepts instead, especially if the ontology is designed using a design principle that enforces this property [19]. However, if a partial alignment contains the correspondence *Paper-Article*, then one can use this insight to ones advantage. For instance, given the concept *Accept_Paper* a profile similarity is more likely to match it to its appropriate counterpart *Approve_Article* if the profile of *Accept_Paper* contains the term *'Article'*.

A partial alignment PA is a set of correspondences, with each correspondence asserting a semantic relation between two concepts of different ontologies. The types of relations modelled in a partial alignment, e.g., $\sqsupseteq$, $\perp$, $\sqcap$ and $\equiv$, are typically also modelled in an ontology and thus exploited in the construction of a profile. Thus, by semantically annotating the given ontologies O_1 and O_2 with the correspondences of PA it becomes possible to exploit these newly asserted relations for the creation of the concept profiles. This enables us to construct the profiles of O_1 using a subset of the terminology of O_2, increasing the probability of a terminological overlap between the profiles of two corresponding concepts. This idea is illustrated in Figure 2.

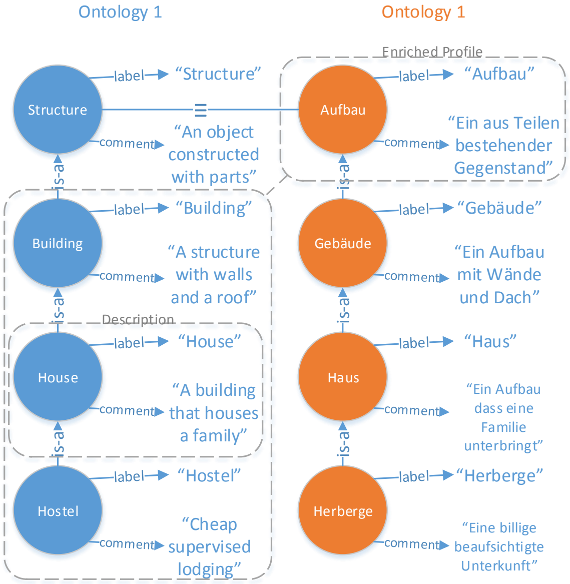

Ontology 1
Ontology 1
Enriched Profile
Structure
label
"Structure"
≡
comment
"An object constructed with parts"
Aufbau
label
"Aufbau"
comment
"Ein aus Teilen bestehender Gegenstand"
is-a
Building
label
"Building"
comment
"A structure with walls and a roof"
Gebäude
label
"Gebäude"
comment
"Ein Aufbau mit Wände und Dach"
Description
House
label
"House"
comment
"A building that houses a family"
Haus
label
"Haus"
comment
"Ein Aufbau dass eine Familie unterbringt"
Hostel
label
"Hostel"
comment
"Cheap supervised lodging"
Herberge
label
"Herberge"
comment
"Eine billige beaufsichtigte Unterkunft"

Figure 2. Two equivalent concepts being compared to a series of anchors.

Before we introduce our approach, we need to define a series terms and symbols that will be used in the following sections:

Correspondence A 5-tuple $< id, e_1, e_2, t, c >$ asserting the semantic relation t between entity $e_1 \in O_1$ and $e_2 \in O_2$ with a confidence of $c \in [0, 1]$.

Mapping/Alignment A set of correspondences, each asserting a relation between $e_1 \in O_1$ and $e_2 \in O_2$.

Partial Alignment A subset of an ideal alignment between ontologies O_1 and O_2.

Anchor A correspondence belonging to a partial alignment.

Collection of words: A list of unique words where each word has a corresponding weight in the form of a rational number.

+: Operator denoting the merging of two collections of words.

×: Operator denoting element-wise multiplication of term frequencies with a weight.

depth(x): The taxonomy depth of concept x within its ontology.

D: The maximum taxonomical depth of a given ontology.

Next, it is necessary to provide a definition of a basic profile similarity upon which we can base our approach. For this, we provide a definition similar to the work by Mao et al. [10]. Neighbouring concepts are explored using a set of semantic relations, such as *isChildOf* or *isParentOf*. A base function of a profile similarity is the description of a concept, which gathers the literal information encoded for that concept. Let x be a concept of an ontology, the description $Des(x)$ of x is a collection of words defined as follows:

$$\begin{array}{rcl} Des(x) & = & \text{collection of words in the name of } x \\ & & +\text{collection of words in the labels of } x \\ & & +\text{collection of words in the comments of } x \\ & & +\text{collection of words in the annotations of } x \end{array} \tag{1}$$

We define the profile of x as the merger of the description of x and the descriptions of semantically related concepts:

$$\begin{array}{rl} \textit{Profile}(x) = & Des(x) + \sum_{p \in P(x)} Des(p) + \\ & \sum_{c \in C(x)} Des(c) + \sum_{r \in R(x)} Des(r) \end{array} \tag{2}$$

where

$$\begin{array}{l} P(x) = \{p | x \textit{ isChildOf } p\} \\ C(x) = \{c | c \textit{ isChildOf } x\} \\ R(x) = \{r | r \textit{ isRelatedTo } x \wedge r \notin P(x) \cup C(x)\} \end{array}$$

In order to compute the similarity between two profiles, they are parsed into a vector-space model and compared using the cosine similarity [20]. To bridge the terminological gap we aim to exploit the semantic relations provided by a given partial alignment PA, such that we can enhance the profile of a concept $x \in O_1$ using the terminology of O_2. We refer to this enlarged profile as the *anchor-enriched-profile*. For this, we explore the parents, children and properties of a concept x (or ranges and domains in case x itself is a property). If during this exploration a concept y is encountered which is mapped in a correspondence in PA to a concept $e \in O_2$, then $\textit{Profile}(x)$ is merged with $Des(e)$.

We will define the set that describes the extended collection of *parentally-anchored-descriptions* (PAD) with regard to concept x in three variations. These gather the descriptions of anchored concepts from the ancestors of x. To measure the improvement caused by the addition of these sets, we also define the omission of any such descriptions. They are defined as follows:

$$\begin{array}{l} \text{PAD}_0(x, PA) = \emptyset \\ \text{PAD}_1(x, PA) = \sum_{e \in E} Des(e); \textit{ where} \\ \quad E = \{e | \exists < id, y, e, t, c > \in PA; y \textit{ isAncestorOf } x\} \\ \text{PAD}_2(x, PA) = \sum_{e \in E} \omega \times Des(e); \textit{ where} \\ \quad E = \{e | \exists < id, y, e, t, c > \in PA; y \textit{ isAncestorOf } x\} \\ \quad \wedge \omega = \frac{D - |depth(x) - depth(y)|}{D} \end{array} \tag{3}$$

An interesting point to note is that PAD$_2$ utilizes the same set of concepts than PAD$_1$, but weighs their descriptions with respect to the concept's relative distance to x, such that the descriptions of closer concepts receive a higher weight.

Exploring the children of x, we define the merged collection of *child-anchored-descriptions* (CAD) in a similar way:

$$\begin{array}{l} \text{CAD}_0(x, PA) = \emptyset \\ \text{CAD}_1(x, PA) = \sum_{e \in E} Des(e); \textit{ where} \\ \quad E = \{e | \exists < id, y, e, t, c > \in PA; y \textit{ isDescendantOf } x\} \\ \text{CAD}_2(x, PA) = \sum_{e \in E} \omega \times Des(e); \textit{ where} \\ \quad E = \{e | \exists < id, y, e, t, c > \in PA; y \textit{ isDescendantOf } x\} \\ \quad \wedge \omega = \frac{D - |depth(x) - depth(y)|}{D} \end{array} \tag{4}$$

Lastly, we can explore the relations defined by the properties of the ontology, being *isDomainOf* and *isRangeOf*. Defining O_c as the set of concepts defined in ontology O and O_p as the set of properties of O, we define the merged collection of *relation-anchored-descriptions* (RAD) in two variations as follows:

$$\begin{array}{l} \text{RAD}_0(x, PA) = \emptyset \\ \text{RAD}_1(x, PA) = \\ \quad \begin{cases} \sum_{e \in E} Des(e); \textit{ where} \\ \quad E = \{e | \exists < id, y, e, t, c > \in PA; x \textit{ isDomainOf } y\} \\ \quad \textit{if } x \in O_c \\ \sum_{e \in E} Des(e); \textit{ where} \\ \quad E = \{e | \exists < id, y, e, t, c > \in PA; y \textit{ isDomainOf } x \vee \\ \quad y \textit{ isRangeOf } x\} \textit{ if } x \in O_p \end{cases} \\ \text{RAD}_2(x, PA) = \\ \quad \begin{cases} \sum_{e \in E \cup F} Des(e); \textit{ where} \\ \quad E = \{e | \exists < id, y, e, t, c > \in PA; x \textit{ isDomainOf } y\} \\ \quad \textit{and } F = \{f | \exists < id, y, f, t, c > \in PA \; \exists z \in O_p; \\ \quad x \textit{ isDomainOf } z \wedge y \textit{ isRangeOf } z\} \textit{ if } x \in O_c \\ \sum_{e \in E} Des(e); \textit{ where} \\ \quad E = \{e | \exists < id, y, e, t, c > \in PA; y \textit{ isDomainOf } x \vee \\ \quad y \textit{ isRangeOf } x\} \textit{ if } x \in O_p \end{cases} \end{array} \tag{5}$$

The noteworthy difference between RAD$_1$ and RAD$_2$ is that if x is a concept and the domain of property z, then every range y of z will be explored as well. As an example, assume we are given the concepts *Car* and *Driver* being linked by the property *ownedBy*. Constructing the anchor-enriched-profile of *Car* using the set RAD$_1$ would mean that we only investigate if *ownedBy* is mapped in PA. Using RAD$_2$ means we also investigate *Driver*, which could provide additional context.

Given a partial alignment PA between ontologies O_1 and O_2, and given a concept x, we define the *anchor-enriched-profile* of x as follows:

$$\begin{array}{l} \textit{Profile}^{AE}_{\kappa,\lambda,\mu}(x, PA) = \textit{Profile}(x) + \text{PAD}_\kappa(x, PA) + \\ \text{CAD}_\lambda(x, PA) + \text{RAD}_\mu(x, PA) \end{array} \tag{6}$$

V. Experiments

In this section, we will detail the performed experiments to test the effectiveness of our approach and discuss the obtained results. A widely used way of evaluating a mapping A is by comparing it to a reference alignment R by calculating the standard measures of Precision, Recall and F-Measure [21]. When matching with a partial alignment PA the newly computed correspondences are typically merged with PA in order to create a complete mapping. However, this action creates a bias with respect to the measured alignment quality.

For instance, if PA contains half the correspondences of R, then the resulting Recall score cannot go below 0.5. It would hence be desirable to use a measure which only focuses on the correspondences that are contributed to PA to get a better indication of their quality. To achieve this, we will use adapted variants of Precision, Recall and F-Measure, which take the presence of a partial alignment into account. Given a computed alignment A, a reference alignment R and a partial alignment PA, the adapted measures of Precision and Recall are computed as follows:

$$P^*(A, R, PA) = \frac{\mid A \cap R \cap \overline{PA} \mid}{\mid A \cap \overline{PA} \mid} \tag{7}$$

$$R^*(A, R, PA) = \frac{\mid A \cap R \cap \overline{PA} \mid}{\mid R \cap \overline{PA} \mid} \tag{8}$$

The adapted F-Measure can then be computed as follows:

$$F^*(A, R, PA) = \frac{2 * P^*(A, R, PA) * R^*(A, R, PA)}{P^*(A, R, PA) + R^*(A, R, PA)} \tag{9}$$

A. *Multi-Farm*

In this section we will present the results of our evaluation on the *Multi-Farm-sameOnto* dataset. This data-set stems from the OAEI 2014 [21] competition. The terminologies of the ontologies in this dataset vary greatly since it is designed to be a cross-lingual dataset. The set consists of 8 ontologies that are modelled using 9 languages (including English). For each pair of ontologies a set of mapping tasks exists consisting of every possible combination of selecting different languages. We generate the partial alignments by randomly sampling the reference alignment with the condition that $R(PA, R) = 0.5$ and aggregate the results of 100 evaluations for each task. This evaluation is repeated for every possible combination of κ, λ and μ. The result of this evaluation is presented in Table I.

Table I. AGGREGATED ADAPTED PRECISION, RECALL AND F-MEASURE FOR ALL VARIATIONS ON THE MULTI-FARM DATASET.

κ	λ	μ	P^*	R^*	F^*
0	0	0	0.418	0.278	0.326
0	0	1	0.657	0.433	0.510
0	0	2	0.630	0.405	0.481
0	1	0	0.500	0.324	0.381
0	1	1	0.675	0.469	0.543
0	1	2	0.666	0,453	0.529
0	2	0	0.512	0.333	0.393
0	2	1	0.688	0.475	0.552
0	2	2	0.678	0.457	0.535
1	0	0	0.521	0.376	0.423
1	0	1	0.667	0.529	0.583
1	0	2	0.659	0.518	0.574
1	1	0	0.594	0.409	0.470
1	1	1	0.691	0.559	0.611
1	1	2	0.688	0.555	0.609
1	2	0	0.601	0.417	0.478
1	2	1	**0.699**	0.565	0.619
1	2	2	0.695	0.562	0.615
2	0	0	0.523	0.385	0.433
2	0	1	0.674	0.538	0.592
2	0	2	0.661	0.522	0.577
2	1	0	0.591	0.411	0.471
2	1	1	0.690	0.562	0.614
2	1	2	0.685	0.554	0.607
2	2	0	0.597	0.421	0.481
2	2	1	0.698	**0.570**	**0.622**
2	2	2	0.692	0.562	0.614

First, by comparing the performance of the baseline configuration $Profile^{AE}_{0,0,0}$ to any configuration of our approach we can easily see that our approach improves upon the performance of the baseline. Adding the sets PAD or CAD using either variation typically resulted in an F-Measure of 0.39-0.43, an improvement of 0.07 to 0.11 when compared to the baseline. Curiously, enriching the profiles using RAD alone typically resulted in a F^* score of approximately 0.5. This could indicate that for this dataset the concept annotations more often contain terms of related concepts than ancestors or descendants.

Looking at dual-combinations between PAD, CAD and RAD we can see a consistent increase in performance. Of these combinations, $Profile^{AE}_{1,1,0}$ resulted in the lowest F-Measure of 0.47, while $Profile^{AE}_{1,0,1}$ resulted in the highest F-Measure of 0.583. We can also observe that combinations which include a variation of the RAD-set in the enriched profiles typically performed better than combinations that didn't.

Lastly, we can observe using all three types of description sets resulted in the highest measured F^* score. We can see that every combination of PAD, CAD and RAD resulted in an F^* score higher than 0.6. The best performing combination was $Profile^{AE}_{2,2,1}$ with an F^* score of 0.622.

Comparing RAD_1 with RAD_2 reveals that combinations which utilized RAD_1 performed slightly better than combinations which used RAD_2 instead. This implies that concepts which are related through properties are less likely to share terms, leading to many impact-less terms being added to the concept profiles.

B. *Comparison with Lexical Enrichment Systems*

The main goal behind this work is to provide an approach that allows the enrichment of concept profile by exploiting the relations of a provided partial alignment. The reason behind this is that current enrichment methods exploit primarily lexical resources, which rely on the presence of an appropriate resource. In the previous sections, we have established the performance of our approach using different configurations, datasets, and partial alignment sizes. In this section, we will provide some interesting context for these results. Specifically, we aim to compare the results of our approach with the performances of matching systems tackling the same dataset while exploiting lexical resources. This allows us to establish whether an approach exploiting a partial alignment can produce alignments of similar quality as approaches exploiting lexical resources. To do this, we will compare the performance of our approach on the Multi-Farm dataset [21] to the performances of the OAEI participants which competed in the 2014 evaluation. Here we will make the distinction between approaches utilizing no external resources, lexical resources and partial alignments. This allows us to see the benefit of exploiting a given type of external resource.

Furthermore, to provide an upper boundary for the potential performance on this dataset, we will also evaluate a method utilizing both lexical resources and partial alignments. To achieve this, we will re-evaluate the best performing configuration from sub-section V-A. However, the profiles of this re-evaluation will be additionally enriched by translating the concept labels using the *Microsoft Bing* translator. This will provide an indication of how well a system may perform when utilizing both appropriate lexical resources and partial alignments. The comparison can be seen in Table II. Performances of approaches utilizing partial alignments are denoted in adapted precision, recall and F-Measure.

Table II. PERFORMANCE COMPARISON BETWEEN OUR APPROACH AND THE OAEI 2014 COMPETITORS (MULTI-FARM DATASET).

Lex.	P. Align.	Matcher	Precision	Recall	F-Measure
yes	yes	$Profile^{AE}_{2,2,1}$ + Bing	*0.849*	*0.838*	*0.843*
yes	no	AML	0.95	0.48	0.62
yes	no	LogMap	0.94	0.27	0.41
yes	no	XMap	0.76	0.40	0.50
no	yes	$Profile^{AE}_{2,2,1}$	*0.698*	*0.570*	*0.622*
no	no	AOT	0.11	0.12	0.12
no	no	AOTL	0.27	0.01	0.02
no	no	LogMap-C	0.31	0.01	0.02
no	no	LogMapLt	0.25	0.01	0.02
no	no	MaasMatch	0.52	0.06	0.10
no	no	RSDLWB	0.34	0.01	0.02

From Table II, we can make several observations. First, we can observe that every system utilizing either lexical resources or partial alignments performs significantly better than systems which do not. This is an expected result given the nature of this dataset. Of the system which do not exploit resources AOT has the highest performance with an F-Measure of 0.12.

Comparing the performance of $Profile^{AE}_{2,2,1}$ to the performance of system exploiting only lexical resources reveals an interesting observation. Specifically, we can see that the performance of these systems is comparable. While the performances of LogMap and XMap were lower than $Profile^{AE}_{2,2,1}$, with an F-Measure of 0.62 the performance of AML is very close to the performance of $Profile^{AE}_{2,2,1}$. However, AML distinguishes itself from our approach by having a notably higher precision and a somewhat lower recall. In fact, all systems utilizing only lexical resources are characterized with a high precision, which implies that enriching ontologies using these resources only rarely leads to false-positive matches in terminology.

Lastly, we can observe the performance of our approach when paired with a lexical resource, specifically *Bing Translator*. The produced alignments reached an F^* score of 0.843, which is significantly higher than the OAEI participants. This implies that the correct correspondences which lexical-based systems find differ significantly from the correct correspondences of a partial-alignment-based system. From this we can conclude that the two types of resources are complementary for matching problems with significant terminological gaps.

VI. Conclusion

In this paper, we presented a technique aimed at tackling ontology mapping problems with significant terminological heterogeneities between the given ontologies. This technique exploits an existing partial alignment by enriching the given ontologies with the relations asserted in the partial alignment. We establish the performance of the approach on a dataset characterized by terminological heterogeneous mapping problems. A comparison with other matching systems reveals that the approach performs similarly to systems utilizing lexical resources. Combining our approach with a lexical resource reveals that a significantly higher performance can be achieved if both partial alignments and lexical resources are utilized.

References

[1] F. Bellifemine, A. Poggi, and G. Rimassa, "Jade–a fipa-compliant agent framework," in Proceedings of PAAM, vol. 99. London, 1999, p. 33.

[2] Y. Lei, V. Uren, and E. Motta, "Semsearch: A search engine for the semantic web," in Managing Knowledge in a World of Networks. Springer, 2006, pp. 238–245.

[3] J. Euzenat, "Towards a principled approach to semantic interoperability," in Proceedings of the IJCAI-01 Workshop on Ontologies and Information Sharing, 2001, pp. 19–25.

[4] E. Jiménez-Ruiz and B. Cuenca Grau, "Logmap: logic-based and scalable ontology matching," in ISWC 2011, 2011, pp. 273–288.

[5] F. C. Schadd and N. Roos, "Anchor-profiles: Exploiting profiles of anchor similarities for ontology mapping," in Proceedings of the 26th Belgian-Dutch Conference on Artificial Intelligence (BNAIC 2014), 2014, pp. 177–178.

[6] Y. Qu, W. Hu, and G. Cheng, "Constructing virtual documents for ontology matching," in Proceedings of the 15th international conference on World Wide Web, ser. WWW '06. ACM, 2006, pp. 23–31.

[7] I. Cruz, F. Antonelli, and C. Stroe, "Agreementmaker: efficient matching for large real-world schemas and ontologies," Proc. VLDB Endow., vol. 2, no. 2, Aug. 2009, pp. 1586–1589.

[8] J. Li, J. Tang, Y. Li, and Q. Luo, "Rimom: A dynamic multistrategy ontology alignment framework," IEEE Transactions on Knowledge and Data Engineering, vol. 21, no. 8, 2009, pp. 1218–1232.

[9] D. Ngo, Z. Bellahsene, and R. Coletta, "Yam++-a combination of graph matching and machine learning approach to ontology alignment task," Journ. of Web Sem., 2012.

[10] M. Mao, Y. Peng, and M. Spring, "A profile propagation and information retrieval based ontology mapping approach," in SKG 2007. IEEE, 2007, pp. 164–169.

[11] F. C. Schadd and N. Roos, "Anchor-profiles for ontology mapping with partial alignments," in Proceedings of the 12th Scandinavian AI Conference (SCAI 2013), 2013, pp. 235–244.

[12] D. Spohr, L. Hollink, and P. Cimiano, "A machine learning approach to multilingual and cross-lingual ontology matching," in ISWC 2011. Springer, 2011, pp. 665–680.

[13] X. Su and J. A. Gulla, "Semantic enrichment for ontology mapping," in Proc. of the 9th International Conference on Applications of Natural Languages to Information Systems, NLDB 2004, vol. 3136, 2004, p. 217.

[14] C. D. Manning, P. Raghavan, and H. Schütze, Introduction to information retrieval. Cambridge university press Cambridge, 2008, vol. 1.

[15] J. Euzenat and P. Shvaiko, Ontology Matching. Springer Berlin, 2007, vol. 18.

[16] G. A. Miller, "Wordnet: a lexical database for english," Communications of the ACM, vol. 38, November 1995, pp. 39–41.

[17] O. Bodenreider, "The unified medical language system (umls): integrating biomedical terminology," Nucleic acids research, vol. 32, no. suppl 1, 2004, pp. D267–D270.

[18] R. Navigli, "Word sense disambiguation: A survey," ACM Comput. Surv., vol. 41, no. 2, Feb. 2009, pp. 10:1–10:69.

[19] Y. Sure, S. Staab, and R. Studer, "Methodology for development and employment of ontology based knowledge management applications," ACM SIGMOD Record, vol. 31, no. 4, 2002, pp. 18–23.

[20] P.-N. Tan, M. Steinbach, and V. Kumar, Introduction to Data Mining, 1st ed. Addison Wesley, May 2005.

[21] Z. Dragisic, K. Eckert, J. Euzenat, D. Faria, A. Ferrara, R. Granada, V. Ivanova, E. Jimenez-Ruiz, A. Kempf, P. Lambrix et al., "Results of theontology alignment evaluation initiative 2014," in OM 2014, 2014, pp. 61–104.

From a Consensual Conceptual Level to a Formal Ontological Level

A case study in healthcare organizations

Fabrício Martins Mendonça
Maurício Barcellos Almeida
Universidade Federal de Minas Gerais
Belo Horizonte, Brazil
E-mail: fabriciommendonca@gmail.com
E-mail: mba@eci.ufmg.br

António Lucas Soares
Cristóvão Polido Sousa
INESC TEC
Universidade do Porto
Porto, Portugal
E-mail: als@fe.up.pt
E-mail: cristovao.dinis@gmail.com

***Abstract* — Knowledge representation depend on experts, even though such professionals do not have skills to provide the formalized knowledge needed to build formal ontologies. In this paper, we present a case study in which we investigate aspects and challenges in formalizing medical knowledge in a healthcare organization. Our experiment used two different instruments to conceptualize and formalize knowledge: i) for conceptualizing knowledge consensual, we used a collaborative framework called ConceptMe; ii) for analysing and formalizing of the knowledge collaboratively conceptualized, we used principles of the Basic Formal Ontology. Even though the process of formalizing knowledge is not a novelty, we try to explore how this task has been done in the scenario of Semantic Web and ontological engineering. We concluded that there is a strong and a sound complementarity between the two aforementioned frameworks, since the first provides a well-done approach for collaborative conceptual modelling and the second provides a way of establishing rules for carrying semi-formal knowledge to a formal level in ontologies. As main contributions, we emphasize the description of how to use the collaborative environment and the organization of a set of rules, as well as their application in real situations.**

Keywords - collaborative conceptualization; formal ontology.

I. INTRODUCTION

People create models using their cognitive skills in a process of meaning construction, in general, called *conceptualization* [1]. Conceptualization is generally conducted by knowledge engineers along with experts (doctors, engineers, lawyers, to name a few). In the recent field of Semantic Web and ontological engineering, conceptualization is considered a cornerstone [2]. However, different specialists have different views of the world, which may result in different concurrent conceptualizations, all of them correct [3]. Thus, the conceptualizing process should be collaborative, and carried out in an environment that allows for consensual definitions [2], with the aim of reaching a reasonable representation of the needs of users.

A challenge for the general methodologies for building ontologies – such as *Methontology* [4] or *NeON* [5] – is to find the best way to perform the transition of knowledge from a *conceptual level* (informal and semi-formal levels) to a level in which constraints are used to reduce ambiguity in the meaning of terms (formal ontological level). The conceptual level is, in general, comprised of unstructured knowledge, obtained from knowledge acquisition from experts. While this conceptual level is essential to building a shared view of world, one must add constraints if the goal is to build formal ontologies.

In this paper, we present an ongoing research that explores how the transition from the conceptual level to the formal level occurs in a medical organization. We conducted a case study with the aim of verifying flaws and proposing improvements to the transition. Studies on knowledge formalization are not a novelty, for example, in artificial intelligence [6]. However, the new context of ontologies in Semantic Web and the increasing participation of experts in modelling activities (for example, in standards like OpenEHR [7]) suggest the need for new research.

In our experiment, we adopted approaches that deal with both the conceptual and formal ontological levels. The first approach is *Conceptualization Modelling Environment* (ConceptMe), an environment that includes a set of principles, resources and tools that allows collaborative development of a shared, consensual, semi-informal conceptual representation [8]. For the second approach, we follow principles, methods, criteria and ontological restrictions proposed by Munn and Smith [9], which represents the foundations of *Basic Formal Ontology* (BFO) [10].

In order to accomplish the first stage of our experiment, we applied the underlying methods of ConceptMe in the ontology for blood transfusion called HEMONTO [11], in order to check the existing relations between candidate terms to the ontology, and then to come up with a shared conceptual model. In the second stage, the conceptual relations defined by ConceptMe were evaluated through the application of a set of ontological restrictions. So, we were able to investigate the transition, problems, flaws and improvements in the formalization process. Ultimately, results obtained indicate that the underlying method of ConceptMe is very efficient to work with knowledge in conceptual level and very useful for dealing with experts. However, the sort of rules embedded in ConceptMe do not permit direct construction of formal ontologies. Indeed, some conceptual relations could not be transformed into ontological relations for several reasons. For example, many relations did not include distinctions between specific types, mainly *part-of* and *is-a* relations, on which we focused on our investigation. On the other hand, the tests with the framework ConceptMe associated with the ontological level enabled us to reach new alternatives to be considered in the formalization processes for ontologies. Our findings indicate the need of complementarity between the approaches in order to deal with both experts and knowledge engineers.

The remainder of this article is organized as follows: the second section presents a research background, which highlights both the theory that underlies ConceptMe, as well as the essentials and ontological criteria required at the formal level. The third section describes our research methodology in applying ontological criteria to formalize knowledge about human blood, in the context of the blood bank. The fourth section presents results obtained with the application of ontological criteria to the model developed through the ConceptMe, emphasizing the possibility of constraining the meaning of terms at the conceptual level. Finally, the fifth section presents some remarks, our conclusions and suggestions for future works.

II. Background

The background of this research involves two main parts: (i) the approach used in the conceptual viewpoint, which includes the theory that based the ConceptMe and (ii) the characterization of the relations of the ontological viewpoint using principles of the top-level ontology BFO.

A. Conceptual Level: the theory underlying the ConceptMe

In this section, we present the approach used in this research of the conceptual viewpoint, that corresponds to the theory used in the ConceptMe.

The ConceptMe was developed based on the method *ColBlend* [12], that supports to process of collaborative conceptualization in the inter-organizational context and it is based in a theory of the cognitive semantic called as *Conceptual Blending Theory* [13]. In this theory, the conceptual integration is more than the sum of its constituent parts, because it should involve also new structures or emergent structures, namely, new information deriving from the process of negotiation.

The process of negotiation of the meaning in the ConceptMe follows the method ColBlend and involves the following semantic spaces [12]: i) the input spaces - private to each party involved in the conceptualization process where the knowledge models proposals are built; b) the blend space - which contains the proposal resulting from the analysis of the input spaces, which is presented for discussion. Moreover it proposes new concepts (originally not identified) from an overall analysis of the current content of the spaces and; c) the generic space - which contains the common domain knowledge model composed by all parts of the universally accepted proposals that were "published" to this shared space.

The ConceptMe also introduces a multidimensional and structured view of the conceptualization process that encompasses four main phases: (i) elicitation of concepts, (ii) organization of concepts, (iii) sharing of concepts, and (iv) negotiation of concepts. It considers two main types of processes: (i) terminological processes, and (ii) processes of knowledge representation [14]. The conceptual relations are treated in the phase of concept organization and as part processes of knowledge representation, within a module called as *Conceptual Relations Reference Model (CRRM)*, which supports domain specialists in the definition of basic conceptual relations between objects in that domain.

The CRRM assists domain specialists in the phase of elicitation of conceptual relations, which is considered one of the most difficult problems in the conceptualization process [15]. Auger and Barrière [14] realized a review of the literature about conceptual relations in different scientific domains (Artificial Intelligence, Information Science, Linguistics, Formal ontology, Cognitive Semantic) and from this, it was identified a set of basic conceptual relations to be used in the CRRM.

Obviously, in approaching different knowledge areas, the names, definitions and use of the relations mentioned vary widely from one application context to the other. In the case of the framework ConceptMe, the strategy used to approach this diversity of relations was to map them (and summarise them), for a set of most basic relations that could represent the most common types of relationship between the objects of the given domain. The following set of relations for the ConceptMe were defined [14]:

(i) Constitution and containment dependence: part-whole conceptual relation;

(ii) Generic dependence: generic-specific conceptual relation;

(iii) Historical dependence: it was separated into spatial conceptual relation and temporal conceptual relation.

(iv) Participation conceptual relation;

(v) Causal conceptual relation or cause-effect conceptual relation.

For each relation, the ConceptMe contains a specific template of the relation and also a set of competency questions that are show to the user of the framework (a specialist, for example) for your choice of the relation more appropriate within analysed context. The competency questions for parthood relation, for example, are [14]: Are A and B physically engaged? Is B a component/constituent or attached to A? Are A and B nested?

As previously mentioned, the theory on which ConceptMe is based is the approach adopted in this research for conceptual viewpoints, however, it is not sufficient for the building of formal ontologies. To address the formal part, we adopted criteria and ontological restrictions of methods regarding development of ontologies. This topic is dealt with in the next section of this background.

B. Ontological level: principles and restrictions for the characterization of ontological relations

For dealing the knowledge of a given domain at formal level, it is necessary to better characterize such relations. We did it, mainly, following the approach presented in

[9][16][17]. Like this, the next paragraphs present a characterization of ontological relations.

The first characterization discern the types of ontological relations from its related entities (relatas), evaluating if they are *universals* or *particulars*: *universals* or *types* are the kinds of things that exist in real world, that is, recurrent entities sharing some characteristics that could be instantiated or exemplified by more than one particular thing; *particulars* refers to a specific object in the real world. Particulars are also called *instances* or *tokens* or *individuals* [10]. Considering the types of relation with *universals* and *instances*, we have: (i) <universal, universal>: for example, *subsumption*: "whole portion of blood is_a portion of body substance"; (ii) <instance, universal>: for example, *instantiation*: "John's blood instance_of whole portion of blood"; (iii) <instance, instance>: for example, *participation*: "John's blood participates_in John's blood transfusion").

A second important aspect is to evaluate if the relation can be considered ontological from four essential criteria: (i) the relations must be genuine ontological relations, in other words, they must be obtained from entities in reality, independently of our experience or methods of learning about them; (ii) the relations must be domain-neutral relations or domain independent; (iii) the relations must be obtained universally: a statement of the form A relation B must obtain for all instances of A, and not just (for example) for some statistically representative selection; (iv) The relation must be definable in a simple, yet rigorous way. This means that intuitive definitions (for example, *functionally_related_to* or *physically_related_to* of the UMLS) should not be made and also some definition is required.

A third characterization refers to the distinction between entities *continuants* and *occurrents*: *continuants or endurants* are entities that continue to exist through time maintaining their identity and do not have temporal parts; *occurrents or perdurants* are entities that occur in time and they unfold themselves through a period of time in such a way that they can be divided into temporal parts or phases [10]. For each relation, it is required to define the domain and range of the relation and to define if the domain and the range must contain a *continuant* or an *occurrent*. For example, the relation *participates_in* always must involve a continuant in the domain and an occurrent in the range.

The last and fourth characterization corresponds to basic logical properties of each relation, analyzing the related entities (relata). The basic ontological properties are known as meta-properties in the literature of the area [18][19]: *(i) reflexivity; (ii) transitivity; (iii) symmetry*. The relation *part-of*, for example, can be characterized as a relation *irreflexive, anti-symmetric* and *transitive*.

These four characterizations presented are general and applicable to all ontological relations; however, for each specific relation, there are set of specific restrictions that must be considered. In this research, we focus our work around of two relations: *is_a* and *part_of*, because they are the most used in the development of ontologies and their use occurs, oftentimes, without concern with their real meaning. Thus, in the next paragraphs, we explain the different types of relations *is_a* and *part_of*.

For the relation *is_a*, used in the building of taxonomies of the ontologies, it is necessary to consider the following types of relations[4][9]: *Instantiation: is_a* used as synonym of *instance_of*; *Specification or specialization: is_a* used as synonym of *subclass_of*; *Synonymy: is_a* used as synonym of *same_as*.

Regarding relation *part_of*, we used some types of this relation from a taxonomy presented in [16], which encompasses a set of types of the mereological and meronymic viewpoints, studied and addressed in [19][20][21].

The taxonomy proposed by Keet and Artale [16] is fairly complete with respect to existing types of relations *part-of*, however, it is necessary to adapt such an approach to treat cases of the relation *part-of* that involve temporality and spatial localization simultaneously, in addition to relations between non-material and material entities, which are, extremely, important in biomedical domains. For dealing with these types of relations *part-of*, we follow the approach proposed by Schulz et al. [17], that include, for example, relations as: (i) *Temporary-Part-Of (A, B, t)*: Amputated toe Temporary-Part-Of Body Human; (ii) *Permanent-Part-Of (A, B, t):* My brain Permanent-Part-Of my head; among others.

III. Methodology

This section describes the methodological steps adopted to make it possible the conceptual-formal transition of the knowledge about the domain addressed. This domain corresponds to the process of blood transfusion, which encompasses the components extracted from human blood for certain therapeutic recommendations and also the processes used to obtain blood components for the transfusion.

To better describe the methodological steps performed, we divide them in four distinct phases: the phases 1 and 2 address the knowledge at the conceptual level, using the framework ConceptMe; the phase 3 corresponds to the conceptual-formal ontological transition and where we apply the criteria and ontological restrictions of the approach adopted; and the phase 4 presents the results of this transition in the domain addressed.

The ontology about blood transfusion (HEMONTO [11]) has been developed within the scope of the Blood Project, using the software Protégé 4.2 [22]. For the purposes of this paper, parts of the HEMONTO were reconstructed in the framework ConceptMe, using its interface of conceptual graphs. The objective here is to test the theory underlying the ConceptMe, using the knowledge of the blood transfusion domain.

Using this approach, the phase 1 was developed encompassing the conceptualization of the domain addressed from two distinct semantic spaces, each one with a specific perception of the domain: (i) "specialist space":

doctors of the Hemominas Foundation developed a conceptual model of the domain based on their specialist technical knowledge; (ii) "ontologist space": the own authors of this paper, with expertise in ontological engineering and having studied the blood domain in the recent years [11], developed a conceptual model of the domain based on the extraction of information from corpus, using the following documents: (a) the guidebook about blood components of the Brazilian Health Minister [23]: (b) the international standard "*ISBT 128: Standard Terminology for Blood, Cellular Therapy, and Tissue Product Descriptions"* [24]; (c) the "Technical *Manual about Blood and Cellular Therapy"* from the international organization *AABB* [25]; and (d) the textbook about clinical hematology: "*Wintrobe's Clinical Hematology* 12th edition [26].

Phase 2 involves negotiation of the semantic meaning of the concepts defined in the prior phase. This process is realized semi-automatically by ConceptMe based on the theory *Conceptual Blending Theory* [12], as explained above; as a result of the negotiation it produces a common concept model accepted by the groups involved in the conceptualization process, called in the theory of model of the generic space.

Phase 3 corresponds to formalization process, where we applied the criteria and ontological restrictions recovered of the literature of the area and used at the conceptual-formal transition (see Table 1). It represents the main contribution of this paper. The strategy adopted here involved the selection of criteria and ontological restrictions researched that it could be applied in the evaluation of the conceptual relations of the type *is_a* and *part_of* of the model developed, in order to allow its transition to the formal level. We created a code and a name for each criterion and its description was made based on the literature review, as explained below:

- The letter "O" is used to denote a basic ontological criterion, for example, O1 to ontological nature or O2 to universality.
- The letter "I" indicates a common criterion for the relations *is_a*, such as its ontological properties, for example, I3 to asymmetry.
- The letter "P" indicates a common criterion for the relations *part_of*, for example, P2 to transitivity.
- For the specific types (subtypes) of the relations *is_a* and *part_of* were used abbreviations of the names of these subtypes, such as I.IN for *instance_of* and P.ST for *structural_part_of*.
- The criteria defined must be used for relations under the forms *A relation B* and *B relation C* for universals, and *a relation b* and *b relation c* for particulars.

TABLE I. CRITERIA AND ONTOLOGICAL RESTRICTIONS

Code	**Criterion**	**Description**
O1	Ontological nature	The relation must be identified in the reality, independent of human constructions.
O2	Universality	The relation must be obtained universally.
O3	Non-intuitiveness	The name of the relation must not be defined in an intuitive way.
I1	Is-a Reflexivity	The entity A is a type of itself. (A *is_a* A)
I2	Is-a Transitivity	The relation is transitive between three entities of the domain, such as: If A *is_a* B and B *is_a* C then A *is_a* C
I3	Is-a Asymmetry	If entity A is a subtype or instance of other entity B, the inverse is not true: If A *is_a* B then not (B *is_a* A). That propriety must only be applied for the relations *instance_of* and *subclass_of* and not for the relation *same_as*.
I.IN	Type Instance-of	Relation *is_a* when a particular instantiates a universal A. if *a* is a continuant A must also be, if *a* is a occurrent A also must be.
I.SC	Type Subclass-of	Relation *is_a* between two universals (an entity is kind of the other entity. If A is a continuant B must also be, if A is an occurrent B must also be.
I.SY	Type Same-as	Relation *is_a* between two entities that are identical, between particulars, universals, continuants, occurrents; if a, A are continuants b, B must also be; and if a, A are occurrents b, B must also be.
P1	Part-of Reflexivity	The entity A is part of itself. (A *part_of* A)
P2	Part-of Transitivity	Relation transitive: If A *part_of* B and B *part_of* C then A *part_of* C
P3	Part-of Asymmetry	If entity A is a part of B, the inverse is not true: If A part_of B then not (B part_of A).
P.ST	Type structural-part-of	Relation part-of between two continuants in which the part composes the structure of the whole, functionally or structurally.
P.CO	Type contained-in	Relation part-of between two continuants in which the part occupies a region 2D inserted within the region 3D occupied by the whole.
P.LO	Type located-in	Relation part-of between two continuants in which the part occupies a portion of the space occupied by the whole.
P.IN	Type involved-in	Relation part-of between two occurrents in which a part represents a step of the whole.
P.ME	Type member-of	Relation part-of between continuants such that a part is a physical object that composes a whole (a non-physical social object).
P.CN	Type constitutes	Relation part-of between continuants such that the part is an amount of matter that constitutes the whole (a physical object).
P.SQ	Type sub-quantity-of	Relation part-of between continuants that are portions of matter and the part is a lower portion of the whole portion.
P.TP	Type temporary-part-of	Relation part-of between continuants or occurrents, in which the part, only at some instant, is located as part of the whole.
P.FP	Type immaterial-	Relation part-of between two continuants so

ISBN: 978-1-61208-420-6

	part-of	that the part is a immaterial object and it is connected to the whole (a material entity).

Lastly, we have in the phase 4 the results obtained with an application of the criteria and ontological restrictions in treated domain, in our case, the blood transfusion. In this phase, we extracted conceptual statements from the model developed in the ConceptMe that contain the relations *is_a* and *part_of*. ConceptMe represents such statements in the form of a conceptual graph, then, we converted these statements in conceptual graph to text format so that they could be evaluated according to ontological criteria. The statements evaluated positively could be transformed in ontological relations and composed the final ontological model.

IV. Analysis of the Relations and Discussion

In order to test and show the practical applicability of the proposal presented, we lead a case study in a healthcare organization that works with the blood transfusion domain named of the Hemominas Foundation. The results of the application of this proposal of conceptual-formal ontological transition in the domain addressed are presented in this section.

For the presentation of these results, we used some conceptual statements extracted of the model developed in the ConceptMe and evaluated them under the criteria and ontological restrictions presented in the methodology (Table 1).

Each conceptual statement is evaluated from criteria used and the result of this evaluation is a short “YES” or “NO” answer to inform if the statement meets or no the criterion. The entire sample of all assessment criteria under the conceptual statement results in the transition of the statement from the conceptual level to ontological (formal) level, considering the possibility that in some cases this transition is not possible and the statement being classified as “non-ontological”.

Hereafter, we present some examples of analysis of conceptual statements extracted from the model developed in the ConceptMe. The results of this analysis are presented in the Table 2.

1) portion of blood *has_quality* blue colour
2) haemoglobin *has_format* circular
3) fresh frozen plasma *is_a* blood component
4) albumin *is_a* protein
5) cryoprecipitate *is_a* blood component
6) leukocyte *is_a* white blood cell
7) portion of venous blood *is_a* blood in vein
8) circulatory system *part_of* human body
9) blood in coronary artery *part_of* heart
10) blood cells *part_of* portion of plasma
11) portion of blood in capillary *part_of* portion of blood of human body
12) centrifugation *part_of* process for obtaining erythrocytes concentrate
13) erythrocytes *part_of* whole portion of blood
14) nutrients *part_of* portion of blood
15) water *part_of* portion of blood
16) platelet *part_of* platelets concentrate
17) blood component for transfusion *part_of* portion of body substance
18) portion of blood collected by venipuncture *part-of* portion of body substance
19) lumen of coronary artery *part_of* heart
20) cavity of ventricle *part_of* heart

TABLE II. RESULTS OF THE ANALYSIS OF THE RELATIONS IN THE BLOOD DOMAIN

Relation	O1	O2	O3	I123	P123	Analysis
1)	N	-	-	-	-	Non-ontological
2)	Y	N	N	-	-	Non-ontological
3)	Y	Y	Y	YYY	-	Instance_of
4)	Y	Y	Y	YYY	-	Instance_of
5)	Y	Y	Y	YYY	-	SubClass_of
6)	Y	Y	Y	YYY	-	Same_as
7)	Y	Y	Y	YYY	-	Same_as
8)	Y	Y	Y	-	YYY	Structural_part_of
9)	Y	Y	Y	-	YYY	Contained-in
10)	Y	Y	Y	-	YYY	Contained-in
11)	Y	Y	Y	-	YYY	Located-in
12)	Y	Y	Y	-	YYY	Involved-in
13)	Y	Y	Y	-	YYY	Member-part-of
14)	Y	Y	Y	-	YYY	Constitutes
15)	Y	Y	Y	-	YYY	Constitutes
16)	Y	Y	Y	-	YYY	Subquantity-part-of
17)	Y	Y	Y	-	YYY	Temporary-part-of
18)	Y	Y	Y	-	YYY	Temporary-part-of
19)	Y	Y	Y	-	YYY	Immaterial-part-of
20)	Y	Y	Y	-	YYY	Immaterial-part-of

V. Conclusions

In this article, we described the background and essentials of two different approaches involved in ontologies development in the scope of healthcare organizations. The first one deals with the conceptual level and second one deal with the formal level. Then, we analyzed relations and entities extracted from an ontology about blood transfusion under construction, considering the principles underlying these two approaches. We systematized a set of rules to convey knowledge from the conceptual level to the formal level using logical constraints. Finally, we presented partial results the experience of formalization in the case study.

In the scope of Semantic Web, studies on ontologies many times have often emphasized a relevant question in knowledge representation, namely, the balance between expressivity and computability. The set of principles required for a biomedical ontology to become a member of the OBO Foundry [27] repository is a good example of an initiative like this, and that can mitigate the so-called data-silo problem, that is, the situation in which systems can not automatically interoperate because of different ways of modelling. The use of ontological principles seems to be a good bet to improve the quality of information systems. These approaches have been researched worldwide, and the results obtained are expressive.

It is worth observing that these technical-oriented approaches, in general, focus on evaluating ontologies and their characteristics as software artefacts. We believe that when ontologies are developed collaboratively, it is essential to consider the way people see the world and to understand the social processes that have led to the development. In this scenario, we believe that approaches like ConceptMe are essential for ontological engineering.

For future works, we will seek new ways of integrating formal and conceptual approaches in organizational environments. We believe that, in order to attain sound ontologies and ontology-based systems, we should foster the complementarity between these approaches. While some may claim that this is widely know, we have not observed this reality in organizations, which justifies research on ontologies oriented to their particular social dynamics.

Acknowledgments

This work is partially supported by Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG), Governo do Estado de Minas Gerais, Brazil, Rua Raul Pompéia, nº101 - São Pedro, Belo Horizonte, MG, 30.330-080, Brazil.

This research was financed by the North Portugal Regional Operational Programme (ON.2 - O Novo Norte), under the National Strategic Reference Framework (NSRF), through the European Regional Development Fund (ERDF), and by national funds, through the Portuguese funding agency, Fundação para a Ciência e a Tecnologia (FCT), within project NORTE-07-0124-FEDER-000057. The authors would like to thank INESC TEC for its support and the BEST CASE Project for the input and contributions.

References

[1] C. Pereira, C. Sousa, and A. L. Soares, "Supporting conceptualisation processes in collaborative networks: a case study on an R&D project," Inter. Journal of Computer Integ. Manufacturing, vol. 26, no. 11, Nov. 2013, pp. 1066–1086..

[2] T. Tudorache, J. Vendetti, and N. Noy, Web-Protégé. "A lightweight OWL ontology editor for the Web", In: C. Dolbear, A. Ruttenberg, and U. Sattler (Eds.), Proceedings of the Fifth Workshop on OWL: Experiences and Directions, vol. 432 of CEUR Workshop Proceedings. CEUR-WS, 2008.

[3] M. Berzell, Eletronic Healthcare Ontologies: philosophy, the real world and IT structures, PhD thesis, Linkoping University, Division of Health and Society, Departament of Medical and Health Science, Linkoping, Sweden, 2010, pp. 163.

[4] A. Gómez-Pérez, M. Fernandéz, and A. Vicente, "Towards a method to conceptualize domain ontologies", In: ECAI Workshop on ontological engineering, 1996, Budapest.

[5] M. Suárez-Figueroa. NeOn Methodology for Building Ontology Networks, Madrid: Facultad de Informatica da Universidad Politécnica de Madrid, 2010.

[6] A. Newell. "The knowledge level". Artificial intelligence v. 18, n. 1, 1982, pp. 87-127.

[7] D. Kalra, T. Beale, and S. Heard, The OpenEHR Foundation. London: IO Press, 2005.

[8] C. Sousa, C. Pereira, and A. Soares. "Collaborative Elicitation of Conceptual Representations: A Corpus-Based Approach". In: Advances in Inf. Systems and Technology, vol. 206, Á. Rocha, A. M. Correia, T. Wilson, and K. A. Stroetmann, Eds. Berlin: Springer, 2013, pp. 111–124.

[9] K. Munn and B. Smith, Applied Ontology: An Introduction. Heusenstamm, Germany: Ontos Verlag, 2008.

[10] P. Grenon, B. Smith, and L. Goldberg, "Biodynamic Ontology: Applying BFO in the Biomedical Domain", In: Pisanelli (ed.), Ontologies in Medicine, Amsterdam: IOS, 2004, pp. 20–38.

[11] F. Mendonça and M. Almeida, "Hemocomponents and hemoderivatives ontology (HEMONTO): an ontology about blood components", In: ONTOBRAS, 2013, Belo Horizonte, 6º Sem. Pesquisas em Ontologias Brasil, 2013, v. 1, pp. 11-23.

[12] C. Pereira, C. Sousa, and A. Soares, "A socio-semantic approach to support conceptualisation processes: a case study in an R&D project", International Journal of Computer Integrated Manufacturing 2012 (July 2), pp. 1–21.

[13] G. Fauconnier, and M. Turner, "Conceptual Integration Networks", Cognitive Science, vol. 22 (2), 1998, pp. 133-187.

[14] C. Sousa, A. Soares, C. Pereira, and R. Costa, "Supporting the identification of conceptual relations in semi-formal ontology development", In: ColabTKR 2012 - Terminology and Knowledge Representation Workshop at International Conference on Language Resources and Evaluation, Istanbul.

[15] A. Auger and C. Barrière, "Probing semantic relations," Probing Semantic Relations: Exploration and Identification in Specialized Texts, vol. 23, 2010, pp. 1-14.

[16] C. Keet and A. Artale, "Representing and reasoning over a taxonomy of part-whole relations". Applied Ontology 3 (1-2), 2008, pp. 91-110.

[17] S. Schulz, A. Kumar, and T. Bittner, "Biomedical ontologies: What part-of is and isn´t", Journal of Biomedical Informatics 39, 2006, pp. 350–361.

[18] B. Smith, "Relations in Biomedical Ontologies", Genome Biology, 6, R46, 2005.

[19] T. Bittner and M. Donnely, "Logical properties of foundational relations in bio-ontologies". Artificial Intelligence in Medicine, vol. 39, n. 3, 2007, pp. 197–216.

[20] A. Varzi, "Parts, wholes, and part-whole relations: the prospects of mereotopology, Data and Knowledge Engineering, vol. 20, 1996, pp. 259-286.

[21] M. Winston, R. Chaffin, and D. Herrmann, "A taxonomy of part-whole relations". Cognitive Science, 11(4), 1987, pp. 417-444.

[22] Stanford Center for Biomedical Informatics Research. [Online]. Available from: http://protege.stanford.edu// 2015.06.11

[23] BRAZIL, Ministério da Saúde. Guia para o uso de hemocomponentes, Brasília, Brazil, 2008.

[24] ICCBBA – ISBT 128 Standard, Standard Terminology for Blood, Cellular Therapy and Tissue Product Descriptions, v 3.33, January 2010.

[25] American Association of Blood Banks (AABB) 17th edition Technical Manual, Bethesda, Maryland: AABB 2011.

[26] J. Greer, J. Foerster, G. Rodgers, F. Paraskevas, B. Glader, D. Arber, and R. Means, Wintrobe's Clinical Hematology 12th Edition, Philadelphia: Lippincott Williams & Wilkins, 2009.

[27] B. Smith. "The OBO foundry: Coordinated evolution of ontologies to support biomedical data integration". Nature Biotechnology, vol. 25, n. 11, 2007, pp. 1251–1255.

News Curation Service using Semantic Graph Matching

Ryohei Yokoo, Takahiro Kawamura, Yuichi Sei, Yasuyuki Tahara, Akihiko Ohsuga

Graduate School of Information Systems
University of Electro-Communications
Tokyo, Japan
Email: {r-yokoh,kawamura}@ohsuga.is.uec.ac.jp, {sei,tahara}@is.uec.ac.jp, ohsuga@uec.ac.jp

***Abstract*—In recent years, “News Curation Services” that recommend news articles on the Internet to users are getting attention. In this paper, we propose a news curation service that collects and recommends “news articles” that users feel interested by using semantic relationships between terms in the articles. We define “interested” news articles as articles that users have curiosity and serendipity. The semantic relations between events terms are represented by Linked Data. We create News Articles Linked Data (candidates for recommendation to users) and User’s preferences Linked Data (users’ preferences). In order to recommend news articles to users, we first search common subgraphs between two kinds of Linked Data. The experiment showed that the curiosity score is 3.30 (min:0, max:4), and the serendipity score is 2.93 in our approach, although a baseline method showed the curiosity score is 3.03, and the serendipity score is 2.79. Thus, we confirmed that our approach is more effective than the baseline method.**

Keywords*–*Semantic Relation; Linked Data; News Recommendation.

I. INTRODUCTION

Recently, web services, such as paper.li [1] and The Tweeted Times [2] that automatically gather news articles and recommend to users have been popular. The users can easily get interested information by those services called “News Curation Services”. In this paper, we propose a semantic graph application for “News Curation Services”, which recommends interested news articles according to users’ preferences. We define “interested news articles” as articles that user has curiosity and serendipity. A lot of content-based recommendation approaches, such as tf-idf use only words or terms in news articles for features of recommendation. In contrast, our approach applies semantic relation between the terms as the features. Thus, our contribution is that we extract users’ preferences more accurately than other approaches, and then recommend news articles to the users. The semantic relations between terms are represented in Linked Data.

We create two kinds of Linked Data in this paper. First, we create News Articles Linked Data, composed of sentences of news articles, which are candidates for recommendation to users. Next, we create users’ preferences Linked Data, composed of sentences of news articles that users feel interested. In order to recommend news articles to users, we search news articles by finding common subgraphs, that is, triples like term-relation-term between two kinds of Linked Data. If there is a common subgraph. we recommend news articles, which are associated with the subgraph in News Articles Linked Data to the users.

The remainder of the paper is organized as follows. Section II describes related works, and Section III describes our approach. In Section IV, we show experiments and evaluation. Finally, we conclude this paper with discussion and the future work in Section V.

II. RELATED WORK

Most of previous studies for recommendation systems based on contents have applied terms in sentences [3][4]. These recommendation systems need Bag-of-Words vectors as features. They recommend contents with frequent terms in text that users feel interested.

Capelle et al. [5] studied content-based recommendation system, which focused on terms semantics. They developed a system by applying similar terms for news articles that users already read or not. The similarity of terms was calculated by WordNet and a search engine Bing.

There is also a study for constructing Linked Data from news articles. Radinsky et al. [6] extracted news topics from sentences in news article titles for 150 years, and then constructed News Linked Data with causal relationships. Then, they tried to expect future events by tracing the Linked Data.

Ohsawa et al. [7] proposed a method for expecting for the number of “Like” in Facebook pages. They applied the information in DBpedia and made the expectation model with words similarities between Facebook pages.

As recommendation systems by using Linked Data, Khrouf et al. [8] targeted event information. They converted meta information on the event news sites, such as location, time, genre and so on to Linked Data, and recommended the event information to users. The information is searched by a hybrid approach of similarities of events’ structures and a collaborative filtering technique.

Moreover, Mirizzi et al. [9] have applied movie information in DBpedia to Vector-Space-model, and recommended movies, which users feel interested by similarity of movie information, such as genre, director and actor, etc.

Elahi et al. [10] proposed a picture recommendation system with DBpedia infomation.

Passant et al. [11] showed a musician recommendation system by information about musicians in DBpedia. They proposed a method for measuring semantic similarity between Linked Data as Linked Data Semantic Distance (LDSD), and then this method is applied to a lot of recommendation systems with Linked Data.

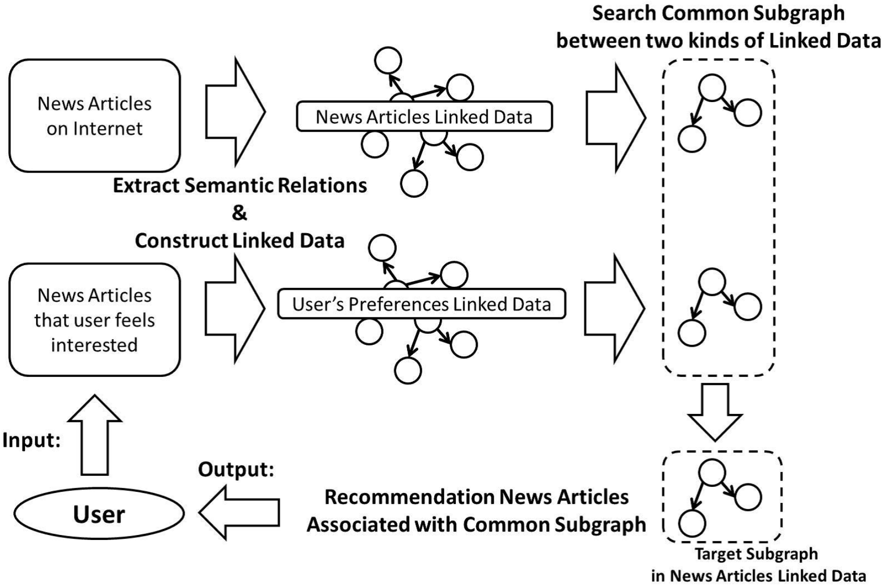

Search Common Subgraph
between two kinds of Linked Data
News Articles on Internet
News Articles Linked Data
Extract Semantic Relations
&
Construct Linked Data
News Articles that user feels interested
User's Preferences Linked Data
Input:
Output:
User
Recommendation News Articles
Associated with Common Subgraph
Target Subgraph
in News Articles Linked Data

Figure 1. FLOW OF OUR APPROACH.

On the other hand, we put specific labels on terms in the text as Semantic Role Labeling [12] to extract semantic relations of the text, and then convert them to Linked Data. WordNet and VerbNet are used in Semantic Role Labeling.

There are many recommendation systems based on contents and Linked Data. However, to the best of our knowledge, there is no news recommendation system by using semantic relations in Linked Data.

III. Proposed Approach

We recommend news articles to users by using semantic relations of terms, since we assume that some news articles that users prefer, indicates the users' interest. Thus, we discuss how to extract the semantic relations and to recommend news articles to users in this section. Figure 1 indicates a flow of our approach.

First, we collect news articles, that users indicated obvious interest from social bookmark sites and others, and then extract the semantic relations from the articles. The semantic relations are combinations of terms with their relations in each sentence of the articles. We assume these semantic relations include users' preferences, and we construct User's Preferences Linked Data.

Next, we crawl a large amount of news articles on the Internet, and extract semantic relations as well, and then construct News Articles Linked Data.

In order to recommend the news articles to the users, we search common subgraphs between User's Preferences Linked Data and News Articles Linked Data. At this time, we also apply an "Entity Linking technique" for matching the terms (nodes of graph). Finally, we recommend the news articles associated with the subgraph in News Articles Linked Data to the users.

In details, the extraction of semantic relations of news articles is described in Section III-A. Section III-B describes how to find common subgraphs between two kinds of Linked Data. Then, we show the technique of Entity Linking in Section III-C.

A. Construction of Linked Data

1) Definition of Semantic Relation: Semantic relations are extracted from each sentence of news articles. In our previous work, Nguyen et al. [13] extracted behavioral properties from Web pages and Tweets to acquire users' behavioral information in a specific event like a disaster. They defined event's properties as Who, Action, What, When, Where, and so on. However, we aim to recommend news articles to users, and thus semantic relations must be simple in order to increase recommendation results. Therefore, we newly defined six new properties in this paper as follows.

- Subject (subject of an event)
- Activity (activity of an event)
- Object (object of an activity)
- Date (date an event occurred)
- Time (time an event occurred)
- Location (where an event occurred)

For example, if a news article has a sentence "Keisuke Honda has been elected to the Worst Eleven in Serie A May 21, 2014", its semantic relations are represented in Linked Data like Figure 2. Our semantic relations are composed of multiple triples, which connect terms in the sentence. The triple is a meta-deta model, which represents the relationship between two resources with a property "resource → (property) → resource". In this case, triples are "elected → (Object) → Worst Eleven" and "Keisuke Honda → (Activity) → elected". Note that, a semantic relation between Subject and Activity is represented as "Subject's term → (Activity) → Activity's term", although other relations are represented as "Activity's term → (property) → term".

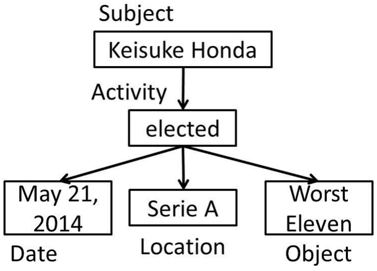


Figure 2. EXAMPLE OF SEMANTIC RELATION.

Terms	Dependency Info.	POS Info.	Labels
Keisuke	1	Noun	B-Subject
Honda	1	Noun	I-Subject
has	2	Auxiliary verb	
been	2	Verb	
elected	2	Verb	B-Activity
to	2	Adposition	
the	3	Article	
Worst	3	Noun	B-Object
Eleven	3	Noun	I-Object

Figure 3. EXAMPLE OF MANUAL LABELING.

2) Pre-processing: In this paper, we adopted Japanese news articles as sources. Then, parentheses frequently appear in news articles and dazzle someone's eyes. Also, they make semantic relations in a sentence more difficult. Therefore, we removed the parentheses in pre-process steps. We first split a sentence with the parentheses to string outside the parentheses and string inside parentheses to simplify the sentence. But, our previous work showed the string inside parentheses are often useless, and thus we deleted them all.

Also, we registered 7,572 locations in Japan and 150,90,897 titles of all Japanese Wikipedia articles as of December, 2014 in our dictionary.

3) Semantic Role Labeling with CRF: In order to extract the semantic relations from news articles, we apply Conditional Random Field (CRF) [14]. CRF is a machine learning technique to solve sequential labeling problems. CRF has been used in morphological analysis, part-of-speech (POS) tagging, named entity recognition [15], and group activity recognition [16], etc.

First, we extract dependency information between terms, and POS information of a sentence, and then convert them to a feature vector format for CRF. We get the dependency information from Cabocha [17], and the POS information from Mecab [18].

As a training dataset for CRF learning phase, we used sentences manually labeled in advance. Figure 3 shows an example of training data, "Keisuke Honda has been elected to the Worst Eleven". I is internal of a chunk, B shows beginning of the chunk. In estimation phase, we use a CRF's model constructed by the training data, and automatically put properties to each sentence.

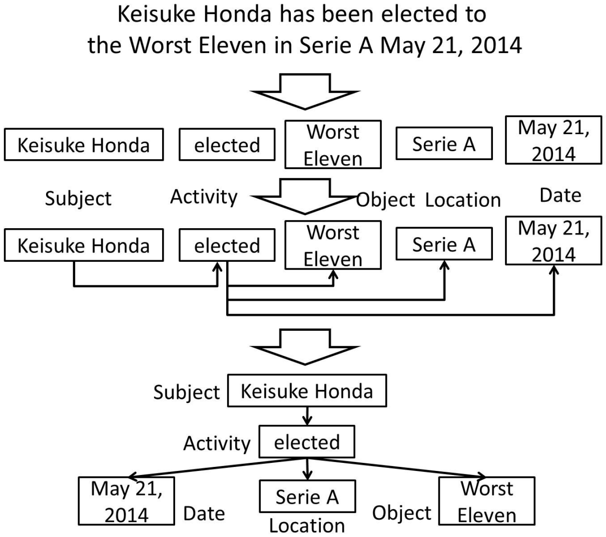


Figure 4. PROCEDURES OF SEMANTIC RELATION EXTRACTION.

As a preliminary experiment, we collected 98 sentences from 13 news articles in Japanese for our the training dataset. The articles are collected from Japanese news site, Asahi.co.jp [19] on Oct. 3, 2014. Details of the dataset are shown in Table I. We then tried to estimate properties (labels) in test sentences, but the accuracy by 10-fold cross-validations was not enough when applying CRF as it is. Especially, Subject, Time, and Location indicate low accuracies. Hence, we devised some heuristics for Time and Location. The heuristic rules are executed on the CRF results based on the dependency and POS information.

As a result, Table II shows the average accuracies for each labels become more than 80%. "Weighted Average" means the average accuracy for all labels.

4) Construction of Semantic Relation: In Figure 4, we show how to construct the semantic relations from the labeled sentences. The figure indicates a procedures of semantic relation construction from a sentence"Keisuke Honda has been elected to the Worst Eleven in Serie A May 21, 201". First, we extract the labeled terms in the sentence. Then, we gather the terms for each semantic relation using dependency information. Finally, we connect these terms with semantic relations, and then convert it to Linked Data in Resource Description Framework (RDF).

B. Recommendation of News Articles using Common Subgraph

In order to recommend news articles to users, we search "common subgraphs" between News Articles Linked Data and User's Preferences Linked Data. We define "subgraphs" in Linked Data as one or more linked triples. We find common subgraphs by finding at least a common triple between two kinds of Linked Data. Common triples need common Subject, Value and Property between two triples, and thus we first try to find them for searching common subgraphs. Then, we get news articles associated with the common subgraphs in News Articles Linked Data.

We show an example of the common subgraph in Figure 5. The subgraph "Keisuke Honda (Subject) ← Activity ←

TABLE I. SUMMARY OF TRAINING DATA

sentences	terms	all labels	Subject	Activity	Object	Date	Time	Location
98	2,479	1,888	265	718	754	79	37	35

TABLE II. ACCURACY OF LABELING

	Subject	Activity	Object	Date	Time	Location	Weighted Average
Precision	67.80%	91.22%	87.41%	81.23%	82.46%	97.77%	86.48%
Recall	85.61%	87.20%	82.22%	90.03%	87.50%	85.71%	86.59%
F-measure	75.67%	89.16%	84.74%	85.40%	84.90%	91.34%	86.53%

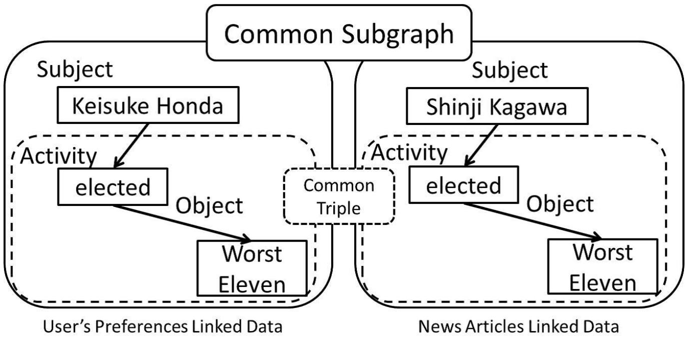


Figure 5. EXAMPLE OF COMMON SUBGRAPH.

elected (Activity) ← Object ← Worst Eleven (Object)" in User's Preferences Linked Data was extracted from a sentence "Keisuke Honda has been elected to the Worst Eleven". Similarly, the subgraph "Shinji Kagawa (Subject) ← Activity ← elected (Activity) ← Object ← Worst Eleven (Object)" in News Articles Linked Data was extracted from a sentence "Keisuke Honda has been elected to the Worst Eleven". These subgraphs have a common triple "elected (Activity) ← Object ← Worst Eleven (Object)", and so this corresponds to a common subgraph. Moreover, each subgraph has a partial match "x (Subject) ← Activity ← elected (Activity)" linked to the common triple. Therefore, we recommend a news article associated with the subgraph "Shinji Kagawa (Subject) ← Activity ← elected (Activity) ← Object ← Worst Eleven (Object)" to users.

Figure 6 shows an algorithm for searching common subgraphs between two kinds of Linked Data. Inputs are *UserGraph* and *NewsGraph*. *UserGraph* is a set of triples in User's Preferences Linked Data. Similarly, *NewsGraph* indicates a set of triples in News Articles Linked Data. First, we check whether *user_triple* and *news_triple* have a common triple or not by using *SIMTRIPLE*. Details of *SIMTRIPLE* are shown in Figure 7. If these triple are determined as a common triple, we get other triples include terms (Subject or Value) of each triple. Then, we search triples that have a common Property between *u_graph* and *n_graph* by *PartialMatch*. In addition, we gather them to *X*. Outputs are common subgraphs in News Articles Linked Data that are linked to *n_triple* and *x*. Finally, we recommend news articles associated with the common subgraphs.

In our approach, we can collect common subgraphs not only in the case that we were able to entirely extract semantic relations in a news articles but also the cases that we partially extracted the semantic relations. We use subgraphs for matching, which have at least two triples with two properties and three nodes. Therefore, common subgraph search in our approach works with news article if the extracted semantic relations have at least two linked triples.

However, the defined schema for Linked Data has Activity as a hub as shown in Figure 2. Therefore, the common subgraph search cannot work if the semantic relations do not include Activity's terms.

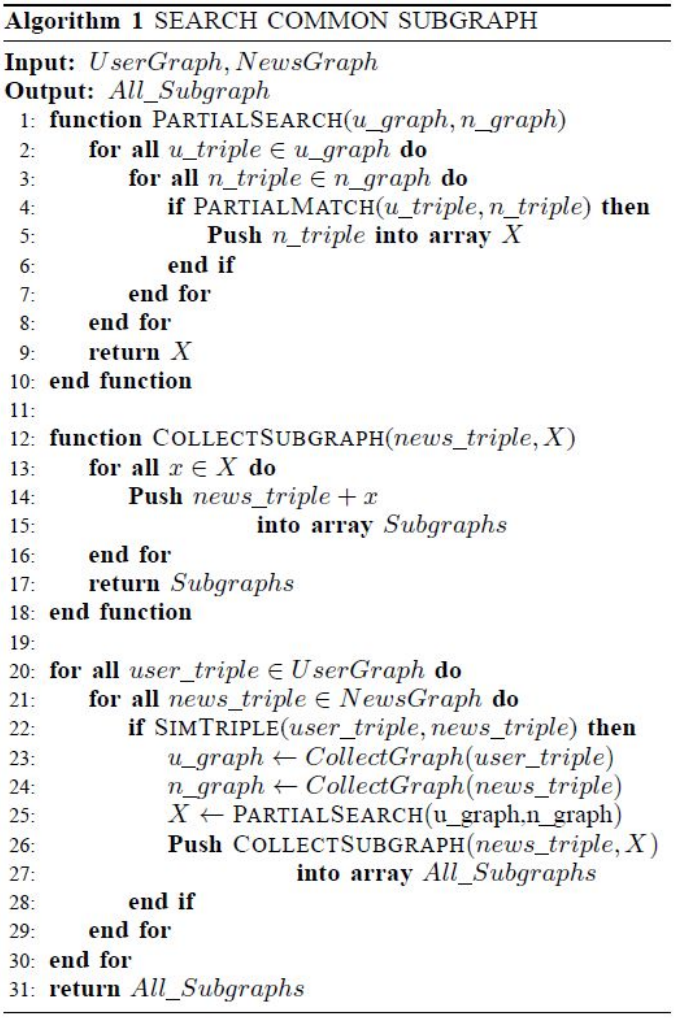
C. Entity Linking

In order to search common subgraphs between User's Preferences Linked Data and News Articles Linked data, the common subgraphs need the same Subject, Property, and Value. However, the number of common subgraphs is very little if we search the common subgraphs with exact matching of terms. Also, this causes to miss an opportunity to find similar subgraphs, and thus leads to a matter of no recommendation.

Therefore, we apply "Entity Linking" for common subgraphs search. Entity Linking is a task for searching common terms by applying synonyms of Entity (terms) in sentences. Entity Linking usually needs an expression dictionary, and a similarity measure between terms. For example, if there is a sentence includes "be elected to the Worst Eleven", "be elected" has the same meaning as "be chosen", and"elected", etc. We get much more common subgraphs than the exact matching by applying such an Entity Linking technique. Study of Bunnescu et al. [20] is a pioneer of Entity Linking. Bunnescu has proposed a method for resolving the word-sense disambiguation by using hyperlink structure between articles of Wikipedia. Also, Hoffart [21] developed an Entity Linking framework AIDA for named entity extraction and word-sense disambiguation. Hoffart's Entity Linking is similarity calculation for terms by using contexts in sentences.

In this paper, we applied Jaccard index and Japanese WordNet for similarity calculation. Jaccard index is a string matching techniques. Equation (1) indicates a formula for Jaccard index, which represents a ratio of common elements of the two sets: A and B. Here, we calculate a similarity score between the two terms by using their surfaces. Inputs are two terms and output indicates a similarity score between [0-1]. If the score is 1, A and B are matched exactly. We a set threshold score of Jaccard index as 0.5.

$$Jaccard(A,B) = \frac{A \cap B}{A \cup B} \tag{1}$$



Algorithm 1 SEARCH COMMON SUBGRAPH

```
Input: $UserGraph, NewsGraph$
Output: $All\_Subgraph$
function PARTIALSEARCH($u\_graph, n\_graph$)
    for all $u\_triple \in u\_graph$ do
        for all $n\_triple \in n\_graph$ do
            if PARTIALMATCH($u\_triple, n\_triple$) then
                Push $n\_triple$ into array $X$
            end if
        end for
    end for
    return $X$
end function

function COLLECTSUBGRAPH($news\_triple, X$)
    for all $x \in X$ do
        Push $news\_triple + x$
                  into array $Subgraphs$
    end for
    return $Subgraphs$
end function

for all $user\_triple \in UserGraph$ do
    for all $news\_triple \in NewsGraph$ do
        if SIMTRIPLE($user\_triple, news\_triple$) then
            $u\_graph \leftarrow CollectGraph(user\_triple)$
            $n\_graph \leftarrow CollectGraph(news\_triple)$
            $X \leftarrow$ PARTIALSEARCH(u_graph,n_graph)
            Push COLLECTSUBGRAPH($news\_triple, X$)
                      into array $All\_Subgraphs$
        end if
    end for
end for
return $All\_Subgraphs$
```

Figure 6. SEARCH COMMON SUBGRAPH ALGORITHM.

By applying Jaccard index, we can determine that “elected” and “elect” are identical. However, “elected” and “chosen” are not solved only by Jaccard index. Therefore, we also applied WordNet, and search similar terms for covering a string matching’s weak point.

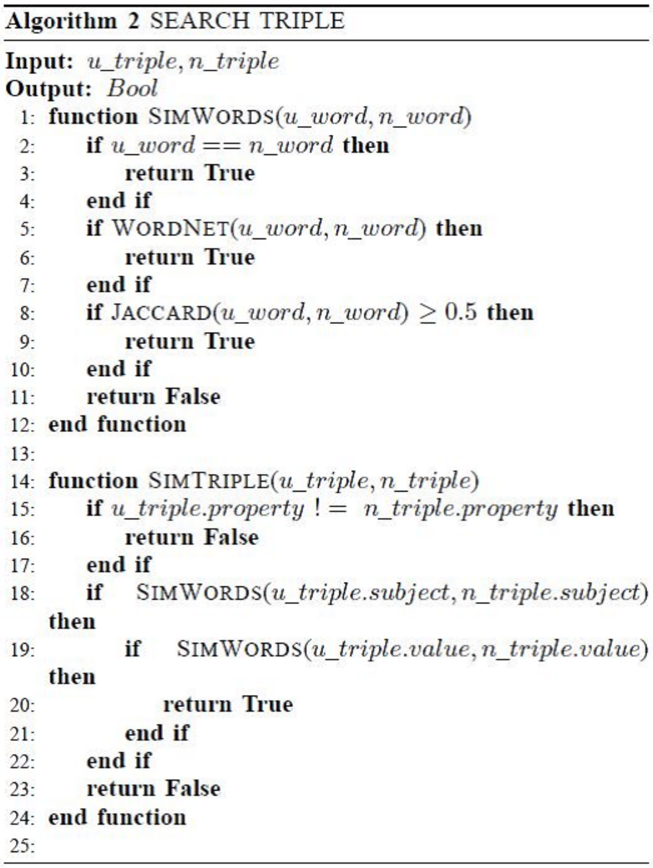
We show a method for searching common triples with Entity Liking in Figure 7 (*SIMTRIPLE* in Figure 6). Inputs are a triple in User’s Preferences Linked Data *u_triple* and a triple in News Articles Linked Data *n_triple*. Note that the triples must have the same Property. Thus, we calculate terms similarity of Subject terms (*u_triple.subject* and *n_triple.subject*) and Value terms (*u_triple.value* and *n_triple.value*) in the order of exact match, WordNet, and Jaccard index.

IV. EXPERIMENT

We recommended “interested” news articles to test users with our approach. We define “interested” means curiosity and serendipity. Therefore, we set as metrics “curiosity”, “serendipity”, and “relevance” (similarity) as reference information.

A. Dataset

In order to construct News Articles Linked Data, we applied 21,105 news articles from Oct. 4, 2014 to Jan. 10, 2015. It took about an hour to construct the Linked Data with



Algorithm 2 SEARCH TRIPLE

```
Input: $u\_triple, n\_triple$
Output: $Bool$
function SIMWORDS($u\_word, n\_word$)
    if $u\_word == n\_word$ then
        return True
    end if
    if WORDNET($u\_word, n\_word$) then
        return True
    end if
    if JACCARD($u\_word, n\_word$) $\geq 0.5$ then
        return True
    end if
    return False
end function

function SIMTRIPLE($u\_triple, n\_triple$)
    if $u\_triple.property\ != \ n\_triple.property$ then
        return False
    end if
    if SIMWORDS($u\_triple.subject, n\_triple.subject$) then
        if SIMWORDS($u\_triple.value, n\_triple.value$) then
            return True
        end if
    end if
    return False
end function
```

Figure 7. SEARCH TRIPLE ALGORITHM.

the articles. Similarly, we applied 1,471 news articles from Jan. 11, 2015 to Jan. 19, 2015 for User’s Preferences Linked Data, which was constructed in a few minutes. The news articles that construct both Linked Data are collected from Japanese news site, Asahi.co.jp. A summary of our dataset for News Articles Linked Data is shown in Table III, and a summary for User’s Preferences Linked Data is shown in Table IV.

B. Experimental Setting

We found 978 common subgraphs from the two datasets. These subgraphs were found between 142 news articles in User’s Preferences Linked Data and 578 news articles in News Articles Linked Data. Thus, 578 news articles associated with the common subgraphs could be recommended to test users. The calculation time is about 3,577 sec. However, checking a large number of articles is almost impractical for test users. Therefore, in order to reduce the news articles, we excluded the following common subgraphs.

- Properties in common triples are Date and Time.
- Number of terms in common triples is 2 or less.
- Common triple’s terms indicate tense alone.
- Common triple includes only short length terms.

The number of the reduced common subgraph was 166. These common subgraphs are composed of 62 news articles in User’s Preferences Linked Data and 126 news articles in News Articles Linked Data. Thus, we used 62 news articles in User’s Preferences Linked Data for evaluation. There were some news

Copyright (c) IARIA, 2015. ISBN: 978-1-61208-420-6

TABLE III. DATASET FOR NEWS ARTICLES LINKED DATA

Articles	Nodes	Labels	Subject	Activity	Object	Date	Time	Location
21,105	42,890	44,869	10,892	12,040	17,994	1,761	749	1,433

TABLE IV. DATASET FOR USER’S PREFERENCES LINKED DATA

Articles	Nodes	Labels	Subject	Activity	Object	Date	Time	Location
1,471	4,526	4,617	1,612	1,612	1,548	172	84	117

TABLE V. EXPERIMENTAL RESULT

	relevance	curiosity	serendipity
Our Approach	3.06	3.30	2.93
Baseline	3.22	3.03	2.79

TABLE VI. PERFORMANCE OF RECOMMEND NEWS

	good	excellent
Our Approach	2.55	1.15
Baseline	2.40	0.65

articles, which can recommend multiple news articles to users. But we recommended a news article from a news article that users feel interested. A news article is selected based on the similarities of triples. If the similarities are the same, we randomly chose a news article.

C. Experiment Procedure

We asked the test user to determine whether or not the recommended articles are relevant (similar to), an article that the user feels interested, and has curiosity, serendipity.

We defined the interesting articles, which users that users get attracted to and make discovery from. We then regarded the articles, which users get highly attracted as the curiosity articles, and the articles, which users make an important discovery as the serendipity articles. There are 20 test users, in which 13 test users are our university students. The test users answered in 4 levels: “I think so”, “I think so a little”, “I don’t think so a little”, and “I don’t think so”. We also conducted comparison with a baseline method using tf-idf. The method is the most famous approach for extracting feature words of sentences and it has been used for a lot of studies [22][23]. It needs term (word) frequency as tf and inverse Document Frequency as idf for calculating weights of the words. We extracted top three weighted words from an article that test users feel interested. All three words are nouns. The baseline method searches a news article contains those three words from dataset for News Articles Linked Data, and then recommends the news articles to each test user.

D. Evaluation

Table V indicates the average scores of our approach and the baseline method. Our approach showed relevance:3.06, curiosity:3.30, and serendipity:2.93 in average. In contrast, the baseline method showed relevance:3.22, curiosity:3.03, serendipity:2.79. As a result, the curiosity and the serendipity score of our approach were higher than the baseline method, although the relevance is lower than the baseline.

The reason why the baseline had a high relevance score was that the baseline method recommended news articles, which include three frequent nouns. However, semantic relations in our approach include terms of noun, verb and adjective, and so on. As a result, the baseline method directly retrieved topics represented in nouns of news articles. Our common subgraphs include several terms, which are not directly relevant to the news articles that users feel interested. However, these terms have the same semantic relations from a certain topic terms as in the news articles the users feel interested. In a sense, we believe that these *variables* contributed to raise the curiosity and the serendipity score, decreasing the relevance score.

Finally, we checked how many “interested” news articles were recommended to the users. If a test user determined that the curiosity and serendipity score are more than 3, we counted the news article as “good”. Then, if a test user answered that both scores are 4 , we counted the new article as “excellent”. We show the result in Table VI. Our approach and the baseline method are almost the same in “good”, but our approach recommended more “excellent” news articles than the baseline method. We thus confirmed that, our approach is superior to the conventional content-based method.

V. Conclusion and Future Work

In this paper, we proposed a new “News Curation Service” by using semantic relations in news articles. Semantic relations are represented as Linked Data. We proposed an approach for constructing Linked Data from news articles and recommend news articles to users based on common subgraphs between User’s Preferences Linked Data and News Articles Linked Data. Through the experiments, we confirmed our approach can incorporate more users’ interest than the existing approach.

In the future work, we will improve accuracy of the CRF labeling and Entity Linking. In addition, we will examine patterns of common subgraphs for news recommendation. Also, we reconstruct our Linked Data schema to find more common subgraphs between two kinds of Linked Data.

Acknowledgment

This work was supported by JSPS KAKENHI grant Numbers 24300005, 26330081, 26870201.

References

[1] Paper.li team: “Paper.li – Be a publisher”, http://papper.li, 2015.06.11.

[2] Tweeted Times team: “The Tweeted Times | Content curation and publishing”, http://tweetedtimes.com, 2015.06.11.

[3] W. Lee, K. Oh, C. Lim, and H. Choi: “User profile extraction from Twitter for personalized news recommendation”, Proceedings of the 16th Advanced Communication Technology, 2014, pp. 779-783.

[4] W. IJntema, F. Goossen, F. Frasincar, and F. Hogenboom: “Ontology-based News Recommendation”, Proceedings of the 2010 EDBT/ICDT Workshops, 2010, pp. 16:1-16:6.

[5] M. Capelle, F. Hogenboom, and A. Hogenboom: “Semantic News Recommendation Using WordNet and Bing Similarities”, Proceedings of the 28th Annual ACM Symposium on Applied Computing, 2013, pp. 296-302.

[6] K. Radinsky, S. Davidovich, and S. Markovitch: “Learning causality for news events prediction”. Proceedings of the 15th international conference on World Wide Web, 2012, pp. 909-918.

[7] S. Ohsawa and Y. Matsuo: Like Prediction: “Modeling Like Counts by Bridging Facebook Pages with Linked Data”. Proceedings of the 22Nd International Conference on World Wide Web Companion, 2013, pp. 541-548.

[8] H. Khrouf and R. Troncy: “Hybrid event recommendation using linked data and user diversity”, Proceedings of the 7th ACM conference on Recommender systems, 2013, pp. 185-192.

[9] R. Mirizzi, T. D. Noia, A. Ragone, V. C. Ostuni, and E. D. Sciascio: “Movie Recommendations with DBpedia”, IIR, volume 835 of CEUR Workshop Proceedings, 2012, pp. 101-112.

[10] N. Elahi, R. Karlsen, and E. J. Holsb?: “Personalized Photo Recommendation By Leveraging User Modeling On Social Network”. Proceedings of International Conference on Information Integration and Web-based Applications, 2013, pp. 68-71.

[11] A. Passant: “dbrec: music recommendations using DBpedia”, ISWC’10 Proceedings of the 9th International Semantic Web Conference on The Semantic Web - Volume Part II, 2010, pp. 209-224.

[12] Y. Matsubayashi, N. Okazaki, and J. Tsujii: “Generalization of Semantic Roles in Automatic Semantic Role Labeling”, Information and Media Technologies, 2014, pp. 736-770.

[13] T. M. Nguyen, T. Kawamura, Y. Tahara, and A. Ohsuga: “Self-supervised capturing of users´ activities from weblogs”, International Journal of Intelligent Information and Database Systems, Vol.6, No.1, 2012, pp. 61-76.

[14] J. Lafferty, A. McCallum, and F. C. N. Pereira: “Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data”, ICML ’01 Proceedings of the Eighteenth International Conference on Machine Learning, 2001, pp. 282-289.

[15] G. Zhu, T. J. Bethea, and V. Krishna: “Extracting Relevant Named Entities for Automated Expense Reimbursement”, Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2007, pp. 1004-1012.

[16] T. Kaneko, M. Shimosaka, S. Odashima, R. Fukui, and T. Sato: “Consistent collective activity recognition with fully connected CRFs”, Proceedings of the 21st International Conference on Pattern Recognition, 2012, pp. 2792-2795.

[17] Taku Kudo: “CaboCha: Yet Another Japanese Dependency Structure Analyzer”, http://taku910.github.io/cabocha/, 2015.06.11.

[18] Taku Kudo: “MeCab: Yet Another Part-of-Speech and Morphological Analyzer”, http://taku910.github.io/mecab/, 2015.06.11.

[19] The Asahi Shimbun Company: “Asahi Shinbun Degital: The news cite of Asahi”, http://asahi.com, 2015.06.11.

[20] R. Bunnescu and M. Pasca: “Using Encyclopedic Knowledge for Named Entity Disambiguation”. Proceesings of the 11th Conference of the European Chapter of the Association for Computational Linguistics, 2006, pp. 9-16.

[21] J. Hoffart et al.: “Robust disambiguation of named entities in text”. Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2011, pp. 782-792.

[22] J. H. Paik: “A novel tf-idf weighting scheme for effective ranking”. In Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR ’13, 2012, pp. 343-352.

[23] L. F. S. Teixeira, G. P. Lopes, and R. A. Ribeiro: “An extensive comparison of metrics for automatic extraction of key terms”. In Joaquim Filipe and Ana L. N. Fred, editors, ICAART 2012, 2012, pp. 55-63.

Temporal RDF System for Power Utilities

Mohamed Gaha, Arnaud Zinflou, Alexandre Bouffard, Luc Vouligny,
Mathieu Viau, Christian Langheit and Etienne Martin

Institut de Recherche
d'Hydro-Québec
Varennes, QC, Canada
Email: {gaha.mohamed|zinflou.arnaud|bouffard.alexandre|vouligny.luc}@ireq.ca
{viau.mathieu|langheit.christian|martin.etienne}@ireq.ca

***Abstract*—Temporal data is a critical component in many applications. This is especially true in analytical applications for the smart grid. The analytical process often requires uncovering and analysing data and complex relationships from heterogeneous and distributed data sources that change over time. A common approach for this task is the use of relational databases or even data warehouses, which unfortunately do not allow reasoning and inference. Ontologies and semantic technologies are proving useful to leverage the value already embodied in existing systems without replacing the enterprise systems. In this paper, we describe how the usage of a formal representation of knowledge can support elaborated processes such as storing and extracting temporal data. The first example uses a semantic approach to capture and manage time series changes. The second example is a direct application of the first one. It consists on an efficient RapidMiner extension that allows end users to transparently extract temporal data from heterogeneous data sources.**

***Keywords*–Ontology; Heterogeneous data source; Versioning; Data-Mining tools; Temporal data.**

I. Introduction

As more smart technologies are deployed across the electrical grid, it generates unprecedented data volume. To manage and use this information, utility companies such as Hydro-Québec must be capable of high-volume data management and advanced analytics designed to transform data into actionable insights. In this context, it is often necessary that the analysis process spans across multiple heterogeneous data sources. Ontologies and semantic metadata standards help and facilitate the aggregation and the integration of this content [1]. In addition, standard models for metadata representation on the World Wide Web, such as the Resource Description Framework (RDF), model relationships as first class objects making it very natural to query and analyze entities based on their relationships.

With the advent of semantic technologies and widely shared ontologies, it becomes possible to build an enterprise unified information view over heterogeneous and distributed data sources [2]. In the electric power industry, there exists the Common Information Model (CIM) [3] ontology that has been adopted by the International Electrotechnical Commission (IEC). It is the most complete and widely accepted ontology that offers a common language to exchange information between applications in the electrical field domain. The CIM is defined through a set of IEC international standards, mainly 61970-301 and 61968-11. The first release was standardized in 2003 and now contains more than a thousand concepts covering generation, transmission and distribution of power utilities.

The use of a common language like the CIM presents a significant opportunity to overcome the semantic barriers between existing information islands. The CIM can reap advantages from a formal representation of knowledge in order to support complex processes. So far, ontologies like the CIM and semantic analytics tools have primarily focused on static data, in the sense that entities represented in these ontologies do not change over time. However, in real context, temporal data are often critical to analyze because they refer to evolving phenomena. As a consequence, managing temporal data are necessary for an effective application of ontologies and semantic technologies in the power industry.

In this paper, we investigate the use of ontologies and semantic technologies to support the storage and the extraction of temporal data in industrial context. Two significant issues have been explored: a versioning mechanism for the RDF data and a method to extract the temporal datasets, successfully applied to several sources in a transparent way for the end user. To the best of our knowledge, despite the proposal of different approaches to support RDF versioning [4], and capture and monitor changes [5], a real application on large scale problems was still missing. We describe concrete examples from the power industry.

This paper is divided as follows: Section 2 presents a non exhaustive literature review of recent ontology evolution and versioning techniques. Section 3 describes the context where our research occurs. Section 4 presents the basis for storing RDF versions in triple store according to previous work. In Section 5, we describe our experimental results and we present an application that combines a semantic triple store and a time series database. Finally, we end with a conclusion and future work.

II. Background

This section reviews the existing work on ontology evolution and versioning. Most of the studies on ontology versioning focus on the validity, interoperability and management of all versions. OntoView [6] allows ontology engineers to compare versions of an ontology and to specify how the ontology concepts in two versions are related. SemVersion [7] is an RDF-based ontology versioning system that supports query answering across multiple versions and the differences between arbitrary versions. PromptDiff [8] compares different versions of ontologies using heuristics, and provides the user with their deltas. However, all these versioning systems store all snapshots in a repository so that the deltas between versions must be recomputed on the fly whenever the change information is required. In other words, they do not consider the space overhead in supporting versions. That is, if we redundantly store every version in a separate storage space, the space requirement would be enormous, especially in a large scale ontology system. Furthermore, this approach also has a limitation in that it recalculates the changes between versions whenever the user queries the ontology.

Tzitzikas et al. [9] focus on the storage space in the RDF repositories and propose a storage index, called Partial Order Index (POI), which provides an efficient RDF version insertion algorithm in main memory. Since this storage scheme is based on partial orders of triple sets, it is the most efficient for storage space in which the new version is a subset or superset of the existing versions. However, the new version cannot be a subset or superset when it has both added and removed triples compared to the existing versions. In addition, in order to construct a specific version, it needs to traverse all the ancestor elements of the given element in the POI. Thus, it is not scalable as the data size increases.

Recently, IM et al. [10] proposed a versioning framework for the RDF data model based on relational databases. This scheme stores the original RDF version and the deltas between each two consecutive versions. They store the deltas separately in a *delete* and *insert* tables, and construct a logical version on the fly using SQL statements that join the version from the original version and the relevant delta tables. The proposed framework is promising but needs to be implemented in a relational database and not in a triple store. Therefore, inference and reasoning is not directly allowed on all versions.

III. Context

In this work, our application context is the Hydro-Québec Distribution network Division (HQD). We used four heterogeneous datasets from the HQD systems. These systems are IRD (French acronym for Inventory of Distribution Network), GSS (French acronym for Underground Structure Management System), GIS (Geographic Information System) and SAP-BW (SAP Business Information Warehouse). The four systems contain data on the distribution network, such as: connectivity, equipment, geographical position, electrical characteristics, etc.

The data of the four systems is not static, but rather changes as a function of time. For each database, a new data dump takes place every weekend. Hence, every week, all the four systems are updated with new datasets. In accordance with the weekly updates, we map the four relational databases to the CIM ontology and *incrementally* export the resulting triples into Oracle 12c RDF Semantic Graph (OSG) triple store. The export process is done by using the *D2R dump-rdf* tool [11]. All the RDF data is bulk loaded into the OSG triple store. The latter was installed on a HP Xeon E7-2830 (2.13 GHz, 8 cores with hyper-threading) processors with 2TB of RAM and 4 ioDrive2 flash block devices of 1.2TB each managed with Oracle ASM. OSG is a secure and scalable platform that supports large RDF graphs of billions of triples and includes capabilities for using forward-chaining inference via RDFS, RDFS++, OWL-SIF, OWL-Prime, OWL2-RL and user-defined rules. It also supports parallel queries.

The weekly stream of new data generates a huge volume (more than 200,000,000 triples) of data and leads to an increase in the complexity of processing. To make valuable business decisions over changing and evolving databases, we decided at the research center of Hydro-Québec (IREQ) to build an architecture capable of dealing efficiently with a vast amount of heterogeneous time series. We developed a set of tools to allow non IT experts to extract and process the time series.

In the next section, we will test three versioning mechanisms, and we will share our experiences on effective means of building a semantic application for heterogeneous time series.

IV. RDF Versioning Management in a triple store

In this section, we do not propose a new RDF versioning system, but rather the basis for storing RDF versions in a triple store according to previous work presented in Section II.

In the context of an RDF triple store, there are a number of ways in which to implement a version control system. A primary choice and probably the simplest is to store the different versions in separate spaces. This approach called the *All Snapshots approach* [12] is very effective for querying but requires excessive storage space. A second choice is to use a *Delta-Based approach* to overcome the excessive space requirements of the All Snapshots approach. Two kinds of delta can be implemented: the *Sequential Delta* [12] and the *Aggregated Delta* [10].

The Sequential Delta approach consists in storing an original version and the delta of each subsequent version separately. Formally, given the original version V_i, let V_{i+1} be the logical version and $\Delta_{i,i+1}$ be the set of change operations between V_i and V_{i+1}. Then, V_{i+1} can be represented as follow:

$$[V_{i+1} = \Delta_{i,i+1}(V_i)] \tag{1}$$

Thus, in order to access a specific logical version, we must construct the logical version on the fly by applying the deltas between the original version and the logical version.

Instead of executing all the in-between deltas in sequence, the aggregated delta can create a logical version directly by storing all of the possible deltas in advance. In other words, given a sequential delta $\Delta_{i,i+1}, \Delta_{i+1,i+2}, ..., \Delta_{j-1,j}$ between V_i and V_{i+1}, an aggregated delta is defined as follow:

$$[\sum_{n=i}^{j-1} \Delta_{n,n+1} = \sum_{n=i}^{j-1} \Delta^{-}_{n,n+1} \cup \sum_{n=i}^{j-1} \Delta^{+}_{n,n+1}, (i < j)] \tag{2}$$

$$[\Delta_{g(i,j)} = \sum_{n=i}^{j-1} \Delta_{n,n+1} - C_t] \tag{3}$$

Where C_t is the set of change operations with overlapped triples in all stored delta.

V. Experiments

A. Experimental settings

We implemented all the version schemes in OSG. Table I summarizes the characteristics of our real data sets. The datasets used in the experiments have between 35,000,000 and 125,000,000 triples.

TABLE I. SIZE OF DATASETS

	IRD	GSS	GIS	SAP-BW
#triples	100,000,000	3,560,230	90,155,000	125,236,540

B. *Versioning and monitoring temporal data changes*

In this section, we compare the performance of three RDF version management methods for our application context: the All Snapshots approach (as used in SemVersion [7]), the Sequential Delta (based on the change detection between consecutive versions) and the Aggregated Delta.

For this part of the experiment we consider only the power transformer equipments from the IRD dataset. This category of equipments represent more than 600,000 equipments in the distribution network, and the delta, the difference between each week, is less than 1%.

Figure 1 shows the number of triples required to store power transformers for each RDF versioning approach. The number of triples of each scheme includes the total number of triples in all the versions and, if any, all the deltas schema. In Sequential Delta and Aggregated Delta, we consider the first version as the original version. With the All Snapshots approach, as shown in Figure 1, the number of triples increases linearly as the number of versions increases, since this scheme stores all the version snapshots in the triple store. In contrast, the Sequential Delta and Aggregated Delta approaches require less triples than the All Snapshots, because they store only the original version and the deltas. When we compare the Sequential Delta and the Aggregated Delta to each other, we notice in Figure 2 that the Aggregated Delta requires more triples than the Sequential Delta. This is because there are duplicated triples stored by the aggregated delta procedure. In terms of number of triples, the Sequential Delta procedure requires less storage space than the Aggregated Delta.

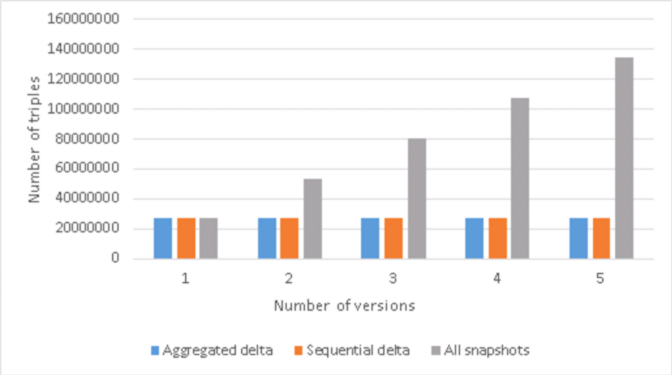


Figure 1. Number of triples for the version management

Figure 3 shows the construction time of versions in the Sequential Delta and the Aggregated Delta. The y-axis represents the construction time of versions in seconds, and the x-axis denotes the specific versions to be constructed. In order to generate versions which are not stored physically, we need to construct logical versions on the fly from the original version. As shown in Figure 3, while the construction time in the Sequential Delta is proportional to the number of versions we need to trace backwards, the Aggregated Delta can compute any specific version at almost a constant time. This is because the Aggregated Delta recreates any version by applying only a corresponding aggregated delta to the original version. Since the relevant version needs to be constructed on the fly for a given query and it occurs very frequently, the Sequential Delta performance is very critical.

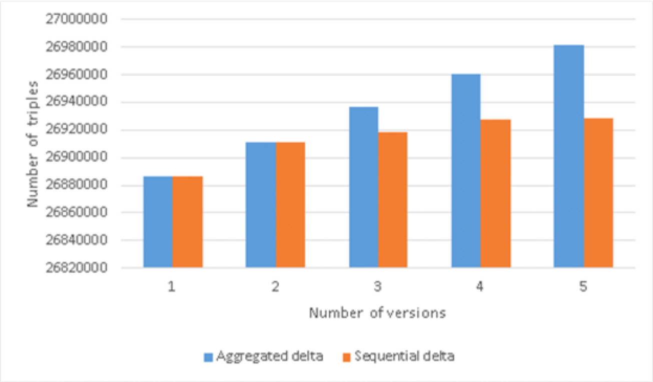


Figure 2. Number of triples for the Sequential Delta vs the Aggregated Delta

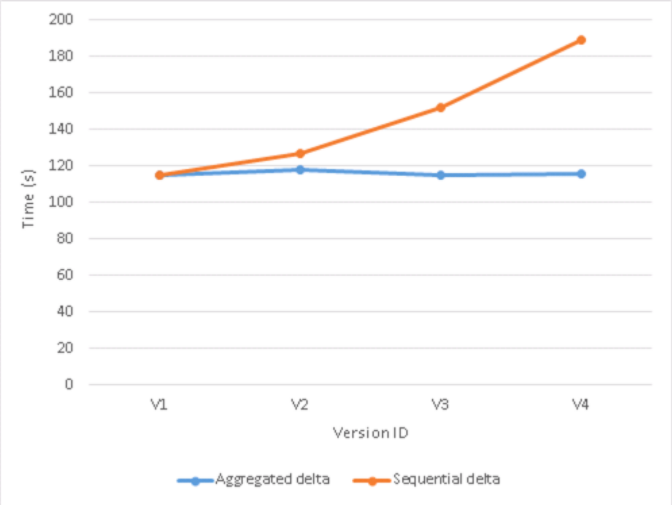


Figure 3. Construction time for the Delta-Based versioning approaches

We also evaluated the query performance of various version storage schemes using the queries in Table II. All the queries are in SPARQL. The SPARQL Query Language is a language for querying the RDF data [13]. Basically, the queries in Table II simply count the number of power transformers for a specific version. Figure 4 shows the average response time of ten executions of the queries of Table II. We notice that the All Snapshots approach is superior to both Delta-Based methods for these queries except for the first version. This can be easily explained by the fact that the sample queries used here require the computation in a specific version and the All Snapshots approach physically stores all the versions. With the Delta-Based approaches, we first need to construct the version on the fly and then query against it.

C. *Managing heterogeneous time series*

With the advent of the smart meters in the distribution network, power system engineers and utility operators began

TABLE II. SAMPLE QUERY

Method	Query
All Snapshot	SELECT (count(?s) as ?total) where {?s a cim:PowerTransformer}

Seq. delta	SELECT (count(?s) as ?total) where {
	graph <V0>{ ?s a cim:PowerTransformer}
	minus {graph <delete v3_v4>{ ?s a cim:PowerTransformer}}
	union {graph <insert v3_v4 >{ ?s a cim:PowerTransformer }}
	minus {graph <delete v0_v1 >{ ?s a cim:PowerTransformer}}
	union {graph <insert v0_v1>{ ?s a cim:Terminal}}
	}

Aggre. delta	SELECT (count(?s) as ?total) where {
	graph <V0 >{?s a cim:PowerTransformer}
	minus {graph <delete v0_v4 >{ ?s a cim:PowerTransformer}}
	union {graph <insert v0_v4>{ ?s a cim:PowerTransformer}}
	}

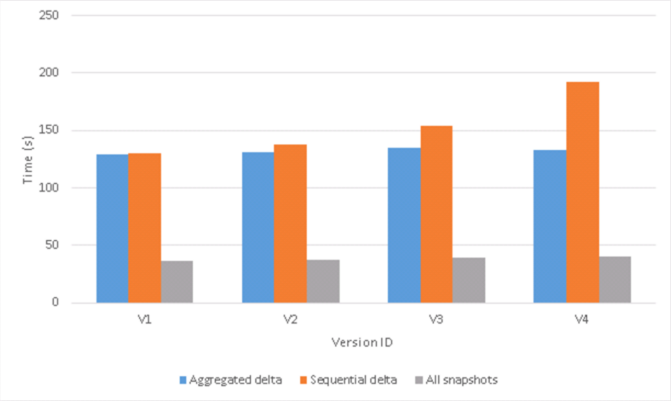


Figure 4. Computation time for the version management.

to extensively study the electrical measures of smart meters. In the province of Québec, there will be a total of 4 million smart meters installed. To make things even more challenging, one smart meter can produce a dozen measurements per customer in a short period of time (approximately every 15 minutes). In fact, it can record dozens of values such as the voltage, the current and the energy consumption. Thus, the huge amount of data newly generated has to be computed in order to make it highly accessible and available for electrical engineers to conduct electrical studies.

One way to deal with the vast volume of data generated by smart meters is to store the time series in a specialized infrastructure. We used the PI historian by OSIsoft for the management of real-time data and events. The PI system is widely used in the power industry.

As stated previously, the data related to the power distribution network, such as the equipment connectivity, the equipment location and their characteristics is stored in the OSG semantic database. On the other hand, the measurement data is stored in the PI historian. The latter is a real-time data infrastructure solution that can capture, store and analyze real-time data. It intends to deal with a massive amount of data while being able to offer a good velocity. At IREQ, we use the PI historian as the main repository for time series such as electrical measurements.

To take advantage of PI scalability and OSG flexibility, it becomes important to bind the two technologies. In fact, OSG is not optimized to efficiently store time series, and PI has not an evolved inference engine as the semantic data base. Thus, the data of the distribution network has to remain in OSG and the measurements in PI.

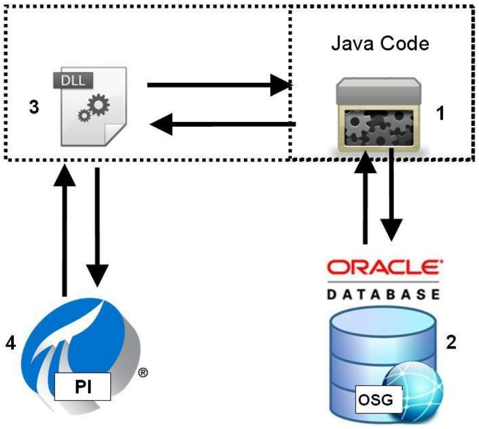


Figure 5. OSG and PI application

To bridge the gap between the distribution network data and measurement values, we have decided to develop an application that combines the power of both OSG and PI. In Figure 5, we present a high level view of our application composed by four modules, as follows.

The *Java Code* module (1) is responsible for processing SPARQL user queries. It can read and alter the user queries in order to detect which data are located in external data sources. We use the Dublin Core Metadata Initiative (DCMI) [14] and W3C Provenance meta-ontology [15] to inform where the data is located and how to extract it from external data sources. The *Oracle Semantic Graph Triplestore* (2) maintains the ontology snapshot and the meta-ontology of the distribution network. The *DLL module* (3) is a C# *DLL COM service* that behaves as web services. It receives custom queries from the Java code module and extracts values from the *PI module* (4). The latter receives a query from the DLL module with the parameters of the smart meters identifiers and the time duration of the measurement values to extract.

To help the reader understand how the two data sources are binded, we describe step by step the execution trace of our system. We show how the SPARQL user query is shared between the data sources and how the results are merged.
Step 1: The Java code receives an initial query from the user (see Table III).
Step 2: The original user query is altered and optional triples are added to detect if the data is located within external data sources via the property *cim:UsagePoint* (see Table III).
Step 3: The external data sources are described using the DCMI and W3C-Provenance ontology.
Step 4: A subset of the ontology extracted from OSG contains part of the user query results and the related metadata.
Step 5: The Java code reads and analyzes the metadata. The latter describes how to extract the external data and what to extract (i.e., the PI tag names). As a consequence, the Java code sends a compact query to the DLL component with the following parameters: the beginning date, the duration (in days), the measurement type (the voltage in the current example) and the PI tag names (see Table III). The PI tag

names are the unique identifiers for the smart meters and the time duration represents the beginning and the end-time markers of the measurement values.
Step 6: The DLL component queries the PI infrastructure and extracts the related measurement values.
Step 7: The DLL component formats the values and sends them back to the Java code. The data is sructured in order to reduce the amount of data transmitted between the DLL component and the Java code (see Table III).
Final Step 8: The Java code converts the received values into an RDF ontology and merges it with the resulting ontology of Step 2. The original user query is applied to the newly merged ontology and the result is returned to the user.

TABLE III. QUERY EXECUTION TRACES

Step	Query

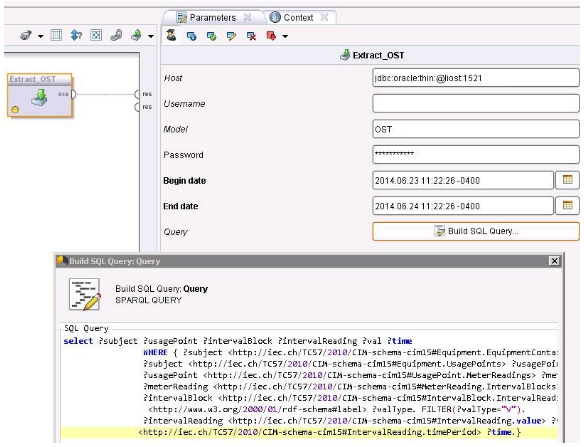
1	SELECT ?c ?v where {
	?c cim:MeterReading.IntervalBlocks ?iB.
	?iB cim:IntervalBlock.IntervalReadings ?iR;
	rdf:label "Volts".
	?iR cim:IntervalReading.value ?v.}

2	CONSTRUCT { ?c cim:UsagePoint ?p. ?v cim:UsagePoint ?p. }
	where { ?c cim:MeterReading.IntervalBlocks ?iB.
	?iB cim:IntervalBlock.IntervalReadings ?iR;
	rdf:label "Volts". ?iR cim:IntervalReading.value ?v.}
	OPTIONAL { ?c a cim:IntervalReading; cim:UsagePoint ?p. }
	OPTIONAL { ?v a cim:IntervalReading; cim:UsagePoint ?p. }
	}

5	query:[2013/01/01 12:00; 1; [V]; [smartMeter1,smartMeter2,...]] {
7	smartMeter1:[V;[2013/01/01 12:00,...,2013/01/02 12:00];[220, 219, ...]] smartMeter2:[V;[2013/01/01 12:00,...,2013/01/02 12:00];[219, 219, ...]]

We embedded our application with RapidMiner [16], a code free modern analytics platform that includes machine learning, data mining, text mining, predictive analytics and business analytics. According to the 15th annual KDnuggets Software Poll [17], released in 2014, RapidMiner remains the most-used free dataming tool. Data mining is the process of analyzing and turning large collections of data into useful knowledge. It can be seen as a natural evolution of information technology, where huge volumes of data accumulated in databases are analyzed, classified and characterized over time.

In Figure 6, we can see the visual interface of the RapidMiner extension. The *begin date* and the *end date* inform about the time duration and the query window allows the user to edit a SPARQL query. We tested our RapidMiner extension by detecting via a K-Nearest Neighbors (K-NN) all the outlier voltage measurements for a subset of customers during one week. The K-NN Global Anomaly Score assigns an anomaly score to each instance prior to the distance between the instance and a K number of neighbors. The higher is the score and the more likely the instance is an outlier. Visually, we can see in Figure 7 that some voltage measurements were detected by the K-NN algorithm as outliers; the bubbles size are proportional to the outlier score.



Extract_OST

Host: jdbc:oracle:thin:@liost:1521
Username:
Model: OST
Password: ***********
Begin date: 2014.06.23 11:22:26 -0400
End date: 2014.06.24 11:22:26 -0400
Query

Build SQL Query: Query

Build SQL Query: **Query**
SPARQL QUERY

SQL Query

```
select ?subject ?usagePoint ?intervalBlock ?intervalReading ?val ?time
               WHERE { ?subject <http://iec.ch/TC57/2010/CIM-schema-cim15#Equipment.EquipmentConta:
               ?subject <http://iec.ch/TC57/2010/CIM-schema-cim15#Equipment.UsagePoints> ?usagePoi
               ?usagePoint <http://iec.ch/TC57/2010/CIM-schema-cim15#UsagePoint.MeterReadings> ?me
               ?meterReading <http://iec.ch/TC57/2010/CIM-schema-cim15#MeterReading.IntervalBlocks:
               ?intervalBlock <http://iec.ch/TC57/2010/CIM-schema-cim15#IntervalBlock.IntervalRead:
                <http://www.w3.org/2000/01/rdf-schema#label> ?valType. FILTER(?valType="V").
               ?intervalReading <http://iec.ch/TC57/2010/CIM-schema-cim15#IntervalReading.value> ?
              <http://iec.ch/TC57/2010/CIM-schema-cim15#IntervalReading.timePeriod> ?time.}
```

Figure 6. RapidMiner extension

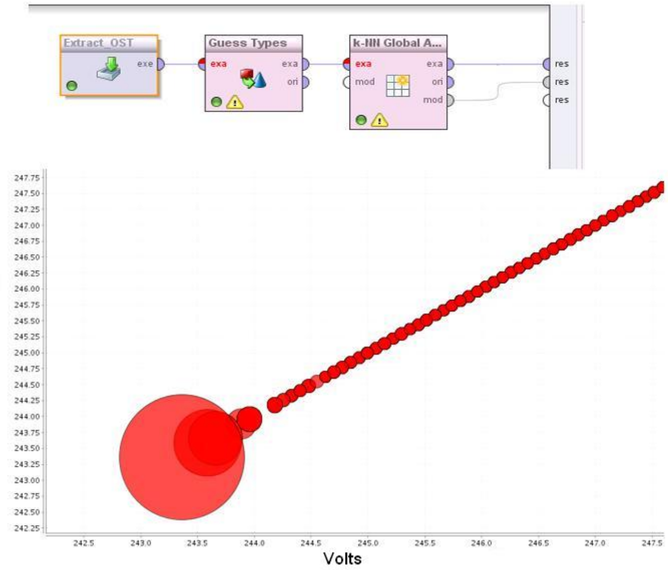


Figure 7. Voltage outliers detection

By combining RapidMiner to a temporal RDF ontology, our goal is to support advanced analytics process and to transform data into actionable insights. In fact, data mining tools offer methods and algorithms that help organizations analyzing large amount of data in order to extract valuable knowledge. For Hydro-Québec, analyzing complex situations and identifying the best solutions for forecasting demand, shaping customer usage patterns, preventing outages, optimizing assets and more is extremely valuable. Transforming a high volume of data into valuable decisions becomes a reality for any business that intends to succeed.

VI. Conclusion

The obtained results are promising and highlight the potential of using an ontology-based approach in an industrial context like electric power utilities. In addition to federate heterogeneous data sources across multiple enterprise systems, the semantic architecture proposed by IREQ goes well beyond that. The use of semantic technologies and versioning offers a new way of analyzing information. It gives a better idea on how information changes and evolves over the time. In fact, when heterogeneous data sources are adequately federated and versioned, it becomes possible to monitor the changes between the data sources and to take corrective actions when required.

Finally, the use of a common semantic model enables additional valuable information and knowledge to be inferred and extracted. The number of databases and information systems in use by utilities reveals the importance of a common language and semantic. This is particularly true for electric power utilities where information is growing fast and will continue to increase because of the introduction of smart grid technologies.

As future work, we are planning to improve the RDF versioning system in order to be efficient both in time and space needed for storage. We also plan to include No-SQL (Hadoop, Cassandra, etc.) data sources in our federated approach.

Acknowledgment

The authors would like to thank all the individuals who participated in the design and development of the architecture.

References

[1] A. Zinflou, M. Gaha, A. Bouffard, L. Vouligny, C. Langheit, and M. Viau, "Application of an ontology-based and rule-based model in electric power utilities," in 2013 IEEE Seventh International Conference on Semantic Computing, Irvine, CA, USA, September 16-18, 2013, 2013, pp. 405–411.

[2] M. Gaha, A. Zinflou, C. Langheit, A. Bouffard, M. Viau, and L. Vouligny, "An ontology-based reasoning approach for electric power utilities," in Web Reasoning and Rule Systems - 7th International Conference, RR 2013, Mannheim, Germany, July 27-29, 2013. Proceedings, 2013, pp. 95–108.

[3] W. W. Group. Cim primer for network models. [Online]. Available: http://cimug.ucaiug.org/default.aspx [retrieved: 04, 2015]

[4] N. Popitsch and B. Haslhofer, "Dsnotify - a solution for event detection and link maintenance in dynamic datasets." J. Web Sem., vol. 9, no. 3, 2011, pp. 266–283.

[5] T. Käfer, J. Umbrich, A. Hogan, and A. Polleres, "Dyldo: Towards a dynamic linked data observatory." in LDOW, ser. CEUR Workshop Proceedings, C. Bizer, T. Heath, T. Berners-Lee, and M. Hausenblas, Eds., vol. 937. CEUR-WS.org, 2012.

[6] M. Klein, D. Fensel, A. Kiryakov, and D. Ognyanov, "Ontology versioning and change detection on the web," in Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web, ser. Lecture Notes in Computer Science, A. Gómez-Pérez and V. Benjamins, Eds. Springer Berlin Heidelberg, 2002, vol. 2473, pp. 197–212.

[7] M. Völkel and T. Groza, "Semversion: Rdf-based ontology versioning system," in In Proceedings of the IADIS International Conference WWW/Internet 2006 (ICWI 2006), 2006. [VKZ + 05, 2006.

[8] N. F. Noy and M. A. Musen, "Promptdiff: A fixed-point algorithm for comparing ontology versions," in Eighteenth National Conference on Artificial Intelligence (AAAI-2002), 2002, pp. 744–750.

[9] Y. Tzitzikas, Y. Theoharis, and D. Andreou, "On storage policies for semantic web repositories that support versioning," in The Semantic Web: Research and Applications, ser. Lecture Notes in Computer Science, S. Bechhofer, M. Hauswirth, J. Hoffmann, and M. Koubarakis, Eds. Springer Berlin Heidelberg, 2008, vol. 5021, pp. 705–719.

[10] D.-H. IM, S.-W. Lee, and H.-J. Kim, "A version management framework for rdf triple stores," International Journal of Software Engineering and Knowledge Engineering, vol. 22, no. 01, 2012, pp. 85–106.

[11] R. Cyganiak. Accessing relational databases as virtual rdf graphs. [Online]. Available: http://d2rq.org/ [retrieved: 04, 2015]

[12] D. Zeginis, Y. Tzitzikas, and V. Christophides, "On the foundations of computing deltas between rdf models," in The Semantic Web, ser. Lecture Notes in Computer Science, K. Aberer, K.-S. Choi, N. Noy, D. Allemang, K.-I. Lee, L. Nixon, J. Golbeck, P. Mika, D. Maynard, R. Mizoguchi, G. Schreiber, and P. Cudré-Mauroux, Eds. Springer Berlin Heidelberg, 2007, vol. 4825, pp. 637–651.

[13] A. Seaborne and E. Prud'hommeaux, "SPARQL query language for RDF," W3C, W3C Recommendation, January 2008, http://www.w3.org/TR/2008/REC-rdf-sparql-query-20080115/.

[14] D. Core. Dublin core metadata initiative. [Online]. Available: http://dublincore.org/ [retrieved: 04, 2015]

[15] W. W. Group. An overview of the prov family of documents. [Online]. Available: http://www.w3.org/TR/prov-overview/ [retrieved: 04, 2015]

[16] RapidMnier. Rapidminer - analytics for anyone. [Online]. Available: https://rapidminer.com/ [retrieved: 04, 2015]

[17] G. Piatetsky. Kdnuggets 15th annual analytics, data mining, data science software poll: Rapidminer continues to lead. [Online]. Available: http://www.kdnuggets.com/2014/06/kdnuggets-annual-software-poll-rapidminer-continues-lead.html [retrieved: 04, 2015]

DataBearings: An Efficient Semantic Approach to Data Virtualization and Federation

Artem Katasonov
VTT Technical Research Centre of Finland
Tampere, Finland
e-mail: artem.katasonov@vtt.fi

Abstract—**In this paper, we describe and evaluate DataBearings, which is a lightweight platform for heterogeneous data integration from various sources such as databases, Web services, and files. DataBearings is based on an efficient semantic data virtualization and federation mechanism. We demonstrate that DataBearings is as fast as non-semantic data integration solutions such as Denodo Platform, making it the first practical semantic alternative to those, given that the comparable semantic solutions such as Virtuoso and TopBraid Composer fall well behind in terms of their run-time performance. We also demonstrate that DataBearings is very lightweight, as well as provides some unique functional features allowing easier and cheaper development and maintenance of data integration systems.**

Keywords–semantic web; enterprise information integration; data virtualization; data federation; internet of things

I. INTRODUCTION

Enterprises own an ever-growing number of databases with heterogeneous data originating from different business functions or processes. Emerging Internet of Things technologies (wireless sensors, etc.) enable enterprises to collect a variety of real-time data from the physical world, pushing the number of heterogeneous datasets even further. In addition, due to globalization and the pervasiveness of the Internet, different supply chains are increasingly integrated with each other and transforming into supply networks, requiring the information systems of different enterprises to work together, with this issue being increasingly significant not only for large scale enterprises but for companies of all sizes [1]. In other words, data relevant to an enterprise operation are often found not only in in-house databases but also in external data sources, which can be the business partners' sources (usually exposed as Web services) or even Open Data sources on the Internet. In the market, there is a great need for novel applications and better capability to provide services to customers in order to differentiate and compete. As a result, enterprises are seeking possibilities to exploit ever-growing and diverse data efficiently and dynamically to provide new and better services.

Several approaches to tackling the data integration problem have been developed, including integrated packages (e.g., SAP), messaging (e.g., WS-* services), data warehouses (also known as Extract-Transform-Load, ETL), and the Enterprise Information Integration (EII) approach [2, sec.7]. The two former approaches require implementing a custom software adapter or wrapper for each constituent data source, while the two latter approaches aim at providing a generic platform which can be configured for a particular integration case without a programming effort.

The vision underlying EII is to provide tools for integrating data from multiple sources without having to first load all the data into a central warehouse [2, sec.1]. The two central problems in EII are data virtualization and data federation. The former is about accessing data without requiring knowledge of how they are formatted or where they are physically located. The latter is about retrieving data from multiple non-contiguous data sources with a single query, even if the constituent sources are heterogeneous. EII generalizes on the principles of federated databases [3], that is, it involves creating a unified data model that encompasses the schemas of participating data sources. Users or applications formulate their queries in terms of this unified model, and each query is automatically reformulated into one or more queries to the data sources.

Data virtualization provides the business benefits of reducing the integration costs by allowing leveraging existing data sources in new ways without data replication or software development expenses, enabling new applications on the intersection of existing data sources, including external ones, as well as access to live data. When considering integration with external data sources, especially when their interfaces constantly and independently evolve, virtualization may be the only approach viable. Replicating all external data into own warehouse may just not be possible, while hard-coded adapters to external sources are expensive to maintain.

In this paper, we describe and evaluate DataBearings, a lightweight data integration platform that is based on an efficient semantic data virtualization and federation mechanism. The evaluation is done comparatively to three commercial data integration products, non-semantic Denodo Platform by Denodo Technologies, and semantic Virtuoso by OpenLink Software and TopBraid Composer by TopQuadrant. This evaluation is concerned with the run time performance, memory footprint, as well as virtualization-related functional features.

Denodo is a leading tool in data virtualization. It is based around the relational data model. Both Virtuoso and TopBraid are semantic solutions that enable virtualization of non-semantic data, in principle. Both realize it via a query-time ETL, where all source data is transformed into Resource Description Framework (RDF) and loaded into a temporary RDF storage, just to be read from there in the next step that is the execution of a SPARQL query. In contrast, DataBearings realizes a more pure data virtualization approach. It does not transform the source data into RDF, but rather searches for the answer to the target semantic query directly from non-semantic source data. To the best of our knowledge, DataBearings is the only semantic data virtualization solution available at present that is not based on ETL while also being capable of working with Web data and not only relational databases.

We demonstrate that DataBearings is very fast, running as fast or even faster than the non-semantic Denodo, and much faster than the comparable semantic solutions such as Virtuoso and TopBraid. Moreover, DataBearings is very lightweight, with a significantly smaller memory footprint than other systems. Finally, in addition to providing known evolution-related benefits of the semantic technology, DataBearings enables even easier and cheaper development and maintenance of data integration systems through a set of advanced features not available in Virtuoso, TopBraid, or Denodo.

The comparative evaluation of systems, reported in this paper, uses a very simple and understandable data integration case. A description of more complex and practical cases that were realized with DataBearings for the parking domain can be found in [4], [5].

The rest of the paper is structured as follows. Section II describes a simple data integration scenario that we use as a running example, as well as an evaluation case. Section III analyses the existing data virtualization approaches and example systems, both non-semantic and semantic, including how our running example is handled in these. Section IV describes the DataBearings platform, while Section V provides a comparative evaluation of DataBearings in terms of its run time performance. Finally, Section VI concludes the paper.

II. Running Example

As a running example, as well as a comparative evaluation case, we use the following simple data integration scenario that involves two data sources.

The Finnish state-owned railway monopoly, VR, publishes data on their trains via a feed at *http://188.117.35.14/TrainRSS/TrainService.svc/AllTrains ?showspeed=true*. The content is XML, with a record about a train found at XPath */rss/channel/item*. A record includes such elements as the train identifier, category, origin, destination, current location and speed. A specific complication comes from the fact that a location is given within a single XML element as a whitespace-separated string of a latitude and a longitude, e.g., *<georss:point>60.91658 26.17051</georss:point>*, instead of two separate elements for the latitude and the longitude. Henceforth, we refer to this data source as *DS1*.

The second data source, *DS2*, contains data, which we collected ourselves, on all major cities of Finland and the approximate bounding rectangles of their metro areas. The data is published in a simple comma-separated-values (CSV) format, with a row, e.g., *Tampere, 61.615563, 23.424657, 61.378988, 24.145634* (name, north, west, south, east).

The data integration task is then to extend the train records with an additional attribute containing the name of the city, in the metro area of which the train is currently located. That is, a join of two data sources is to be performed with the following condition: *ds1.lat>=ds2.south & ds1.lat<=ds2.north & ds1.lng>=ds2.west & ds1.lng<=ds2.east*.

III.Related Work

EII industry was born in late 90's and branded as a market category in 2002 [2, ch.6]. In the present, a number of big IT companies provide a data virtualization solution, with notable examples being IBM Cognos, Cisco Composite Information Server, and Denodo Platform by Denodo Technologies. Some products, e.g., IBM Cognos, only support databases but not Web services, while others, e.g., Denodo, are able to virtualize data from a variety of sources including relational and NoSQL databases, Web Services, files including CSV and MS Excel, and even some semi-structured and non-structured sources. Due to its rich feature set and the availability of an evaluation version (Denodo Express), we use in this paper Denodo Platform as a representative example of this category of products.

Denodo, as other traditional EII products mentioned above, works within the relational data model. This means that every constituent data source is represented by a relational view (a virtual relational database table) and all the following operations including data integration are performed as Structured Query Language (SQL) commands (Denodo defines an SQL extension called Virtual Query Language, VQL).

As to our running example, the *DS2* CSV data source on cities, after connecting it to Denodo, is straightforwardly represented by a virtual table view *ds2* with five columns. The *DS2* XML data source on trains is also automatically given a virtual table view *ds1* with seventeen columns, of which twelve correspond to the elements of a train record and the other five repeat for each record the values of the elements and attributes of encompassing *rss* and *channel* XML tags. As the current location of a train is given with a single whitespace-separated value, we need to define a secondary projection/selection view *p_ds1*, in which we define two new columns: 'lat' as *cast('float', substring(ds1.point, 0, instr(ds1.point, ' ')))* and 'lng' as *cast('float', substring(ds1.point, instr(ds1.point, ' ')+1))*, as well as preserve only the columns of interest. Finally, we define a join view *p_join*, with inner join conditions *p_ds1.lat>=ds2.south*, *p_ds1.lat<=ds2.north*, *p_ds1.lng>=ds2.west* and *p_ds1.lng<=ds2.east*. An execution of this view produces the result we seek.

While providing an efficient solution to our integration problem, the main disadvantage of this approach is low modifiability. This is due to the fact that the relational model always requires an explicit schema (even if it is automatically produced by Denodo) and the links between views are hard-coded to that schema via SQL constructs. In fact, in Denodo, after renaming a few output fields in *ds1*, we were just not able to fix the case without completely removing and re-doing *p_ds1* and then *p_join*.

Several authors in [2] argued for a need to exploit the benefits of the semantic technologies in EII. One reason for applying semantics to the data integration problem is a 'softer' nature of links in semantic models, as links and entities can be added or removed without breaking the rest of the model. This enables an agile and interactive evolution of data integration and integrated data analytics cases, with a faster return on investment [6].

In [2, ch.6], it was stated that none of the existing at the time EII tools used formal semantics, but predicted that EII will adopt the foundational technologies of the Semantic Web. Efforts towards semantic EII were reviewed later in [7], referencing, however, only a handful of research projects. Even at the time of writing this paper, to the best of our knowledge, the only available practical semantic data virtualization solutions are those that only support working with relational databases as virtual RDF graphs and cannot be used for access to Web data, such as D2RQ [8]. All solutions that support a variety of data source types rely on ETL instead, that is extraction of non-semantic data from their original data sources, explicit transformation of those data into RDF, and loading it into an RDF data warehouse. Notable commercial products include Data Unleashed Federator by Blue Slate Solutions, Virtuoso by OpenLink, and TopBraid Composer by TopQuadrant, with Linked Stream Middleware [9] and the ontology-based mediator in [10] deserving a mention on the research side.

We use in this paper Virtuoso and TopBraid as representative examples of the state-of-art in semantic data integration of heterogeneous data. Both come the closest to being data virtualization products as they support query-time ETL. That is, after a SPARQL query is received, they access relevant data sources, transform received data into RDF, load RDF into an in-memory RDF storage, and then execute the query on that storage. The repetition of the ETL step is avoided for static sources that did not change since the last query.

Let us use TopBraid to explain the specifics. As to our running example, the *DS2* CSV data source is mapped to

RDF via SemTables, which is TopBraid's own simple ontology consisting of three properties: sheetIndex, rowIndex, and columnIndex. With this approach, mapping data onto an arbitrary semantic structure is not supported, but only onto the most straightforward one: each data sheet corresponds to a class, every row to an instance of that class, and every column to a property of that class. Each data row in *DS2* is, thus, transformed into RDF (Turtle notation) as follows: *[a c:City] c:name "Tampere"; c:north 61.615563; c:west 23.424657; c:south 61.378988; c:east 24.145634*. *DS1* XML data source is mapped to RDF via SXML, which is also TopBraid's own ontology for describing the structure of XML documents. The resulting semantic structure is rather complex and not flexible, with train attributes represented as classes rather than properties, as follows (only the id and the location attributes included): *[] a vr:item; composite:child [a vr:guid; composite:child [sxml:text "IC10"]]; composite:child [a vr:point; composite:child [sxml:text "60.37992 25.09723"]]*.

After both data sources are mapped to RDF, the target integration case is realized via the multi-graph SPARQL query in Figure 1 (simplified here by selecting only the id and the location of a train plus omitting the full URIs of the two graphs). Submitting this query to the TopBraid's SPARQL endpoint produces the result we seek.

```
SELECT *
WHERE {
  GRAPH <ds1> {
    [ ] a vr:item;
      composite:child [a vr:guid; composite:child [sxml:text ?id]];
      composite:child [a vr:point; composite:child [sxml:text ?loc]].
  }.
  BIND (xsd:decimal(strbefore(?loc,’ ’)) as ?lat) .
  BIND (xsd:decimal(strafter(?loc,’ ’)) as ?lng).
  GRAPH <ds2> {
    [ ] c:name ?name; c:north ?n; c:south ?s; c:west ?w; c:east ?e
  }.
  BIND (xsd:decimal(?n) as ?no). BIND (xsd:decimal(?s) as ?so).
  BIND (xsd:decimal(?w) as ?we). BIND (xsd:decimal(?e) as ?ea).
  FILTER (?lat<=?no && ?lat>=?so && ?lng>=?we && ?lng<=?ea).
}
```

Figure 1. SPARQL query for the running example.

The main disadvantages of this approach are low performance and scalability (due to the nature of ETL) and a need to mix all processing and integration steps in a single SPARQL query. Note the explicit instructions for splitting the location into the latitude and the longitude, which are now part of the final query, while in the case of a relational tool like Denodo hidden into an intermediary projection view. For data statically residing in the TopBraid RDF storage, one could implement this step with inference rules, but this option is not available for data loaded on-demand from external non-semantic sources. The need to explicitly address the graphs corresponding to each data source is also a disadvantage as it does not allow protecting a user from data distribution details.

IV. DataBearings Approach

To the best of our knowledge, DataBearings is the only available at present semantic solution for data integration which is not based on an explicit extract-transform-load of data as RDF while also being capable of working with Web data and not only relational databases. Another distinctive feature is that, instead of restricting itself to the capabilities of standard semantic technologies such as RDF and SPARQL, DataBearings uses a more expressive and powerful data model, Tim Berners-Lee's Notation3 (N3) [11]. N3 can be easiest explained as RDF with nesting. Each N3 statement is necessarily an RDF triple, but the subject and/or object of it are allowed to be nested N3 models containing other statements (see Figure 3 for an example of N3 data). An important convention is that only statements at the top level are treated as facts, while statements in a nested model are considered only in the context set by the containing statement.

DataBearings features a fast in-memory N3 data storage, which can contain RDF or N3 data, data source annotations, as well as production rules and other imperative constructs in N3-based Semantic Agent Programming Language (S-APL). S-APL is developed around the central idea of N3Logic [12], that is to have N3 as a single data model for all of data, queries, and rules. However, S-APL drops the monotonicity assumption of N3Logic and, similarly to SPARQL, includes constructs for negation, solutions aggregation, as well as for facts removal. S-APL was introduced in [13] and later formalized in [14].

An overview of the DataBearings' semantic data virtualization and federation approach is given in Figure 2. In [5], we provided a detailed description of how this approach is realized exploiting the capabilities of S-APL. In this paper, we focus rather on the practical aspects and benefits of using DataBearings for implementing data integration cases.

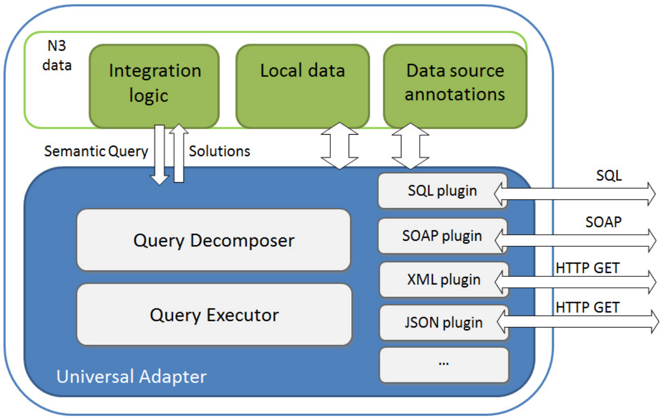


Figure 2. Semantic data virtualization in DataBearings.

Not unlike other data virtualization systems, DataBearings features a central component, we refer to as *Universal Adapter*, with dynamically loaded adapter plugins for different types of data sources. DataBearings currently comes with plugins for SQL databases, SOAP Web services, XML/JSON/CSV Web services or local files, as well as MS Excel files, and provides an API for developing additional plugins. A plugin is instantiated using an explicit N3-based *data source annotation*, an example of which, for *DS1* from our running example, is given in Figure 3. Such an annotation specifies the class of the plugin (*o:type*), class-specific connection parameters (*o:service*), class-specific data syntax (*d:tree* in *o:semantics*), data mapping to an ontology (the rest of *o:semantics*), an applicability-to-query pattern (*o:getPattern*) and, optionally, an applicability precondition (*o:precondition*). Instantiated adapter plugins that act as ontological virtualizations of data sources we refer to as *ontonuts*, a concept we first introduced in [15].

The data source annotation for *DS2* is done in a similar fashion. It uses sapl.shared.eii.CsvOntonut plugin, describes the syntax via d:table, d:row, and d:value properties and maps data to the following semantic structure: *[a ex:City] ex:hasName ?name; ex:hasBounds [ex:north ?n; ex:south ?s; ex:west ?w; ex:east ?e]*.

This data source mapping approach alone offers a number of advantages over the rigidness of mapping in existing

```
ex:VR_Ontonut a o:Ontonut
  ; o:type ”sapl.shared.eii.XmlOntonut”
  ; o:service [ o:uri ”http://%%ip%%/TrainRSS/...” ]
  ; o:precondition { ex:VR ex:hasIP ?ip }
  ; o:semantics {
     {
       * d:tree { * d:row {
          [ d:element ”rss”] d:branch {
            [ d:element ”channel” ] d:branch {
               [ d:element ”item” ] d:branch {
                 [ d:element ”guid”] d:branch {* d:value ?id}.
                 [ d:element ”point”] d:branch {* d:value ?loc}.
          } } } }.
          ?lat s:expression ”substring(?loc,0,indexOf(?loc,’ ’))”.
          ?lng s:expression ”substring(?loc,indexOf(?loc,’ ’)+1)”.
     } => {
       [a ex:Train] ex:hasID ?id; ex:hasLocation [ex:lat ?lat; ex:lng ?lng]
     }
  }
  ; o:getPattern { * a ex:Train } .
```

Figure 3. A data source annotation for the running example.

products like TopBraid, Virtuoso, or Denodo (see Section III):

- The explicit variable-based mapping allows data to be mapped to arbitrary semantic structures as dictated by target ontologies. This is in contrast to being forced to deal with over-simplified (for *DS2*) and over-complex (for *DS1*) temporary semantic structures, created just for the integration job, in TopBraid or Virtuoso.
- Transformations like splitting a location into latitude and longitude can be handled already at the data source mapping level. Note the two *s:expression* operations in Figure 3. This, again, allows mapping data to existing ontologies, as well as in contrast to having to carry these operations into the final SPARQL query in TopBraid or handling them in an intermediary projection view in relational systems like Denodo.
- Data source requests can be parametrized with values obtained from local data (via a precondition as in Figure 3 or via a local starter query, see below). This is in contrast to always having to specify the URIs statically in all of TopBraid, Virtuoso, and Denodo, even while most practical cases require parametrization.

After providing the data source annotations, the S-APL production rule that obtains the result we seek in our running example is as in Figure 4. As can be seen from Figure 4, an additional advantage of DataBearings is that, unlike in TopBraid, a data user is completely insulated from the data distribution details. A person or system specifying a query or a production rule as above does not need to know whether all needed data is found from a single data source or is distributed among two sources. If the situation changes in this regard (data sources are combined or split), only the ontonuts' definitions have to be updated while the queries and rules are not affected.

```
{
  [a ex:Train] ex:hasID ?id; ex:hasLocation [ex:lat ?lat; ex:lng ?lng].
  [a ex:City] ex:hasName ?name; ex:hasBounds
     [ ex:north ?n; ex:south ?s; ex:west ?w; ex:east ?e ].
  ?lat <= ?n. ?lat >= ?s. ?lng >= ?w. ?lat <= ?e.
} => { ... }
```

Figure 4. S-APL query for the running example.

The Query Decomposer part of the Universal Adapter analyses a query found in the head of a production rule against data source annotations, pre-selected using their applicability-to-query patterns (*o:getPattern*). Based on this analysis, all query statements are split into the following groups:

1) Statements covered by one or more ontonuts.
2) Statements which are covered by an ontonut, but could not be handled by that ontonut (e.g., a filter on a property value can always be handled by an SQL source, but by a Web service only if its request interface includes a corresponding parameter).
3) Inter-ontonut join conditions.
4) Local starters: statements not matching any ontonut's semantics that will be run as a query against the local data at the beginning of the execution. The obtained solutions may be used by ontonuts for parametrization.
5) Local join conditions: explicit conditions for performing join of ontonuts-produced solutions with solutions obtained at the beginning via local starters.
6) Local finalizers: statements that will be handled at the end of the execution. These are either solution aggregators (count/min/max/sum) or selection statements (from local data) that depend on variable values produced in ontonuts.

The Query Executor part of the Universal Adapter performs a query evaluation process, in which all the involved statements are handled in the following order: (1) local starters, (2) ontonuts' preconditions, (3) covered statements combined and translated into ontonut-specific form, e.g., SQL, SOAP, HTTP GET, (4) ontonuts' post-processing operations (e.g., splitting a location into latitude and longitude), (5) covered but not handled statements, (6) join conditions (also implicit join by a common variable value is supported, as well as the union operation), (7) local join conditions (again, including implicit), (8) local finalizers. Note that, in this process, at no point external non-semantic data is transformed into RDF. Rather, a semantic query is answered on the combination of external non-semantic and local semantic data. The step 3 is based on a relevant sub-query transformation, not data transformation. This step outputs directly a set of solutions, i.e., a list of variable-value mappings. For SQL sources, this step actually involves translating an S-APL sub-query into SQL. For simple Web services like in our running example, this step involves matching the sub-query with the data source syntax, requesting data, and then picking up relevant values from data and assigning them to variables as needed.

The simple example in Figure 4 does not include any local or not-handled statements, but only ontonut-covered selection statements (group 1) and inter-ontonut join statements (group 3). However, local statements appear in most practical integration cases. Based on a specific question in hand, or the current situation, some initial solutions (via local starters) may need to be obtained and used to determine (via preconditions) what exactly external data sources have to be contacted. Solutions from external sources may need to be filtered locally if the corresponding source cannot do it (covered but not handled statements). Finally, after solutions from external sources are obtained and joined or unionized, one may still want to extend them with extra attributes from the local data or to group solutions by a value and, e.g., count. An ability to flexibly combine virtualized and local data, and, thus, handle these practical cases, is a powerful feature of DataBearings giving it an advantage over most other data integration solutions, both semantic and non-semantic.

V. Evaluation

A. Integration run-time

In this section, we report on an evaluation of the performance of DataBearings, in terms of the integration run-time, in comparison to related commercial integration products, namely non-semantic Denodo 5.5 and semantic Virtuoso 7.2, and TopBraid 4.6 (see Section III).

For this evaluation, we created files with snapshots of data from *DS1* and *DS2* data sources of our running scenario (see Section II). The *DS1* snapshot contains 75 train records, while *DS2* snapshot contains 25 city records. This gives 1875 join pairings to examine and results in 27 solutions (trains currently in a city area). Then, we created copies of the *DS1* snapshot and, by copy-paste of existing data, increased the number of records in each by a factor from 2 to 200. The largest file thus contains 15000 train records encoded in 4.5 megabytes of XML content, and results in 375000 join pairings and 5400 solutions. The same *DS2* snapshot was used in all cases.

Denodo run-times are obtained from its execution trace view. For TopBraid, the integration SPARQL query was used as a SPIN framework inference rule, because the execution times of such rules are reported in TopBraid's SPIN statistics view. To check whether SPIN results in a significant additional overhead, we were also submitting the query directly via the TopBraid's SPARQL endpoint and observed the response time (time to first byte) in Chrome browser's development tools. To avoid an overhead created by formatting a large response, the SPARQL query was modified to return only the number of results instead of the results themselves. We did not observe any significant deviation between such a response time and the corresponding SPIN statistics number for any of the input file sizes, and report the SPIN statistics numbers here. Virtuoso does not seem to have performance self-reporting, so we measured its SPARQL endpoint's response times, same way as described above for TopBraid.

Both TopBraid and Virtuoso use query-time ETL, that is, when receiving a SPARQL query, they access and transform input data into RDF, load it into temporary RDF storage, and then run the query. If the files, however, did not change since the last query, the ETL step is skipped. Therefore, we recorded separately also the run-time of TopBraid and Virtuoso on previously loaded data, and report these numbers as 'TopBraid (QL)' and 'Virtuoso (QL)'. This case only involves executing a SPARQL query on the RDF storage, but requires extracting all trains and all cities and then doing the cross-join.

All experiments were performed on the same Windows 7 PC with 2.3 GHz CPU and 4GB RAM. All the run-time numbers reported are averages over 10 execution runs. Table I and Figure 5 provide the results. The columns correspond to different replication factors of the *DS1* snapshot. Run-times are reported in milliseconds.

TABLE I. INTEGRATION TIME FOR THE RUNNING SCENARIO

	1	10	20	50	100	200
DataBearings	14	42	61	140	265	552
Denodo	41	66	78	178	345	555
TopBraid	124	1322	3892	21212	79284	305832
TopBraid (QL)	78	537	1042	2631	5277	10554
Virtuoso	340	1842	2999	7359	16559	35152
Virtuoso (QL)	25	56	80	156	344	708

As can be seen, DataBearings consistently demonstrates a performance very similar to that of non-semantic Denodo, with a roughly-linear increase in the run-time with the data size growth. In fact, DataBearings even outperformed Denodo in all our test cases, with a bigger gain for smaller data sizes, up to three times for the smallest (original) data. Note that this is achieved even given all the overhead in DataBearings created by flexible query decomposition and match-making to the data sources.

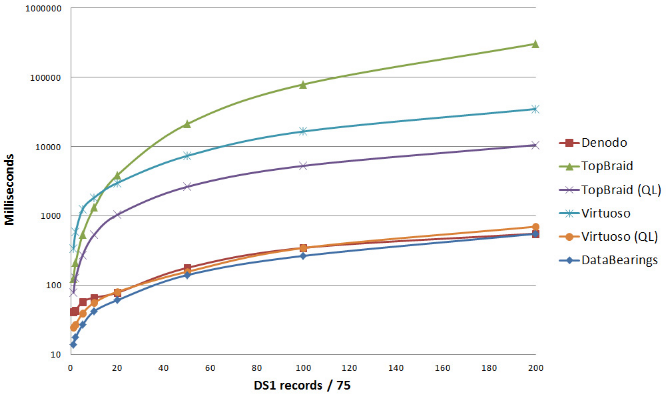


Figure 5. Integration time for the running scenario.

The biggest of the test jobs (15000 train records) is handled by both Denodo and DataBearings in just over 0.5 seconds. On the other hand, Virtuoso needs 35 seconds to do the same job, while TopBraid is out of hand with 5.1 minutes. Note the logarithmic scale in the figure. Considering SPARQL performance on already pre-loaded data, TopBraid still needs 10 seconds with the largest data set, which is surprisingly poor. Virtuoso, however, needs just 0.7 seconds, which is comparable to the total performance of DataBearings and Denodo, but can work only in a static data case.

B. Memory footprint

In addition to measuring the run-time performance, we also measured the memory footprints of DataBearings, Denodo, and TopBraid when executing our running example. These three systems are Java-based, which gives us a possibility to precisely measure their footprints. Virtuoso was excluded from this comparison as it is a native Windows application. We can only say that its total memory footprint, as can be observed in the Windows resource monitor, was always rather significant during the tests, with values in the range of 200 to 800 MB.

The total memory footprint includes permanent generation memory (the program code), which is constant regardless of the handled data size, as well as allocated memory (the heap) which grows with the handled data size. Table II and Figure 6 provide the results, in MB. The first column in the Table II shows the permanent generation footprint alone, while the rest of columns are sums of permanent generation and allocated footprints. As before, all the numbers reported are averages over 10 execution runs.

TABLE II. MEMORY FOOTPRINT FOR THE RUNNING SCENARIO

	perm	1	10	20	50	100	200
DataBearings	10	17	26	32	48	84	118
DataBearings (GC)	10	15	20	21	41	60	89
Denodo	68	120	150	183	280	267	280
TopBraid	72	241	648	602	622	674	738

The permanent generation footprint of DataBearings was measured using Java VisualVM tool, which is a part of the Java Development Kit. The allocated memory footprint was measured via calling java.lang.Runtime's *totalMemory()-freeMemory()* from code, for a better precision. Such a reading was performed in multiple points of the code and the maximum value was taken. The memory footprints (both permanent generation and allocated) of Denodo and TopBraid were measured using Java VisualVM. Note that we left the maximum memory settings of Denodo and TopBraid as default in these systems, which affects the point at which the Java automatic garbage collection starts (visible in the figure).

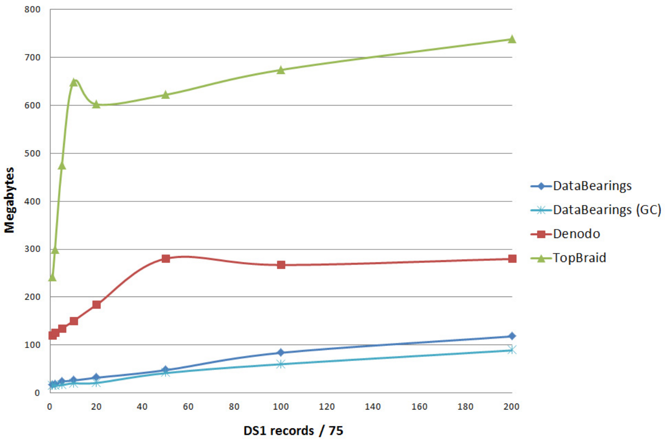


Figure 6. Memory footprint for the running scenario.

As can be seen in Table II, the DataBearings' code (its permanent generation footprint) is seven times lighter than that of Denodo or TopBraid. Also, the total memory footprint of DataBearings was significantly lower than the total footprints of Denodo and TopBraid, in all our experiments.

To further demonstrate the light weight of DataBearings, we used it as a library in an Android application and made a mobile phone to execute the integration job from our running example. The experiments were conducted on a Nexus 5 phone. Table III presents the results, while comparing them to the numbers obtained on a PC (as in Table I).

TABLE III. INTEGRATION TIME ON ANDROID PHONE

	1	2	5	10	20	50	100	200
PC	14	18	27	42	61	140	265	552
Nexus 5	72	122	283	501	1079	2605	5313	10625

Obviously, data integration jobs are better left to be performed on servers. Yet, using DataBearings, small data volumes can be integrated even locally within a smartphone application. So, 1500 trains (factor of 20) x 25 cities are handled in just over a second, which appears to be still an acceptable time for a user to wait.

VI.CONCLUSIONS

Supported by the comparative evaluation presented in this paper, we claim that, to the best of our knowledge, DataBearings is (1) the only semantic data virtualization solution available at present, which is not emulating virtualization via query-time ETL while capable of working with Web data and not only relational databases, (2) the only semantic solution to the data integration problem that is as fast as non-semantic ones, and (3) the only data integration solution, semantic or not, that is so lightweight that it can be run on a smartphone.

Being a semantic solution, DataBearings offers a possibility to exploit the evolution-related benefits of the semantic technology. In addition, DataBearings enables even easier and cheaper development and maintenance of data integration systems through a set of advanced features not available in other semantic systems. These include the ability to map data to arbitrary semantic structures as dictated by target ontologies, the ability to perform source data transformations prior to mapping to an ontology, the ability to parametrize data source requests, as well as the ability to flexibly combine virtualized and local data.

In this paper, we did not touch some other advanced features of DataBearings that go well beyond capabilities of existing data integration systems. These include a support for federated data updates (i.e., write not only read), as well as for abstraction of virtualized data. A discussion of these DataBearings' features can be found in [5].

An interesting result in our experiments was that Virtuoso and TopBraid performed worse than DataBearings even when running the job on already transformed into RDF and preloaded data. This indicates that, even when dealing with static data, when it would be possible to transform all data into semantic form and store as RDF, it may still be not a good idea if integration (join) operations cannot be also performed statically, but have to be done upon a request. Thus, even in such cases, pure virtualization, as in DataBearings, may be the most efficient and thus recommended choice.

ACKNOWLEDGEMENTS

This work is performed in the DIGILE Internet of Things project, which is supported by TEKES.

REFERENCES

[1] L. D. Xu, "Enterprise systems: State-of-the-art and future trends," IEEE Trans. Industrial Informatics, vol. 7, no. 4, 2011, pp. 630–640.

[2] A. Y. Halevy, N. Ashish, D. Bitton, M. Carey, D. Draper, J. Pollock, A. Rosenthal, and V. Sikka, "Enterprise information integration: Successes, challenges and controversies," in Proc. ACM SIGMOD International Conference on Management of Data. ACM, 2005, pp. 778–787.

[3] A. P. Sheth and J. A. Larson, "Federated database systems for managing distributed, heterogeneous, and autonomous databases," ACM Computing Surveys, vol. 22, no. 3, 1990, pp. 183–236.

[4] A. Lattunen, A. Katasonov, and T. Koivuniemi, "Flexible parking data management across enterprise and beyond," in Proc. ITS World Congress, Detroit, 2014.

[5] A. Katasonov and A. Lattunen, "A semantic approach to enterprise information integration," in Proc. 8th IEEE Conf. Semantic Computing, 2014, pp. 219–226.

[6] Introducing Data Unleashed: An Overview of Data Federation Agility, Blue Slate Solutions, 2014, online: http://www.blueslate.net/Dave/DataUnleashedIntroduction_OverviewOfDataFederationAgility/ [retrieved: June, 2015].

[7] J. Zhou, H. Yang, M. Wang, R. Zhang, T. Yue, S. Zhang, and R. Mo, "A survey of semantic enterprise information integration," in Proc. Intl. Conf. Information Sciences and Interaction Sciences (ICIS). IEEE, 2010, pp. 234–239.

[8] C. Bizer and A. Seaborne, "D2RQ – treating non-RDF databases as virtual RDF graphs," in Proc. 3rd International Semantic Web Conference, 2004.

[9] D. L. Phuoc, H. Q. Nguyen-Mau, J. X. Parreira, and M. Hauswirth, "A middleware framework for scalable management of linked streams," J. Web Semantics, vol. 16, 2012, pp. 42–51.

[10] K. Hribernik, C. Hans, C. Kramer, and K.-D. Thoben, "A service-oriented, semantic approach to data integration for an internet of things supporting autonomous cooperating logistics processes," in Architecting the Internet of Things, D. Uckelmann, M. Harrison, and F. Michahelles, Eds. Springer, 2011, pp. 131–158.

[11] T. Berners-Lee, Notation 3: An RDF language for the Semantic Web, online: http://www.w3.org/DesignIssues/Notation3.html [retrieved: June, 2015].

[12] T. Berners-Lee, D. Connolly, L. Kagal, Y. Scharf, and J. Hendler, "N3logic: A logical framework for the world wide web," Theory Pract. Log. Program., vol. 8, no. 3, 2008, pp. 249–269.

[13] A. Katasonov and V. Terziyan, "Semantic agent programming language (S-APL): A middleware platform for the semantic web," in Proc. 2nd IEEE Conf. Semantic Computing, 2008, pp. 504–511.

[14] M. Cochez, "Semantic agent programming language: Use and formalization," Master's thesis, University of Jyväskylä, 2012.

[15] S. Nikitin, A. Katasonov, and V. Terziyan, "Ontonuts: Reusable semantic components for multi-agent systems," in Proc. 5th Intl. Conference on Autonomic and Autonomous Systems. IEEE, 2009, pp. 200–207.

Semantic Location Models for Bioenergy and Biofuel Projects

Krishna Sapkota, Pathmeswaran Raju,
Will Byrne, Craig Chapman,
Centre for Knowledge Based Engineering,
Birmingham City University
Birmingham, UK
e-mail: {krishna.sapkota, path.raju, william.byrne, craig.chapman}@bcu.ac.uk

Lynsey Melville
Centre for Low Carbon Research
Birmingham City University
Birmingham, UK
e-mail: {lynsey.melville}@bcu.ac.uk

Abstract — **The five fundamentals for a successful bioenergy project are feedstock, technology, off-take, finance and help from experts. In order to realize the essence of these fundamentals, recently, many governments and organizations are providing Decision Support Tools (DST) to the experts and project developers. One of such tools is Location Model, which helps the developers to identify the prime location for a project. The model comprises the concepts, relations, logics, constants and equations related to bioenergy and location. The model is currently represented in non-semantic means, such as spreadsheets and programming code. Representing the knowledge in non-semantic format will make the model less reusable and extendable. In order to alleviate the issues, in this paper, we present semantic location models. In particular, we have leveraged the Semantic Web technologies to represent the knowledge about the bioenergy and biofuel plant location and inferred the equations and other values required for location related calculations. The results, observed in two INTERREG IVB projects, have been found encouraging.**

Keywords- bioenergy ontology; location model; semantic location model; ontology-based location model; biomass; bioenergy; renewable energy

I. INTRODUCTION

Bioenergy is one of the most dynamic and the fastest growing renewable energy and promises sustainable solutions to the depleting fossils fuels. Biomass, the solar energy stored in the materials derived from biological sources, is treated with some conversion technologies, and bioenergy is generated. Some examples of biomass are wood fuel, waste wood, energy crops, straw, waste and agricultural waste. Biomass is used as an input while generating bioenergy; therefore, in this paper, biomass is also called feedstock.

Many governments and organizations have recently shown efforts to realize the principle of five fundamentals in bioenergy projects. An example of such efforts is the development of Decision Support Tools (DST), which provides essential information to plant developers efficiently. One of such organizations is INTERREG IVB, which has funded two projects: BioenNW and EnAlgae.

The ongoing North West European projects: BioenNW [1] and EnAlgae [2] aim to facilitate the project developers to start a new bioenergy plant in the region and develop sustainable technologies for algal biomass production respectively. These projects aspire to increase the global share of renewable energy sources by 20% within the EU by 2020. In order to achieve this, they have to consider the steps towards the five fundamentals. One of the crucial step is to provide the farmers in the region with DSTs, which will help them make decisions [3]–[11]. The DSTs provide the answers to various decision-making queries, such as:

1. What is the best place to start a new bioenergy plant?
2. What kind and how much amount of biomass are available in a region?
3. What kind of technology is suitable for a plant in a region?
4. How much investment is needed and how long will it take to return the investment?
5. What are the logistic and other related costs involved?

Location Model is one of the DSTs in these two projects, which helps the developers to identify the prime location for a project. It is represented by considering various factors, such as the concepts, relations, logics, constants, and equations. It helps project developers to analyze the viability of a project by providing them with location information. In particular, it will provide a mechanism to calculate the equations related to the costs, incentives and distance associated with a project or a cultivation technology in a location. The equations and the calculations (evaluation of the equations) in the model are affected by various attributes, such as capacity of the project, incentives in the region, bioenergy conversion technologies used and the quality of the biomass. Providing the necessary information regarding a location to developers prior to the start of a project will help them to make an informed and confident decision about starting the project.

Previous studies have primarily concentrated on implementing non-semantic models, such as hard-coded programming languages or Excel spreadsheets. That is to say, the concepts, relationships, logics, constants and equations are all embedded in some native programming language. For example, in the BioenNW project, the model was represented in MATLAB functions; whereas, in EnAlgae, the spreadsheets have been equally used to represent the model. Representing the model in such a way poses some issues, such as the model being less shareable, less reusable and less extendable, and the data being inconsistent. To alleviate the issues, we propose semantic models for the location determination.

The semantic location model exploits Semantic Web technologies. Ontology can be used to represent the domain knowledge and check the consistency of the model. Since the vocabularies in the ontologies are reusable, and they are explicitly defined, this approach makes the model extensible. In other words, the ontology not only holds the data but also explicitly defines the concepts used for a particular purpose [12]. Using the existing definition of the concept in an ontology, engineers can extend the ontology by creating new concepts that conform to the existing knowledge. In the proposed approach, the ontological axioms and rules are used to define the entities, such as biomass, technology, incentive, tariff, capacity and project. The following are the key benefits of using the proposed approach:

1. Explicit concept definition: It defines the key concepts used in the economic model explicitly, which will make the ontology re-useable and extensible.
2. Consistent data: It provides a mechanism to check the consistency of the data in the model.
3. Inference: It identifies the implicit knowledge in the model by means of ontological inference mechanism.

The rest of the paper is organized as follows. Section II provides the detailed description of the ontology-based location model in a case study. Section III and IV provide the ontology based calculation and discussion respectively. The finding of this research is concluded in Section V.

II. THE ONTLOGY-BASED LOCATION MODEL

One of the objectives of the INTERREG IVB supported two projects BioenNW and EnAlgae is to provide DSTs to the developers in the North West of Europe. The BioenNW project aims to facilitate the farmers in the selected regions to start a new bioenergy plant. The regions have been selected in five countries: the United Kingdom, the Netherlands, Germany, Belgium and France. It has Business Support Centers (BSC) in each country for the regions in the country, where farmers get advice from bioenergy experts. These regions contain a unit of area called a cell, which is a square area, for example, 1KM square area is a cell. The DSTs provide the information about each cell, such as

1. how much it yields in a year,
2. what kind of biomass is available,
3. what regulatory guidelines are applicable in the region,
4. what incentives are being offered in the region.

The EnAlgae project aims to develop sustainable technologies for algal biomass production. The microalgae economic models are developed for various cultivation technologies, such as open pond, flat panel and tubular reactors while the economics of downstream processing is calculated for biodiesel, bioethanol dry milling and methane production.

The DSTs are Web-based tools, which can be accessed remotely by farmers, and encourage them to start a new plant. The DSTs in EnAlgae helps in making decisions about algae biomass production, from understanding the algae growth, cultivation, economics and life-cycle analysis to setting up plants and operating procedures. Some of the DSTs are map-based information tools, while others are dashboards and conversion pathways. Map-based information tools allow users to click on a map to see whether a location is suitable for a particular type of the bioenergy or algae plant. Dashboards provide calculations related costs and economics, such as how much investment is needed and how long it will take to return the investment back. One of the DSTs is the locations model.

The location model either identifies a best location for a plant (location unknown) or provides information about a selected location (location known). The model is presented in a map based web interface, where users interact with various input variables and analyze the output. The input variables in the interface are longitude and latitude of a geographic location, region in a country, amount of biomass needed in tons per annum, biomass type and biomass scenario. Based on these variables, the user is able to see whether there is sufficient biomass around the location. The biomass availability is displayed in a list of cells around the location with different density of the colors. A snapshot and description of the webpage is provided in Figure 7 in a latter section.

The location model is now represented in SW technologies. There are axioms and SWRL rules that define and infer various knowledge about the model. In particular, the reusable knowledge is separated from the calculations and queried with DL-Query and SPARQL in order to make the implicit facts explicit. Figure 1 shows an overall idea of the ontology based location model. The ontology represents the knowledge of the model; the axioms define the concepts and assert constraints; SWRL rules assert new knowledge in the ontology. The reasoners, such as Pellet help to deduce the implicit knowledge and make them explicit. The query engine with the DL-Query and SPARQL will identify the entities required for the location related calculations.

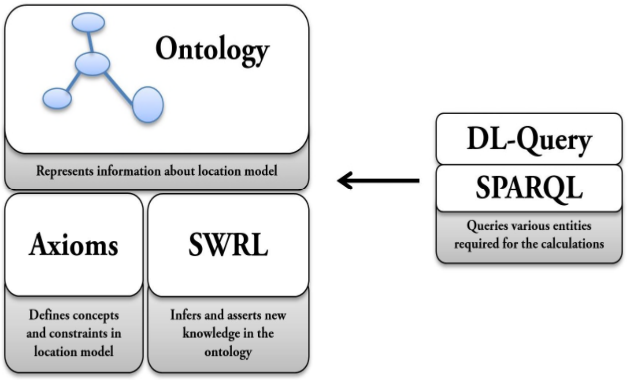

Ontology
Represents information about location model
Axioms
Defines concepts and constraints in location model
SWRL
Infers and asserts new knowledge in the ontology
DL-Query
SPARQL
Queries various entities required for the calculations

Figure 1. Semantic Web technologies used in the location model

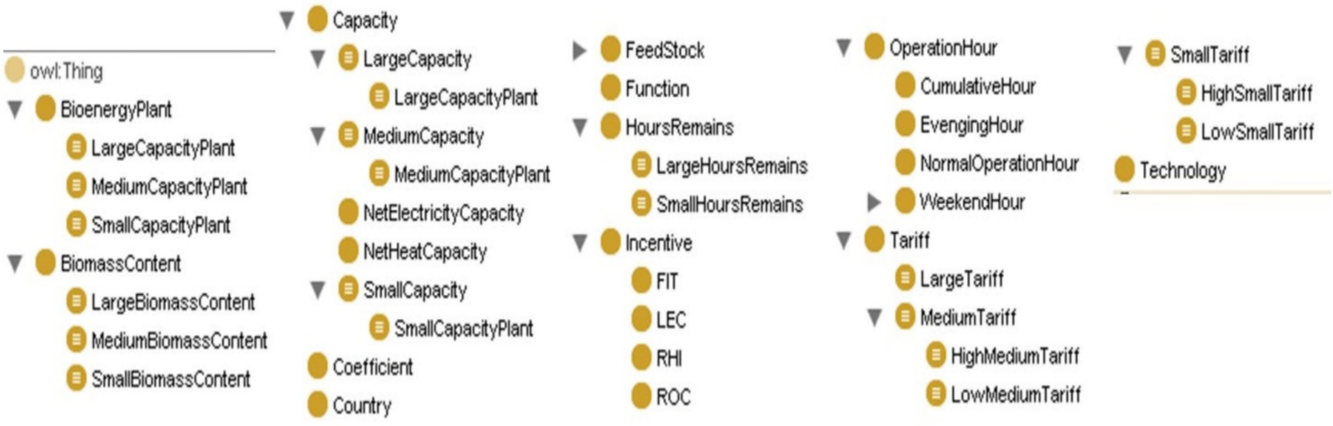


Figure 2. The classes in the location model ontology

Various tools and technologies have been deployed in order to realize the location model. Some of them are Protégé 4, Pellet [17], OWLAPI [18] with Pellet, and Jena [19]with Pellet. Protégé 4 is an ontology editor, and Pellet is a reasoner and is used as a plug-in in Protégé. OWL API with Pellet helps us to run DL-Queries in Java. In addition, Jena API with Pellet reasoner has also been used to run SPARQL queries.

Some classes of the location model ontology are depicted in Figure 2. The important constituents of the ontology are described in the following sections.

A. The Ontology

The location model ontology is created in Protégé 4 and OWL-DL. The experts in the domain have been consulted in order to understand the concepts and the relations. The consultation went through several iteration in order to verify the understanding and representation. The classes, properties, axioms, rule and queries involved in the ontology are described separately in the following sections.

1) Classes

The key classes of the ontology are Feedstock, BioenergyPlant, Technology, Country, Tariff, and Incentive. The Feedstock is a biomass that will be used as an input for heat and power generation. Technology is a process through which a Feedstock is treated in order to generate bioenergy. Depending upon BiomassContent, a feedstock can be treated with different technologies in order to generate energy efficiently. The Incentive and Tariff in a project are affected by other concepts, such as Technology, Country and Capacity of a BioenergyPlant. The other concepts are BiomassContent, Capacity, Coefficient, Function, HourRemains, and OperationHours.

2) Properties

The model exploits the properties and the values for the representation and inference. The important object properties are relate the classes mentioned above, such as BioenergyPlant, Capacity, Country and Technology with each other. For example, a BioenergyPlant is related with Country, Technology and Feedstock with properties locatedIn, hasTechnology and hasFeedstock respectfully. Figure 3 displays some properties in the ontology for the location model.

The values of the data-type properties are mostly double and play the key role in defining concepts and computing simple equations. One of the important data-type properties is hasDataValue, which is associated with many concepts, such as Capacity, BiomassContent and Coefficient. Some data-type properties are used to hold the results of the calculations. For example, hasQaQiValue holds the computed value of the quality assurance calculation.

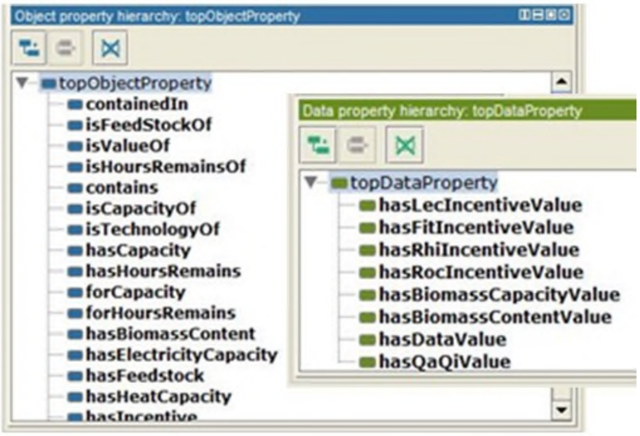


Figure 3. Snapshot showing properties in the location model ontology

3) Axioms

The axioms in the ontology are essential to define the concepts and constraints. In the ontology, most of the axioms contain data-type properties, particularly with double. For example, small medium and large capacity are defined as follows:

SmallCapacity:
Capacity and hasDataValue some double[<200]

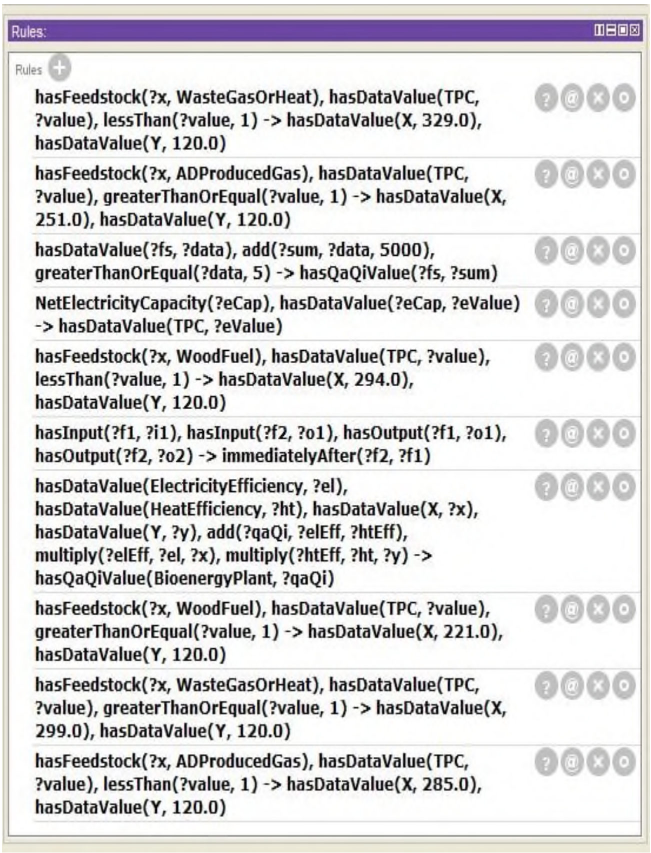
MediumCapacity:
Capacity and hasDataValue some double[>=200.0, <1000.0]

LargeCapacity:
Capacity and hasDataValue some double[>= 1000.0]

In these axioms, the small capacity is defined as a capacity with value less than 200, the medium capacity is a capacity with value greater than equal to 200 and less than 1000, and the large capacity is a capacity with value 1000 and more.

4) Rules

In location model ontology, the SWRL rules are used to assert more knowledge. There are some caveats using the rules. Since OWL and SWRL both do not support non-monotonicity, we cannot change the existing value in the ontology. However, if the value of a property is empty or not filled, it can assert a new value. If we add a new value to a property, it will not replace the existing one, but the property will have two values instead. In this ontology, most of the data-type properties are functional; therefore, any attempt to add a new value to a property that already has a value, will make the ontology inconsistent. Some rules used in the ontology are depicted in Figure 4.

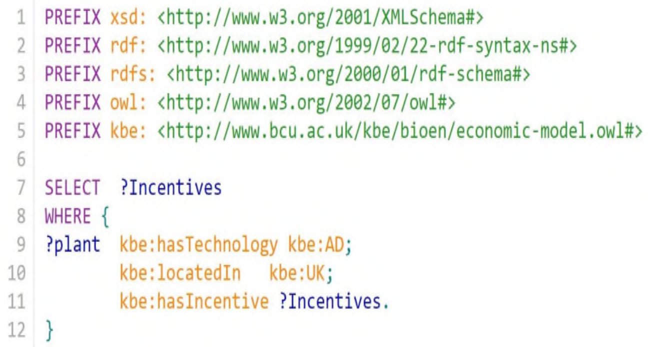


```
hasFeedstock(?x, WasteGasOrHeat), hasDataValue(TPC, ?value), lessThan(?value, 1) -> hasDataValue(X, 329.0), hasDataValue(Y, 120.0)
hasFeedstock(?x, ADProducedGas), hasDataValue(TPC, ?value), greaterThanOrEqual(?value, 1) -> hasDataValue(X, 251.0), hasDataValue(Y, 120.0)
hasDataValue(?fs, ?data), add(?sum, ?data, 5000), greaterThanOrEqual(?data, 5) -> hasQaQiValue(?fs, ?sum)
NetElectricityCapacity(?eCap), hasDataValue(?eCap, ?eValue) -> hasDataValue(TPC, ?eValue)
hasFeedstock(?x, WoodFuel), hasDataValue(TPC, ?value), lessThan(?value, 1) -> hasDataValue(X, 294.0), hasDataValue(Y, 120.0)
hasInput(?f1, ?i1), hasInput(?f2, ?o1), hasOutput(?f1, ?o1), hasOutput(?f2, ?o2) -> immediatelyAfter(?f2, ?f1)
hasDataValue(ElectricityEfficiency, ?el), hasDataValue(HeatEfficiency, ?ht), hasDataValue(X, ?x), hasDataValue(Y, ?y), add(?qaQi, ?elEff, ?htEff), multiply(?elEff, ?el, ?x), multiply(?htEff, ?ht, ?y) -> hasQaQiValue(BioenergyPlant, ?qaQi)
hasFeedstock(?x, WoodFuel), hasDataValue(TPC, ?value), greaterThanOrEqual(?value, 1) -> hasDataValue(X, 221.0), hasDataValue(Y, 120.0)
hasFeedstock(?x, WasteGasOrHeat), hasDataValue(TPC, ?value), greaterThanOrEqual(?value, 1) -> hasDataValue(X, 299.0), hasDataValue(Y, 120.0)
hasFeedstock(?x, ADProducedGas), hasDataValue(TPC, ?value), lessThan(?value, 1) -> hasDataValue(X, 285.0), hasDataValue(Y, 120.0)
```

Figure 4. Some SWRL rules in Protégé to assert new knowledge and calculate the quality assurance

5) SPARQL Queries

One of the query languages used to query the knowledge represented in the ontology is SPARQL. It will provide a mechanism to answer the queries related to the location calculations. For example, provided a set of conditions, such as technology and country, we can search for the suitable incentives applicable in a country. Figure 5 illustrates a SPARQL query for applicable incentives for AD plants in the UK. The result of this query will be RHI, LEC and FIT.



```
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX kbe: <http://www.bcu.ac.uk/kbe/bioen/economic-model.owl#>

SELECT  ?Incentives
WHERE {
?plant  kbe:hasTechnology kbe:AD;
        kbe:locatedIn   kbe:UK;
        kbe:hasIncentive ?Incentives.
}
```

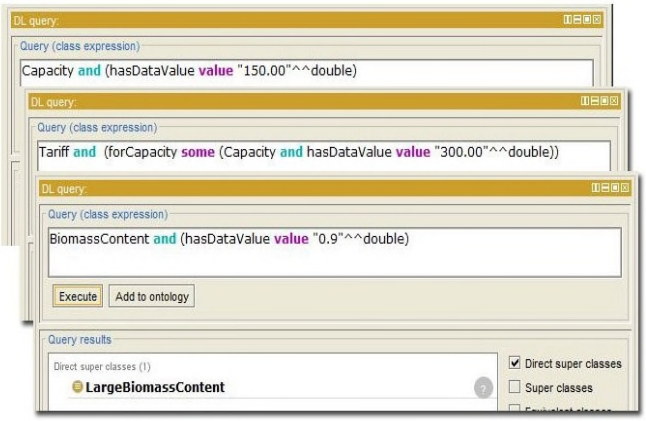
Figure 5. A SPARQL query in the location model ontology to find the applicable incentives for a bioenergy plant in the UK

6) *DL – Queries*

The DL-Queries identify the appropriate category of biomass content, tariff and incentives. For example, in Figure 6, the query asks, "what kind of biomass content is it if it has biomass content value 0.9?", and the query returns LargeBiomassContent as the answer.

In another example, the query asks, "What kind of tariff is applicable to a bioenergy plant if it has a capacity of 150.00?", and MediumTariff is returned as the answer.

Likewise, SmallCapacity is the answer to the query "what category of capacity does my bioenergy plant have if the capacity value is 150.00?".



```
Capacity and (hasDataValue value "150.00"^^double)
```

```
Tariff and  (forCapacity some (Capacity and hasDataValue value "300.00"^^double))
```

```
BiomassContent and (hasDataValue value "0.9"^^double)
```

```
LargeBiomassContent
```

Figure 6. Three DL-Queries in the location model ontology to determine small, medium and large biomass content

III. Ontology Based Calculation

The idea of ontology based location model is to separate the domain knowledge of the model from the programing code and calculations. The knowledge is represented in the ontology for the model, and the ontology will provide the crucial information to carry out the actual calculations. Once the answers or values are obtained, there will be further queries to obtain detailed information; in particular, many numerical values, required for the calculations, are generated by the ontology. Hence, it will allow separating the logics from calculations, and the calculations will be the only tasks carried out in the programming code.

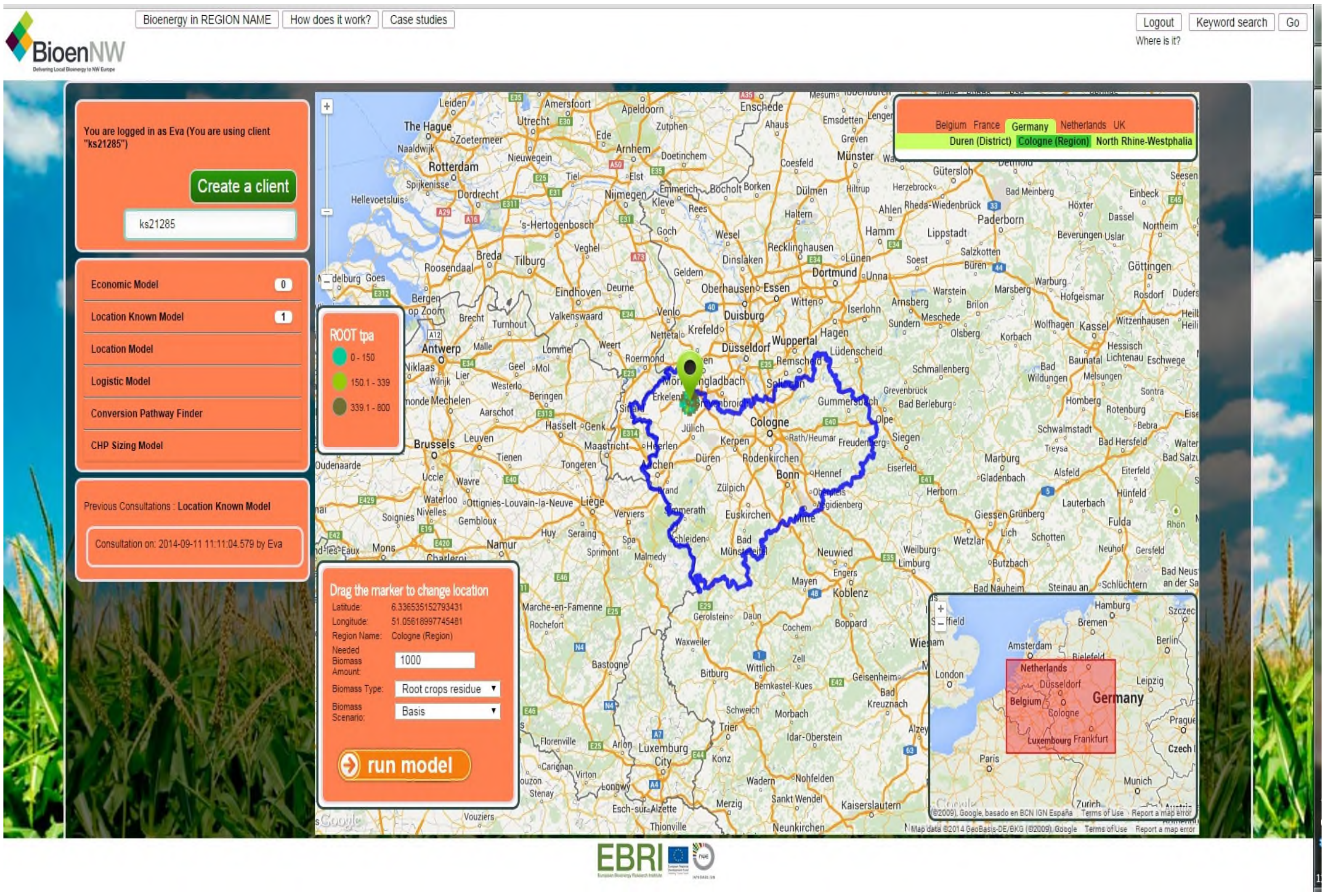
In the ontology, we can apply sequential inference rules to deduce the values required for the calculation. For example, in the rules in Figure 4, identify the QaQi values are identified by using a sequence of inferences. The values obtained from the ontology are computed in Java methods, and the values are sent to clients as RESTful services. The clients request the output, in this case in JSON format, to the server, and when they receive the output, they imbibe it in their implementation and provide the information – the output variables, in particular – to the users in a web page.

Figure 7 shows the map of Cologne region in Germany. The polygons around the pin in the region represent the cells containing the required biomass for the following requested input variables:

- Latitude = 6.336535
- Longitude = 51.056189
- Region Name = Cologne
- Needed Biomass = 1000 tons per annum
- Biomass Type = Root crops residue
- Biomass Scenario = Basis

The system also provides a mechanism to store the consultation containing the input and output variables under a client's name. The users, then, retrieve the saved results, compare and analyze them in order to get a better view of the potential project.

If a client chooses a place and places the pointer where there is not enough biomass of the requested type and amount, the system will suggest the client that there is not enough biomass available and displays the nearest places where the client might be interested to choose instead.



You are logged in as Eva (You are using client "ks21285")

ks21285

Previous Consultations : Location Known Model

Consultation on: 2014-09-11 11:11:04.579 by Eva

Drag the marker to change location

Latitude:	6.336535152793431
Longitude:	51.05618997745481
Region Name:	Cologne (Region)
Needed Biomass Amount:	1000
Biomass Type:	Root crops residue
Biomass Scenario:	Basis

Figure 7. Google map showing 1000 tons of root crop residue around a point in Cologne region

IV. Discussion

The results from the ontology-based location model have shown that the approach has a promising solution. The model is one of the DSTs for the bioenergy and algae plants, which helps the plant developers or the farmers to identify appropriate location for their projects. In the model, there are various incentives and tariffs based on various factors, such as location, feedstock and technology. The model needs to execute a series of calculations integrating all the information. Currently, the domain knowledge of the location model is hard coded with programming code, which entails the model with some limitations, for example the model being less explicit, less reusable, less sharable and less extendable. An alternative to these limitations is to separate the knowledge from the code and represent it in a logic based explicit format. One of such representations format is Ontology. Ontology is a part of SW technologies.

In this paper, we explained how the concepts, relations and logic behind the calculations could be represented semantically. SW technologies allow us to define the concepts, relations and their constraints, thus making the knowledge explicit, sharable, reusable and extendable.

V. Conclusion

In this paper, we described how the concepts, relationships and logics could be separated from the location model for bioenergy and biofuel projects, and represented them in an ontology, which made the model more shareable, reusable and extendible. Ontology allows an efficient mechanism to specify formal concepts with axioms and rules. Since the information in ontology is represented in description logic, there are well-known inference engines available to infer new knowledge from existing knowledge. By inferring the knowledge in the ontology for the location model, we generated the required information for calculations, such as the variables: incentives, tariffs and other attributes. This approach is useful for extending the model and checking whether the knowledge in the model is consistent.

In future work, we aim to integrate the bioenergy ontology, which is developed as part of the BioenNW project, into the ontology for the location model. If the time permits, we will also integrate MathML into our system, and infer the equations relevant to a particular scenario in such a way that they will be executed in a correct sequence.

VI. Acknowledgements

This study was undertaken as part of the bioenergy project BioenNW (http://bioenergy-nw.eu) and algae project EnAlgae (http://enalgae.eu). These projects are funded by the European Union INTERREG IVB program. The data needed for the location model is provided by the European Bioenergy Research Institute (EBRI), UK and Karlsruhe Institute of Technology (KIT), Institute for Technology Assessment and Systems Analysis (ITAS), Germany. The data for EnAlgae are provided by ACRRES, the Application Centre for Renewable Resources from Wageningen UR.

References

[1] Bioenergy Website, "About BioenNW," *About BioenNW*, 2015. [Online]. Available: http://bioenergy-nw.eu/about-bioennw/. [Accessed: 23-May-2015].

[2] EnAlgae Website, "About EnAlgae," *About Enalgae*, 2015. [Online]. Available: http://www.enalgae.eu/about-us.htm. [Accessed: 23-May-2015].

[3] C. P. Mitchell, "Development of decision support systems for bioenergy applications," in *Biomass and Bioenergy*, 2000, vol. 18, pp. 265–278.

[4] T. Buchholz, E. Rametsteiner, T. A. Volk, and V. A. Luzadis, "Multi Criteria Analysis for bioenergy systems assessments," *Energy Policy*, vol. 37, pp. 484–495, 2009.

[5] K. Sternberg, M.-M. Brinker, P. Raju, K. Sapkota, C. Chapman, and L. Melville, "Who Does What ? The Enalgae Map on Algae Activities in North West Europe," *BE Sustainable, The magazine of bioenergy and the bioeconomy*, pp. 23–25, Jun-2014.

[6] Food and Agriculture Organisation, "A Decision Support Tool for Sustainable Bioenergy," *Energy*, 2010. [Online]. Available: http://www.fao.org/docrep/013/am237e/am237e00.pdf. [Accessed: 31-Oct-2014].

[7] K. Sapkota, W. Byrne, L. Melville, and C. Chapman, "An Ontology-Based Model of the Bioenergy Project Development Domain," in *International Bio-Energy Conference*, 2014, pp. 1–2.

[8] K. Sapkota, W. Byrne, P. Raju, C. Chapman, L. Melville, D. Wright, and J. Scott, "Ontology-Based Pathways Generation for Biomass to Bioenergy Conversion," in *The 10th IEEE International Conference on e-Business Engineering (ICEBE 2014), Sun Yat-sen University, Guangzhou, China, November 5 - 7 , 2014*, 2014, pp. 213–219.

[9] K. Sapkota, P. Raju, C. Chapman, W. Byrne, and L. Melville, "Bioenergy Ontology for Automatic Bioenergy Pathway Generation," in *International Conference on Knowledge Engineering (ICKE 2014), Singapore, February 2 - 3, 2015*, 2015, pp. 1–7.

[10] K. Sapkota, P. Raju, W. Byrne, and C. Chapman, "Ontology-Based Economic Models for Bioenergy and Biofuel Projects," in *Ninth IEEE International Conference on Semantic Computing (IEEE-ICSC 2015) February 7 – 9, 2015 Anaheim, CA, USA*, 2015, pp. 397–404.

[11] K. Sapkota, A. Aldea, M. Younas, D. A. Duce, and R. Banares-Alcantara, "Semantic Knowledge Mapping: An Extension of Compendium with Semantic Knowledge Representation," *Int. J. Artif. Intell. Appl.*, vol. 3, no. 5, pp. 1–12, 2012.

[12] T. R. Gruber, "A Translation Approach to Portable Ontology Specifications by A Translation Approach to Portable Ontology Specifications," *Knowl. Creat. Diffus. Util.*, vol. 5, no. April, pp. 199–220, 1993.

[13] I. Niles and A. Pease, "Towards a standard upper ontology," in *In The 2nd International Conference on Formal Ontology in Information Systems (FOIS-2001)*, 2001.

[14] I. Horrocks, P. F. Patel-schneider, H. Boley, S. Tabet, B. Grosof, and M. Dean, "SWRL : A Semantic Web Rule Language Combining OWL and RuleML," *W3C Member submission 21*. 2004.

[15] E. Prud'hommeaux and A. Seaborne, "SPARQL Query Language for RDF," *W3C Recomm.*, vol. 2009, pp. 1–106, 2008.

[16] Protege Website, "DL Query Tab," *DL Query Tab*, 2015. [Online]. Available: http://protegewiki.stanford.edu/wiki/DLQueryTab. [Accessed: 23-May-2015].

[17] E. Sirin, "Pellet Reasoner," *Clark Parsia LLC*. pp. 1–11, 2008.

[18] M. Horridge and S. Bechhofer, "The OWL API: A Java API for OWL ontologies," *Semant. Web*, vol. 2, no. 1, pp. 11–21, 2011.

[19] M. Grobe, "RDF, Jena, SparQL and the 'Semantic Web,'" in *Proceedings of the ACM SIGUCCS Fall Conference on User Services Conference (SIGUCCS '09)*, 2009, p. 131.

A Semantic Data Fragmentation Approach for RDF Data Management

Meisam Booshehri, Peter Luksch
Institute of Computer Science
University of Rostock
Rostock, Germany
e-mail: {firstname.lastname}@uni-rostock.de

***Abstract*— Efficient management of Resource Description Framework (RDF) data is one of the significant factors in realizing the semantic web vision. However, current RDF data management systems scale poorly, having performance limitations. In this PhD work, a new kind of data fragmentation in the context of RDF data is proposed based on the idea of ontology modularization. The proposed approach indicates dividing an ontology into several modules, applying RDF storing methods on ontology modules rather than on the whole ontology. By using this approach, three contributions can be introduced as follows. First, it will reduce the amount of data to be worked on at any specific point of time in order to achieve less load time and higher performance. Second, it will provide some kind of improved locality that reduces the need for interaction across the nodes of a distributed system, resulting in less message traffic. Third, according to the nature of data fragmentation we will expect higher concurrency as well. In order to show the feasibility of the approach, the main components of a suitable architecture is proposed and discussed in detail. For the evaluation, we intend to implement our proposed architecture as a layer over existing prominent open source storage systems to support the proposed fragmentation and verify the contributions. The proposed metrics would be query-time and system throughput. The former is expected to decrease while the latter is expected to increase.**

Keywords-RDF data; The semantic web database systems; Data fragmentation; Ontology modularization; Concurrency; Load time; Data traffic on the network; Performance.

I. INTRODUCTION

In order to realize the semantic web vision, it is essential to provide high-performance and scalable solutions for RDF data storage and retrieval. On the other hand, current state-of-the-art solutions have yet to be improved regarding the tremendous influx of RDF data. Current state-of-the-art methods that can be used for RDF storage and indexing could be classified into four categories:

1- *Relational Schemes*, which use Relational Database Management Systems (RDBMS) for storing RDF data.
2- *Native Schemes*, which build RDF-specific stores and indexes from scratch.
3- *Not Only SQL (NoSQL) database systems*, which are not built primarily on tables, and generally do not use SQL for data manipulation[1].
4- *Hybrid storage approaches*, which originally are aimed at the integration of NoSQL systems (such as Hadoop) with relational database technology in order to make an analytical platform for Big Data [2][3][4]. Obviously, this approach can be used for storing RDF triples.

As for the RDF data, native schemes perform well because of their tailored design, which makes the reasoning process over the semantic data easier and more straightforward. This is because of eliminating the need for some extra processes during the query process, such as query rewriting and the transformation of data to a suitable semantic format; however, relational schemes are preferred yet from the perspective of maturity, generality and scalability [5]. On the other hand, NoSQL database systems are generally more scalable than the relational database systems while NoSQL systems have some disadvantages including lack of ACID (Atomicity, Consistency, Isolation, Durability) properties and lack of SQL support. Consequently, the hybrid storage approaches have emerged as aforementioned. This evolution shows the significance of relational database technology insofar as they are being integrated into new technologies, such as NoSQL systems. Consistently, in this research we are exploring new ways for improving both for relational schemes and hybrid storage approaches.

One of the key questions in this context is the following: "How should we design tables for storing RDF triples?" The most well-known storage methods for row-oriented relational database systems are Horizontal Table [6], Vertical Table [7][8], Horizontal Class [9], Table per Property [9] and Hybrid Designs [9]. As for the column-oriented relational database systems, several prominent storage and indexing techniques have been proposed, including vertically partitioning method [10] and sextuple indexing technique [5][11], which beats row-oriented methods in terms of performance according to the recent experiments [12]. The common characteristic among all the above-mentioned methods is that they all are applied to the whole ontology data.

In this study, we specifically intend to explore the effect of a new semantic data fragmentation approach for storing RDF triples, which is elementally based on ontology modularization. As maintaining large ontologies is a difficult task and reusing the whole ontology is time-consuming and costly, the notion of an *ontology module* has been proposed.

[13]. The assumption we consider as a basis for our discussion in this paper is that “a module is considered to be a significant and self-contained sub-part of an ontology” [14]. Therefore, the vocabulary of an ontology module is a sub-set of the whole ontology vocabulary. And a module would represent a smaller ontology plus inter-module links. Moreover, a module is considered to be self-contained whenever reasoning tasks over a module can be done within the module without having accessing to other modules[15][16].

Overall, the hypothesis we are going to verify is the following: *we could use the ontology modules as the database design basis in the* ***Relational Data base Management Systems*** *or in the* ***Hybrid Storage Systems*** *instead of considering the whole ontology in order to decrease the amount of data to be worked on at any specific point of time. This will result in increasing concurrency and performance and reducing the message traffic on the network at the same time. This approach is considered as a new type of data fragmentation in the context of RDF data management systems.*

The rest of the paper is organized as follows. The second section is to review the background and some popular related works. Next, in the third section our proposed approach is described. Then, in the fourth section a customized evaluation design is proposed, and the expected results are discussed. Finally, the fifth section is to present the conclusion.

II. Related Work

We categorize the related work to this PhD thesis into four groups: ontology modularization strategies, Criteria for ontology modularization, modularity and databases, and ontology based data access systems. A detailed discussion of each category comes in the sections below.

A. *Ontology Modularization Strategies*

According to Parent and Spaccapietra, Ontology modularization strategies fall into three classes: Semantics-Driven Strategies, Structure-Driven Strategies, and Machine Learning Strategies [17]. There are also another classifications and interpretations regarding ontology modularization, including logic-based approaches and Graph theory based approaches [14][15][17][18][19][20][21][22][23][24]. However, all the classifications fall into the categories introduced by Parent and Spaccapietra on which we draw mainly in our whole research. Semantics-Driven Strategies let the ontologies be driven by the semantics of the application domain. This method relies on human expert knowledge regarding the application domain while the responsibility of the machine is usually limited to recording the allocation of knowledge items to the modules. Structure-driven strategies, on the other hand, do not rely on the human input. These methods look at the ontology as a graph structure and use graph partitioning techniques to extract ontology modules. Machine learning strategies establish another category for ontology modularization, which is considered as an alternative to human-driven modularization. In this approach, a combination of machine learning techniques can be used for knowledge processing in order to extract the ontology modules.

B. *Criteria for Ontology Modularization*

According to Mathieu d’Aquin et al. [14], there are different criteria for modularization, including logical criteria and structural criteria.

Logical criteria can be expressed in terms of local correctness and local completeness. Local correctness states that every axiom being entailed by the module should also be entailed by the original ontology, meaning that nothing has been added in the module that was not originally in the ontology. Local completeness, on the other hand, indicates the reverse property of local correctness.

Structural criteria include some measures like the the size of a module and the intra-module distance. Indeed, the relative size of a module (number of classes, properties and individuals) has a strong effect on its maintainability and, therefore, on the robustness of the applications relying on it. The intera-module distance is another important structural measure, which computes how the terms described in a module move closer to each other compared to the original ontology, for instance, by counting the number of relations in the shortest path from one entity to the other.

C. *Modularity and Databases*

Abadi et al. propose vertically partitioned method for storing RDF triples in a column-oreinted relational database system where they have observed that the query-time have dropped from minutes to several seconds [12]. Accordingly, Booshehri et. al. propose the vertically partitioned module method for the column-oriented databases in order to achieve better performance by creating the tables based on ontology modules [25]. The perspective proposed by Booshehri et. al's approach is the most related work to this PhD thesis. However, the new perspective described in this PhD work is a thoroughly refined idea of Booshehri et. al's approach. In contrast to Booshehri et. al's, the new approach described in this paper is a more generalized approach, which is not limited to relational database management systems and can be adapted with different database systems, including NoSQL systems, native schemes for RDF storage, and hybrid storage systems as well. In this research, however, we focus on relational schemes and hybrid schemes. As further is discussed, this is going to be realized by implementing different ontology based data access systems.

D. *Ontology Based Data Access Systems*

Ontology based data access (OBDA) is a technology for mapping a relational database into an ontology so that we can answer queries over the target ontology. Currently, there are two approaches for implementing an OBDA [26]: query rewriting and materialization. In the materialization approach, the input relational database is used to derive new facts based on an ontology and a set of mapping rules; then, it will be stored in a new database, which is the materialization of the data in the first database.

Sequeda et. al [26] provide a new OBDA system called Ultrawrap^{OBDA}, which combines the query rewriting and materialization approach. This combinatorial approach has been shown to achieve better performance comparing against another prominent OBDA system, namely Ontop [27]. On the other hand, to the best of our knowledge, most of OBDA systems aim at mapping a relational database into an ontology while a question will still remain open: How could we design an OBDA system for a hybrid storage system, such as Hadapt [28], which provides the capability of running SQL queries over Hadoop. Therefore, we have proposed the notion of an OBDA system for hybrid storage systems, as further is discussed in the next section.

III. Proposed Approach

The main idea of the proposed approach is to make use of ontology modularization as a semantic data fragmentation. Consequently, we expect less load time and higher performance, higher degree of concurrency and system throughput, and less message traffic on the network. We discuss these objectives in detail in the next sections.

We are motivated to design and implement our proposed approach as a layer over existing traditional RDBMSs and Hybrid Storage Systems. Accordingly, the main components of an architecture, which can show the feasibility of the approach is proposed and discussed in detail. First, a component is needed to create a *partitioned schema* based on the ontology modules instead of the whole ontology. Next, an *OBDA* system should be provided to convert the queries into queries over the new partitioned schema. Finally, a data fragmentation unit should be provided in order to fragment the ontology data according to the portioned schema. We discuss these components in more details in the sections below.

A. Schema Partitioning Component

The *Schema Partitioning component* converts the original schema which is based on the whole ontology into a partitioned schema which is based on the ontology modules. For the proposed system, we intend to provide two options for the end users. The first option is to introduce the ontology modules to the system manually and the second option is to make use of prominent approaches for automatic ontology modularization.

B. OBDA Unit

When the database schema is converted into a partitioned schema, consequently, a query rewriter should be provided in order to convert the original SQL queries into queries over the new partitioned schema. As discussed in the related work section, materialization is another approach for implementing an OBDA system, which also can be used in combination with the query rewriting techniques. We intend to design optimized OBDA systems, which support embedding our fragmentation approach into both relational database systems and hybrid storage systems. Moreover, we aim at implementing an OBDA system, which combines query rewriting and materialization in order to achieve better performance.

Regarding the partitioned schema, two types of properties can be defined for ontologies: *intra-module properties* and *inter-module properties*. Intra-module properties refer to those which are only related to the concepts and individuals within an ontology module and inter-module properties are those which connect couples of concepts or individuals from different ontology modules. It is obvious that we may have both of these two types of properties within an ontology. Accordingly, considering this classification the queries also can be classified into two categories which are *intra-module queries* and *inter-module queries*. An intra-module query is applied only on the data and ontology elements within a specific module. On the other hand, an inter-module query is applied on the information and ontology elements that connect different ontology modules. Of course, an inter-module query could be a combination of some intra-module queries as well as some inter-module queries.

Considering these classifications, whenever a query is applied to a database, the OBDA unit is responsible to recognize whether the query is inter-module or intra-module.

C. Data Fragmentation Unit

Now that we have a portioned schema, the ontology data should be fragmented and allocated to different workstations in the network.

As for a RDBMS, we have to redesign the tables according to the extracted modules. Then, it would be the responsibility of the OBDA unit to reason over the fragmented database.

In case of exploiting a hybrid storage system, the responsibility of the data fragmentation unit would be generating specialized map-reduce functions in order to fragment the data according to the ontology modules. Then, the OBDA unit will be responsible for reasoning over the database.

IV. Research Objectives and Discussions

There are three main objectives for this research:

A. Less load time and higher performance.

The proposed approach emphasizes on ontology modules as the database design basis. It is obvious that the number of extracted tables from an ontology module is less than the number of extracted tables from the whole ontology. Consequently, existing data in the tables of an ontology module is less than existing data in the corresponding tables of the whole ontology. It means that focusing on modules instead of the whole ontology, may result in a decrease in the size of the information to be worked on at any specific point of time. Hence, we expect ontology modularization to cause less load time and higher performance for column-oriented RDBMSs, row-oriented RDBMSs and Hybrid Storage Systems.

B. Increasing the degree of concurrency and system throughput.

As previously mentioned, in this research module extraction is considered as a new type of data fragmentation in the context of RDF database systems. Therefore, the more precise the module extraction algorithms are the more suitable semantic data fragmentation we have. Naturally, increasing the degree of concurrency and system throughput are two important benefits of data fragmentation in distributed databases [29][30]. Therefore, module extraction and use of ontology modules as database design basis is expected to make us closer to these two benefits. Considering the self-contained feature of the ontology modules, dividing ontologies into modules is a justifiable data fragmentation.

On the other hand, there are two important disadvantages for data fragmentation as follows:

- If there are some requirements which are in conflict with data fragmentation, the performance would decrease. For instance it is costly to retrieve several different parts of data that must be joined or unioned from different sites [30].
- During data fragmentation some attributes that is related to an association relationship may be separated into several parts and located in distinct sites. This will cause the problem of difficulty in semantic control of data and difficulty in integrity control as well [30].

However, according to the self-contained feature of an ontology module, we can say that the problems mentioned above are not serious about ontology modules.

C. Reducing the message traffic on the network with respect to intelligent allocation of data to the cluster nodes in distributed systems.

As discussed before an ontology module is self-contained meaning that special reasoning tasks such as inclusion relation or query answering within a module are possible without need to access other modules. Concerning the self-contained feature of an ontology module, it seems that allocating the data of each ontology module to a single cluster node in distributed database systems is an intelligent allocation that brings us less message traffic on the network over a specified period. This is because of the majority of intra-module queries in comparison to inter-module queries. The fewer inter-module queries leads to less message traffic between cluster nodes on the network.

V. EVALUATION

To formulate the evaluation methodology we will consider the following tasks:

1. Design a suitable benchmark to generate large-scale datasets based upon a big ontology like SWEET ontology. SWEET Ontology [31] is a highly modular ontology containing more than 200 modules.
2. Design several benchmark queries that cover all important RDF join patterns.
3. Selecting a storage system. It could be an open source column-oriented RDBMS, an open source row-oriented RDBMS, or a hybrid storage system.
4. Implementing an OBDA system so that we could implement our approach and reason over the selected ontology.
5. Evaluation of the proposed approach in terms of performance (query time) and system throughput in several working periods of the system.

After performing the above mentioned steps we expect to have the following outcomes:

1. Decrease in the running time of queries
2. Increase in the system throughput over specified working periods.

In case of achieving the expected outcomes, we will replace the first step of the evaluation methodology with an alternative step in which instead of selecting an ontology which has specified modules at the beginning, we will divide an ontology into modules automatically by using both structure driven strategies and machine learning techniques in order to test the effect of ontology modularization algorithms on the achieved outcome.

VI. CONCLUSION

Tremendous influx of RDF data calls for highly scalable and high-performance storage methods which is essential for realizing the semantic web vision. In this proposal, we are investigating the answer to the following question: “Could we improve current state-of-the-art methods for RDF storage by using ontology modules as the database design basis instead of considering the whole ontology? “

We consider the process of dividing large ontologies into modules as a kind of semantic data fragmentation. Based upon this perspective, a general architecture is proposed in order to show the feasibility of the approach. The fragmentation approach is not limited to one kind of storage system; however, we deepen our research by focusing on relational schemes and hybrid storage schemes. Next, we will try to improve them in terms of performance, concurrency and data traffic on the network.

As for the evaluation of the proposed approach, we have suggested an evaluation methodology in which different aspects of the proposed approach are verified thoroughly.

ISBN: 978-1-61208-420-6

REFERENCES

[1] A. Bialecki, M. Cafarella, D. Cutting, and O. O'MALLEY, "Hadoop: a framework for running applications on large clusters built of commodity hardware," Wiki at http://hadoop.apache.org, vol. 11, 2005.

[2] A. Abouzeid, K. Bajda-Pawlikowski, D. Abadi, A. Silberschatz, and A. Rasin, "HadoopDB: an architectural hybrid of MapReduce and DBMS technologies for analytical workloads," Proceedings of the VLDB Endowment, vol. 2, 2009, pp. 922-933.

[3] K. Bajda-Pawlikowski, D. J. Abadi, A. Silberschatz, and E. Paulson, "Efficient processing of data warehousing queries in a split execution environment," in Proceedings of the 2011 ACM SIGMOD International Conference on Management of data, 2011, pp. 1165-1176.

[4] A. Abouzied, D. J. Abadi, and A. Silberschatz, "Invisible loading: access-driven data transfer from raw files into database systems," in Proceedings of the 16th International Conference on Extending Database Technology, 2013, pp. 1-10.

[5] X. Wang, S. Wang, P. Du, and Z. Feng, "Storing and Indexing RDF Data in a Column-Oriented DBMS," in DBTA, 2010, pp. 1-4.

[6] R. Agrawal, A. Somani, and Y. Xu, "Storage and querying of e-commerce data," in VLDB, 2001, pp. 149-158.

[7] D. BeckettandJ. Grant, "Swad-europe deliverable 10.2: Mapping semantic web data with rdbmses," W3C Semantic Web Advanced Development for Europe (SWAD-Europe), 2003.

[8] M. Stonebraker, *et al.*, "C-store: a column-oriented DBMS," in Proceedings of the 31st international conference on Very large data bases, 2005, pp. 553-564.

[9] Z. PanandJ. Heflin, "Dldb: Extending relational databases to support semantic web queries," DTIC Document2004.

[10] D. J. Abadi, A. Marcus, S. R. Madden, and K. Hollenbach, "Scalable semantic web data management using vertical partitioning," in Proceedings of the 33rd international conference on Very large data bases, 2007, pp. 411-422.

[11] C. Weiss, P. Karras, and A. Bernstein, "Hexastore: sextuple indexing for semantic web data management," Proceedings of the VLDB Endowment, vol. 1, 2008, pp. 1008-1019.

[12] D. J. Abadi, A. Marcus, S. R. Madden, and K. Hollenbach, "SW-Store: a vertically partitioned DBMS for Semantic Web data management," The VLDB Journal—The International Journal on Very Large Data Bases, vol. 18, 2009, pp. 385-406.

[13] A. SchlichtandH. Stuckenschmidt, "A flexible partitioning tool for large ontologies," in Web Intelligence and Intelligent Agent Technology, 2008. WI-IAT'08. IEEE/WIC/ACM International Conference on, 2008, pp. 482-488.

[14] M. d'Aquin, A. Schlicht, H. Stuckenschmidt, and M. Sabou, "Criteria and evaluation for ontology modularization techniques," in Modular ontologies: Springer, 2009, pp. 67-89.

[15] P. Doran, "Ontology reuse via ontology modularisation," in KnowledgeWeb PhD Symposium, 2006.

[16] B. Konev, C. Lutz, D. Walther, and F. Wolter, "Logical Difference and Module Extraction with CEX and MEX," in Description Logics, 2008.

[17] C. ParentandS. Spaccapietra, "An overview of modularity," in Modular Ontologies: Springer, 2009, pp. 5-23.

[18] H. StuckenschmidtandM. Klein, "Structure-based partitioning of large concept hierarchies," in The Semantic Web–ISWC 2004: Springer, 2004, pp. 289-303.

[19] J. Bao, D. Caragea, and V. G. Honavar, "Modular ontologies–a formal investigation of semantics and expressivity," in The semantic web–ASWC 2006: Springer, 2006, pp. 616-631.

[20] B. Cuenca Grau, "Automatic Partitioning of OWL Ontologies Using E− connections," 2005.

[21] J. Pathak, T. M. Johnson, and C. G. Chute, "Survey of modular ontology techniques and their applications in the biomedical domain," Integrated computer-aided engineering, vol. 16, 2009, pp. 225-242.

[22] I. Palmisano, V. Tamma, T. Payne, and P. Doran, "Task Oriented Evaluation of Module Extraction Techniques," in The Semantic Web - ISWC 2009. vol. 5823, A. Bernstein, *et al.*, Eds.: Springer Berlin Heidelberg, 2009, pp. 130-145.

[23] J. SeidenbergandA. Rector, "Web ontology segmentation: analysis, classification and use," in Proceedings of the 15th international conference on World Wide Web, 2006, pp. 13-22.

[24] B. C. Grau, I. Horrocks, Y. Kazakov, and U. Sattler, "Extracting modules from ontologies: A logic-based approach," in Modular Ontologies: Springer, 2009, pp. 159-186.

[25] M. Booshehri, K. Zamanifar, and S. Shariatmadari, "A new approach for storing RDF triples based on ontology modularization," in The 2011 International Conference on Semantic Web and Web Services, Las Vegas, Nevada, 2011, pp. 119-125.

[26] J. F. Sequeda, M. Arenas, and D. P. Miranker, "OBDA: Query Rewriting or Materialization? In Practice, Both!," in The Semantic Web–ISWC 2014: Springer, 2014, pp. 535-551.

[27] M. Rodríguez-Muro, R. Kontchakov, and M. Zakharyaschev, "Ontology-based data access: Ontop of databases," in The Semantic Web–ISWC 2013: Springer, 2013, pp. 558-573.

[28] [retrieved:June, 2015]. http://hadapt.com/

[29] A. Silberschatz, H. F. Korth, and S. Sudarshan, Database system concepts, Sixth ed.: McGraw-Hill, 2010.

[30] M. T. Özsuand, P. Valduriez, Principles of distributed database systems: Springer Science & Business Media, 2011.

[31] [retrieved:June, 2015]. https://sweet.jpl.nasa.gov/

τOWL-Manager: A Tool for Managing Temporal Semantic Web Documents in the τOWL Framework

Abir Zekri, Zouhaier Brahmia
University of Sfax
Sfax, Tunisia
emails: abir.zekri@fsegs.rnu.tn,
zouhaier.brahmia@fsegs.rnu.tn

Fabio Grandi
University of Bologna
Bologna, Italy
email: fabio.grandi@unibo.it

Rafik Bouaziz
University of Sfax
Sfax, Tunisia
email: raf.bouaziz@fsegs.rnu.tn

***Abstract*—Several semantic web-based applications (e.g., e-commerce, e-government and e-health applications) require temporal versioning of ontology instances, in order to represent, store and retrieve time-varying ontologies. However, commercial systems do not provide any support for creating and updating temporal ontologies. In this paper, we propose a prototype system, named Temporal OWL 2 Web Ontology Language Manager (τOWL-Manager), which implements our τOWL framework and supports temporal versioning of ontology instances. It allows (i) creating and validating a temporal semantic web document, by augmenting an OWL 2 ontology schema with a set of logical and physical annotations, and (ii) creating and maintaining time-varying ontology instance documents, by generating a new timestamped version of each ontology instance document when updates are applied.**

Keywords–Semantic Web; Ontology; OWL 2; τXSchema; Logical annotations; Physical annotations; Temporal database; XML Schema; XML

I. Introduction

Due to the dynamic nature of the Web, ontologies [2]—like other components of the Web 3.0 including databases and Web pages—evolve over time to reflect and model changes occurring in the real-world. Furthermore, several Semantic Web-based applications (like e-commerce, e-government and e-health applications) require keeping track of ontology evolution and versioning with respect to time, in order to represent, store and retrieve time-varying ontologies.

Unfortunately, while there is a sustained interest for temporal and evolution aspects in the research community [3], existing Semantic Web [4] standards, state-of-the-art ontology editors and knowledge representation tools do not provide any built-in support for managing temporal ontologies. In particular, the W3C OWL 2 recommendation [5][6] lacks explicit support for time-varying ontologies, at both schema and instance levels. Thus, a Knowledge Base Administrator (KBA), i.e., a knowledge engineer or a maintainer of semantics-based Web resources, must use ad hoc techniques when there is a need, for example, to specify an OWL 2 ontology schema for time-varying ontology instances.

On the other hand, in order to handle temporal ontology evolution in an effective and systematic manner and to allow historical queries to be efficiently executed on time-varying ontologies, a built-in temporal ontology management system is needed. For that purpose, we proposed in our previous work [1] a framework, called τOWL, for managing temporal Semantic Web documents, through the use of a temporal OWL 2 extension. In fact, we want to introduce with τOWL a principled and systematic approach to the temporal extension of OWL 2, similar to that Snodgrass and colleagues did with their Temporal XML Schema (τXSchema) [7][8] to the eXtensible Markup Language (XML) and XML Schema [9]. τXSchema is a powerful framework (i.e., a data model equipped with a suite of tools) for managing temporal XML documents, well known in the database research community and, in particular, in the field of temporal XML [10]. Moreover, in the previous work [11], with the aim of completing the framework, we augmented τXSchema by defining necessary schema change operations.

Being defined as a τXSchema-like framework, τOWL allows creating a temporal OWL 2 ontology from a conventional (i.e., non-temporal) OWL 2 ontology specification and a set of logical (or temporal) and physical annotations. Logical annotations identify which components of a Semantic Web document can vary over time; physical annotations specify how the time-varying aspects are represented in the document. By using temporal schema and annotations to introduce temporal aspects in the conventional Semantic Web, our framework (i) guarantees logical and physical data independence [12] for temporal ontologies and (ii) provides a low-impact solution since it requires neither modifications of existing Semantic Web documents nor extensions to the OWL 2 recommendation and Semantic Web standards.

Furthermore, while there is a lot of research works on managing temporal ontologies [13][14][15][16], only two research tools have been proposed to handle some particular aspects: Stock Recommendations Aggregation System (SRAS) [17], which is centered around the aggregation of stock recommendations and financial data, and CHRONOS [18], which is a reasoner over temporal information in OWL ontologies. Current commercial solutions in the Semantic Web area (Oracle Semantic Technology [19], IBM Scalable Ontology Repository (SOR) [20], and IBM DB2 Resource Description Framework (RDF) [21]) do not include features for supporting time in ontologies.

In order to (i) show the feasibility of our τOWL approach [1], (ii) facilitate a KBA when he/she has to create a temporal ontology and manipulate its instances, and (iii) fill the lack of support noticed in commercial knowledge management systems, we propose in this paper a prototype system, named τOWL-Manager, which allows a KBA (i) to create and validate τOWL ontology schemata, and (ii) to create and update τOWL ontology instance documents. When modified, instance documents are augmented with timestamps to support temporal versioning.

With regard to our previous work [1], the current one focuses on implementing our τOWL framework; the result, τOWL-Manager, could be a first step towards providing commercial support for temporal ontologies.

The remainder of the paper is organized as follows. Section II describes our τOWL framework, previously proposed in [1]: the architecture of τOWL is presented and details on all its components and support tools are given. Section III illustrates the use of τOWL through an example. Section IV proposes our prototype tool, τOWL-Manager: its architecture and some screenshots showing its functioning are provided. Section V provides a summary of the paper and some remarks about our future work.

II. The τOWL Framework

In this section, we present our τOWL framework for handling temporal Semantic Web documents. We describe the overall architecture of τOWL. Since τOWL is a τXSchema-like framework, we were inspired by the τXSchema architecture and tools while defining the architecture and tools of τOWL. More details on our framework can be found in [1] and [22].

The τOWL framework allows a KBA to create a temporal OWL 2 schema for temporal OWL 2 instances from a conventional OWL 2 schema, logical annotations, and physical annotations. Since it is a τXSchema-like framework, τOWL use the following principles: separation between (i) the conventional (i.e., non-temporal) schema and the temporal schema, and (ii) the conventional instances and the temporal instances; (iii) use of logical and physical annotations to specify temporal and physical aspects, respectively, at schema level.

Figure 1 illustrates the architecture of τOWL. The framework is based on the OWL 2 language [5][6], which is a W3C standard ontology language for the Semantic Web. It allows defining both schema (i.e., entities, axioms, and expressions) and instances (i.e., individuals) of ontologies.

The KBA starts by creating the *conventional schema* (box 7), which is an OWL 2 ontology that models the concepts of a particular domain and the relations between these concepts, without any temporal aspect. To each conventional schema corresponds a set of conventional OWL 2 instances (box 12). As recommended in the the OWL 2 specification [6], τOWL deals with OWL 2 ontologies with an RDF/XML syntax [23].

After that, the KBA augments the conventional schema with *logical* and *physical annotations*, which allow him/her to express, in an explicit way, all requirements dealing with the representation and the management of temporal aspects associated to the components of the conventional schema, as described in the following.

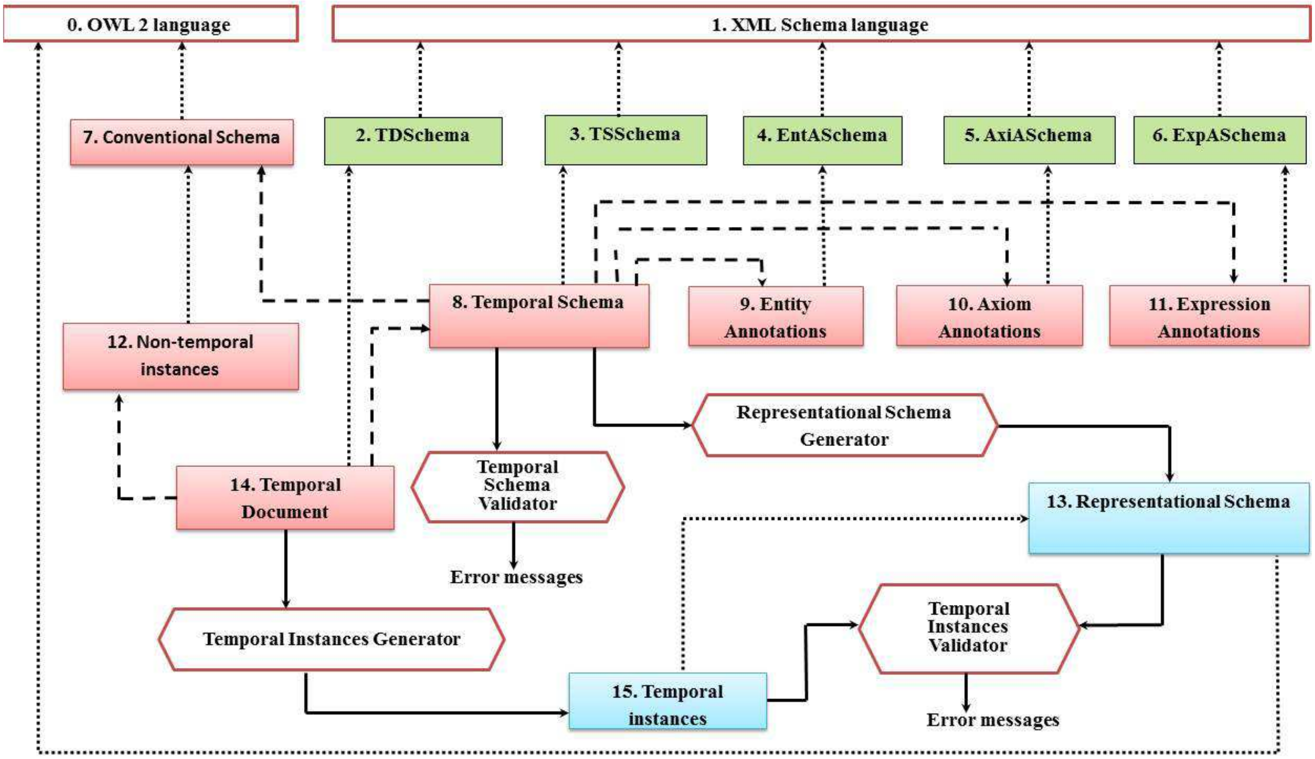


Figure 1. Overall architecture of τOWL.

Logical annotations [8] allow the KBA to specify (i) whether a conventional schema component varies over valid time and/or transaction time, (ii) whether its lifetime is described as a continuous state or a single event, (iii) whether the component may appear at certain times (and not at others), and (iv) whether its content changes.

Physical annotations [8] allow the KBA to specify the timestamp representation options chosen, such as where the timestamps are placed and their kind (i.e., valid time or transaction time) and the kind of representation adopted. Timestamps can be located either on time-varying components (as specified by the logical annotations) or somewhere above such components. Two OWL 2 documents with the same logical information will look very different if we change the location of their physical timestamps.

Finally, when the KBA finishes annotating the conventional schema and asks the system to save his/her work, this latter creates the *temporal schema* (box 8) in order to provide the linking information between the conventional schema and its corresponding logical and physical annotations. The temporal schema is a standard XML document which ties the conventional schema, the entity annotations, the axiom annotations, and the expression annotations together. In the τOWL framework, the temporal schema is the logical equivalent of the conventional OWL 2 schema in a non-temporal context. This document contains sub-elements that associate a series of conventional schema definitions with entity annotations, axiom annotations, and expression annotations, along with the time span during which the association was in effect. The schema for the temporal schema document is the XML Schema Definition document *TSSchema* (box 3).

To complete the picture, after creating the temporal schema, the system creates a *temporal document* (box 14) in order to link each conventional ontology instance document (box 12), which is valid to a conventional ontology schema (box 7), to its corresponding temporal ontology schema (box 8), and more precisely to its corresponding logical and physical annotations (which are referenced by the temporal schema). A temporal document is a standard XML document that maintains the evolution of a non-temporal ontology instance document over time, by recording all of the versions (or temporal slices) of the document with their corresponding timestamps and by specifying the temporal schema associated to these versions. This document contains sub-elements that associate a series of conventional ontology instance documents with logical and physical annotations (on entities, axioms, and expressions), along with the time span during which the association was in effect. Thus, the temporal document is very important for making easy the support of temporal queries working on past versions or dealing with changes between versions. The schema for the temporal document is the XML Schema Definition document *TDSchema* (box 2).

III. ILLUSTRATIVE EXAMPLE

In order to show the functioning of the τOWL approach and how management of temporal ontology document versions is dealt with in it, we provide an example concerning the evolution of an ontology based on Friend Of A Friend (FOAF). The FOAF [24] project is creating a Web of machine-readable pages describing people, the links between them and the things they create and do.

Suppose that a Web site “Society-Web” publishes the FOAF definition for their users and that the webmaster of this Web site wants to keep track of the changes performed on FOAF RDF [25] information. We will focus in this example on one user whose name is “Khalid Sinan”.

Suppose that on January 15, 2014, the KBA creates a conventional ontology schema, named “PersonSchema_V1.owl” (Figure 2), and a conventional ontology instance document, named “Persons_V1.rdf” (Figure 3), which is valid with respect to this schema. We assume that the KBA defines also a set of logical and physical annotations, associated to that conventional schema; they are stored in an ontology annotation document titled “PersonAnnotations_V1.xml” as shown in Figure 4.

Notice that the conventional (i.e., non-temporal) schema (Figure 1) for the FOAF RDF document (Figure 2) is the schema for an individual version, which allows updating and querying individual versions. The conventional ontology instance document describes, according to the FOAF ontology, the personal information of “Khalid Sinan” (i.e., name and nickname) and the information about his online accounts on diverse sites (i.e., the home page of the site, and the account name of the user). In this example, we only consider the user account on the “Facebook” Web site.

```
<rdf:RDF>
  <owl:Ontology  rdf:about="http://purl.org/
                           az/foaf#">
   <rdfs:Class  rdf:about="#Person">
     <rdf:type  rdf:resource="http://www.w3.org/
                             2002/07/owl#Class"/>
   </rdfs:Class>
   <rdf:Property  rdf:about="#holdsAccount">
     <rdf:type  rdf:resource="http://www.w3.org/
                    2002/07/owl#ObjectProperty"/>
     <rdfs:domain  rdf:resource="#Person"/>
     <rdfs:range  rdf:resource="#OnlineAccount"/>
   </rdf:Property>
   <rdf:Property  rdf:about="#accountName">
     <rdf:type  rdf:resource="http://www.w3.org/
                  2002/07/owl#DatatypeProperty"/>
     <rdfs:domain rdf:resource="#OnlineAccount"/>
   </rdf:Property>
   …
</rdf:RDF>
```

Figure 2. An RDF/XML extract from the OWL 2 FOAF ontology.

```
…
<foaf:Person rdf:ID="#Person1">
  <foaf:name>Khalid Sinan</foaf:name>
  <foaf:nick>Khal</foaf:nick>
  <foaf:holdsAccount>
   <foaf:OnlineAccount rdf:about="
          https://www.facebook.com/Khalid.Sinan">
     <foaf:accountName>Khal_Sinan
      </foaf:accountName>
   </foaf:OnlineAccount>
  </foaf:holdsAccount>
</foaf:Person>
…
```

Figure 3. A fragment of Khalid FOAF RDF document on January 15, 2014.

```
<?xml version=”1.0” encoding=”UTF-8”?>
<ontologyAnnotationSet>
  <logicalAnnotations>
    <item target=”/Person/nick”>
      <validTime kind=”state” content=”varying”
                 existence=”constant”/>
    </item>
  </logicalAnnotations>
  <physicalAnnotations>
    <stamp target=”Person/nick”
           dataInclusion=”expandedVersion”>
      <stampkind timeDimension=”validTime”
                 stampBounds=”extent”/>
    </stamp>
  </physicalAnnotations>
</ontologyAnnotationSet>
```

Figure 4. The annotation document on January 15, 2014.

After that, the system creates the temporal ontology schema in Figure 5 (that ties “PersonSchema_V1.owl” and “PersonAnnotations_V1.xml” together), which is stored in an XML file named “PersonTemporalSchema.xml”. Consequently, the system uses the temporal ontology schema of Figure 5 and the conventional ontology document in Figure 3 to create a temporal document as in Figure 6, that lists both versions (i.e., temporal “slices”) of the conventional ontology documents with their associated timestamps. The squashed version of this temporal document, which could be generated by the Temporal Instances Generator, is provided in Figure 7.

```
<?xml version=”1.0” encoding=”UTF-8”?>
<temporalOntologySchema>
  <conventionalOntologySchema>
    <sliceSequence>
      <slice location=”PersonSchema_V1.owl”
             begin=”2014-01-15” />
    </sliceSequence>
  </conventionalOntologySchema>
  <ontologyAnnotationSet>
    <sliceSequence>
      <slice location=”PersonAnnotations_V1.xml”
             begin=”2014-01-15” />
    </sliceSequence>
  </ontologyAnnotationSet>
</temporalOntologySchema>
```

Figure 5. The temporal schema on January 15, 2014.

```
<?xml version=”1.0” encoding=”UTF-8”?>
<td:temporalRoot temporalSchemaLocation=
                      ”PersonTemporalSchema.xml”/>
  <td:sliceSequence>
    <td:slice location=”Persons_V1.rdf”
              begin=”2014-01-15” />
  </td:sliceSequence>
</td:temporalRoot>
```

Figure 6. The temporal document on January 15, 2014.

On February 08, 2014, Khalid modified his nickname from “Khal” to “Elkhal” and his account name of Facebook from “Khal_Sinan” to “Elkhal_Sinan”. Thus, the system updates the conventional ontology document “Persons_V1.rdf” to produce a new conventional ontology document named “Persons_V2.rdf” (Figure 8). Since the conventional ontology schema (i.e., PersonSchema_V1.owl) and the ontology annotation document (i.e., PersonAnnotations_V1.xml) are not changed, the temporal ontology schema (i.e., PersonTemporalSchema.xml) is consequently not updated. However, the Temporal Instances Generator tool updates the temporal document, in order to include the new slice of the conventional ontology document, as shown in Figure 9. The squashed version of the updated temporal document is provided in Figure 10.

Obviously, each one of the squashed documents (Figure 7 and Figure 10) must conform to a particular schema, that is the representational schema, which is generated by the Representational Schema Generator from the temporal schema shown in Figure 5.

```
<foaf:Person rdf:ID="#Person1">
  <foaf:name>Khalid Sinan</foaf:name>
  <nick_RepItem>
    <nick_Version>
      <timestamp_ValidExtent begin=”2014-01-15”
                             end=”now” />
      <foaf:nick>Khal</foaf:nick>
    </nick_Version>
  </nick_RepItem>
  <foaf:holdsAccount>
    <foaf:OnlineAccount rdf:about="
          https://www.facebook.com/Khalid.Sinan">
      <accountName_RepItem>
        <accountName_Version>
          <timestamp_ValidExtent
                 begin=”2014-01-15” end=”now” />
          <foaf:accountName>Khal_Sinan
          </foaf:accountName>
        </accountName_Version>
      </accountName_RepItem>
    </foaf:OnlineAccount>
  </foaf:holdsAccount>
</foaf:Person>
```

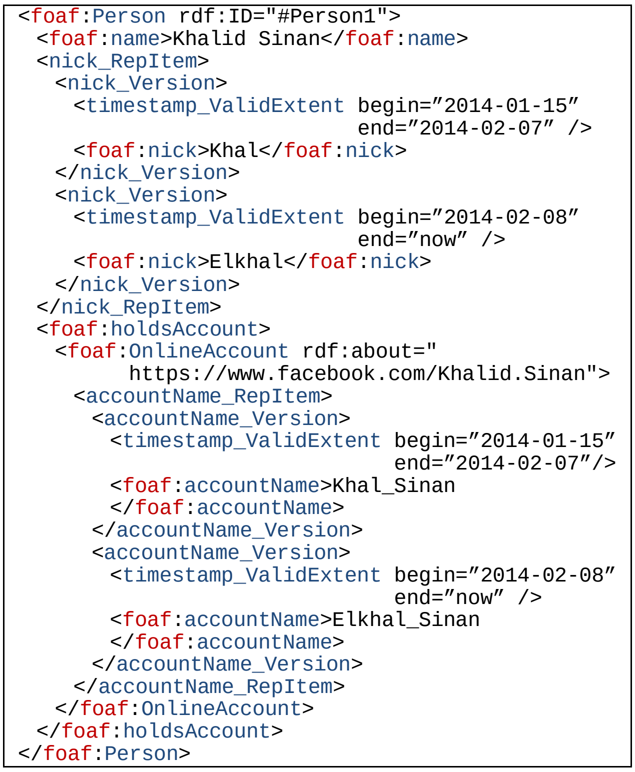
Figure 7. The squashed document correponding to the temporal document on January 15, 2014.

```
…
<foaf:Person rdf:ID="#Person1">
  <foaf:name>Khalid Sinan</foaf:name>
  <foaf:nick>Elkhal</foaf:nick>
  <foaf:holdsAccount>
    <foaf:OnlineAccount rdf:about="
          https://www.facebook.com/Khalid.Sinan">
      <foaf:accountName>Elkhal_Sinan
      </foaf:accountName>
    </foaf:OnlineAccount>
  </foaf:holdsAccount>
</foaf:Person>
…
```

Figure 8. A fragment of Khalid FOAF RDF document on February 08, 2014.

```
<?xml version=”1.0” encoding=”UTF-8”?>
<td:temporalRoot temporalSchemaLocation=
                      ”PersonTemporalSchema.xml”/>
  <td:sliceSequence>
    <td:slice location=”Persons_V1.rdf”
              begin=”2014-01-15” />
    <td:slice location=”Persons_V2.rdf”
              begin=”2014-02-08” />
  </td:sliceSequence>
</td:temporalRoot>
```

Figure 9. The temporal document on February 08, 2014.



```
<foaf:Person rdf:ID="#Person1">
  <foaf:name>Khalid Sinan</foaf:name>
  <nick_RepItem>
    <nick_Version>
      <timestamp_ValidExtent begin=”2014-01-15”
                             end=”2014-02-07” />
      <foaf:nick>Khal</foaf:nick>
    </nick_Version>
    <nick_Version>
      <timestamp_ValidExtent begin=”2014-02-08”
                             end=”now” />
      <foaf:nick>Elkhal</foaf:nick>
    </nick_Version>
  </nick_RepItem>
  <foaf:holdsAccount>
    <foaf:OnlineAccount rdf:about="
            https://www.facebook.com/Khalid.Sinan">
      <accountName_RepItem>
        <accountName_Version>
          <timestamp_ValidExtent begin=”2014-01-15”
                                 end=”2014-02-07”/>
          <foaf:accountName>Khal_Sinan
          </foaf:accountName>
        </accountName_Version>
        <accountName_Version>
          <timestamp_ValidExtent begin=”2014-02-08”
                                 end=”now” />
          <foaf:accountName>Elkhal_Sinan
          </foaf:accountName>
        </accountName_Version>
      </accountName_RepItem>
    </foaf:OnlineAccount>
  </foaf:holdsAccount>
</foaf:Person>
```

Figure 10. The squashed document correponding to the temporal document on February 08, 2014.

IV. IMPLEMENTATION

In this section, we describe a prototype system, named τOWL-Manager, which implements our τOWL approach and shows its feasibility. It allows (i) the specification and validation of τOWL ontologies schemata, and (ii) the creation and maintenance of τOWL ontology instance documents. Each update operation on an instance document gives rise to a new version of this document with its corresponding timestamp.

τOWL-Manager is a Java (JDK 1.7) application, developed in the Integrated Development Environment (IDE) “Eclipse Helios”, using (i) the OWL Application Programming Interface (API) [26], which is a Java API and a reference implementation, for creating and manipulating OWL ontologies, and (ii) the Java Document Object Model (JDOM) API for creating and manipulating XML files. In the following, we first describe the architecture of τOWL-Manager and then provide some screenshots showing its use. Notice that these screenshots deal with the same example presented in Section III.

A. Architecture of τOWL-Manager

The overall architecture of τOWL-Manager is depicted in Figure 11. It is composed of three layers: presentation layer, business layer, and storage layer.

The **presentation layer** includes an interface for constructing temporal ontologies and an interface for creating and updating ontology instances.

The **business layer** contains two modules: one for managing temporal ontologies, named “Temporal Ontology Manager”, and the other for managing ontology instances, named “Ontology Instance Document Manager”. The “Temporal Ontology Manager” first generates the files corresponding to the temporal ontology schema, that is the conventional schema file and the annotation document file, from the specifications expressed by the KBA in its interface. Then, it checks the validity of the generated files and creates the temporal schema file, which ties together the two other files.

The **storage layer** contains the repository of resources making up temporal ontologies and associated instances, named τOWL Repository.

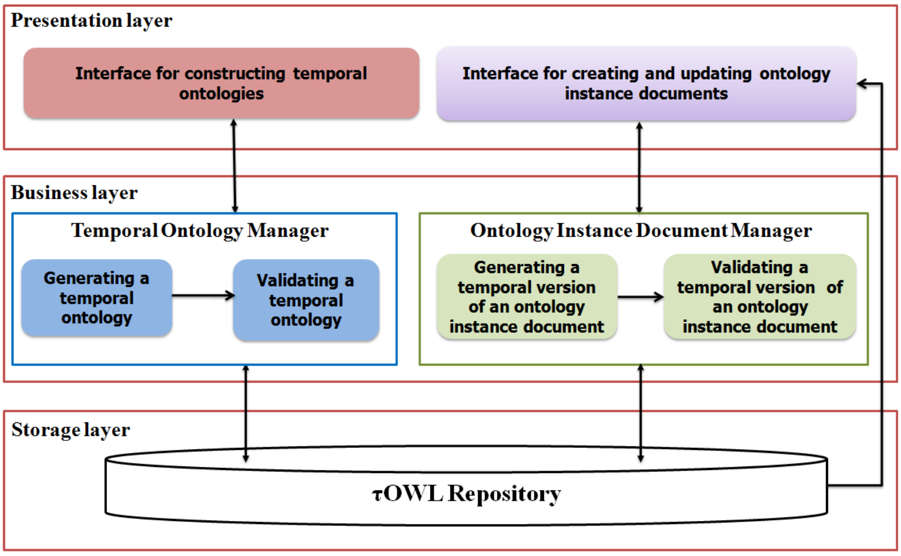

Presentation layer
Interface for constructing temporal ontologies
Interface for creating and updating ontology instance documents
Business layer
Temporal Ontology Manager
Generating a temporal ontology
Validating a temporal ontology
Ontology Instance Document Manager
Generating a temporal version of an ontology instance document
Validating a temporal version of an ontology instance document
Storage layer
τOWL Repository

Figure 11. Architecture of τOWL-Manager.

B. *Screenshots of τOWL-Manager*

Currently, τOWL-Manager allows a KBA to perform two activities: (i) creating and validating temporal ontologies, and (ii) creating and updating ontology instances. In the following, we illustrate its functioning and show its use for each one of the two activities, via the example of Section III.

1) Constructing and validating temporal ontologies

To construct a new temporal ontology, the KBA has to perform the following tasks:

i) He/she starts by creating a τOWL project. To this aim, the KBA has to provide a reference to an existing valid conventional ontology schema (definition of an ontology schema from scratch is not supported in the current version of τOWL-Manager). Assume here that the KBA has chosen the FOAF ontology.

ii) After that, the KBA annotates the new conventional ontology schema by some logical and physical annotations. Figure 12 shows the specification of some logical annotations on the class "Person" and Figure 13 shows the specification of some physical annotations on the same class.

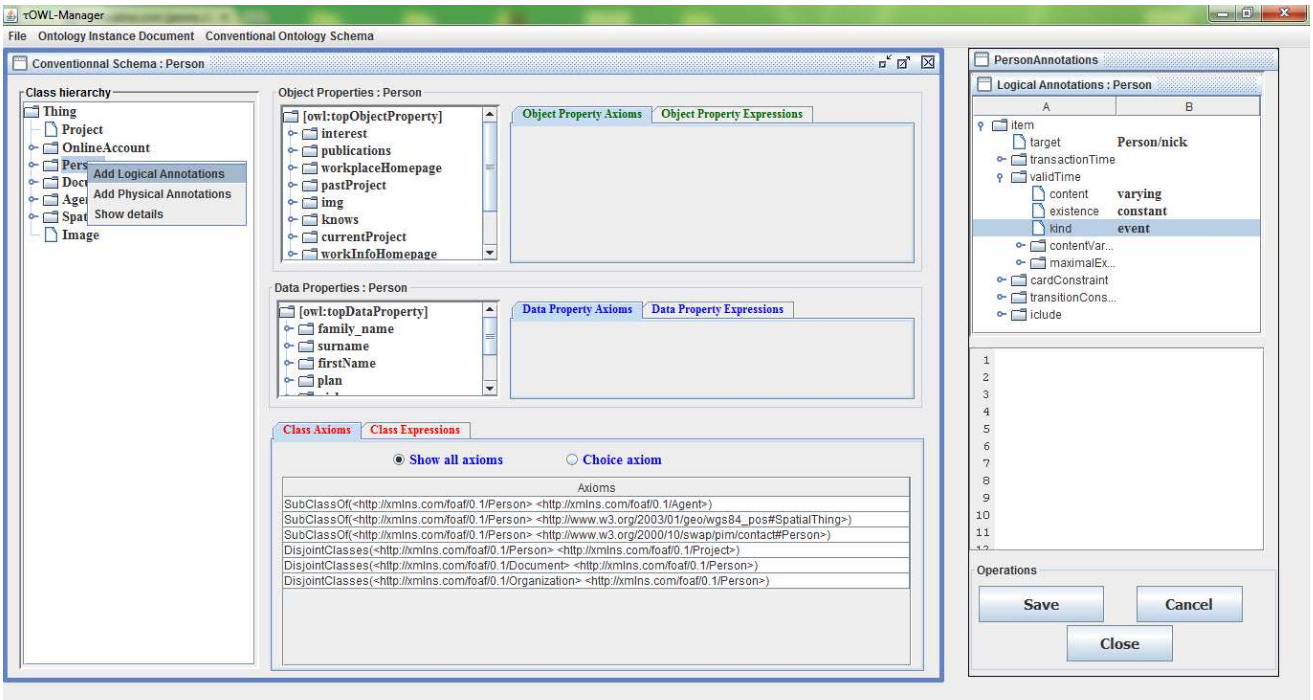
Notice that a τOWL project is a set of folders:

- Annotations: it contains the file corresponding to the logical and physical annotation document of a τOWL ontology;
- Conventional Ontology Instance Documents: it stores all the versions of conventional ontology instance documents;
- Conventional Ontology Schema: it includes the conventional ontology schema file of a τOWL ontology;
- Representational Schema: it stores the representational schema file;
- Temporal Document: it includes the temporal document (which is generated automatically);
- Temporal Ontology Instance Documents: it contains all the versions of temporal ontology instance documents.
- Temporal Schema: it contains the temporal schema file.

2) Creating and versioning ontology instance documents

We show in Figure 14 an ontology instantiation. After the KBA has chosen a τOWL ontology schema, he/she can create its instances (Figure 14). Finally, he/she should save his/her work, through the "Save" button. Consequently, the system generates an RDF file corresponding to the conventional ontology instances which have been created. Such a file is generated using the OWL API and validated using the Pellet reasoner. Furthermore, the system automatically updates the temporal document in order to add a new slice corresponding to the new version of the ontology instance document.

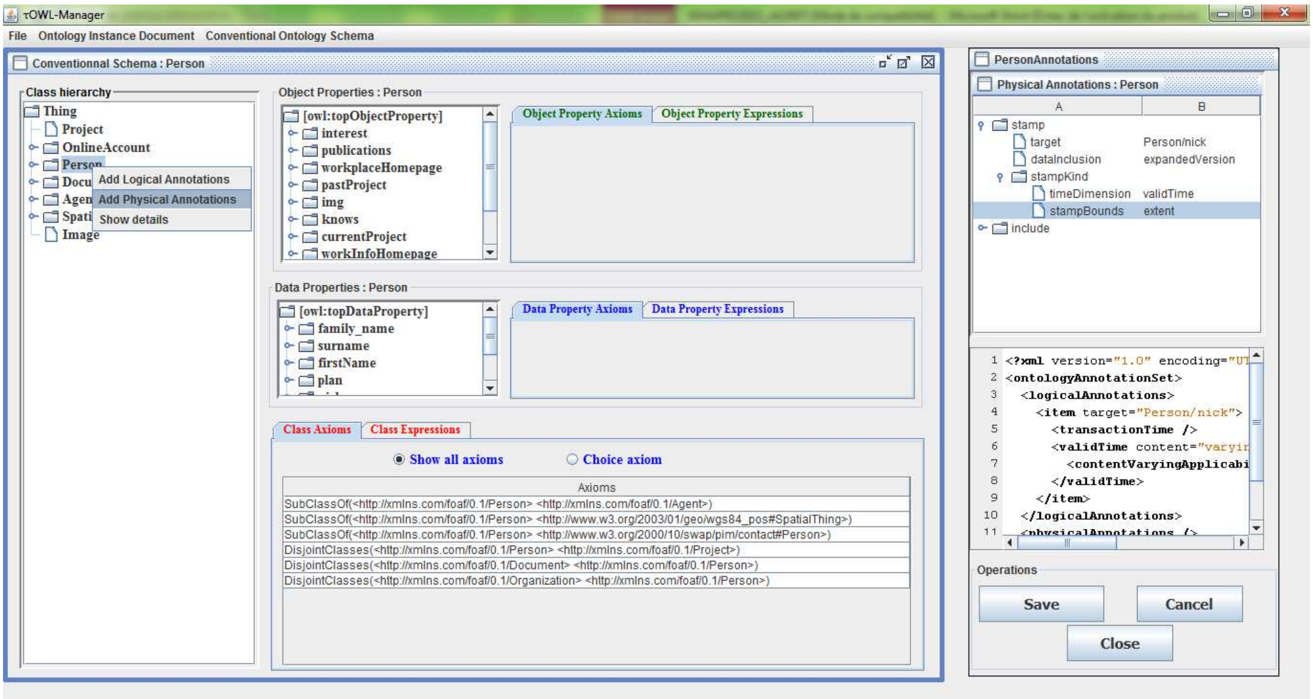
Moreover, τOWL-Manager allows keeping track of ontology instances when they evolve over time. Figures 15 and 16 show an example of maintaining the history of an ontology instance evolution: first the KBA chooses "Persons_V1.rdf" as the ontology instance document version that must be updated (Figure 15).



Axioms
SubClassOf(<http://xmlns.com/foaf/0.1/Person> <http://xmlns.com/foaf/0.1/Agent>)
SubClassOf(<http://xmlns.com/foaf/0.1/Person> <http://www.w3.org/2003/01/geo/wgs84_pos#SpatialThing>)
SubClassOf(<http://xmlns.com/foaf/0.1/Person> <http://www.w3.org/2000/10/swap/pim/contact#Person>)
DisjointClasses(<http://xmlns.com/foaf/0.1/Person> <http://xmlns.com/foaf/0.1/Project>)
DisjointClasses(<http://xmlns.com/foaf/0.1/Document> <http://xmlns.com/foaf/0.1/Person>)
DisjointClasses(<http://xmlns.com/foaf/0.1/Organization> <http://xmlns.com/foaf/0.1/Person>)

A	B
target	Person/nick
content	varying
existence	constant
kind	event

Figure 12. Specifying some logical annotations on the conventional ontology schema.

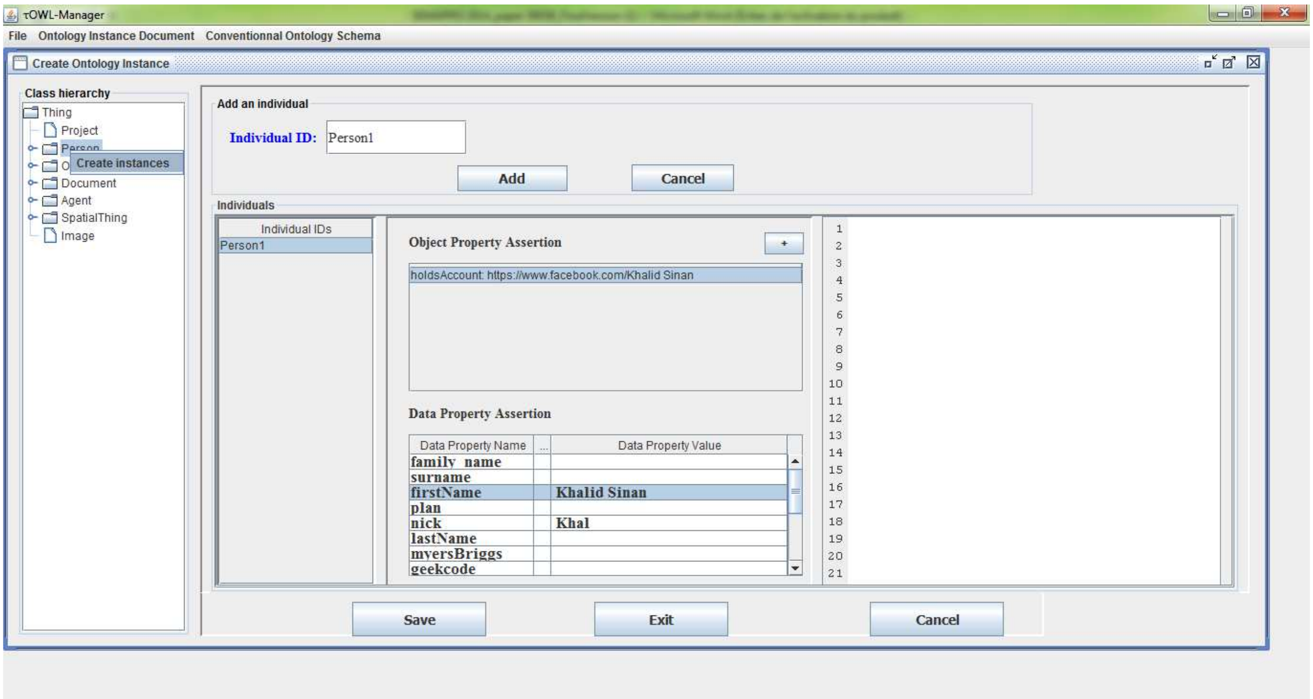


Axioms
SubClassOf(<http://xmlns.com/foaf/0.1/Person> <http://xmlns.com/foaf/0.1/Agent>)
SubClassOf(<http://xmlns.com/foaf/0.1/Person> <http://www.w3.org/2003/01/geo/wgs84_pos#SpatialThing>)
SubClassOf(<http://xmlns.com/foaf/0.1/Person> <http://www.w3.org/2000/10/swap/pim/contact#Person>)
DisjointClasses(<http://xmlns.com/foaf/0.1/Person> <http://xmlns.com/foaf/0.1/Project>)
DisjointClasses(<http://xmlns.com/foaf/0.1/Document> <http://xmlns.com/foaf/0.1/Person>)
DisjointClasses(<http://xmlns.com/foaf/0.1/Organization> <http://xmlns.com/foaf/0.1/Person>)

A	B
stamp	
target	Person/nick
dataInclusion	expandedVersion
stampKind	
timeDimension	validTime
stampBounds	extent
include	

```
<?xml version="1.0" encoding="UT
<ontologyAnnotationSet>
  <logicalAnnotations>
    <item target="Person/nick">
      <transactionTime />
      <validTime content="varyin
        <contentVaryingApplicabi
      </validTime>
    </item>
  </logicalAnnotations>
  <physicalAnnotations />
```

Figure 13. Specifying some physical annotations on the conventional ontology schema.



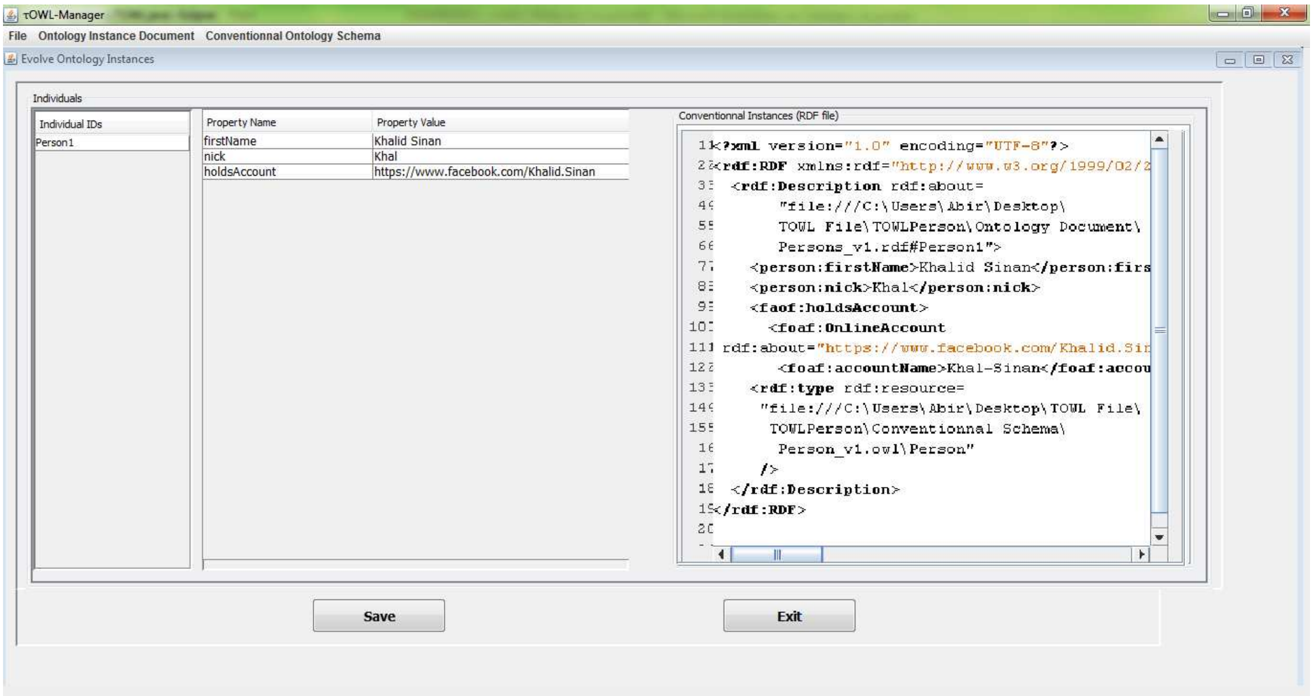
Individual ID: Person1

Individual IDs
Person1

holdsAccount: https://www.facebook.com/Khalid Sinan

Data Property Name	...	Data Property Value
family_name		
surname		
firstName		Khalid Sinan
plan		
nick		Khal
lastName		
myersBriggs		
geekcode		

Figure 14. Populating a conventional ontology.


τOWL-Manager
File
Ontology Instance Document
Conventionnal Ontology Schema
Evolve Ontology Instances
Individuals
Conventionnal Instances (RDF file)
Save
Exit

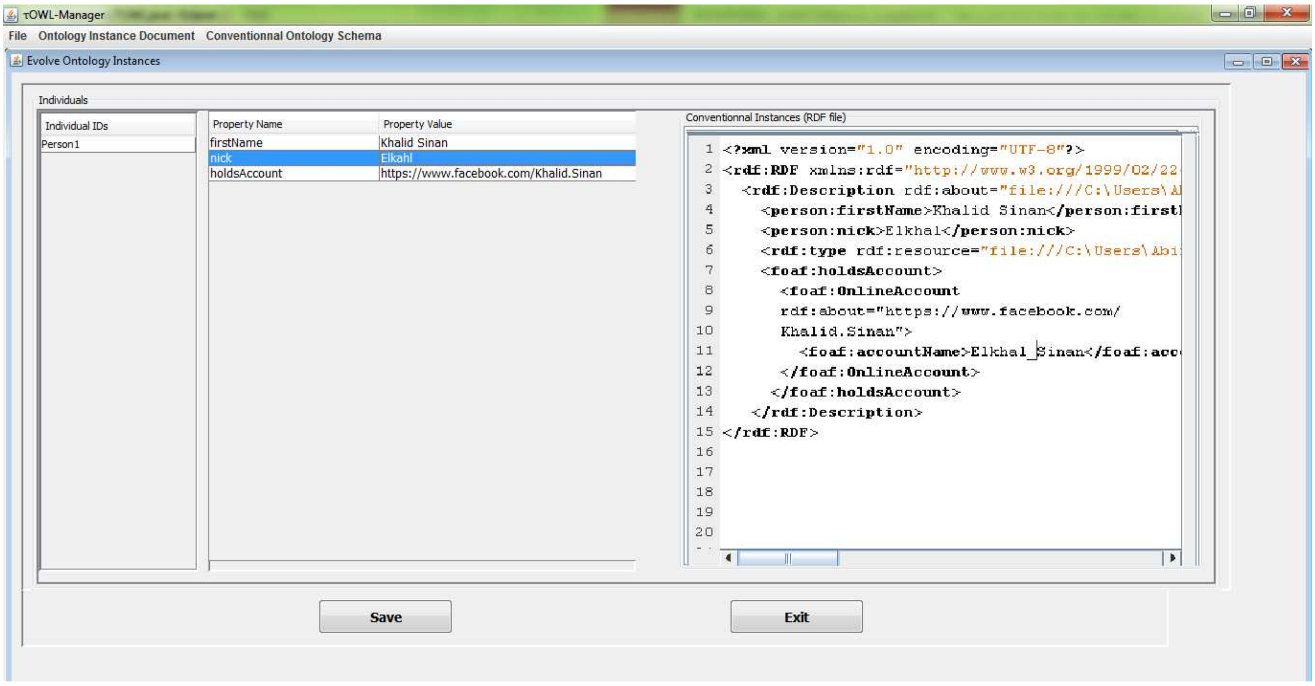
Individual IDs
Person1

Property Name	Property Value
firstName	Khalid Sinan
nick	Khal
holdsAccount	https://www.facebook.com/Khalid.Sinan

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/2
  <rdf:Description rdf:about=
        "file:///C:\Users\Abir\Desktop\
        TOWL File\TOWLPerson\Ontology Document\
        Persons_v1.rdf#Person1">
    <person:firstName>Khalid Sinan</person:firs
    <person:nick>Khal</person:nick>
    <faof:holdsAccount>
      <foaf:OnlineAccount
rdf:about="https://www.facebook.com/Khalid.Sin
        <foaf:accountName>Khal-Sinan</foaf:accou
    <rdf:type rdf:resource=
     "file:///C:\Users\Abir\Desktop\TOWL File\
      TOWLPerson\Conventionnal Schema\
       Person_v1.owl\Person"
     />
  </rdf:Description>
</rdf:RDF>
```

Figure 15. Showing the chosen conventional ontology instance document version.

Figure 16 shows that the KBA has modified the chosen ontology instance document version (by modifying the nick and the account name of the Person “Khalid Sinan”). Thus, the system automatically generates a new version of the ontology instance document. After verifying that the new ontology instance document version is different from its predecessor, the system adds it to the folder “Conventional Ontology Instance Documents” of the τOWL project. Moreover, the creation of a new version of an ontology instance document causes an automatic update of the temporal document.


τOWL-Manager
File
Ontology Instance Document
Conventionnal Ontology Schema
Evolve Ontology Instances
Individuals
Conventionnal Instances (RDF file)
Save
Exit

Individual IDs
Person1

Property Name	Property Value
firstName	Khalid Sinan
nick	Elkahl
holdsAccount	https://www.facebook.com/Khalid.Sinan

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22
  <rdf:Description rdf:about="file:///C:\Users\A
    <person:firstName>Khalid Sinan</person:first
    <person:nick>Elkhal</person:nick>
    <rdf:type rdf:resource="file:///C:\Users\Abi
    <foaf:holdsAccount>
      <foaf:OnlineAccount
      rdf:about="https://www.facebook.com/
      Khalid.Sinan">
        <foaf:accountName>Elkhal_Sinan</foaf:acc
      </foaf:OnlineAccount>
     </foaf:holdsAccount>
   </rdf:Description>
</rdf:RDF>
```

Figure 16. Updating the chosen conventional ontology instance document version (changing the nick and the account name of Khalid).

V. Conclusion and Future Work

In this paper, we have presented τOWL-Manager, a prototype tool for specifying temporal ontologies and temporal instance versioning, in the τOWL framework, demonstrating its feasibility. It helps a KBA to create temporal ontologies and manipulate its instances, overcoming the lack of support detected in state-of-the-art commercial knowledge management systems and research tools. Thus, it could be considered as a first step towards providing commercial support for temporal ontologies.

Our future work aims at extending τOWL-Manager to also support temporal versioning of the schema itself, in the τOWL framework. Such extension requires, as a first step, the definition of necessary schema change operations, that is operations acting on conventional schema, annotations and temporal schema. A subset of these operations has been defined in our recent work [27].

References

[1] A. Zekri, Z. Brahmia, F. Grandi, and R. Bouaziz, “τOWL: A Framework for Managing Temporal Semantic Web Documents,” Proceedings of the 8th International Conference on Advances in Semantic Processing (SEMAPRO 2014), Rome, Italy, 24-28 August 2014, pp. 33-41.

[2] N. Guarino (Ed.), Formal Ontology in Information Systems, IOS Press, Amsterdam, 1998.

[3] F. Grandi, “Introducing an Annotated Bibliography on Temporal and Evolution Aspects in the Semantic Web,” SIGMOD Record, vol. 41, December 2012, pp. 18-21.

[4] T. Berners-Lee, J. Hendler, and O. Lassila, “The Semantic Web,” Scientific American, vol. 284, May 2001, pp. 34-43.

[5] W3C, OWL 2 Web Ontology Language – Primer (Second Edition), W3C Recommendation, 11 December 2012. <http://www.w3.org/TR/owl2-primer/> [retrieved: June, 2015]

[6] W3C, OWL 2 Web Ontology Language – Document Overview (Second Edition), W3C Recommendation, 11 December 2012. <http://www.w3.org/TR/owl2-overview/> [retrieved: June, 2015]

[7] F. Currim, S. Currim, C. E. Dyreson, and R. T. Snodgrass, “A Tale of Two Schemas: Creating a Temporal XML Schema from a Snapshot Schema with tXSchema,” Proceedings of EDBT’2004, Heraklion, Crete, Greece, 14-18 March 2004, pp. 348-365.

[8] R. T. Snodgrass, C. E. Dyreson, F. Currim, S. Currim, and S. Joshi, “Validating Quicksand: Schema Versioning in τXSchema,” Data and Knowledge Engineering, vol. 65, May 2008, pp. 223-242.

[9] W3C, XML Schema Part 0: Primer Second Edition, W3C Recommendation, 28 October 2004. <http://www.w3.org/TR/2004/REC-xmlschema-0-20041028/> [retrieved: June, 2015]

[10] C. E. Dyreson and F. Grandi, “Temporal XML,” in L. Liu and M. T. Özsu (Eds.), Encyclopedia of Database Systems, Springer US, 2009, pp. 3032-3035.

[11] Z. Brahmia, F. Grandi, B. Oliboni, and R. Bouaziz, “Schema Change Operations for Full Support of Schema Versioning in the τXSchema Framework,” International Journal of Information Technology and Web Engineering, vol. 9, April-June 2014, pp. 20-46.

[12] T. Burns et al., “Reference Model for DBMS Standardization, Database Architecture Framework Task Group (DAFTG) of the ANSI/X3/SPARC Database System Study Group,” SIGMOD Record, vol. 15, March 1986, pp. 19-58.

[13] C. Gutiérrez, C. A. Hurtado, and A. A. Vaisman, “Introducing time into RDF,” IEEE Transactions on Knowledge and Data Engineering, vol. 19, February 2007, pp. 207-218.

[14] F. Grandi and M. R. Scalas, “The valid ontology: A simple OWL temporal versioning framework,” Proceedings of the 3rd International Conference on Advances in Semantic Processing (SEMAPRO 2009), Sliema, Malta, 11-16 October 2009, pp. 98-102.

[15] M. J. O’Connor and A. K. Das, “A method for representing and querying temporal information in OWL,” In Biomedical Engineering Systems and Technologies, volume 127 of Communications in Computer and Information Science, pp. 97-110. Springer-Verlag, Heidelberg, Germany, 2011.

[16] V. Milea, F. Frasincar, and U. Kaymak, “tOWL: A Temporal Web Ontology Language,” IEEE Transactions on Systems, Man, and Cybernetics, Part B, vol. 42, February 2012, pp. 268-281.

[17] V. Milea, F. Frasincar, and U. Kaymak, “Knowledge Engineering in a Temporal Semantic Web Context,” Proceedings of the 8th International Conference on Web Engineering (ICWE 2008), Yorktown Heights, New York, USA, 14-18 July 2008, pp. 65-74.

[18] E. Anagnostopoulos, S. Batsakis, and E. G. M. Petrakis, “CHRONOS: A Reasoning Engine for Qualitative Temporal Information in OWL,” Proceedings of the 17th International Conference in Knowledge-Based and Intelligent Information & Engineering Systems (KES 2013), Kitakyushu, Japan, 9-11 September 2013, pp. 70-77.

[19] Z. Wu et al., “Implementing an Inference Engine for RDFS/OWL Constructs and User-Defined Rules in Oracle”, Proceedings of the 24th International Conference on Data Engineering (ICDE 2008), Cancún, México, 7-12 April 2008, pp. 1239-1248.

[20] J. Lu et al., “SOR : a practical system for ontology storage, reasoning and search”, Proceedings of the 33rd International Conference on Very Large Data Bases (VLDB 2007), University of Vienna, Austria, 23-27 September 2007, pp. 1402-1405.

[21] IBM, “Developing RDF Applications for IBM Data Servers”, January 2013. <ftp://ftp.software.ibm.com/ps/products/db2/info/vr101/pdf/en_US/DB2DevRDFdb2rdfe1011.pdf> [retrieved: June, 2015]

[22] A. Zekri, Z. Brahmia, F. Grandi, and R. Bouaziz, “τOWL: A Framework for Managing Temporal Semantic Web Documents Supporting Schema Versioning,” International Journal On Advances in Software, in press.

[23] W3C, RDF/XML Syntax Specification (Revised), W3C Recommendation, 10 February 2004. <http://www.w3.org/TR/2004/REC-rdf-syntax-grammar-20040210/> [retrieved: June, 2015]

[24] The Friend of a Friend (FOAF) project. <http://www.foaf-project.org/> [retrieved: June, 2015]

[25] W3C, Resource Description Framework (RDF), Semantic Web Standard. <http://www.w3.org/RDF/> [retrieved: June, 2015]

[26] M. Horridge and S. Bechhofer, “The OWL API: A Java API for OWL Ontologies”, Semantic Web, vol. 2, February 2011, pp. 11-21.

[27] A. Zekri, Z. Brahmia, F. Grandi, and R. Bouaziz, “Temporal Schema Versioning in τOWL,” Proceedings of the 2nd International Conference on Knowledge Management, Information and Knowledge Systems (KMIKS 2015), Hammamet, Tunisia, 16-18 April 2015, pp. 81-92.

Semantic OLAP with FluentEditor and Ontorion Semantic Excel Toolchain

Dariusz Dobrowolski
Faculty of Mathematics, Physics and Computer Science
Maria Curie Sklodowska University
Lublin, Poland
e-mail: dariusz.dobrowolski@umcs.lublin.pl

Paweł Kapłański
Department of Applied Informatics in Management
Gdansk University of Technology
Gdansk, Poland
e-mail: pawel.kaplanski@pg.gda.pl
Cognitum
e-mail: p.kaplanski@cognitum.eu

Andrzej Marciniak
University of Economics and Innovation
Lublin, Poland
e-mail: andrzej.marciniak@wsei.lublin.pl

Zdzisław Łojewski
Faculty of Mathematics, Physics and Computer Science
Maria Curie Sklodowska University
Lublin, Poland
e-mail: zdzislaw.lojewski@umcs.lublin.pl

***Abstract*—Semantic technologies appear as a step on the way to creating systems capable of representing the physical world as real time computational processes. In this context, the paper presents a toolchain for an ontology based knowledge management system. It consists of the ontology editor, FluentEditor and the distributed knowledge representation system, Ontorion. FluentEditor is a comprehensive tool for editing and manipulating complex ontologies that uses Controlled Natural Language (CNL). Its main feature is the usage of Controlled English as a knowledge modelling language. Ontorion is a Distributed Knowledge Management System with Natural Language interfaces (CNL) and a built-in rules engine. The Ontorion system is equipped with plugins for connection with other software environments, for example rOntorion using an R language package to access ontologies. It is exemplified with the semantic extension of On Line Analytical Processing (OLAP) using R language.**

Keywords- Semantic OLAP, Semantic, OLAP, Semantic Web, Ontorion, FluentEditor.

I. INTRODUCTION

Business Intelligence (BI) is a technology that enables the business to make intelligent, data-driven decisions. Intelligence here is governed by the laws of statistics that are applied on loosely coupled statistical variables, however to understand the meaning of data we need to link statistical variables to the real-life entities. This improvement can be implemented nowadays with aid of semantic technologies. As a result, we obtain the “smarter” BI system that represents the physical world as real time computational processes.

The remainder of the paper is structured as follows. In Section II, we present the semantic knowledge management framework that can be built with particular solutions available on the market. In Section III, we present the idea of OLAP - a powerful BI tool and its possible implementation in the R language. Semantic OLAP, the result of our research on bridging together both semantic toolkit and OLAP, is introduced and discussed in the Section IV, followed by the conclusion, in Section V.

II. SEMANTIC KNOWLEDGE MANAGEMENT FRAMEWORK

The expectations of business and science require new, global, flexible and much more effective technology of data exchange and processing. When the whole world is braided with effective communication links, what we need is a new efficient middleware working in the existing infrastructure but possessing new possibilities. After a decade of using file exchange systems, much experience was acquired. Very simple and easy rules of metadata connection gained great popularity.

Some factors should be mentioned which are important from our point of view:

- Easy exploitation: the end clients do not have any barriers;
- Accessibility: they could be used anywhere on many platforms and media;
- Effectiveness: acceptable from the point of view of the data receiver;
- Stability: information about resources must always be reliable;
- Independence: each node is completely autonomous within the system;
- Limitation of platform co-share: data are of a very simple form and the system is not able to provide co-sharing of more complex information.

The factors mentioned above indicate a tendency to recognize the meaning of a given resource, and in a later stage to the machine “understanding” of its content (i.e., ascribing semantic qualifiers to it enabling automatic decisive processes). The systems working in this layer use many technologies, which can be divided into the following categories:

- Natural language processing;
- Artificial intelligence and teaching machines;
- Ontologies;
- Meta-information, standardization and tagging documents.

By modelling domain ontologies with Semantic Web Rule Language (SWRL) [1] rules we are able to define a knowledge scheme in any semantic knowledge base. The store for the knowledge base can be implemented in Not only SQL (NoSQL) technology (e.g. Cassandra [2], Azure Tables [3]) or in Resource Description Framework (RDF) [4] data stores (e.g. AllegroGraph, Virtuoso) [5][6][7]. A relatively simple interface to model ontologies is supported by Protégé [8] or NeON [9] editors. Although these interfaces are rather simple for experienced practitioners, they are not for common users that do not know the nuances of ontology engineering. On the other hand, Semantic Rules Representation in CNL using FluentEditor [10] is the simplest way to represent knowledge in a natural language way. Nevertheless, using natural language is unattainable for the current technology and thus for the machines that should understand this knowledge. The most appropriate solution seems to use controlled natural languages.

A. *Ontorion SDK*

Ontorion [11] is a Distributed Knowledge Management System with Natural Language interfaces (CNL) and a built-in rules engine. It is compatible with Web Ontology Language 2 (OWL2) [12] and Semantic Web Rule Language (SWRL) and can be hosted in the Cloud or OnPremise environments. Ontorion is a family of products of server and client-side components for desktop and web allowing for the broad integration of custom software and existing corporate infrastructure. Ontorion performs real-time reasoning over the stream of data with the aid of an ontology that expresses the meaning of the given data (see Figure 1).

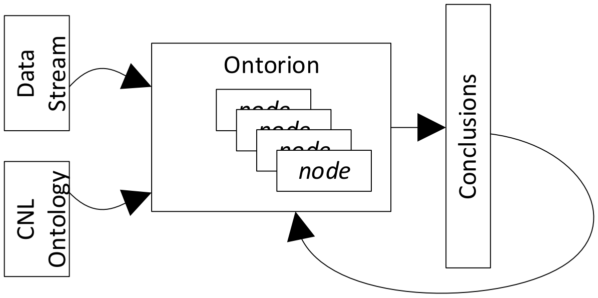


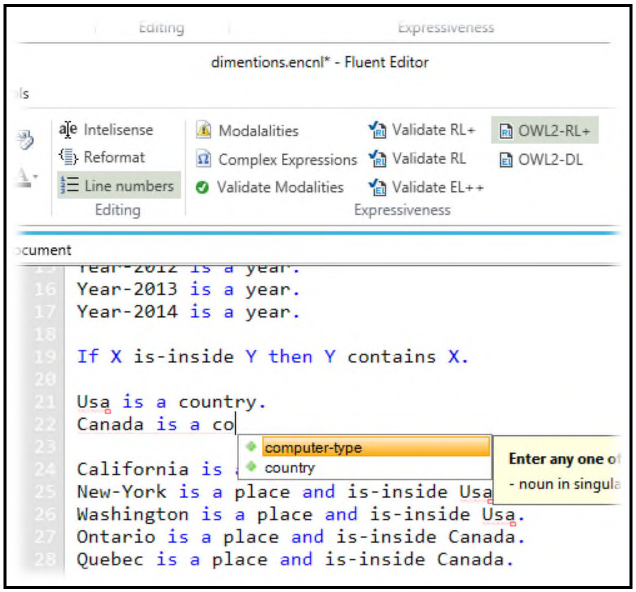
Figure 1. Ontorion [11] - Knowledge Management System

Ontorion is a set of components equipped with algorithms that allows one to build large, scalable solutions for the Semantic Web. The scalability is realized by both the NoSQL, symmetric database and the ontology Modularization algorithm. Modularization algorithm splits the problem into smaller pieces that are able to be processed in parallel by the set of computational nodes, therefore; Ontorion is a symmetric cluster of servers, able to perform reasoning on large ontologies. Every single Ontorion Node is able to make the same operations on data. It tries to get the minimal suitable ontology module (component) and perform the desirable task on it. Symmetry of the Ontorion cluster provides the ability for it to run in the "Computing Cloud" environment, where the total number of nodes can change in time depending on the user needs.

B. *FluentEditor 2014*

FluentEditor 2014, an ontology editor, is a comprehensive tool for editing and manipulating complex ontologies that uses CNL [13].

FluentEditor, shown in Figure 2, provides a more suitable alternative for human users to eXtensible Markup Language (XML)-based OWL editors. Its main feature is the usage of Controlled English as the knowledge modelling language. Supported via Predictive Editor, it prohibits one from entering any sentence that is grammatically or morphologically incorrect and actively helps the user during sentence writing.



```
Year-2012 is a year.
Year-2013 is a year.
Year-2014 is a year.

If X is-inside Y then Y contains X.

Usa is a country.
Canada is a co

California is
New-York is a place and is-inside Usa
Washington is a place and is-inside Usa.
Ontario is a place and is-inside Canada.
Quebec is a place and is-inside Canada.
```

Figure 2. Ontology of dimensions edited in FluentEditor 2014

Controlled English is a subset of Standard English with restricted grammar and vocabulary in order to reduce the ambiguity and complexity inherent in full English.

Main features:

- CNL OWL implementation: The implementation of CNL OWL - FluentEditor grammar is compatible with OWL-DL and OWL2
- OWL 2.0 full compliance: Full compliance with OWL 2.0 standard from W3C

- OWL API: Compatible with OWL API, which allows it to be used in cooperation with other tools
- SWRL compliance: The user can import existing ontologies from OWL files
- Dynamic referencing of external OWL ontologies: CNL documents can dynamically reference external OWLs from Web or disk.
- Predictive Edition Support: Users have enhanced support from the predictive editor
- Built-in dictionary: The built-in dictionary makes it easier to avoid misspelling errors

Some examples of other features are:

- Advanced user Interface, in order to open up semantic technologies for inexperienced users,
- In-place error resolving support - direct information about possible errors with hints on how to resolve them,
- Importing existing ontologies – users can directly import to CNL any external ontology
- Ambiguity resolution - it keeps track of ambiguities of concepts and/or instance names imported from different external ontologies.

C. *R language and Ontologies*

R language [14] is a widely used tool for statistical analysis. Combining ontologies and statistics opens an efficient way for the quantitative-qualitative analysis of data. It is possible to use both approaches conveniently in a single place by using an R language package to access ontologies (rOntorion). rOntorion R package allows direct access to ontologies created with FluentEditor and opens them for semantic processing in the R environment.

The R language plugins for FluentEditor are shown in Figure 3.

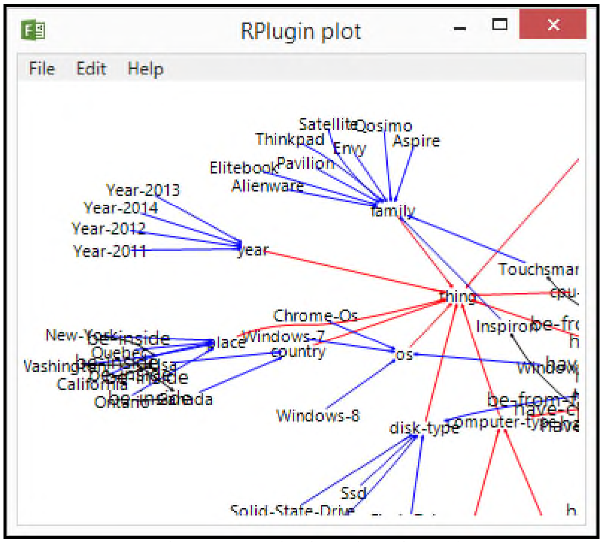
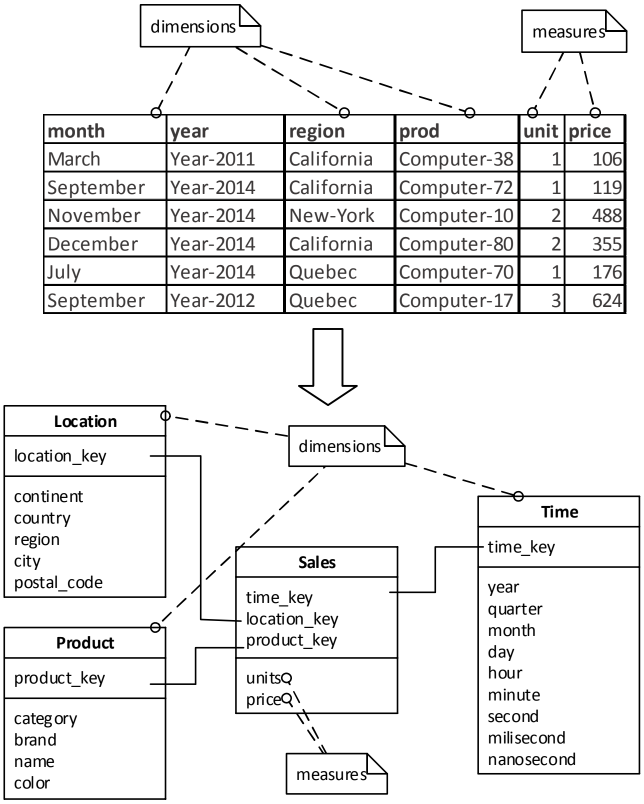


Figure 3. Graph of ontology from Figure 2

Beside development of analytical models with R and rOntorion it is also possible to build plugins for FluentEditor with R language. These plugins have direct access to the ontology within the editor host and can use any available R package. Plugins can display graphical results or textual output directly in FluentEditor.

III. Introduction To OLAP

OLAP is a well-known method [15] used in Business Analytics to provide decision makers with Online Access to Analytical Capabilities. It is based on the concept of data-cubes, multidimensional cubes of data that if equipped with tools allow the data and problems wherein to be explored. To create a datacube, we need data that can be represented in a STAR schema. The central table contains "measures" while "dimensions" are placed in surrounding tables (see Figure 4).



month	year	region	prod	unit	price
March	Year-2011	California	Computer-38	1	106
September	Year-2014	California	Computer-72	1	119
November	Year-2014	New-York	Computer-10	2	488
December	Year-2014	California	Computer-80	2	355
July	Year-2014	Quebec	Computer-70	1	176
September	Year-2012	Quebec	Computer-17	3	624

Figure 4. Transformation of a given dataset into the STAR schema (example)

To turn the data into a hypercube, we need to denormalize the STAR (by creating a single table) and what is put in each cell in the data-cube (hypercube) represents an aggregate value of measurements for a unique combination of each dimension. The aggregate is a function, e.g.: SUM, AVERAGE, MAX, MIN, COUNT, etc. See Figure 5.

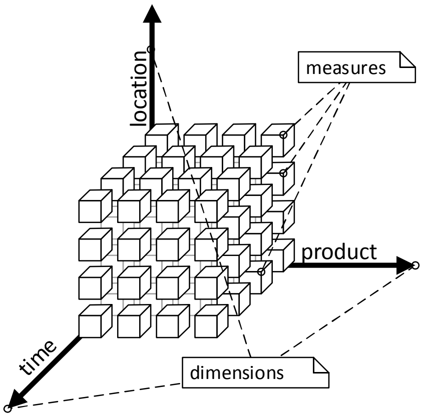


Figure 5. Extracting the data hypercube

Having the datacube we can slice and dice it (filter values), and rollup/drill-down/pivot over dimensions (see Figure 6).

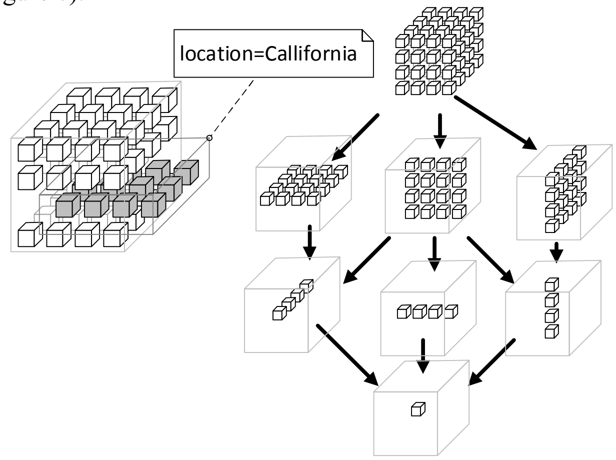


Figure 6. Slicing the data-cube over dimensions

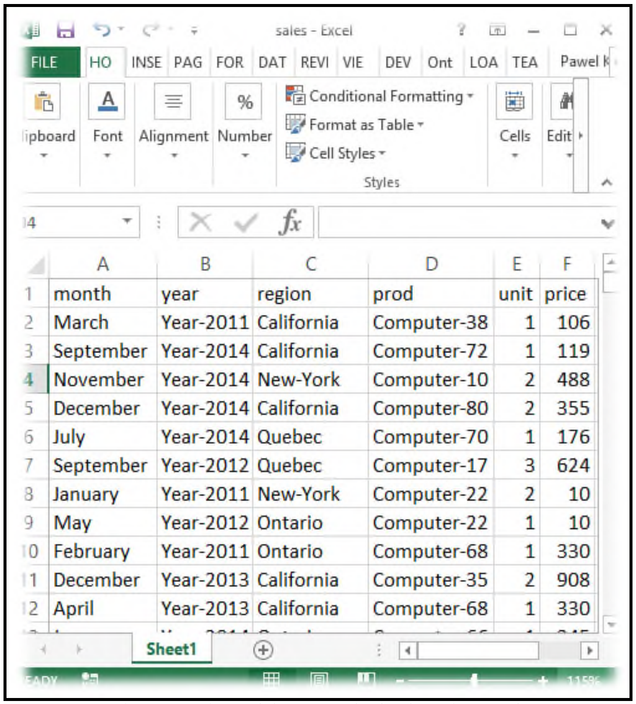
IV. The Semantic OLAP

By using the toolchain of FluentEditor and Ontorion SDK, it is possible to create something more than OLAP. We call it "Semantic OLAP", however, a solution delivered by Infotopics [16] is similar and it is called "natural query language".

In our case, the example application that implements the Semantic OLAP approach was built on top of the following tools:

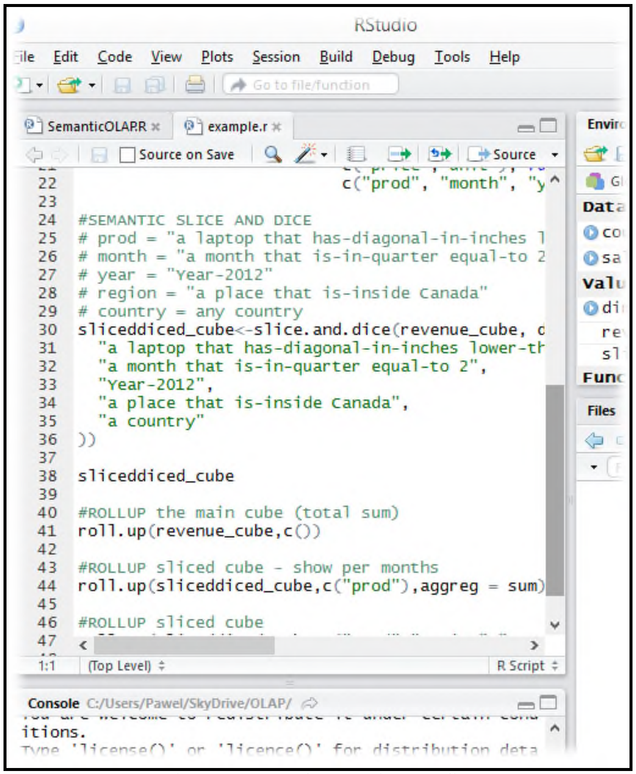
- Excel [17] – to create the database (see Figure 7)
- RStudio [18] – an open source integrated development environment (IDE) for R – to develop the software.

A piece of the code of queries is shown in Figure 8 and the result of the query from Figure 8 is displayed in Figure 9.



	A	B	C	D	E	F
1	month	year	region	prod	unit	price
2	March	Year-2011	California	Computer-38	1	106
3	September	Year-2014	California	Computer-72	1	119
4	November	Year-2014	New-York	Computer-10	2	488
5	December	Year-2014	California	Computer-80	2	355
6	July	Year-2014	Quebec	Computer-70	1	176
7	September	Year-2012	Quebec	Computer-17	3	624
8	January	Year-2011	New-York	Computer-22	2	10
9	May	Year-2012	Ontario	Computer-22	1	10
10	February	Year-2011	Ontario	Computer-68	1	330
11	December	Year-2013	California	Computer-35	2	908
12	April	Year-2013	California	Computer-68	1	330

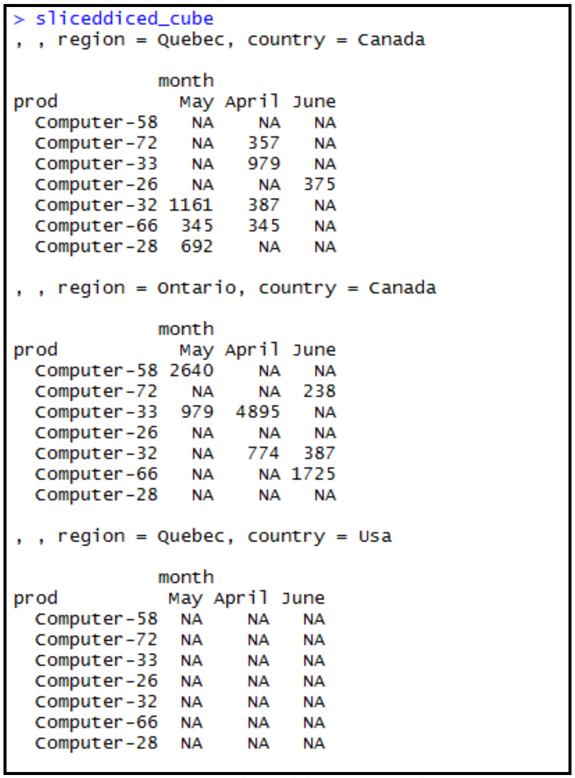
Figure 7. View of the example database in Excel



```
              c("prod", "month", "y

#SEMANTIC SLICE AND DICE
# prod = "a laptop that has-diagonal-in-inches l
# month = "a month that is-in-quarter equal-to 2
# year = "Year-2012"
# region = "a place that is-inside Canada"
# country = any country
sliceddiced_cube<-slice.and.dice(revenue_cube, c
  "a laptop that has-diagonal-in-inches lower-th
  "a month that is-in-quarter equal-to 2",
  "Year-2012",
  "a place that is-inside Canada",
  "a country"
))

sliceddiced_cube

#ROLLUP the main cube (total sum)
roll.up(revenue_cube,c())

#ROLLUP sliced cube - show per months
roll.up(sliceddiced_cube,c("prod"),aggreg = sum)

#ROLLUP sliced cube
```

```
itions.
Type 'license()' or 'licence()' for distribution deta
```

Figure 8. Example query in RStudio



```
> sliceddiced_cube
, , region = Quebec, country = Canada

             month
prod           May April June
  Computer-58   NA    NA   NA
  Computer-72   NA   357   NA
  Computer-33   NA   979   NA
  Computer-26   NA    NA  375
  Computer-32 1161   387   NA
  Computer-66  345   345   NA
  Computer-28  692    NA   NA

, , region = Ontario, country = Canada

             month
prod           May April June
  Computer-58 2640    NA   NA
  Computer-72   NA    NA  238
  Computer-33  979  4895   NA
  Computer-26   NA    NA   NA
  Computer-32   NA   774  387
  Computer-66   NA    NA 1725
  Computer-28   NA    NA   NA

, , region = Quebec, country = Usa

             month
prod          May April June
  Computer-58 NA    NA   NA
  Computer-72 NA    NA   NA
  Computer-33 NA    NA   NA
  Computer-26 NA    NA   NA
  Computer-32 NA    NA   NA
  Computer-66 NA    NA   NA
  Computer-28 NA    NA   NA
```

Figure 9. View of the query result

I. CONCLUSION

The semantic extension of OLAP is proved to be fully functional using the toolchain of domain ontology, FluentEditor and the distributed knowledge representation system, Ontorion combined with, e.g., Excel as a source of data and RStudio for OLAP. Moreover, it created the foundations for already available on the market, developed and maintained by Cognitum, a solution called Ask Data Anything (ADA) [19]. The ADA allows exploring data by using natural language directly, rather than by using CNL, therefore we classify ADA as a tool that allows to explore data with natural language.

The modern approach to BI called BigData, is currently understood to face the problem of *“(...) growing number of insights that are being produced by big data through automated forms of analysis (...) What happens to the thousands of insights that are being generated automatically by all of those nifty machine learning algorithms? How do they find their way to a person at the right time?“* [20]. Semantic OLAP as well as its successor called ADA proves that the problem can be solved with support of semantic technologies.

REFERENCES

[1] SWRL: A Semantic Web Rule Language Combining OWL and RuleML. [Online]. Available: http://www.w3.org/-Submission/SWRL/ [retrieved: 1 june, 2015]

[2] A. Lakshman and P. Malik, “Cassandra: a decentralized structured storage system,” *Operating Systems Review*, vol. 44, no. 2, pp. 35–40, 2010. [Online]. Available: http://-dblp.uni-trier.de/db/journals/sigops/-sigops44.html#LakshmanM10

[3] S. Krishnan, *Programming Windows Azure*. " O’Reilly Media, Inc.", 2010.

[4] W3C. Rdf 1.1 primer. [Online]. Available: http://-www.w3.org/TR/2014/NOTE-rdf11-primer-20140225/ [retrieved: 1 june, 2015]

[5] Franz Inc. (2010) AllegroGraph RDFStore Web 3.0’s Database. [Online]. Available: http://www.franz.com/-agraph/allegrograph/

[6] O. Erling and I. Mikhailov, “RDF support in the virtuoso DBMS networked knowledge - networked media,” ser. Studies in Computational Intelligence, T. Pellegrini, S. Auer, K. Tochtermann, and S. Schaffert, Eds. Berlin, Heidelberg: Springer Berlin / Heidelberg, 2009, vol. 221, ch. 2, pp. 7–24. [Online]. Available: http://dx.doi.org/-10.1007/978-3-642-02184-8_2

[7] D. Dobrowolski and M. Lesnik, “Social graphs in acquiring knowledge,” *Zeszyty Naukowe Uniwersytetu Szczecinskiego. Ekonomiczne Problemy Uslug*, no. 87, pp. 34–41, 2012.

[8] M. Musen, N. Noy, C. Nyulas, M. O’Connor, T. Redmond, S. Tu, T. Tudorache, J. Vendetti, and S. S. of Medicine, “Protege,” 2010, [http://protege.stanford.edu]. [Online]. Available: http://protege.stanford.edu

[9] P. Haase, H. Lewen, and R. Studer, “The neon ontology engineering toolkit.”

[10] Cognitum, “Fluent Editor 2014 - Ontology Editor.” [Online]. Available: http://www.cognitum.eu/semantics/-FluentEditor/ [retrieved: 1 june, 2015]

[11] ——. Ontorion Semantic Knowledge Management Framework. [Online]. Available: http://www.cognitum.eu/-semantics/ontorion/ [retrieved: 1 june, 2015]

[12] P. Hitzler, M. Krotzsch, B. Parsia, P. F. Patel-Schneider, and S. Rudolph, “OWL 2 Web Ontology Language Primer,” World Wide Web Consortium, W3C Recommendation, October 2009. [Online]. Available: http://www.w3.org/TR/owl2-primer/

[13] A. Wroblewska, P. Kaplanski, P. Zarzycki, and I. Lugowska, “Semantic rules representation in controlled natural language in fluenteditor,” in *Human System Interaction (HSI), 2013 The 6th International Conference on*. IEEE, 2013, pp. 90–96.

[14] R. Gentleman and R. Ihaka. R lanugage. [Online]. Available: http://www.r-project.org/ [retrieved: 1 june, 2015]

[15] S. Chaudhuri and U. Dayal, “An overview of data warehousing and olap technology,” *ACM Sigmod record*, vol. 26, no. 1, pp. 65–74, 1997.

[16] Infotopics. Natural query language. [Online]. Available: http://www.infotopics.nl/infotopics-tableau-blog/entry/-project-stel-een-vraag-aan-tableau [retrieved: 1 june, 2015]

[17] D. Z. Meyer and L. M. Avery, “Excel as a qualitative data analysis tool,” *Field Methods*, vol. 21, no. 1, pp. 91–112, 2009.

[18] RStudio, “Rstudio ide - a powerful and productive user interface for r.” [Online]. Available: https://-www.rstudio.com/ [retrieved: 1 june, 2015]

[19] Cognitum. (2015) Ask data anything. [Online]. Available: http://techblog.cognitum.eu/2015/05/ask-data-anything.html [retrieved: 1 june, 2015]

[20] D. Woods. (2015) Why big data needs natural language generation to work. Forbes. [Online]. Available: http://-www.forbes.com/sites/danwoods/2015/07/09/why-big-data-needs-natural-language-generation-to-work/ [retrieved: 1 june, 2015]

Information Extraction from Unstructured Texts by means of Syntactic Dependencies and Constituent Trees

Raoul Schönhof, Axel Tenschert, Alexey Cheptsov

High Performance Computing Center Stuttgart,

University of Stuttgart

Stuttgart, Germany

e-mail: raoul.schoenhof@b-f-u.de, tenschert@hlrs.de, cheptsov@hlrs.de

Abstract—**This paper presents a technology of automated knowledge extraction from unstructured text corpora by leveraging computer linguistic tools and cross-fertilizing them with the semantic ontologies techniques. In our approach, the quality of information (e.g., in form of OWL ontologies) that is derived by semantic analysis techniques from large domain-specific text corpora can be considerably improved by incorporating linguistic analysis tools that help gain a deeper insight into the grammatical structure of the analysed texts and thus allow the reasoning engines to cover a much wider set of rules and patterns, also positively impacting the performance. The novelty of our approach lies in a possibility of its application to the domains that require a very high quality of the knowledge extraction and analysis, such as reasoning for legacy data collections. We propose a system architecture for the implementation of our approach and illustrate its use on a practical use case for legislative and regulatory information analysis.**

Keywords-***Knowledge Representation; Legal Systems; Ontology; OWL; Big Data; Reasoning; DreamCloud Project.***

I. INTRODUCTION

Many domains use texts collected and stored in the natural language as a primary (or, in some cases, the only) trustful source of the domain-specific knowledge. In some cases, this is caused by historical reasons, when the knowledge collection had started long time before the computer standards that allows for a certain level of automation were introduced. In the other cases, the automated storage and processing was impossible by commodity analysis tools due to the complex grammatical constructions used in the texts, as well as their sizes. Whereas the newly-emerged standards like Resource Description Framework (RDF) [1] have enabled tackling with the complex issues of textual representation in the ontologically-understandable format of a logical triple "subject→predicate→object", the information extraction from grammatically-complex texts remains a major challenge, especially for the domains that require a high precision of the information representation and formalization like law system, patent management, etc. The Semantic Web approach has shown how the information from unstructured sources on the web can be extracted and then used in a wide range of applications, from search and filtering engines to complex reasoning systems that aim to derive new knowledge that is not explicitly provided in the initial variant of texts. A lot of satellite techniques have also appeared around this topic, including the ontology construction tools like Protégé [2], semantic databases like Jena [3], reasoners like Pellet [4], etc. However, the issue of dealing with complex grammatical constructions remains being an essential drawback to promoting those technologies into a wider range of application domains that deal with complex texts analysis.

Most of the information retrieval methods and techniques, such as the language modeling [5], do not consider the grammatical structure of the sentence. However, the latest advances of natural language processing (NLP) technologies, e.g., from the Stanford NLP Group [6], allow those techniques to take some advantages of the grammar-based analysis. In particular, the analysis technologies can be leveraged in the following ways:

- generative grammar tools [7] can be used for extracting functional terms used in the text,
- dependency grammar tools [8] can be used to derive complex connections in form of relations between two words within a sentence.

The remainder of the paper is structured as follows. Section II introduces the state of the art, with focus on computer linguistic tools and related semantic web technologies (such as OWL and SWRL). Section III discusses our analysis approach and presents the design of our system's prototype. Section IV describes an exemplary scenario based on legislative and regulatory information [9][10] analysis domain, demonstrating the usage of the system's prototype. Section V presents conclusions.

II. STATE OF THE ART

A. *Information Retrieval Systems for Ontology Generation*

The amount of information is constantly increasing but only available in an unstructured format. Mostly, the information is hiding in natural language texts. There have been numerous approaches to retrieve information from documents and texts, deriving an RDFS/OWL graph. To the most popular approaches belong Text2Onto [11], OntoLearn/OntoLearn Reloaded [12] OntoMiner [13] and OntoLT [14]. The OntoMiner approach analyzes regularities from HTML Web documents. A substantial disadvantage is the requirement of a handpicked set of web sites within the admired field of interest. The output taxonomy is strictly hierarchical, which is appropriate to classify entities, but it cannot find a considerable amount of relations between entities inside a level in the hierarchy. Interconnections are necessary performing complex reasoning tasks. The same situation looms with regards to the OntoLearn Reloaded approach. Much more promising is the approach of Text2Onto and OntoLT. Text2Onto combines machine learning strategies with basic NLP methods, particularly tokenization, lemmatizing and shallow parsing [11], allowing the application to analyze a natural language text more detailed. Testing Text2Onto has demonstrated, that the retrieved amount of information was not enough, with regards to the field of interest. Beside Text2Onto, even OntoLT was using NLP technology, above the task of named-entity-recognition, to generate semantic networks [15]. Hereby, OntoLT was using predefined mapping rules for every desired annotation tag. OntoLT then constructs an OWL ontology according to the given mapping rules [15]. According to our knowledge, OntoLT has not been extended since 2007.

The presented approach affiliates this concept of grammatical-driven information retrieval, implements state of the art NLP tools and expands it by considering grammatical dependencies for information retrieval to achieve a higher precision and applying it to the field of law. Using dependencies for information retrieval, the approach benefits by additional information about the semantic content of text [16]. Our approach is based on three pillars: Linguistic, Computer Science, and Law. The first two pillars offer technologies while the third one a use case. In the following subsections, we concentrate on technological challenges of the analysis technologies.

B. *Linguistic Tools and Syntax Theories*

P. G. Otero [16] presents an approach for exploiting human-written text by computers, according to which it is necessary to examine the structure of each sentence considering the dependency syntax. In the last decade, the linguistic tool development has been established very well, especially with regard to grammatical parsers. For example, the Stanford NLP Group offers a comprehensive toolset for various aspects of grammatical sentence parsing [6]. For our goals, we took a closer look at four types of computer linguistic tools: i) constituency parsers, based on the generative grammar, ii) dependency parsers, based on the dependency grammar, iii) named entity recognizers, used for locating and classifying entities in text, and iv) sentence splitters.

Constituency Parser. Constituency parsers are based on the idea of splitting a sentence in functional units called constituents [7]. The resulting tree of superior and subordinated constituents generates a tree-like structure, which is mapped as an Augmented Transition Network (ATN) [17]. ATN offers a flexible and scalable technology to represent the grammatical structure of sentences. It disassembles a sentence into constituents (see Figure 1) and tags them. A very common constituency parser is the Stanford Parser [18]. It supports various languages, including English, German, Chinese, and Arabic. An example of ATN for the sentence "*A computer is a machine.*" is shown in Figure 1, by using the constituent tags from the Penn Treebank Project [19][20]. The sentence (S) is divided in two "sub-constituents", here the noun phrase (NP) and the verbal phrase (VP). These contain either atomic words or other constituents. Here, the determiner (DT) "*A"*, the noun (NN) "*computer"*, the verb (VBZ) "*is",* the noun "*machine*" represent atomic words, whereas "*A computer*" or "*a machine*" form a noun phrase. In combination with a verb, the constituent VBZ and NP, here "*is a machine*", form a verbal phrase.

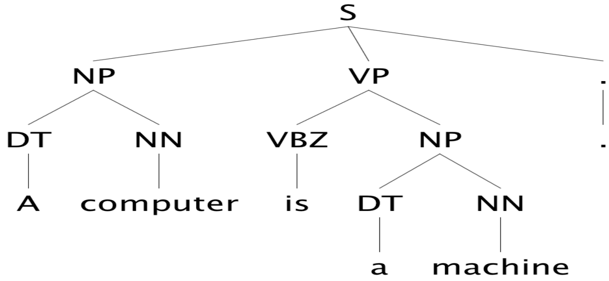


Figure 1. ATN Example based on Stanford Parser GUI

Dependency Parser. Dependency parsers are based on the dependency grammar [8], which focuses on relationships between words and their functional role within a sentence [21]. Relations can be represented in a form of a directed graph, which makes it possible to derivate a hierarchy [21]. Because the structure of the hierarchy is only depending on the grammatical syntax, it is also possible to conclude to the semantic [16], e.g., Figure 2 shows an example sentence with its dependencies and constituents.

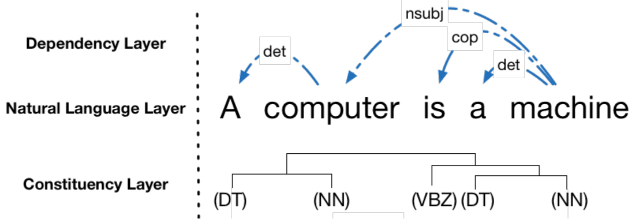


Figure 2. Pattern of a sentence

The dependencies in a sentence are presented as a tree of connected word tags being knots. Hereby, the dependency tags *det*, *nsubj* and *cop* stand for *determiner*, *nominal subject* and *copula* [22] and provide additional information about the type of grammatical relations. Basically, this pattern is representative for a sentence of the type "*Object*

→ *Subject*”. Figure 2 shows the resulting dependency pattern. The abstract pattern helps finding sentences with the known information structure. The identified words then need to be transferred into a more machine-recognizable format. This is not only useful for identification of classes and their subclasses but also with regard to the "valence theory" [8]. Origin of this theory is the empirical knowledge of the structure-determining characteristic of verbs as presented by L. Tesnière. According to this and exposed by H. M. Mueller et al. [21] and V. Ágel [24], each word or word group is typically associated with a verb in the sentence. Therefore, dependencies could also help identifying the actions (= verbs) of individuals in the sentence.

Named Entity Recognition. Named Entity Recognizers (NER) are tools to identify typical non-context related individuals, e.g., locations ("Berlin", "Hong Kong"), organizations ("UNICEF", "NASA") or person names ("Lisa", "Rouven"). Therefore, NERs, like the Stanford NER, are using Conditional Random Fields (CRFs) to identify entities [25]. With regards to our approach, NERs are not essential but an improvement area to gather additional information helping to find some individuals, which could not be found by only focusing on ATN-trees or dependency structures.

Sentence Splitting. Typically, a text contains many sentences; in order to analyze them, they have to be separated. This task is performed by sentence splitters. One of the most popular sentence splitters is provided by A Nearly-New Information Extraction System (ANNIE) [26] - a software package of the GATE project. This splitter can distinguish between a full stop and any other point.

C. *Working with Information*

While ATN, NER, and other dependency parsers can derive some useful information about the texts' structure, the ontology languages facilitate information representation. Ontologies can be leveraged to text to identify classes, individuals, or even properties in them. Alongside with that, ontology-based analysis frameworks provide tools that allow for querying the retrieved information.

1) *Web Ontology Language*

OWL provides a framework to store and handle information by ontologies [27]. OWL is based on the RDF [1] and equipped with an additional vocabulary [28]. Each OWL ontology can represent different kinds of information, e.g., classes, individuals or properties. While classes express abstract concepts, individuals are existing members of one or more classes. The relations between other individuals are defined by their properties. Therefore, OWL is predestinated to use ontologies with reasoning algorithms. [27]

2) *Semantic Web Rule Language*

As a special sublanguage of OWL, the SWRL represents abstract rules associating OWL individuals to any desired OWL class. Special forms of these rules are built-in relations. These rules consist of an antecedent, called "*body*", and a consequence, called "*head*". Several OWL individuals of an ontology can hereby be associated with another class [29]. This enables the use of very complex rules. A little example to illustrate: "*If a device contains a CPU, then it is a computer*". Therefore, an individual of the class "*device*" is defined as "*computer*" if this individual is connected to another individual of the class "*CPU*" by the object property "has*Contain*". The resulting SWRL Rule would be (1).

$$device(?x) \wedge hasContain(?x,?y) \wedge CPU(?y) \\ \Longrightarrow computer(?x) \tag{1}$$

III. SYSTEM ARCITECTURE AND DESIGN

A. *General Overview*

The system concept aims to identify applicable laws for a given use case by extracting information fully automated from natural texts. The whole system contains three components shown in Figure 3: the Sentence Processing Unit, the Pattern Interpreter and the Reasoner, which is currently in progress.

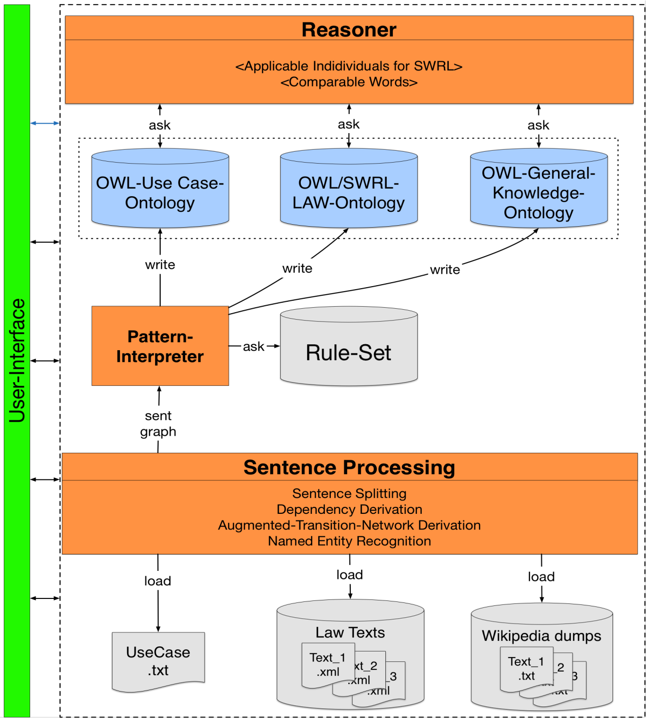


Figure 3. System Architecture

The first component represents the Sentence Processing Unit. It is basically a conglomeration of different language processing tools containing the sentence splitter from ANNIE/GATE [26], the dependency and constituency parser from the Stanford NLP Group [23], as well as a Named Entity Recognizer. The second component is the Pattern Interpreter (see Figure 3). It builds three OWL models out of natural texts. The first ontology contains the information about the questionable use case (OWL - Use

Case Ontology). The second one contains the laws, respectively the legal prerequisites, represented as SWRL Rules (OWL/SWRL - LAW Ontology). The third ontology contains general knowledge, mainly about classes and subclasses (OWL - General - Knowledge). Finally, the third component is the reasoning process, respectively the reasoner. Hereby, the reasoner tries to match the given information based on the Use Case Ontology with the rules from the LAW Ontology. Because laws are written in a notional way, it is necessary to establish a connection between the individual of the use case and the SWRL rule. The General Knowledge Ontology provides this connection. The strict separation between the Use Case Ontology and the General Knowledge Ontology is necessary because the correctness of the given information in a random use case cannot be assumed.

B. Sentence Processing

Starting point is the raw data, which contains texts with one or many sentences. The source of the texts might be Wikipedia [30], law texts [10], or any other texts related to the topic of our use case. These texts have to be processed, so the sentence structure, defined as pattern p, can be mapped. Each pattern $p \in P = \{d, A\}$ is represented by a subset of dependencies $d \in D = \{s, p, o\}$ and an ATN A. Hereby, d is described by triples, consisting of a subject s, a predicate p and an object o. While s and o are words, p belongs to a dependency tag, also shown in Figure 2. Therefore, each text passes through the ANNIE sentence splitter of the text-engineering tool GATE [26]. The constituent and dependency parsers then analyze the isolated sentences. Afterwards, the atomic words will be exchanged against their lemma projecting the numerous variants of a concept to a single lemma. Therefore, the complexity of the dictionary is reduced.

C. Pattern Interpreter & Rule-Set

The Pattern Interpreter translates a given sentence to a machine-recognizable OWL ontology, based on its pattern. The resulting OWL ontology is representing the base for any reasoning attempts. Hereby, the Pattern Interpreter compares the grammatical structures of a given sentence from a set of predefined grammatical patterns, called Rule-Set, to derive the OWL ontologies mentioned in Figure 3. If a pattern could be identified, the Pattern Interpreter converts the words as OWL Classes, OWL Individuals or SWRL Rules and interconnects them. The axioms are stored in different OWL ontologies. This is essential because the given information from a use case do not have to be true. One of the most difficult tasks is the development of the Rule-Set. This set contains patterns of typical sentence structures, as well as corresponding instructions. They can be described as follows:

Let $rs \in RS = \{p, i\}$ be the Rule-Set, which contains pattern p and instruction i. The instruction describes the connection between the words as and OWL model, by generating OWL's Classes, Individuals, and Predicates. Because of the complexity of natural language, the patterns cannot exist statically (therefore, one pattern for each type of sentence) but must be composed from different rules. This process could be demonstrated at the following example.

Let's apply Rule-Set that to the text mentioned in Figure 2 ("a computer is a machine"). The first rule $rs_1(p_1, i_1)$ contains pattern p_1 that describes a noun (*computer*) referencing to another noun (*machine*) using the dependency *nominal subject*. The corresponding instruction i_1 defines the first noun as a subclass of the second one:

$$i_1 := SubClassOf(Computer, Machine). \quad (1)$$

Now, let's add to our Rule Set another rule rs_2 specifying the connection between two nouns by means of the dependency "*compound*", like shown in Figure 4. The pattern is typical for compound nouns like "*computer system*" or "*street light*".
The instruction for this rule will be the following:

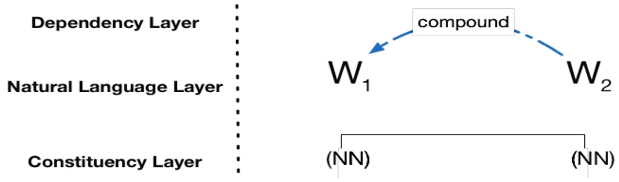


Figure 4. Pattern of rs$_2$

$$i_2 := Class(ComputerSystem). \quad (2)$$

Now, when trying to apply these both rules rs_1 and rs_2 to a more complex sentence like "a computer system is a machine", see Figure 5, we'll see that none of them is able to cope with the more complex grammatical structure of the new text. Therefore, the initial rule set should extended by more complex rules, based on the simple patterns discussed above.

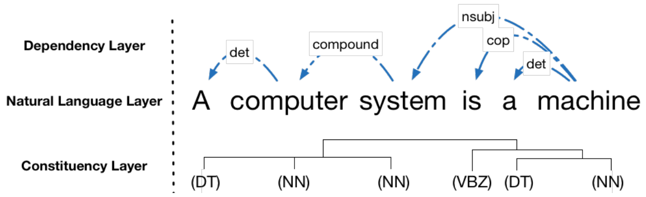


Figure 5. Joint pattern of rs$_1$ and rs$_2$

Hereby, the selection of several applicable rules follows the principle of speciality, according to which a more complex rule can be created based on the more simple one. The described patterns exist currently just in hard-coded form to proof the concept. Later, it has to be derivated by automated or semi-automated machine learning algorithms.

D. Reasoner with OWL Ontologies

Main task of the reasoner is the identification of connections between the given case and the law ontology. Therefore, the reasoner has to find a conclusive path through the OWL tree. The results of the pattern interpreter are, depending of the input source, three OWL ontologies.

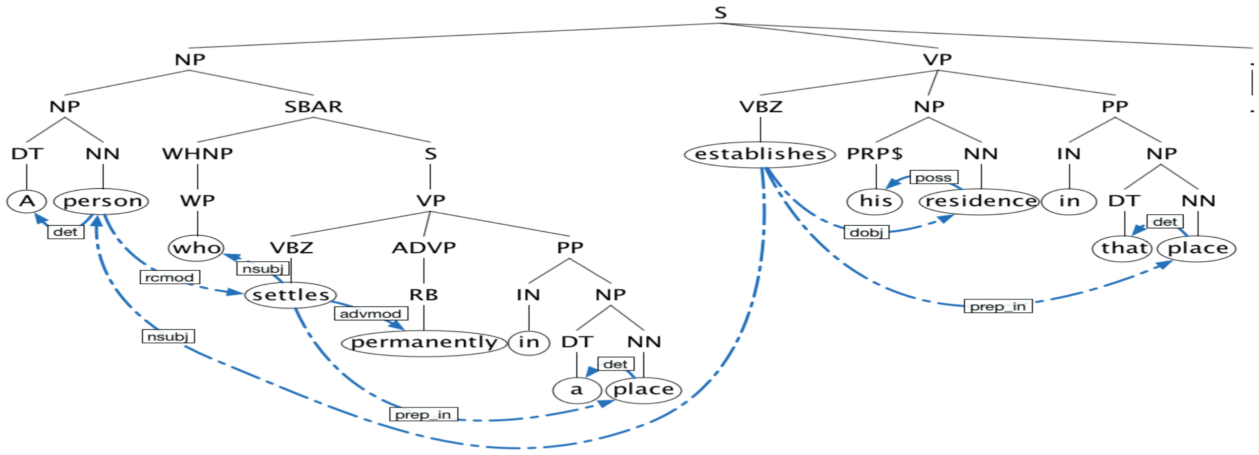


Figure 6. § 7 I BGB, parsed by the Stanford Parser GUI complemented with the dependency relations based on the Stanford Dependency Parser

The record information, like individuals and their actions, is represented in the use case ontology. Information about the laws is given in the law ontology, mainly as classes and SWRL built-in rules. In this state, it would be impossible to find a connection between the given case and the abstract rule. Therefore, it is necessary to bridge the missing links through additional information about the given case. Classes must be linked to hyper- and subclasses, properties like verbs must be associated with other properties. This information shall be extracted by analyzing wikidump files and stored to the General Knowledge Ontology [30]. The following example shall illustrate the interaction.

If an individual named "*bicycle*" is given in the Use Case Ontology, as well as a SWRL rule requiring an individual of the class "*thing*"; the General Knowledge Ontology contains necessary information about the hyperclasses of "*bicycle*". One of them is the hyperclass "*thing*". Therefore, the individual of the class "*bicycle*" can be used for a SWRL rule, which requires an individual of the class "*thing*".

When working with large amount of information by converting texts from natural language to an OWL model, it is likely to find an inconsistency. This circumstance is not only the result of potential mistakes in the information extraction process, but also inducted by contradictory statements in a text. The problematic becomes obvious with regard to paragraph 90a of the German Civil Code [9]. It declares that animals are not things even though laws for things shall be applicable for animals as well. Therefore, the reasoning process will have to work with such types of inconsistencies. This problem could be solved by creating and solving two ontologies in parallel, where just one critical statement at a time is given. The result of this type of reasoning would not be a logical but a conclusive solution.

IV. EXEMPLARY USE CASE SCENARIO

We would like to illustrate the application of our system for the analysis of the following text from a paragraph (§7) of the German Civil Code: "*A person who settles permanently in a place establishes his residence in that place.*" [9]. At first the sentence passes through the sentence processing unit, which derives an ATN and the dependencies, shown in Figure 6. In addition, the words (tree leaves) are exchanged to their lemma in order to reduce the complexity for reasoning tasks. Root point of the ATN is the constituent sentence (S). It consists of a noun phrase (NP), as well as a verbal phrase (VP). Here, the ATN depicts the difference between the legal prerequisite, the noun phrase (NP), and the legal consequence, the verbal phrase (VP). The dependency tree shows the relation between words. Root point of this dependency tree is the verb "*establishes".* The root point is outstanding, because it has no dependency pointing at it, but one or more, which point away from it. This verb declares the action "*establishes*" for the nominal subject (nsubj) "*person*". But this noun is restricted by a sub-ordered conjunction (SBAR) [19]. Here, the noun "*person*" is getting conditioned by the clause "*settle permanently in a place*". Hereby, "*settle*" itself refers firstly to "*place*" via the preposition "*in*" (prep) and secondly to its modifier "*permanent*". The legal consequence of this sentence is contained in the verbal phrase. The verb "*establish*" in connection with the direct object (dobj) "*residence*". The main task of the pattern interpreter is to look if patterns, given from the pattern set, could match in this constituents and dependency tree. At this point of time, the actual words, respectively the content of this sentence, does not matter anymore. Depending on the pattern, nouns are converted to OWL classes or individuals. Here, Figure 6 shows three types of nouns: "*person*", "*place*" and "*residence*". By treating these nouns as OWL classes, it is possible to associate individuals to them. Beside nouns, verbs are converted to OWL object properties. The given sentence contains just the two verbs "*establish*" and "*settle*", which is restricted by the adverbial modifier (advmod) "*permanent*". In SWRL, the first noun phrase is true if there are two individuals, one of the class "*person*" and one of the class "*place*", which are connected by an object property "has*SettlePermanent*", see equation 8. The antecedent of the SWRL rule contains classes "*person*"

$$person(?x) \land hasSettlePermanent(?x,?y) \land$$
$$place(?y) \Longrightarrow \; hasEstablishResidence(?x,?y) \qquad (3)$$

and "*place*", which are connected by the object property "has*SettlePermanent*", as consequence the individual of the

ISBN: 978-1-61208-420-6

class "*place*" is also declared as individual of the class "residence". Also the new object property "*EstablishResidence*" will be inserted and connects then "*person*" and "*place*".

V. Conclusion

In the paper, we showed how the ontology-based reasoning techniques can be improved by leveraging the syntactical analysis tools. A system architecture, as well as a use case scenario from the law domain were presented. As a proof-of-concept, a prototype implementing the system architecture was implemented based on the Java toolset from the DreamCloud project [31] was equipped with a hard coded rule set. The prototype was used to identify i) abstract concepts as OWL classes, ii) persons and specific entities as OWL individuals, and iii) verbs as OWL object properties correctly. The resulting ontology was tested with the Pellet reasoner and further, the use of the presented approach for handling simple unstructured texts was performed successfully. The described work serves mainly as a foundation for further research and development activities.

Future tasks will focus on several issues like implementing the reasoner and enhancing the presented approach by not only considering isolated sentences but extending the sentence analysis by broadening its scope and applying it on paragraphs as a whole and full texts. In addition, the currently hard coded rule set will become a flexible more complex one containing a wide range of rules customized to the given context through adapting automated methods composed by making use of machine learning concepts and algorithms for generating tailor made rules. After the rule set is more flexible, a detailed evaluation will be done. Besides the full text analysis and the enhanced rule set generation the presented approach will be extended by taking into account a thesaurus for improving the general knowledge ontology and thus providing the reasoner with additional information regarding language and meaning of terms.

The work done in the scope of this paper and the future developments will conclude in a flexible, syntactic dependencies and constituent tree handling, as well as meaning aware reasoning system being able to handle laws and further being applicable to other unstructured text types.

Acknowledgment

Authors would like to thank the consortium of the EU-ICT research project DreamCloud (Dynamic Resource Allocation in Embedded and High-Performance Computing) for the support with the Java platform and first prototype of our software system development.

References

[1] About RDF: www.w3.org/RDF/ (retrieved: 03, 2015).

[2] About Protégé: www.protege.stanford.edu/ (retrieved: 02, 2015).

[3] About Jena: jena.apache.org/ (retrieved: 03, 2015).

[4] About Pellet: clarkparsia.com/pellet/ (retrieved: 03, 2015).

[5] J. M. Ponte and W. B. Croft, "A Language Modeling Approach to Information Retrieval", University of Massachusetts, http://ciir.cs.umass.edu/pubfiles/ir-120.pdf (retrieved: 03, 2015).

[6] Stanford NLP Tools: http://nlp.stanford.edu/software/index.shtml.

[7] A. Carnie, "Synatx - A Generative Introduction", Third Edition, chapter 1 - 3, Published by Wiley-Blackwell 2013, ISBN 978-0-470-65531-3.

[8] L. Tesnière, "Eléments de syntaxe structurale", Librairie C. Klincksieck, Paris, 1959, Translation: U. Engel: L. Tesnière - Grundzüge der strukturalen Syntax, Published by Klett-Cotta Stuttgart, 1980, ISBN 3-12-911790-3.

[9] N. Mussett, Federal Ministry of Justice and consumer protection Germany: German Civil Code BGB, Date: 01.10.2013, published in: http://gesetze-im-internet.de/englisch_bgb (retrieved: 02, 2015).

[10] Law texts: http://gesetze-im-internet.de/Teilliste_translations.html.

[11] P. Cimiano and J. Völker, "Text2Onto A Framework for Ontology Learning and Data-driven Change Discovery", 2005.

[12] P. Velardi, S. Faralli, and R. Navigli, "OntoLearn Reloaded: A Graph-based Algorithm for Taxonomy Induction", Computer Linguistics, Vol. 39, No. 3, August 2013, pp. 665-707.

[13] H. Davulcu, S. Vadrevu, and S. Nagarajan, "OntoMiner: Bootstrapping and Populating Ontologies From Domain Specific Web Sites", The First International Workshop on Semantic Web and Databases, 2003, pp. 24-33.

[14] P. Buitelaar, D. Olejnik, M. Sintek, "OntoLT: A Protégé Plug-In for Ontology Extraction from Text", International Semantic Web Conference (ISWC), 2003, pp. 31-44.

[15] P. Buitelaar, D. Olejnik, M. Sintek, "A Protégé Plug-In for Ontology Extraction from Text Based on Linguistic Analysis", First European Semantic Web Symposium (ESWS), Heraklion, Greece, May 2004.

[16] P. G. Otero, "The Meaning of Syntactic Dependencies", Linguistik online, 2008, Vol. 35, Issue 3, pp. 33-53, ISSN 1615-3014.

[17] W. A. Woods, "Transition network grammars for natural language analysis", Communications of the ACM, Vol. 13, Issue 10, Oct. 1970, p. 591-602**.**

[18] Stanford Parser: http://nlp.stanford.edu/software/lex-parser.shtml.

[19] M. P. Marcus, B. Santorini, and M. A. Marcinkiewict, "Building a Large Annotated Corpus of English: The Penn Treebank", Computational Linguistics - Special issue on using large corpora, Vol. 19, pp. 313-330, June 1993.

[20] A. Bies, M. Ferguson, K. Katz, and R. MacIntyre: http://www.sfs.uni-tuebingen.de/~dm/07/autumn/795.10/ptb-annotation-guide/root.html.

[21] H. M. Mueller, "Arbeitsbuch Linguistik", Second Edition, Published by Ferdinant Schoeningh 2009, ISBN 978-8252-21969-0.

[22] About Dependencies: http://nlp.stanford.edu/software/ dependencies_manual.pdf (retrieved: 02, 2015).

[23] Stanford Dependency Parser: http://nlp.stanford.edu/software/ stanford-dependencies.shtml (retrieved: 02, 2015).

[24] V. Ágel, "Valenztheorie", Published by Narr Tuebingen, 2009, ISBN 3-8233-4978-3.

[25] About Stanford NER: nlp.stanford.edu/software/CRF-NER.shtml.

[26] About GATE/ANNIE: https://gate.ac.uk/sale/tao/splitch6.html.

[27] About OWL: http://w3.org/TR/owl2-overview (retrieved: 02, 2015).

[28] About OWL: http://w3.org/TR/owl-features (retrieved: 02, 2015).

[29] About SWRL: http://w3.org/Submission/SWRL (retrieved: 02, 2015).

[30] M. Pataki, M. Vajna, and A. Marosi, "Wikipedia as Text", Ercim News - Special theme: Big Data, Vol. 89, 2012, pp. 48-49, About Wiki Dumps: https://dumps.wikimedia.org/backup-index.html.

[31] About DreamCloud: http://www.dreamcloud-project.org.

Data-driven Context Discovery Model for Semantic Computing in the Big Data Era

Takafumi Nakanishi
Center for Global Communications (GLOCOM),
International University of Japan
Tokyo, Japan
e-mail: takafumi@glocom.ac.jp

***Abstract*—We introduce a data-driven context discovery model for semantic computing in the big data era. Our model extracts from data sets the appropriate feature set as the context. We suggest that selection of a target data set is one of the representation processes for this purpose and the context in a big data environment. When a person selects target data from big data, that action latently indicates the context represented by the person's intention. Selecting a feature set from big data constitutes a data-driven context creation. Recently, fragmental data has spread on the Internet. In order to analyze big data, it is necessary to aggregate the appropriate data from data that has been dispersed on the Internet. An aggregation policy represents the purposes or contexts of analysis. In the big data era, it is necessary to focus not only on analysis but also on aggregation. After data aggregation, it is necessary to extract feature sets for semantic computing. This is what our model focuses on.**

Keywords: data-driven; context; feature selection; big data; data set

I. INTRODUCTION

Recently, big data is generated in a number of ways, including Internet browsing, sensors, and smartphones, etc. Most people have said that big data is a big opportunity. However, there are some who get hooked on a flood of big data. We information science researchers have already constructed data sensing, aggregation, retrieval, analysis, and visualization environments via web portals, software, APIs, etc. on the Internet. It is necessary to encourage people to use these big data. The number of information resources available on the Internet has been increasing rapidly. In particular, there is a large amount of fragmental data created by each person's device or created by the number of sophisticated sensors for the sake of scientific curiosity. In short, we not only retrieve but also create these data every day. Mountains of various fragmental data are being created.

One of the most important points is that data has become not only massive but also fragmentary. Currently, most users search contents through a search engine. This means that users acquire pages as contents. As data becomes fragmented, a model that searches for a single page will fail. It is more important to survey the entire data set than to analyze one piece of data deeply, given the large amount of fragmental data.

We observe that the essence of big data is not only massive data processing, but also optimization of the real world through the knowledge acquired from aggregated data. The current tendency of research on big data is how to aggregate a massive amount of data and how to process the data quickly. In the future, research will tend to focus on methods of discovering optimized solutions from big data.

Meanings are relatively determined by the context in a dynamic manner. One of the most important issues is achieving dynamic semantic computing that depends on the context. The dynamic nature is a very important part of the essence, because data that represents the features of each concept changes on each occasion and in each location. In other words, big data has volume, velocity, and variability. In order to compute semantics, a process to determine a context as a viewpoint is required. This means that it is necessary to predefine a space for the measurement of correlation. The space consists of feature sets as axes. Because we cannot predefine the feature set, it is necessary to develop a method of data-driven feature selection for semantic computing. The selected feature set constructs a measurement space. In other words, the measurement space represents the context in semantic computing.

In this paper, we introduce a data-driven context discovery model for semantic computing in the big data era. Our model extracts from data sets the appropriate feature set as the context.

We suggest that selection of a target data set is one of the representation processes for this purpose and the context in a big data environment. When a person selects target data from big data, that action latently indicates the context represented by the person's intention. Selecting a feature set from big data constitutes a data-driven context creation. Recently, fragmental data has spread on the Internet. In order to analyze big data, it is necessary to aggregate the appropriate data from data that has been dispersed on the Internet. An aggregation policy represents the purposes or contexts of analysis. In the big data era, it is necessary to focus not only on analysis but also on aggregation. After data aggregation, it is necessary to extract feature sets for semantic computing. This is what our model focuses on.

The contributions of our paper are as follow.

- We propose a new model of semantic computing by achieving a data-driven feature selection.
- The system applied to our model extracts feature sets corresponding to data sets, because an aggregation policy represents purposes or contexts of analysis.

- Our method reduces the computational cost of measurement of semantic computing because our method reduces the dimension of each vector represented in a certain selected feature set.

This paper is organized as follows. In Section II, we survey the existing work related to our proposed method. In Section II, we present the basic idea of our model. Next, we describe formulation of the design of our model in Section IV. Finally, we present our conclusions in Section V.

II. Related Work

One of the most important issues of semantic computing is correlation and similarity measurement. The most popular and basic method is utilization of the vector space model [1]. The dimensionality reduction techniques of the vector space model have been used for developing traditional vector space models, such as latent semantic indexing [2].

A weighting method is regarded as one of the feature selection techniques. Reference [3] describes a survey of weighting methods, such as binary [4], term frequency (TF) [4], augmented normalized term frequency [4][5], log [5], inverse document frequency (IDF) [4], probabilistic inverse [4][5], and document length normalization [4].

There have been studies defining similarity metrics for hierarchical structures such as WordNet [6]. Rada et al [7] have proposed a "conceptual distance" that indicates the similarity between concepts of semantic nets by using path lengths. Some studies [8][9] have extended and used the conceptual distance for information retrieval. Resnik [10] has proposed an alternative similarity measurement based on the concept of information content. Ganesan et al [11] have presented new similarity measurements in order to obtain similarity scores that are more intuitive than those based on traditional measurements.

In regard to other perspectives, the reference [12] has been surveyed. This survey [12] shows common architecture and general functionality as OBIE from various ontology-based information extraction studies. It consists of an "information extraction module," "ontology generator," "ontology editor," "semantic lexicon," and a number of preprocessors. The researchers are working both on various studies of OBIE system implementation and on studies focused on each module. In this paper, we will mainly introduce research on OBIE system implementation.

Our model processes a dynamic data-driven feature selection corresponding to a context. This means that our model does not have to prepare the space in advance. This is a very important difference, because we cannot create the space or schemas in advance in an open assumption. Currently, we are in the big data era. In a big data environment, we can aggregate a large amount of diverse fragmental data. We cannot predict in advance the kinds of data we will obtain. In fact, an increased key-value store means that the schema cannot be designed in advance. Since data updates are increasing in speed, the space for semantic computations and analyses should change dynamically as well.

One of the more famous methods of feature selection is "bags of keypoints" [13]. The bag of keypoints method is based on vector quantization of affine invariant descriptors of image patches. We can use the bag of keypoints for image classification.

An overview of feature selection algorithms is given in reference [14]. In this case, the feature selection algorithm is a computational solution that is motivated by a certain definition of relevance. It is hard to define the relevance. This [14] represents some roles of feature selection as follows: 1) Search organization, 2) Generation of successors, and 3) Evaluation measure.

Type 1) is in relation to the portion of the hypothesis explored with respect to their total number. This is responsible for driving the feature selection process using a specific strategy. The methods related to type 1) are [15], [16], and [17]. Type 2) proposes possible variants (successor candidates) of the current hypothesis. The method related to type 2) is [18]. Type 3) compares different hypotheses to guide the search process. The methods related to type 3) are [19], [20], and [21].

[14] also represents a general scheme for feature selection. The relationship between a feature selection algorithm and the inducer chosen to evaluate the usefulness of the feature selection process can take three main forms: embedded, filter, and wrapper.

There are some methods without feature selection, such as deep learning [22]. However, it is not possible to ignore feature selection completely. Generally, an artificial intelligence must depend on evaluation functions that are created by humans. The evaluation function is dependent on the manner in which features are selected. Even if more work is done on deep learning, work related to feature selection will still be conducted.

Currently, we are in the big data era. In a big data environment, we can aggregate a large amount of diverse fragmental data. We cannot predict in advance the kinds of data we will obtain. In fact, an increased key-value store means that the schema cannot be designed in advance. Since data updates are increasing in speed, the space for semantic computations and analyses should change dynamically as well.

Our model clearly differs in purpose from other methods. The current method predefines semantics as a measurement space, ontology, etc. By contrast, the system applied in our method extracts an appropriate feature set from a given data set. The given data set is the target data set. We suggest that selection of a target data set is one of the representation processes for this purpose and the context in a big data environment. Meanings are relatively determined by the context in a dynamic manner. One of the most important issues is achieving dynamic semantic computing that depends on the context. The dynamic nature is a very important part of the essence, because data that represents the features of each concept changes on each occasion and in each location.

Selecting a feature set from big data constitutes a data-driven context creation. Recently, fragmental data has spread on the Internet. In order to analyze big data, it is necessary to aggregate the appropriate data from data that has been dispursed on the Internet. An aggregation policy represents the purposes or contexts of analysis. In the big data era, it is necessary to focus not only on analysis but also o aggregation. After data aggregation, it is necessary to extract feature sets for semantic computing. This is what our model focuses on.

We have proposed a new weighting method for the vector space model [23]. This paper presents an overview of the reference [23]. In particular, the system that has been applied to our model extracts feature sets corresponding to data sets, because an aggregation policy represents purposes or contexts of analysis.

III. Basic Idea of Our Model

In this section, we introduce our assumptions and basic ideas for our model: a data-driven context discovery model for semantic computing.

Figure 1 shows an overview of the basic idea for our model.

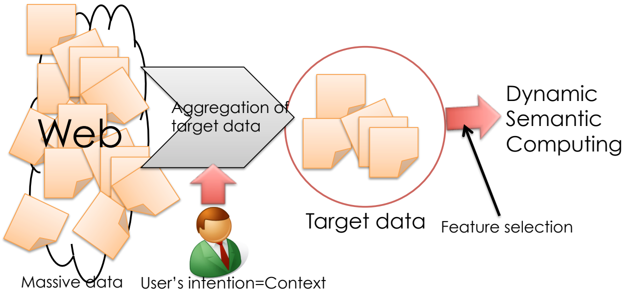


Figure 1. Basic idea of our model.
There is a large amount of data on the Internet. When we would like to analyze something, we try to aggregate data. In this case, we suggest that selection of a target data set is one of the representation processes for this purpose and the context in a big data environment. When a person selects target data from big data, that action latently indicates the context represented by the person's intention. In other words, the system can dynamically extract semantics by extracting feature sets. We can analyze dynamic data-driven semantic computing.

The dynamic nature is a very important part of the essence, because data that represents the features of each concept changes on each occasion and in each location. In other words, big data has volume, velocity, and variability. In order to compute semantics, a process to determine a context as a viewpoint is required. This means that it is necessary to predefine a space for the measurement of correlation. The space consists of feature sets as axes. Because we cannot predefine the feature set, it is necessary to develop a method of data-driven feature selection for semantic computing.

We suggest that selection of a target data set is one of the representation processes for this purpose and the context in a big data environment. When a person selects target data from big data, that action latently indicates the context represented by the person's intention.

Recently, fragmental data has spread on the Internet. In order to analysis big data, it is necessary to aggregate the appropriate data from data that has been dispersed on the Internet. In this case, we use crawler techniques. More specifically, we use focused crawlers. The focused crawler aggregates data corresponding to conditions given by a user. Therefore, this process includes the user's intention. The user's intention is one of the important clues for context detection.

The system applied to our model detects context from aggregated data because of this background. Context detection is achieved through feature selection.

We suggest that feature sets create the context. The feature set can construct measurement space. Each feature is driven by each axis of the measurement space. The measurement space achieves similarity or correlation of semantics. For example, the system detects the context of correlation between climate and another factor when we aggregate temperature data. Therefore, we can identify the context through aggregated data.

In other words, we can extract semantics from data usage logs. Figure 2 shows the relationship between content, context, and semantics.

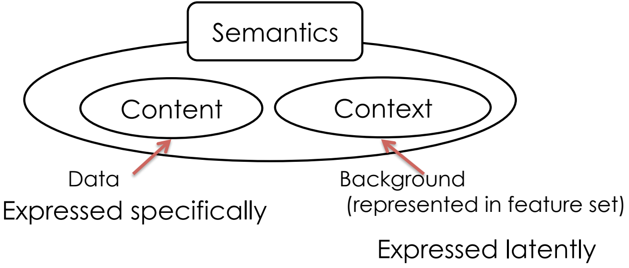


Figure 2. Relationship between semantics, content, and context.
Semantics consist of content and context. Content is something expressed specifically, such as data itself. Context is something expressed latently. The system applied to our model extracts feature sets from data sets. In other words, we can identify the context through data set usage. Semantics are created by data itself and data usage in our model.

Semantics consist of content and context. Content is something expressed specifically, such as data itself. Context is something expressed latently. The system applied to our model extracts feature sets from data sets. In other words, we can identify the context through data set usage. Semantics are created by data itself and data usage in our model.

Data usage logs represent context. It is important to achieve dynamic semantic computing. Semantics consist of content and context. We can aggregate data on the Internet as content. The system applied to our method can extract feature sets as context. Therefore, we can identify semantics by content and context.

Please note that the semantics of data change dynamically through usage of the same data. In this model, data is content. The same data has various ways in which it can be used. When the method of use changes, the context also changes. Therefore, the semantics of data change dynamically.

This is an importance element of dynamic semantic computing. Semantics are relatively determined by the context in a dynamic manner. One of the most important issues is achieving dynamic semantic computing that depends on the context. The dynamic nature is a very important part of the essence, because data that represents the features of each concept changes on each occasion and in each location.

It is difficult to extract context. This paper addresses how to extract context. The system applied to our model is one solution for extracting context. When a person would like to analyze something, he or she selects target data from big data.

IV. Formulation of the Data-driven Context Discovery Model

In this section, we present our method: a data-driven context discovery model for semantic computing in the big data era. Our model extracts the appropriate feature sets as the context from data set. We suggest that selection of a target data set is one of the representation processes for this purpose and the context in a big data environment. When a person selects target data from big data, that action latently indicates the context represented by the person's intention. Selecting a feature set from big data constitutes a data-driven context creation.

First, we introduce Bayesian variance, which is used in our model, in Section IV-A. Next, we design a mathematical formulation in Section IV-B.

A. Variational Bayesian Estimation

In this section, we show one of the estimation methods [24]: variational Bayesian estimation, which is used in this paper. Please note that our method can be applied to other estimation methods. In this paper, we use this as the estimation method for a conditional probability set of $p(v_l|e_m)$.

It is expressed with the stochastic variables X and Z. In addition, X is a known stochastic variable and Z is an unknown variable. The unknown variable Z denotes marginalization as follows.

$$p(X) = \int_Z p(X,Z)dZ$$

$$logp(X) = \int_Z logp(X,Z)\,dZ = L(q) + KL(q||p) \geq L(q)$$

$$L(q) = \int_Z q(Z)\frac{p(X,Z)}{q(Z)}dZ$$

$$KL(q||p) = -\int_Z q(Z)\frac{p(Z|X)}{q(Z)}dZ$$

$KL(q\|p)$ is a Kullback–Leibler divergence. Therefore, the Kullback–Leibler divergence is a minimum value when $q(Z)=p(Z|X)$. However, it is difficult to solve $p(Z|X)$ distribution.

Here, we apply it to the mean field approximation. The mean field approximation is represented as follows when a set of unknown variable $Z=\{z_1,z_2,...,z_k\}$:

$$q'(Z) = \prod_{i=1}^{k} q_i(z_i)$$

$q'(Z)$ can be represented by the Kullback–Leibler divergence. The approximate solution which should be calculated is equivalent to the minimum of the following formula.

$$KL(q'||q) = \int q'(Z)log\frac{q'(Z)}{q(Z)}dZ$$

he $q'(Z)$ is substituted for $L(q)$:

$$L(q) = \int_Z \prod_{i=1}^{k} q_i(z_i)\frac{p(X,Z)}{\prod_{i=1}^{k} q_i(z_i)}dZ$$

$$= \int_Z q_j(z_j)\left\{\int logp(X,Z)\prod_{i\neq j} q_i(z_i)dz_i\right\}dz_j$$

$$-\int_Z q_j(z_j)log\,q_j(z_j)dz_j + const$$

$$= \int_Z q_j(z_j)log\frac{\not{p}(X,z_j)}{q_j(z_j)}dz_j + const = KL(q_j||\not{p}) + const$$

Maximization of L(q) is equivalent to minimization of Kullback-Leibler divergence. The optimal solution $q_j^*(Z_j)$ is calculated as follows.

$$logq_j^*(Z_j) = \int logp(X,Z)\prod_{i\neq j} q_i(Z_i)dZ_i + const = \mathbb{E}_{i\neq j}[logp(X,Z)] + const$$

$$q_j^*(Z_j) = \frac{exp(\mathbb{E}_{i\neq j}[logp(X,Z)])}{\int exp(\mathbb{E}_{i\neq j}[logp(X,Z)])dZ_j}$$

B. Formulation of our model

1) Overview

Figure 3 shows an overview of our model. Figure 3 represents the processes in each step.

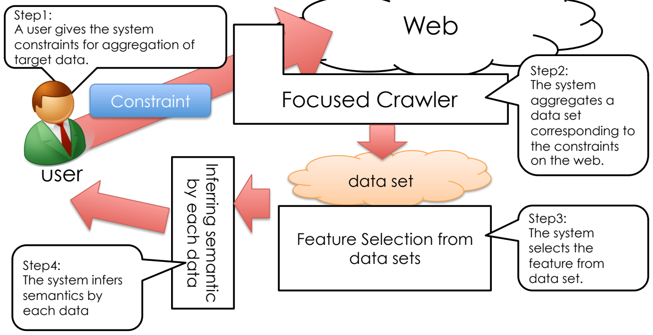

Step1:
A user gives the system constraints for aggregation of target data.
Web
Constraint
Focused Crawler
Step2:
The system aggregates a data set corresponding to the constraints on the web.
user
data set
Inferring semantic by each data
Feature Selection from data sets
Step3:
The system selects the feature from data set.
Step4:
The system infers semantics by each data

Figure 3. Overview of our model.
Our model consists of four steps. Step 1 is giving constraints for data aggregation. Step 2 is aggregation of data corresponding to the constraints on the Internet. Step 3 is feature selection from an aggregated data set. Step 4 is inferring the semantics of each piece of data. In other words, we can add semantics tags for each piece of data

Our model consists of four steps. These steps are as follows.

- Step 1: A user gives the system constraints for aggregation of target data.
  The important point of our model is the utilization of the usage data log. We suggest that feature sets create the context. The feature set can construct measurement space. Each feature is driven by each axis of the measurement space. The measurement space achieves similarity or correlation of semantics. Therefore, first, the user gives the system constraints for the focused crawler. This means that the user defines the usage data.
- Step 2: The system aggregates a data set corresponding to the constraints on the Internet.
  The system aggregates a data set along with the given constraint. The data set represents context. Each piece

of data represents content. When we combine them, we can obtain the semantics of each piece of data.

- Step 3: The system selects features from the data set.
  The system processes the feature selection from the aggregated data set. This step extracts feature sets as semantics axes. The feature set creates the context. We can map each piece of data into a space that is created by the feature set as each axis.
- Step 4: The system infers the semantics of each piece of data.
  We obtain each context and content from the data set and each piece of data. By combining these; we can obtain the semantics of each piece of data by probabilistic weighting.

2) *Formulation*

In this section, we formulate the model in accordance with each step. Figure 4 shows a representation of a graphical model for our model.

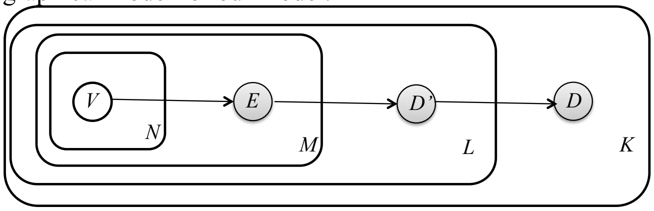


Figure 4. A graphical model of our model
D is a data set on the Internet. *D'* is an aggregated data set corresponding to the given constraint. *E* is an element set of the aggregated data set. *V* is a feature set.

We define the entire data set on the Internet $D=\{d_g\}$, the aggregate data set $D'=\{d'_h\}$, the element set $E=\{e_i\}$, and the feature set $V=\{v_j\}$. We reason $V=\{v_j\}$, when we aggregate D'.

Each element between $\{ d_g \}$ and $\{ d'_h \}$ which is represented in nodes is connected by the edges. Each value of each edge is represented in $p(d_g \mid d_h)$. The number of hit pages of a search engine can predict the values.

Each element between $\{ d'_h \}$ and $\{ e_m \}$ which is represented in nodes is connected by the edges. Each value of each edge is represented in $p(d_h| e_m)$. In the case of a text data set, it is easy to solve. For example, each piece of data has a word. The words are regarded as elements. This means that these values are solved by counting word frequency.

Each element between $\{ v_l \}$ and $\{ e_m \}$ which is represented in nodes is connected by the edges. Each value of each edge is represented in $p(v_l \mid e_i)$.

In conclusion, a conditional probability set of $p(v_l| e_m)$ is a good estimation function of feature selection. In other words, when a conditional probability set of $p(v_l| e_m)$ is bigger than the threshold, we can regard e_m as an appropriate feature v_l.

It is necessary to solve a conditional probability set of $p(v_l| e_m)$. A number of estimation methods in machine learning, such as variational Bayesian estimation [24], etc., are shown in Section III-C. In this paper, we use variational Bayesian estimation [24] as the estimation method for a conditional probability set of $p(v_l| e_m)$.

Finally, we can drive $p(v_l| e_m, d'_h, d_g)$ with the above values. These are represented by the data's metadata. In other words, we can add context-dependent semantics for each piece of data.

V. Conclusion

In this paper, we presented a data-driven context discovery model for semantic computing in the big data era. Our model extracts from data sets the appropriate feature set as the context.

We suggest that selection of a target data set is one of the representation processes for this purpose and the context in a big data environment. When a person selects target data from big data, that action latently indicates the context represented by the person's intention.

Our model clearly differs in purpose from other methods. The current method predefines semantics as a measurement space, ontology, etc. By contrast, the system applied to our method extracts an appropriate feature set from a given data set. The given data set is the target data set. We suggest that selection of a target data set is one of the representation processes for this purpose and the context in a big data environment. Meanings are relatively determined by the context in a dynamic manner. One of the most important issues is achieving dynamic semantic computing that depends on the context. The dynamic nature is a very important part of the essence, because data that represents the features of each concept changes on each occasion and in each location.

In the near future, our model will be applied to a heterogeneous data environment. It is necessary to consider the actual application of our model. Dynamic and automatic feature selection, such as is part of our model, is a more important technology in the big data era.

References

[1] G. Salton, A. Wong, C. S. Yang, "A vector space model for automatic indexing," *Magazine Communications of the ACM CACM* Homepage archive, vol.18(11), pp. 613-620, Nov. 1975.

[2] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, vol. 41(6), pp. 391-407, 1990.

[3] T. G. Kolda, D. P. O'Leary, "A semidiscrete matrix decomposition for latent semantic indexing information retrieval", *Journal ACM Transactions on Information Systems (TOIS)* TOIS Homepage archive vol.16(4), pp. 322-346, Oct. 1998.

[4] G.Salton, C. Buckley, "Term weighting approaches in automatic text retrieval," *Inf. Process. Manage*. 24, pp. 513–523, 1988.

[5] D. Harman, "Ranking algorithms. In Information Retrieval: Data Structures and Algorithms," *W. B. Frakes and R. Baeza-Yates, Eds. Prentice Hall, Englewood Cliffs*, NJ, pp. 363–392, 1992.

[6] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, K. J. Miller. "Introduction to WordNet: An on-line lexical database," *Journal of Lexicography*, vol.3(4), pp. 235-244, January 1990.

[7] R. Rada, H. Mili, E. Bicknell, M. Blettner, "Development and application of a metric on semantic nets," *IEEE Transactions on Systems, Man and Cybernetics*, vol.19(1), pp. 17-30, Jan/Feb 1989.

[8] Y. Kim, J. Kim, "A model of knowledge based information retrieval with hierarchical concept graph," *Journal of Documentation*, vol.46(2), pp. 113-136, 1990.

[9] J. Lee, M. Kim, Y. Lee, “Information retrieval based on conceptual distance in is-a hierarchies,” *Journal of Documentation*, vol.49(2), pp. 188-207, 1993.

[10] P. Resnik, “Using information content to evaluate semantic similarity in a taxonomy” In IJCAI: *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 448-453, 1995.

[11] P. Ganesan, H. Garcia-Molina, J. Widom, “Exploiting hierarchical domain structure to compute similarity,” *ACM Trans. Inf. Syst.*, vol.21(1), pp. 64-93, 2003.

[12] D. Wimalasuriya, D. Dou, “Ontology-based information extraction: An introduction and a survey of current approaches,” *Journal of Information Science*. 36, 3 (June 2010), pp. 306-323, 2010.

[13] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, C. Bray, "Visual Categorization with Bags of Keypoints," In *Proceedings of the Workshop on Statistical Learning in Computer Vision*, 2004.

[14] L.C. Molina, L. Belanche, A. Nebot, "Feature selection algorithms: a survey and experimental evaluation," In *Proceedings*. 2002 IEEE International Conference on Data Mining,(ICDM 2003), pp. 306--313, 2002..

[15] P. Narendra and K. Fukunaga, “A Branch and Bound Algorithm for Feature Subset Selection,” *IEEE Transactions on Computer*. C-26(9):917-922, 1977.

[16] J. Pearl. Heuristics, Addison Wesley, 1983.

[17] H. Liu and H. Motoda, “Feature Selection for Knowledge Discoverv nnd Dam Mining. Kluwer Academic Publishers,” I London. GB, 1998.

[18] D. Koller and M. Sahami, “Toward Optimal Feature Selection,” In *Proceedings of the 13th International Conference on Machine Learning*, pp. 284-292, Bari, IT. 1996.

[19] P.A. Devijver and J. Kittler, “PonernRecognition-A Statistical Appmach,” Prentice Hall, London. GB, 1982.

[20] H. Almuallim and T. G. Dietterich, “Leaming Boolean Concepts in the Presence of Many Irrelevant Features,” Artificial Intelligence, 69(1-2):279-305. 1994.

[21] H. Liu and H. Motoda, “Feature Selection for Knowledge Discovery and Dam Mining,” Kluwer Academic Publishers, London. GB, 1998.

[22] Q. V. Le, M. A. Ranzato, R. Monga, M. Devin, K. Chen, G. S. Corrado, J. Dean, A. Y. Ng, “Building High-level Features Using Large Scale Unsupervised Learning,” In *Proceedings of the 29th International Conference on Machine Learning*, Edinburgh, Scotland, UK, 2012.

[23] T. Nakanishi, “Semantic Context-Dependent Weighting for Vector Space Model,” In *Proceedings of the 2014 IEEE International Conference on Semantic Computing (ICSC)*, pp. 262-266, 2014.

[24] M. Christopher Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

Data Management in Cyber-Physical Work Environments

Influences on a Decision Model Derived from the Example of a Facility Management Support System

Clauß, Michael.; Müller, Egon
Department of Factory Planning and Factory Management
Chemnitz University of Technology
Chemnitz, Germany
e-mail: michael.clauss@mb.tu-chemnitz.de
e-mail: egon.mueller@mb.tu-chemnitz.de

Hofmann, Marcus
Department of Business Informatics
Dresden University of Cooperative Education
Dresden, Germany
e-mail: marcus.hofmann@ba-dresden.de

Götze, Janek.; Schumann, Christian-Andreas
Department of Business Informatics
Zwickau University of Applied Sciences
Zwickau, Germany
e-mail: janek.goetze@fh-zwickau.de
e-mail: christian.schumann@fh-zwickau.de

***Abstract*—Semantic technologies are said to have huge advantages over traditional data keeping approaches regarding flexibility and interpretability that are of increased importance in rather unstructured environments such as cyber-physical systems (CPS). But what are the parameters that influence a decision for or against its application in real-world data integration projects? Based on the findings of the ongoing research project FMstar (Facility Management with semantic technologies and augmented reality), the article derives some relevant influences on a respective decision model using the example of a facility management (FM) scenario.**

Keywords-data management; semantic technologies; factory management; facility management; decision model; parameter identification.

I. INTRODUCTION

Data is said to be the new oil both of the New or Digital Economy and, in a mere reflecting manner, of the Old Economy, too. First of all, why is that? With the development and implementation of approaches like the Internet of Things [1] or the concept of CPS [2], the real world gradually merges with its virtual counterpart. In manufacturing, this means that new insights from analyzing the data can not only be used for supplementary value-added services around the core business [3], but also have an impact on how the core business itself works inside. Approaches for logistic control systems that utilize the ubiquitous availability of data at the runtime of a manufacturing system are essential for modern flexible and changeable manufacturing systems [4]. Unfortunately, these new capabilities come with new dependencies that need an active and foresighted management in order to ensure reliability and profitability. So, the question that may arise is what technology is the best to organize the data in a certain area of application? Since the answer to that general question supposedly is a rather complex one, this paper will focus on practical experiences with data management in a well-defined area of application: facility management. Therefore, the article will follow an inductive approach and is structured as follows. First of all, a compact overview of the state of the art of data management in the factory management domain with an outline of open issues is given in Section II. Subsequently, in Section III, the facility management subdomain with a use case from the FMstar project [5] will be described and reviewed for relevant influences on the decision making process. The outcome of that will be condensed and integrated into a preliminary draft of a decision model interface description in Section IV that summarizes relevant influencing factors. A short summary and conclusion will be given in Section V.

II. STATE OF THE ART

Today’s practice of data generation, storage, usage and management varies among different areas of application. To start with, personal work is often supported by tools such as Excel or individually organized file storage systems. There is no explicit semantics and the principles used to organize the data depend a lot on the personal preferences of the user. In [6], some further dedicated tools for more collaborative tasks such as computer-aided design (CAD), knowledge base engineering (KBE), product data model/ product lifecycle management (PDM/PLM) or enterprise resource planning (ERP) systems are exemplarily identified. They make use of more adequately structured data models that ensure interchangeability, at least to a certain extent. In the majority of cases, relational databases and proprietary storage solutions are used. The latter often can only be accessed through more or less lossy export mechanisms based on particular exchange standards. A full picture is hard to draw at this point. Some further examples will be given in Section III. The basic distinction that can be made so far is the one between relational databases and so-called NoSQL approaches [7]. While relational databases still represent the dominating approach for data storage, the NoSQL approach refers to a larger group of database types (e.g., document-, object- or graph-based databases etc.) that become more and more important in cyber-physical application environments such as factory management. Since they offer potential for distributed architectures and in-memory operations they are more flexible and much faster in certain situations, but still lack powerful mechanisms to update and retrieve data at a large scale. There are pros and cons and further enhancements on both sides [8]. Their coexistence as well as the need for an economically justifiable integration of indeed old-fashioned designed but still indispensable legacy systems can be regarded as a given fact that has to be respected.

However, complexity in CPS design rises due to an increasing variety of requirements that have to be met [9]. Data has to be capable of inter-domain operation and this will have consequences for the scope of rather local or dedicated data management solutions [10]. The integration of data along the industrial value chains and its persistence throughout the whole product life cycle is necessary for legal or automation purposes [11]. The aforementioned PDM/PLM solutions offer part of that functionality, but their capabilities are limited to structures known at build-time. CPS in changeable environments such as modern manufacturing systems require capabilities for an adaption of data structures at run-time of the system. So, instead of or at least complementary to dedicated PLM solutions, an additional integration layer covering all relevant data sources for a specific use case seems like a Swiss army knife-like solution that meets all thinkable global integration demands. That is the point where semantic technologies usually come into play. Moreover, it is exactly the point where it is necessary to evaluate and decide whether a solution based on semantic technologies is really the best option or if traditional data integration approaches that rely on human insights and manual adaptations are the better choice.

III. Facility Maintenance Use Case

For this decision, two basic assumptions that are typical for a realistic scenario have been made in the FMstar project: First of all, the scope of the integration task is limited to the scenario at hand. Higher-level integration is only of theoretical interest unless there are practical guidelines to be used during integration that are supposed to enable such capabilities and usually cause additional, otherwise avoidable efforts. Secondly, the respective data sources cannot be modified in any way without violating their designated application. The consequence of these assumptions is that the syntactic and semantic interoperability can only be provided by some kind of mapping between the different data sources [12] that rely on appropriate schema management mechanisms [13]. A rough distinction between general data integration approaches is illustrated in Figure 1 that shows three alternative solutions for a mapping of the database schemas.

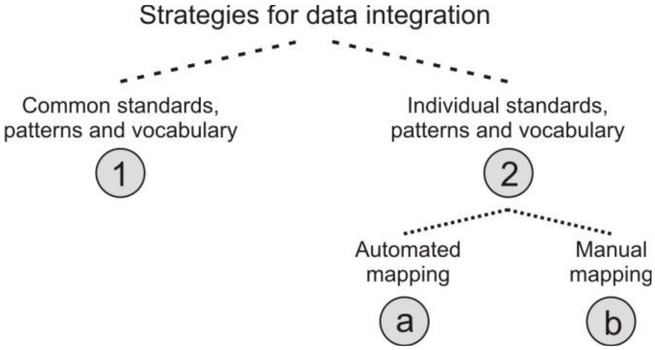

Strategies for data integration
Common standards, patterns and vocabulary
1
Individual standards, patterns and vocabulary
2
Automated mapping
a
Manual mapping
b

Figure 1. Alternative approaches for data integration [5].

The initial situation in the domain of the project can be best described with this apt quotation from [14]: “The building industry is a collaboration environment that requires repeated, iterative data exchanges and communication among different domains and applications in a high frequency. To automate information processing, standardized and qualified data is necessary for efficient working processes.” What the project team found was a mix of proprietary solutions for the management of 3D model data, maintenance task descriptions and dependency models for the technical infrastructure. If explicit schema models are provided, schema languages such as Unified Modeling Language (UML)/ Extensible Markup Language (XML), Resource Description Framework Schema (RDFS) and Data Definition Language (DDL) could be mapped using mapping languages such as XQuery, SPARQL, TRIPLE or Structured Query Language (SQL) [12]. Unfortunately, they were not available and that is also the reason for the solution approach that was selected: referring to Figure 1, the different databases were integrated manually, so the option 2b was chosen. The disadvantages are obvious; this repetitive process is slow, expensive, causes redundancies and does not meet the domain specific requirements described in the quotation above, since the schemas have to be analyzed manually through time-consuming interviews and workshops. Nevertheless, the solution in the project integrates the data sources using Apache Jena, an open source framework for the semantic web [5]. One goal of the project is to utilize semantic technologies for data integration, which is satisfied by implementing the backend of the FM support system based on Jena. A Google Nexus 10 tablet with Android KitKat (4.4) and libGDX for rendering the user interface (UI) is used as frontend. Figure 2 shows the overall architecture of the prototype. The bottom layer shows the UI of the prototype with a pump from the heating system. With that, the maintenance staff can access relevant information by selecting an object of the 3D model. The visualization approach is the second main focus of the project and aims at an intuitive interaction with the data.

Since the solution for the data integration is not suitable for practical use, the idea of using a vendor-neutral, open Building Information Model (BIM), such as the Industry Foundation Classes (IFC) [14], was raised. This could be used as an intermediary language and as a standard reference for mapping between different data schemas [12].

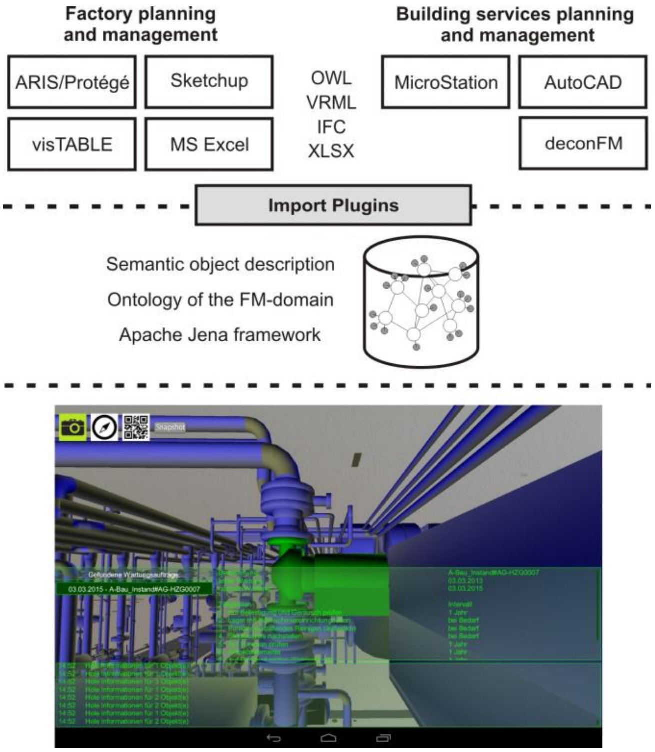

Factory planning and management
Building services planning and management
ARIS/Protégé
Sketchup
visTABLE
MS Excel
OWL
VRML
IFC
XLSX
MicroStation
AutoCAD
deconFM
Import Plugins
Semantic object description
Ontology of the FM-domain
Apache Jena framework

Figure 2. FMstar system architecture [5].

This way the manual mapping efforts could be reduced if each data source would provide a self-description for its schema. Referring to Figure 1, this would enable option 2a and possibly tend to option 1 as a complementary long-term development. However, option 1 represents a rather ideal solution that is very unlikely to be realized since it aims at consistent, interoperable data models at all partners. So, option 2b is probably the desired one, which also provides capabilities for non-redundant, bidirectional data flows between and not only from flexibly integrated systems. The selection of an appropriate data integration strategy depends on a couple of situational influences that will be pointed out based on the introduced use case in the next Section.

IV. CPS Data Management Decision Model

According to [15], data management "is a corporate service which helps with the provision of information services by controlling or coordinating the definitions and usage of reliable and relevant data." So, apparently reliability and relevance of data are central concerns of an appropriate data management strategy. Both concepts depend on the use case that is supposed to be supported. Referring to the central concept of data integration, this would be something related to the target context of the data. Another thing to look at from this definition is the corporate boundaries that more and more lose their limiting character. Data sources and application scenarios can be distributed over several organizations. In other words, the qualities of the source context also influence the decision. As a third area of influences, the transformation process and its capabilities determines what strategy fits best for the project at hand.

A. Influences

The description of the influences will be divided into the three areas just mentioned: source context, target context and transformation process. According to the assumptions described in Section II, the source context cannot be influenced by the project. The target context can be regarded as a kind of run-time environment of the desired solution and can be influenced within the project. Following this logic, the transformation process can be interpreted as the build-time environment.

The first area of influences is the source context. It is determined by the structure of the data, the standardization, the roadmap for further development and its dynamics.

- Structure: As mentioned before, the structure of the data sources is determined by the database schema. Even though it cannot be modified in the project, the more detailed the available schema is, the easier it is to utilize semantic technologies for mapping.
- Standardization: Standard-based languages for schema description or domain-specific standards for data organization support the utilization of semantic mapping technologies.
- Roadmap: A roadmap outlining the further development of a source database helps to evaluate its suitability for automated mapping approaches.
- Dynamics: The more often the structure of the source databases changes, the more difficult manual integration will be. Semantic technologies pay off in high volatile environments.

The second area of influences is the target context. It is determined by the use case and, derived from that, by the required quality of the data and the focus of the solution.

- Use Case: The use cases' complexity in terms of inter-domain operation determines how much it would make sense to provide an integrated view on the data. The more complex the use case, the less it pays off to go for manual integration and semantic approaches are very likely the better choice.
- Quality: The importance of data quality is a central aspect that is referred to in many publications in the context of data integration [6]. The more quality matters, the more the consequences of low data quality should be minimized by auxiliary means if semantic technologies are used.
- Focus: The focus of application of the desired solution can reach from local to global. The wider the focus, the more probable is it to have the need to flexibly integrate new data sources and schema information provided in foreign languages.

Influences from the transformation process, the third area of influences, are the competencies of the people involved.

- Competence: The people involved in the data integration project determine the conceptual power and the technology stack that the solution will be built on. Even if it sounds like a platitude, if no experts with profound competencies for semantic technologies are available, the project is better realized with traditional approaches.

B. Towards a Decision Model for CPS Data Management

In Figure 3, the influences that were just identified are summarized in a structured way. The domain gap between the source and the target of the data plays an important role when it comes to automated mapping approaches. Especially in multi-domain application environments, where the use case requires a combination of technically rather unrelated domains, the integration efforts may increase in a non-linear way if the inhomogeneity exceeds certain limits.

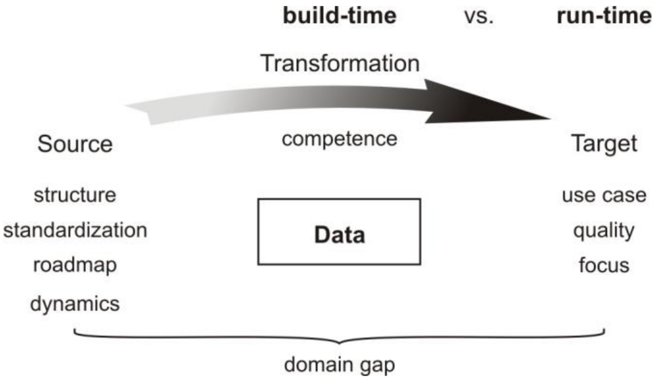


Figure 3. Influences on data management strategy.

However, this first draft of a model provides a very basic structure of the influences on the decision making process. It may support an early stage decision process in a CPS project by providing some relevant influences that should be considered. But for all that, it is supposed to serve as a starting point and to be subject to further refinement.

V. SUMMARY & CONCLUSION

This paper focuses on influences on a decision model for using semantic technologies for CPS engineering projects. While semantic technologies offer a wide range of opportunities to map schemas and integrate data automatically, this does not come for free and even the results are not necessarily satisfying. Influences on the data sources, the capabilities of the transformation process and the desired outcome including the specific use cases have to be analyzed at an early stage of a project in order to make a profound decision of what system architecture and respective data management approach to choose. This pays off in many ways, even though technology evolves away from local data management to cloud-based solutions that provide more powerful capabilities for integration right from the start [16]. However, depending on who is asked, this might be a promising perspective for data integration in the future. In the meantime, a conscientious decision model that balances requirements and specific environmental conditions is inevitable for economically successful CPS projects.

ACKNOWLEDGMENT

The work presented in this paper has been supported by the German Federal Ministry for Economic Affairs and Energy (Bundesministerium für Wirtschaft und Energie – BMWi, VP2333105DB2). We want to thank all the partners that have contributed to the results so far.

REFERENCES

[1] H. Kopetz, Real-Time Systems, 2nd ed. New York: Springer, pp. 307-323, 2011, doi: 10.1007/978-1-4419-8237-7.

[2] M. Broy, Cyber-Physical Systems. Berlin: Springer, pp. 17-32, 2010, doi: 10.1007/978-3-642-14901-6.

[3] Z. Bi, L. D. Xu, and C. Wang, “Internet of Things for Enterprise Systems of Modern Manufacturing,” Industrial Informatics, IEEE, vol. 10, Jan. 2014, pp. 1537-1546, doi: 10.1109/TII.2014.2300338.

[4] H.-P. Wiendahl et al., “Changeable Manufacturing: Classification, Design and Operation,” CIRP Annals Manufacturing Technology, vol. 56, Elsevier, 2007, pp. 783-809.

[5] M. Clauß, J. Götze, E. Müller, and C.-A. Schumann, “Real-time Data Access through Semantic Technologies and Augmented Reality in Facility Management Processes”, Proceedings of International Conference on Innovative Technologies (IN-TECH 2014), Sep. 2014, pp. 45-48, ISBN: 978-953-6326-88-4.

[6] J. Krogstie, „Capturing Enterprise Data Integration Challenges Using a Semiotic Data Quality Framework,“ Business & Information Systems Engineering, vol. 57, Feb. 2015, pp. 27-36, doi: 10.1007/s12599-014-0365-x.

[7] P. J. Sadalage, and M. Fowler, NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence. Upper Saddle River, NJ: Addison-Wesley, 2013, ISBN: 978-0-321-82662-6.

[8] H. Voigt, Flexibility in Data Management. Dresden, 2014, [Online]. Available from: http://nbn-resolving.de/urn:nbn:de:bsz:14-qucosa-136681 [retrieved: June, 2015].

[9] M. M. Bezemer, Cyber-Physical Systems Software Development. Enschede: The Netherlands: University of Twente, 2013, doi: 10.3990/1.9789036518796.

[10] M. D. Ilić, L. Xie, U. A. Khan, and J. M. F. Moura, “Modeling Future Cyber-Physical Energy Systems,” IEEE Power Engineering Society General Meeting, Pittsburgh, PA. Jul. 2008, pp. 1-9.

[11] A. Denger et al., „Organisationaler Wandel durch die Emergenz Cyber-Physikalischer Systeme: Die Fallstudie AVL List GmbH (English title: Organizational Change through the Emergence of Cyber-Physical Systems: The AVL List GmbH Case Study),“ HMD, vol. 51, Oct. 2014, pp. 828-837, doi: 10.1365/s40702-014-0090-4.

[12] H. Adametz, A. Billig, J. Einhaus, and J. Gottschick, Whitepaper Semantische Interoperabilität (English title: Whitepaper Semantic Interoperability). Berlin: Fraunhofer-Institut für Software- und Systemtechnik ISST, 2013.

[13] H. Glatzel, „Schema-Management ohne Schema? – Schema-Verwaltung in NoSQL-Datenbanksystemen (English title: Schema Management without Schema? – Schema Management in NoSQL Database Systems),“ 44. Jahrestagung der Gesellschaft für Informatik, Big Data – Komplexität meistern, Sep. 2014, pp. 2461-2472, ISSN: 1617-5468, ISBN: 978-3-88579-626-8.

[14] C. Zhang, J. Beetz, and M. Weise, “Interoperable Validation for IFC Building Models using Open Standards,“ ITcon – Journal of Information Technology in Construction, vol. 20, 2015, pp. 24-39, ISSN: 1874-4753.

[15] K. Gordon, Principles of Data Management: Facilitating Information Sharing, 2nd ed. Swindon, UK: BCS Learning & Development Limited, 2013, ISBN: 978-1-78017-185-2.

[16] L. Zhao, S. Sakr, A. Liu, and A. Bouguettaya, Cloud Data Management. Cham: Springer, 2014, doi: 10.1007/978-3-319-04765-2.

Mobile Queries using Semantic Processing into Augmented Reality

Itzel Coral, Miguel Martínez, Félix Mata
Computing Mobile Laboratory
IPN-UPIITA
Mexico City, Mexico
{email: olivoscastillo@gmail.com,
mrosales81@gmail.com, mmatar@ipn.mx}

Roberto Zagal, Consuelo García,
Systems Department
IPN-ESCOM
Mexico City, Mexico
{email: rzagalf@ipn.mx,
varinia400@hotmail.com}

***Abstract*— This article presents a mobile system to answer structured queries into a research-academic domain using a spatial Ontology. The answers to queries are displayed on an Augmented Reality (AR) interface. The structure of query is a triplet formed by: interrogative adverb, verb and direct object. The case study is in a university campus. Queries are solved using semantic processing, for example, {Who can advise on vector calculus?}. Then, possible answers (researchers or professors candidates) are obtained applying semantic similarity on attributes defined into Ontology (e.g., level of expertise, research line to which it belongs, topic, etc.). Additionally, spatial parameters are included into the answer, such as: where researcher is located, schedules, colleagues, etc. The functionalities of system are: search persons based on qualitative and spatio-temporal attributes. The combination of a semantic approach with an augmented reality interface provides new possibilities to express queries; not only in text or based on location, but using AR with interactions. It is useful to locate persons in outdoor environments.**

Keywords-Spatial Semantics; Augmented Reality; Mobile queries.

I. INTRODUCTION

Very often, in an academic environment, students are looking for professors, researchers or specialists with knowledge on different topics; even a thesis advisor. Then, when non-local students visit the facilities of a university, they look for professors to answer a particular question or doubt (expressed as a query from smartphone). They do not know researchers' names; in other words, information is imprecise. The only data they have is the specialty that a professor should belong to. Then, they need to ask more information about professors or researchers from other students or people on campus. Therefore, it would be useful to have a system to help find which professors are the experts on particular issues. We have to consider that the professors can be located based on schedule of work. In addition, when the search for a professor includes several criteria, such as level of experience, international recognition, among others, the task becomes a challenge.

This can be solved using a semantic processing approach combined with advantages of navigation using AR. In addition, once students have identified the professors that can support or advise them on a particular topic, it is useful to have a comprehensive tool in order to find out the schedule and specific geographical location within the campus where to find the professor or researcher in question. This functionality is enhanced when is displayed using AR.

This paper introduces a semantic mobile system using augmented reality with the following capabilities: 1) search of professor or researcher by criteria: topic, level expertise among others, 2) schedules and places where the researchers can be found and 3) an interface of navigation and AR. The case study is focused on a campus of IPN (Instituto Politécnico Nacional) in Mexico, in order to assist students with questions regarding to topics of thesis or other subject-matter.

The rest of the paper is organized as follows: Section II shows the related work; Section III explains the methodology used; Section IV describes the obtained results, and finally, the conclusion and future work are outlined in Section V.

II. RELATED WORK

AR technology augments the sense of reality by superimposing virtual objects and cues upon the real world in real-time [1] and indoor environments. AR has been the object of increasing development in outdoor and indoor environments [2][3][4][14]. In [11], an indoor technique is used for AR positioning system for indoor construction application by tracking the coordinates and its angles of vision. In order to, achieve indoor positioning in three dimensions. Nevertheless, no ontologies or semantic processing were used. AR was used in several ways but navigation is not provided by a semantic processing, such as in [12], where AR is used in a self-guided tour; the user can see environment information, sites or buildings, listen to audio touring narratives, or get directions.

ISBN: 978-1-61208-420-6

Ontology is also employed as a method for identifying categories, concepts, relations, and rules [5][6][7]. When combined with query languages, domain ontologies favor the design and development of domain-based search engines and their application to different areas [8]. In contrast, similar semantic approaches have been proposed; for example, GeoSpatial ontologies are applied in emergency systems for indoor disasters, for campus of University of Melbourne [13]. The semantic approach gets a spatial analysis; it is used for emergency management capabilities indoor and outdoors components. But, AR is not assisted using ontologies. Zhang et al. [9] proposed a technique that uses common sense geographic knowledge and qualitative spatial reasoning for the generation of a geographic Ontology. The tools for processing Ontology Web Languages (OWL) are numerous; One of the most popular is Pellet [10], we decided to use it in this research.

III METHODOLOGY FOR QUERYING AND SEMANTIC PROCESSING

In order to solve the queries submitted, a four stages methodology is defined: a) query contextualization, b) displayed results on AR, c) parsing, semantic and spatio-temporal processing, information retrieval, and d) visualization on AR interface. In Figure 1, general architecture of the system is shown.

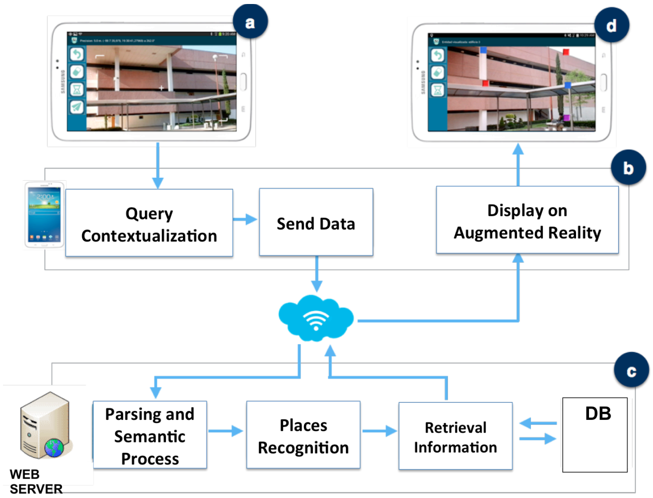


Figure 1. General architecture of the system.

In Figure 1, the labels a) and b) represents the stage of query contextualization, it means, the attributes required to build a query (e..g, “Who knows Electronics?”). This query is sent to stage c) where the query is analized sintactically and semantically; elements of query are associated with concepts in domains of space and time, in order to infer answer(s) to query. These answers, are attributes on time (e.g., hour) and location (e.g., a classroom) that will be sent to modules in stage b) in order to transform them in elements that can be displayed on AR interface, finally the stage d) shows the AR interface with the obtained results. In the case of semantic processing stage, it involves three steps: Ontology design, rules’ definition and reasoner’s implementation. They are described in the next section.

A. Ontology Model Design

The ontological model was designed to represent the knowledge of researchers, professors and their respective field of expertise and workplace. Ontology will be explored in order to answer queries. Ontology is built based on relationships from custom university academic domain. The model was built with the OWL language and using Protegé editor 4.3.0 [16]. The semantic consistency was checked using Pellet reasoner [15] that was coupled as a plug-in in the editor. In Figure 2, the basic hierarchical structure of Ontology is shown; it describes the context of spatial location of the professors at a university. In Ontology, the geographical entity class is the parent concept that defines the set of laboratory classes, classrooms, auditoriums, offices, buildings, etc. Each class, with its identifier, has three aspects: subclass, equivalents and the elements of which are disjoint.

The Ontology was implemented in Spanish; the concepts are related to geographical objects (classroom, level, office, building) events (exposition, conference) and classification of personnel (Academic, administrative, etc). In similar way, Ontology was complemented to spatial objects, temporal aspects and subjects. Figure 2 shows an Ontology fragment (in Spanish) of geographical objects.

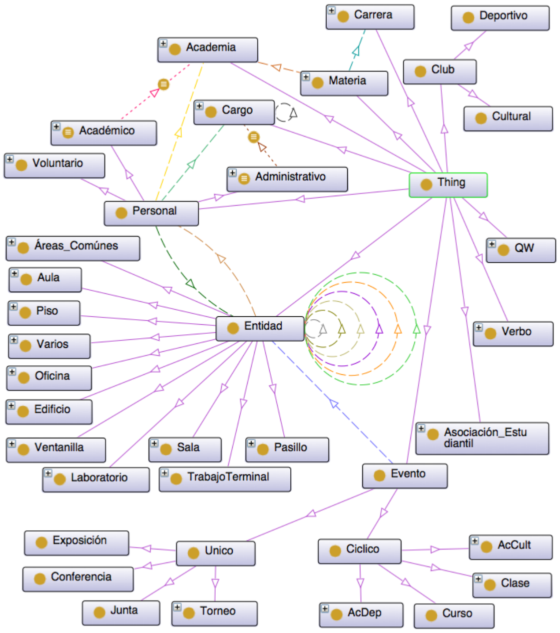


Figure 2. Spatio-Temporal Ontology.

The Ontology is explored in order to contextualize queries (find a matching concept between query and class or instance of Ontology); the Ontology is used to make inferences, too.

B. *Semantic Processing*

The Ontology describes academic knowledge and associated spatio-temporal attributes (e.g., specialist in mathematics, located in basic sciences office, available schedule 13:00 to 17:00). The reasoner Pellet is used to extract information that is not explicitly represented by the data (some inferences). In Table I, object's properties (denoted as P_n where n is the property identifier) are categorized for each type of relationship: spatial (e.g., P_1-withInThe), temporal (P_{13}-ocurredIn) and academic (P_5-knows). For example, “isLocatedIn” (P_{16}) is a spatial relationship to link people with their specialty and identify their specific positions, i.e., one can infer the following knowledge: if the teacher Luis Flores (instance of the class “Academic”) “isLocatedIn” department of advanced technologies (it is an instance of "Office" class) and its location says that this office is *withInThe* academic department (then it follows that the teacher Luis *isLocatedIn* the entity called “Academic department”). Several inferences are made in order to solve imprecise queries (e.g., looking for physics researcher with works in modern physics) in opposite way with a precise query (e.g., looking for researcher Pedro). In Table 1, a sample of different properties of spatio-temporal Ontology’s classes (from Figure 2) is shown.

TABLE I. AXIOMS OF FIGURE 2

	Property	**Meaning**
P_1	houses	Related offices with staff who reside there
P_2	limitShares	Related Contiguous entities
P_3	withCharge	Relates to administrative staff position held
P_5	knows	Personnel related to their area of knowledge (academy)
P_8	withIn	Relating a larger entity that contains
P_{16}	isLocatedIn	Reverse Property Shelter_to
P_{17}	over	Reverse Property below

The properties link entities, and in turn with entities’ domain and their values are used as axioms in reasoning and detecting inconsistencies. This is useful for scalability of Ontology design. The properties defined in the Ontology are type transitive, i.e., the reasoner is able to conclude that if an object O_1 is related to another object O_2 by property P_1 and this in turn is related by the same property with O_3 object, then individuals O_3 and O_1 share the property P_1.

The domain of P16 isLocatedIn relationship is the staff of the institution, its range are different entities (offices, laboratories and staff rooms); where a teacher can be located. Figure 3 shows some of the 162 entities of courses offered by the university. The professors were characterized by defining their equivalents, the academy and career to which they belong. There are 256 people including information, such as names, titles and locations, academic and administrative staff of the institution (see Figure 3).

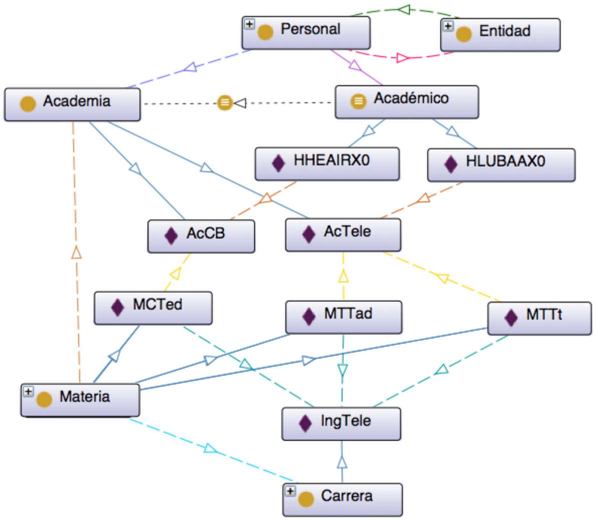


Figure 3. Entities, Relationships and Axioms for Academy Concept.

This fragment of Ontology is used to solve queries related to personnel search by their specialty; an example of modeled knowledge is as follows: Personnel is located in an entity, therefore an entity contains elements of Personnel class. Personnel (professors or researchers) know some element of academy class (e.g., Mathematics) then is of academic type. For example: person HHEAIRX0 (Pablo Hernandez) knows AcCB (basic sciences academy). The element MCTed (differential equations) is of type subject (topic) and belongs to Basic sciences academy, hence professor Pablo Hernandez knows differential equations.

C. *Querying and Semantic Processing*

The system uses two types of processing queries: precise and imprecise; an example of the former {Q_1 = Who knows Electronics?} to solve, the Ontology is explored to retrieve the name (find a semantic matching) of academy teachers belong to electronics subject; as it is to assume that all teachers know subjects of the academy they belong to.

In the case of the latter type of queries, i.e. working with imprecise questions, let us consider the query such as Q_2 = Where is the office 122? To answer it, one use DESCRIBE relation to retrieve or infer facts about the concept "office 122" for example, what class are held here, if it has cubicles teachers, how spatial concepts share limit, on which floor and in which the building it is located, etc.

The overall set of steps to process queries is: 1) SPARQL [17] translation, 2) detection of the object of interest, 3) identification of keywords exploring the Ontology using words of query, which detects the interrogative adverb and verb, and 4) related to the query retrieves information from the Ontology. For example, the query Q_3: Who can help me with an issue in Electronics? is translated into SPARQL query format as follows:

```
PREFIX bibo:
http://www.semanticweb.org/itz/ontologies/2014/7/BIBO_v1#
SELECT ?personnelName ?AcademyName
WHERE {?subject bibo:withName?subjectName
FILTER (str(?nameSubject) = “Electronics”)
{?subject bibo;belongsTo
?Academy.?academy bibo:withName?AcademyName.
?Personnel bibo:KnowsAbout?academy.
?Personnel bibo:withName? PersonnelName}
```

The query Q_3, retrieves teachers from academy where Electronics course is taught. The original query should be analyzed in order to identify the direct object (interest object), and then the terms of original query are compared with the elements of Ontology (using similarity). This way, we determined if the object of interest is referring to a subject, academy, and event or is an unknown concept for the Ontology. To determine how similar a text string is to another, the elements of the Ontology and the phrase are mapped to a vector space that allows the use of the dot product as a measure of similarity: In the first instance, it is required to extract the "universe" of words contained in the collection of items of Ontology and determine the frequency in which they appear.

Below is built a N×M matrix, where N is the number of elements and M to the number of terms (words without repeating) in the "universe". The vector representing the user's phrase, is constructed and the dot product between it and the rows of the matrix is calculated. Accordingly, the pair of vectors whose dot product is the largest will be the most similar to the phrase. Therefore; the system identifies it as the object of interest the user refers to. In the following section the results are discussed.

IV. TEST AND RESULTS

Testing was done using several queries; the first test was to evaluate the answer of the system when a query asks for data not stored in the Ontology. Hence, in order to answer, the processing should use similarity, and, then, the results obtained are shown.

Let us consider the query Q_4={who can help me with issues of vector calculus}

```
2015/02/9 21:04 Monday
Matches: 2
multivariable calculus ... 0.6544891121378675
Answer: There are two teachers who may like
help:
Francisco Perez
Mario Contreras
RA: false
Time: 168 milliseconds
```

For example, in the query Q4, user asked for vector calculus (but no data or concept of calculus vector is present into the Ontology). The system answers by relating Vector calculus with an entity of class named "multivariate calculus", although it has similarity just above of 60%, but not best candidates were found. This relation was made because both subjects-matter (vector and multivariable calculus) are in the area of Basic Sciences according to the hierarchy of Ontology. The next test is a query spatio-temporal, when a user requires to know when and where a subject matter is taught.

Let us consider the query Q5 ={Schedule of network security subject?}

```
 2015/02/11 11:56
 Matches: 6
 network security ... 1.0
 Task: I'm looking for the schedule network security
 Answer: network security in 3TM3 group:
 * Monday at 10:00 Telematics Laboratory II
 * Tuesday at 10:00 in classroom 122
 network security in 3TV3 group:
 * Monday at 16:00 in classroom 102
 * Wednesday at 16:00 laboratory telematics II
 RA: false
Time: 181 milliseconds
```

In query Q_5, a query element has a matching of 100% with the event of Ontology: “network security”. Nevertheless, when is an event occurred in two groups and the query not specified which one is required, then the reasoner retrieves both schedules.

The next test is based on spatial queries using current user’s location and spatial relations (next, in front of, etc). These queries ask for information such as:

Q_6={available professors in building “A”?},

Q_7={researchers located next to telematics academy?},

Q_8={events occurred around to my current position?}.

In Figure 4, the user's position is represented using a user icon. The points marked by the user symbol and the orientation indicated by the sight lines from A to H. The GPS have a 5 meters error (in optimal conditions) that facilitates the deployment of virtual objects on screen (augmented reality) with equivalent margin of error.

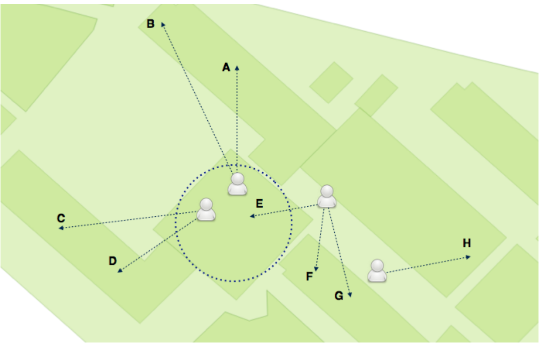


Figure 4. Several User's Position and Sight Line Directions.

The events are retrieved under sight line A, using user's position from GPS, and the neighborhood using spatial relations. The sight line is processed based on the actual position. Then, we calculated that line cross building "A" is located to 11.252 meters from user. The result obtained is as follows:

Data received: (19.511537 -99.126536) at 0 °
Fri 2015/02/20 10:45:00
Sight Line: LINESTRING (-99.126536 19.511537, -99.126431 19.517536)
Against: Building A
Central Crossing: POINT (-99.1265322955 19.5117492255) to 11,252 meters
Estimated Height: 6.97 meters
Information: * Classroom 423
Class: Distributed Systems
Group: 2TM5
Title: José Rodriguez
Floor: P2
Address: Left
Distance: 1555 meters
* Laboratory of Complex Systems
Entity no events registered
Floor: P1
Address: Left
Distance: 1828 meters
* Nanophotonics and laboratory techniques
Entity no events registered
Floor: PB
Address: Left
Distance: 1975 meters
* Classroom 422
entity unoccupied
Floor: P2
Address: Right
Distance: 7786 meters
* Classroom 412
Entity no events registered
Floor: P1
Address: Right
Distance: 7366 meters
Time: 4089 milliseconds

These results are displayed using AR; to achieve that, the building height is computed to determine the vertical position of virtual objects to be displayed on screen (according to corresponding level of building).

Each virtual object can be touched and relevant information of this object or event will be displayed. Hence, the following information is retrieved: event, level, address and distance regarding to cross point between view line and polygon. Figure 5 shows this result.

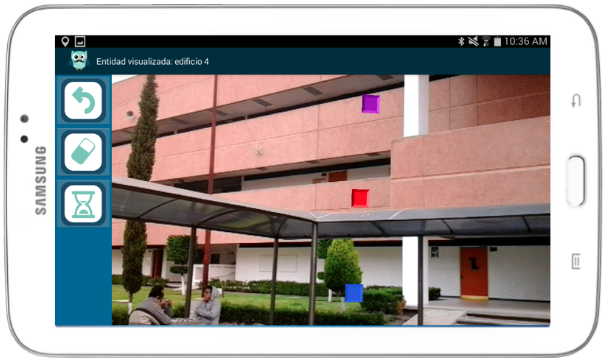
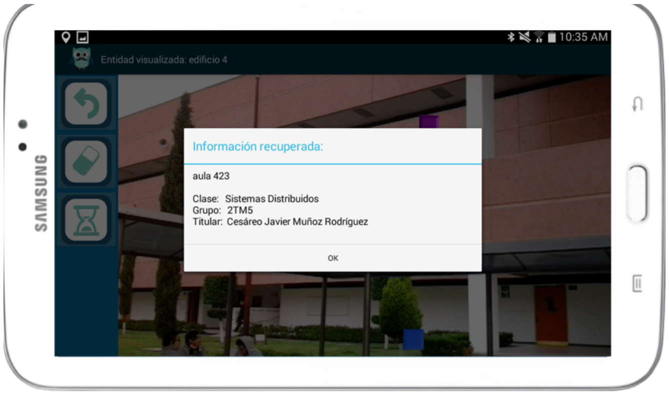


Figure 5. Augmented Reality Navigation for Query Types Q_6, Q_7 and Q_8.

In Figure 5, the app notifies the user that s/he is in front of building A. Then, the interface allows to navigate using AR, when the mobile device points to a different direction, less or equal to 30° degrees regarding to original line of sight (located in top of Figure 4). While, in Figure 6, information is retrieved, when virtual objects are touched. This data corresponds to events belong to time interval (defined by start and finish hour) and the temporal context.



Información recuperada:

aula 423

Clase: Sistemas Distribuidos
Grupo: 2TM5
Titular: Cesáreo Javier Muñoz Rodríguez

OK

Figure 6. Retrieved Information in AR Interface Q_8.

In this case, the AR tests were conducted in portrait mode (vertical position); to display the position of the virtual objects the building height is dividing by three (floors of building). Converting meters to pixels is essential in order for the building to be displayed on screen from foundation to roof, in order to display all the elements on screen (see Figure 7).

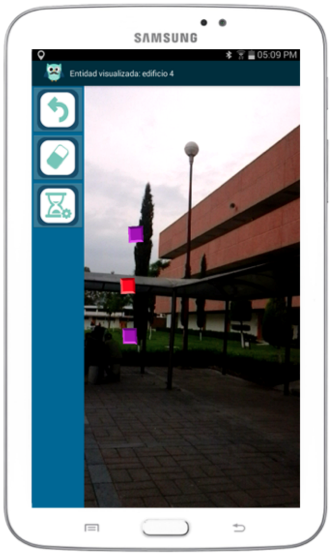


Figure 7. Information retrieval displayed on the AR interface.

In Figure 7, we note that the response to the sight line B presents a significant error (the square icons are displayed out of building). This is because although the system does not expect the user to visualize the entities full front, the algorithm that calculates the position of virtual objects does not consider the perspective introduced by lateral views. To remedy this deficiency, it was decided to deploy graphic elements only on the sub entities that lie within the range formed by an angle of 30° right or left regarding to user's line of sight.

V. CONCLUSION AND FUTURE WORK

The use of semantic processing to find a knowledge profile represents a useful field for tasks of semantic similarity. The resolution of queries over Ontology exploration about spatial and temporal attributes can solve complex queries.

Displaying results in augmented scenarios provide a practical way to locate people. The combination of ontologies and AR for mobile phones represents a field of opportunity for various tasks and scopes. GPS and compass sensor require developing algorithms in order to compensate the error margin or use external devices with great precision, in order to offer a precise augmented navigation. Future work considers including speech recognition and AR using sensors instead of pattern recognition.

ACKNOWLEDGMENT

The authors of this paper wish to thank the Mobile Computing Laboratory-UPIITA, Instituto Politécnico Nacional in Mexico, IPN-UPIITA, IPN-ESCOM, COFAA-IPN, IPN-SIP and CONACYT.

REFERENCES

[1] J. Carmigniani, B. Furht, M. Anisetti, P. Ceravolo, E. Damiani, and M. Ivkovic, Augmented reality technologies, systems and applications. Multimedia Tools and Applications, 2011, vol. 51, no. 1, pp. 341-377.

[2] H. M. Park, S.H. Lee, and J.S. Choi, 2008. Wearable augmented reality system using gaze interaction. In Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR '08). IEEE Computer Society, Washington, DC, USA, 175-176. DOI=10.1109/ISMAR.2008.4637353 http://dx.doi.org/10.1109/ISMAR.2008.4637353

[3] T.N Arvanitis., A. Petrou, Knight J.F., Savas S., Sotiriou S., M. Gargalakos, E. Gialouri, Human factors and qualitative pedagogical evaluation of a mobile augmented reality system for science education used by learners with physical disabilities. 2009, Personal and Ubiquitous Computing 13(3):243–250.

[4] F. Zhou, H. B. L. Duh, and M. Billinghurst, Trends in augmented reality tracking, interaction and display: A review of ten years of ISMAR. In Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality. 2008, pp. 193-202. IEEE Computer Society.

[5] B. Smith, and D. Mark, Geographical categories: an ontological investigation. International Journal of Geographical Information Science, 2001, 15 (7), 591–612.

[6] M. Sorrows and S. Hirtle, The nature of landmarks for real and electronic spaces. Lecture Notes in Computer Science, 1999, 1661, 37–50.

[7] B. Tversky and K. Hemenway, Objects, parts, and categories. Journal of Experimental Psychology: General, 1984, 113 (2), 169–193.

[8] B. Huang, C. Claramunt, Spatiotemporal data model and query language for tracking land use change. 2005, Transportation Research Record, 107-113.

[9] Y. Zhang, Y. Gao, L. Xue, S. Shen, K. Chen, A common sense geographic knowledge base for GIR, Science in China Series E: Technological Sciences, 2008, 51 (1) , pp. 26–37.

[10] E. Sirin, B. Parsia, B. C. Grau, A. Kalyanpur, & Y. Katz, Pellet: A practical owl-dl reasoner. Web Semantics: science, services and agents on the World Wide Web, 2007, 5(2), 51-53.

[11] C. Kuoa, T. Jengb and I. Yangc, An invisible head marker tracking system for indoor mobile augmented reality. Automation in Construction, an International Research Journal. Volume 33, August 2013, Pages 104–115.

[12] T. L. Chou and L. J. ChanLin, Augmented reality smartphone environment orientation application: a case study of the Fu-Jen University mobile campus touring system. Procedia-Social and Behavioral Sciences. 2012, 46, 410-416.

[13] H. Tashakkori, A. Rajabifard, M. Kalantar, A new 3D indoor/outdoor spatial model for indoor emergency response facilitation. Building and Environment. 2015, 89, 170-182.

[14] P.Daponte, L. De Vito, F. Picariello and M. Riccio, State of the art and future developments of measurement applications on smartphones. Journal of the International Measurement Confederation, volume 46, Issue 9, November 2013, Pages 3291–3307.

[15] Pellet Reasoner, http://pellet.owldl.com/, [retrieved: February, 2015].

[16] Protégé Editor, http://protege.stanford.edu/, [retrieved: February, 2015].

[17] SPARQL, http://www.w3.org/TR/rdf-sparql-query/, [retrieved: February, 2015].

Deep Learning for Large-Scale Sentiment Analysis Using Distributed Representations

Kazuhei Katoh
Department of Computer Science
Faculty of Engineering
Ehime University
Matsuyama, Ehime, Japan
Email: katoh@ai.cs.ehime-u.ac.jp

Takashi Ninomiya
Department of Electrical and Electronic Engineering
and Computer Science
Graduate School of Science and Engineering
Ehime University
Matsuyama, Ehime, Japan
Email: ninomiya@cs.ehime-u.ac.jp

***Abstract*—This paper presents the performance evaluations of deep learning classifiers for large-scale sentiment analysis using Rakuten Data. Many NLP theories and applications use 1-of-K representations for representing a word, but 1-of-K representations are difficult to use with many deep learners because they are vectors consisting of millions of dimensions. To reduce the number of dimensions of 1-of-K representations, we used distributed representations for words by using word2vec. Two experiments were conducted: (1) sentiment analysis using a small data set, the IMDB dataset, and (2) sentiment analysis using a large-scale data set, Rakuten Data. In the experiments, we observed that multi-layer neural networks did not work well for the small data set (i.e., neural networks without hidden layers achieved the best result), but multi-layer neural networks worked well for the large-scale data set. In the experiments using Rakuten Data, we tested the neural networks with $0-6$ hidden layers, and neural networks with three hidden layers achieved the best result.**

Keywords–sentiment analysis; deep learning; distributed representations.

I. INTRODUCTION

For the last decade, many kinds of social media on the Internet, such as Twitter, SNS, and blogs, have become available, and more than a billion people use them in their daily life now. As these social media grow, sentiment analysis from these media becomes more important to extract opinions about political issues, events, and some specific commercial products.

In this paper, we present performance evaluations of deep learning classifiers for a large-scale sentiment analysis using Rakuten Data. Deep learning has attracted many researchers because it has achieved significant results in the fields of speech recognition and image recognition [1][2]. The models of deep learning classifiers are defined as multi-layer neural networks having around 3 to 20 hidden layers. For a deep learner, generalized features or concepts are automatically acquired in hidden layers along with the training of whole networks. Though some deep learning tools have already been developed publicly, these tools assume dense and low dimensional vectors as inputs. This is a crucial problem for large-scale natural language processing (NLP) because many NLP theories and applications use 1-of-K representations for representing a word, which has millions of dimensions. To mitigate the problem of 1-of-K representations, we use word2vec [3] to reduce the number of dimensions of 1-of-K representations.

We conducted two experiments on sentiment analysis. In the first experiment, we used Rakuten Data [4] as a data set, and we used the IMDB dataset [5] in the second experiment. Rakuten Data is a large-scale data set, consisting of 16 million reviews, that is written in Japanese. The IMDB dataset consists of 100,000 movie reviews written in English. The IMDB dataset is smaller than Rakuten Data, and hence the deep learner can learn from the IMDB dataset using 1-of-K representations. In the experiments, we evaluated the effectiveness of word2vec by comparing it with 1-of-K representations in the IMDB dataset. Finally, we evaluated the effectiveness of our classifier for the large-scale sentiment analysis using Rakuten Data. In the experiments, we observed that multi-layer neural networks did not work well for the small data set (i.e., neural networks without hidden layers achieved the best result for the IMDB dataset), but multi-layer neural networks worked well for the large-scale data set, Rakuten Data.

Section Ⅱ introduces an overview of related work. Section Ⅲ presents large-scale sentiment analysis based on multi-layer neural networks and distributed representations for words using word2vec. Section Ⅳ describes the procedures and results of the experiments. Section V concludes the document.

II. RELATED WORK

In NLP, 1-of-K representations (or one-hot representations) are generally used for representing a word as a vector. The word vector in 1-of-K representations has the same length of vocabulary size, and each dimension corresponds to each word in a dictionary. The vector for word w in 1-of-K representations takes a form in which only one dimension corresponding to w is given as 1, and other dimensions are given as 0. For example, if we have a dictionary consisting of one hundred thousand words, the word vector takes a form in which only one dimension is given as 1 among one hundred thousand dimensions, and all the remaining dimensions are given as 0. Let n be the vocabulary size and e be the index of dimensions for word w_e. The vector for word w_e in 1-of-K representations is formally given as $(d_1, ..., d_i, ..., d_n)$, where $d_i = 1$ if $i = e$, and $d_i = 0$ otherwise. Therefore, word vectors in 1-of-K representations are large and extremely sparse.

Word2vec [3] is a method for obtaining distributed representations for words by using neural networks. In word2vec, neural networks are defined to solve a pseudo task of predicting a word given surrounding words. After training neural networks using huge size text, weight vectors for each word in a dictionary are retrieved from the neural networks as distributed representations for words.

Two types of neural network models are defined in word2vec: the continuous bag-of-words (CBOW) model and the Skip-gram model. The CBOW model predicts the current word given surrounding words, and the Skip-gram model predicts surrounding words given the current word. Both models generate distributed representations for words by retrieving a weight matrix from neural networks between the input layer and the hidden layer. Therefore, the number of dimensions of the weight vectors is equal to the number of nodes in the hidden layer, around 200 or 400. Thus, word2vec reduces the number of dimensions of word vectors in 1-of-K representations having hundreds of thousands of words in the input layer to hundreds of dimensions. Mikolov et al. [3] have shown that the acquired word vector represents a semantic concept for a word and relationships between words. The relationships between words can be calculated using simple addition and subtraction, e.g., the vector for ‘queen’ is close to the vector for ‘king’ minus ‘man’ plus ‘woman.’

Deep learning involves multi-layer neural networks with efficient learning methods and generalization. In our experiments, we used Caffe [6] as a deep learning tool. Caffe is an efficient implementation, one in which neural network models can be customized in a model file. However, Caffe cannot deal efficiently with large-scale data, such as Rakuten Data, in 1-of-K representations due to memory limitation. Therefore, we reduced the number of dimensions of the dataset by using word2vec.

III. LARGE-SCALE SENTIMENT ANALYSIS BY USING DEEP LEARNING AND DISTRIBUTED REPRESENTATIONS

We present large-scale sentiment analysis based on multi-layer neural networks and distributed representations for words using word2vec. In the experiments, we tested multi-layer neural networks with $0-6$ hidden layers. In the input layer, a document vector $\mathbf{d}$ was used as an input. The document vector is a vector for a review made by adding distributed representations for all words in the review. The distributed representations for words were acquired by using word2vec trained from Rakuten Data. Formally, we have the document vector $\mathbf{d}$ as follows (1).

$$\mathbf{d} = \sum_{w \in r} word2vec(w), \tag{1}$$

where r is a review, w is a word in r, and $word2vec(w)$ is the distributed representation for word w. In the output layer, binary sentiments (positive/negative) were used as an output, and softmax functions were applied to the output from the hidden layers. Figure 1 shows the structures of neural networks that we used in the experiments. In the figure, (A), (B), and (C) draw neural networks with 0, 1, and 2 hidden layers, respectively. Neural networks with 0 hidden layers are equivalent to the logistic regression model without a prior.

TABLE I. RAKUTEN DATA

	data size (GB)	number of reviews	number of words
training set	8.45	13,133,032	656,834,594
development set	1.02	1,655,042	82,123,546
test set	1.12	1,818,107	88,549,699

TABLE II. IMDB DATASET

	data size (MB)	number of reviews	number of words
training set (labeled)	33.158	25,000	5,843,019
training set (unlabeled)	66.557	50,000	14,273,230
test set (labeled)	32.376	25,000	5,711,718

IV. EXPERIMENTS

We conducted experiments to evaluate the performance of the multi-layer neural networks for large-scale sentiment analysis using Rakuten Data.

A. Dataset and Tools

We used two datasets, Rakuten Data [4] and IMDB dataset [5]. Rakuten Data is a large-scale data set consisting of around 16 million reviews written in Japanese[1]. Each review in Rakuten Data is labeled with $0-5$ grade labels: 0 is the most negative, and 5 is the most positive. We converted the $0-5$ grade sentiments into binary sentiments by regarding $0-3$ grades as negative and $4-5$ grades as positive. In the experiments, we evaluated the binary classification task. Table I shows the specifications of Rakuten Data. The IMDB dataset consists of 100,000 movie reviews written in English. In the IMDB dataset, 50,000 reviews are labeled with $1-10$ grade labels: 1 is the most negative, and 10 is the most positive. We converted the $1-10$ grade sentiments into binary sentiments by regarding $1-5$ grades as negative and $6-10$ grades as positive. Table II shows the specifications of the IMDB dataset. The IMDB dataset is smaller than Rakuten Data, and hence the deep learner can learn from the IMDB dataset using 1-of-K representations. We evaluated the effectiveness of the deep learning classifier for the large-scale sentiment analysis using Rakuten Data. We also evaluated the effectiveness of word2vec by comparing it with 1-of-K representations in the IMDB dataset, where we have the document vector $\mathbf{d}$ for 1-of-K representations as follows (2).

$$\mathbf{d} = \sum_{w \in r} \text{1-of-}K(w), \tag{2}$$

where r is a review, w is a word in r, and 1-of-$K(w)$ is the 1-of-K representation for word w.

The number of dimensions for the distributed representations was determined using the development set. We tested 100, 200, 400, and 800 dimensions for Rakuten Data, and 100, 200, 400, 800, and 1600 dimensions for the IMDB dataset. Table III shows the details of other hyper parameters that were determined by using the development data set.

[1] Currently, Rakuten Data 2014 consists of around 64 million reviews for 150 million products. We used Rakuten Data 2010 in the experiments, and it consists of around 16 million reviews.

TABLE III. HYPERPARAMETERS OF WORD2VEC.

hyperparameters	setting
Model	CBOW
Window Size	8
Negative Samples	25
Hierarchical Softmax	none
Iteration	15
Subsampling of Frequent Words	1e-3

TABLE IV. HYPERPARAMETERS OF CAFFE.

hyperparameters	setting
Hidden layer	$0-6$
The number of nodes	500
Test interval	1,000
Max iteration	100,000

We used Caffe as a deep learning tool. However, Caffe cannot efficiently learn from Rakuten Data using 1-of-K representations because it is a large data set and because Caffe does not support sparse vectors. We first trained word2vec using 13,133,032 reviews (656,834,594 words) in Rakuten Data, and then we trained Caffe using 2,684,354 reviews (137,456,326 words) in Rakuten Data. Table IV shows the hyper parameters for Caffe. The batch size was 200 for Rakuten Data and 1000 for the IMDB dataset. The base learning rate was 0.01 for Rakuten Data and 0.005 for the IMDB dataset. Figure 1 shows the structures of the multi-layer neural networks.

We also compared the performance of deep learning with L2-regularized logistic regression. We used Liblinear [7] for evaluating L2-regularized logistic regression. The hyperparameters were tuned by using the development data.

We used Mecab [8] for tokenizing Rakuten data and used Stepp Tagger [9] for tokenizing the IMDB dataset.

B. Results

In the experiments, "LR(1-of-K)" means the result of L2-regularized logistic regression using 1-of-K representations. "LR(w2v)" means the result of L2-regularized logistic regression using distributed representations for words. "NN-Li(1-of-K)" means multi-layer neural networks with i hidden layers using 1-of-K representations. "NN-Li(w2v)" means multi-layer neural networks with i hidden layers using distributed representations for words.

Table V shows the results of the experiments for the test set of Rakuten Data. In the table, neural networks with three hidden layers (NN-L3(w2v)) achieved the best result for Rakuten Data. We can also see that NN-L3(w2v) achieved a better result than that of logistic regression, LR(w2v). In the table, we can observe that the accuracy increased when we used more hidden layers such that the number of hidden layers was less than four, and the accuracy decreased when we used more than four hidden layers.

Table VIII shows the results of the experiments for the test set of the IMDB dataset. In the table, we can see the difference in the neural networks using 1-of-K representations and those using distributed representations. The best result was achieved by the neural networks without hidden layers using the distributed representations. Contrary to our expectation, the multi-layer neural networks using 1-of-K representations were worse than those using the distributed representations.

TABLE V. ACCURACY FOR TEST DATASET OF RAKUTEN DATA.

Model	Accuracy
NN-L0(w2v)	89.130 %
NN-L1(w2v)	90.220 %
NN-L2(w2v)	90.703 %
NN-L3(w2v)	91.015 %
NN-L4(w2v)	91.001 %
NN-L5(w2v)	90.795 %
NN-L6(w2v)	90.727 %
LR(1-of-K)	90.956 %
LR(w2v)	90.124 %

Table VI and Figure 2 show the analyses for the development set of Rakuten Data, and Table VII and Figure 3 show the analyses for the development set of the IMDB dataset.

C. Discussion

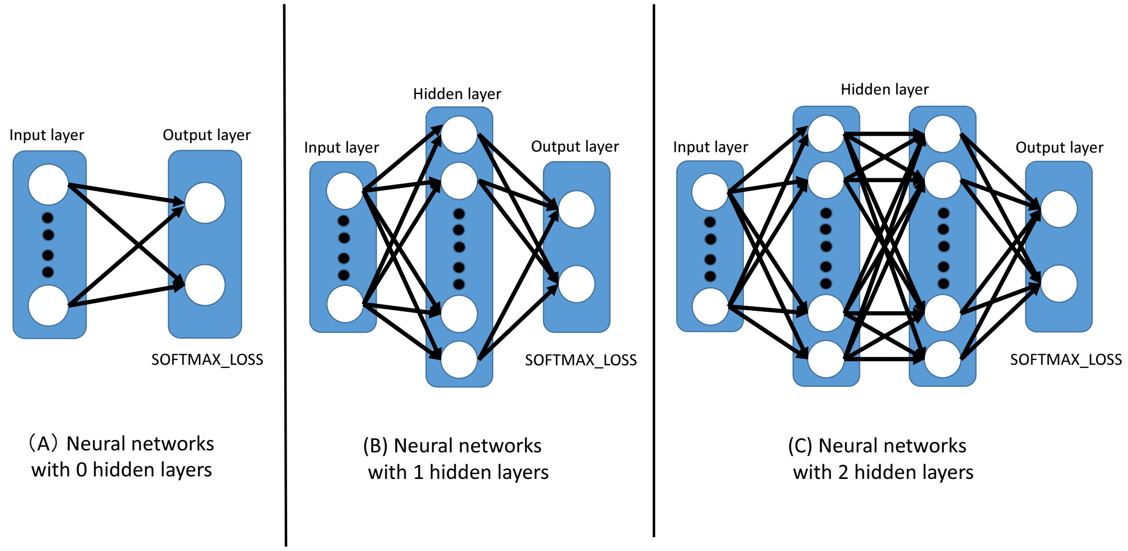
We can see from Table VIII that neural networks with a higher number of hidden layers did not work well for the IMDB dataset, especially in the case of 1-of-K representations. We think that the multi-layer neural networks with 1-of-K representations failed to learn the concepts of words in their hidden layers. This may be because the size of the IMDB dataset was too small to learn them. In the case of training with Rakuten Data, the neural networks with three hidden layers achieved better results than those of the neural networks without hidden layers. We think that these results partially support our hypothesis that extremely large datasets, such as Rakuten Data, enable neural networks to learn their hidden layers well in the task of sentiment analysis.

With these experimental results, we think that in tasks of natural language processing, unlike image recognition or speech recognition, extremely large datasets are needed to learn the concepts of words or phrases in the hidden layers of multi-layer neural networks. In the experiments of the IMDB dataset, we used around 20 million words (75,000 reviews) for word2vec training. But, the data size of the IMDB dataset was much smaller than that of Rakuten Data, which consists of around 650 million words (around 13 million reviews). We think that pre-training of neural networks using an extremely large dataset is a good solution for simultaneously learning the word concepts and the tasks of NLP, such as multi-task learning [10] or a stacked auto-encoder [11].

From Table V and VIII, the accuracy of LR(1-of-K) was better than that of LR(w2v) in both experiments. However, the accuracy of NN-Li(1-of-K) was worse than that of NN-Li(w2v) in the experiment of the LDBM dataset. We think that this also means that neural networks with 1-of-K representations fail to learn the hidden layers. The tendency of how hidden layers are learned from 1-of-K representations can be seen more clearly if we could conduct experiments on the multi-layer neural networks with 1-of-K representations for Rakuten Data. We leave this for future work.

V. CONCLUSION

In this paper, we presented performance evaluations of deep learning classifiers for large-scale sentiment analysis using Rakuten Data. Many NLP theories and applications use 1-of-K



（A）Neural networks with 0 hidden layers

(B) Neural networks with 1 hidden layers

(C) Neural networks with 2 hidden layers

Figure 1. Structure of neural network

TABLE VI. ACCURACY FOR DEVELOPMENT DATASET OF RAKUTEN DATA.

Representation	Number of dimensions	NN-L0	NN-L1	NN-L2	NN-L3	NN-L4	NN-L5	NN-L6	LR
w2v	100	87.978 %	89.363 %	89.962 %	90.119 %	90.135 %	90.149 %	90.088 %	88.949 %
	200	88.111 %	89.947 %	90.478 %	90.551 %	90.595 %	90.538 %	90.541 %	89.392 %
	400	88.689 %	90.257 %	90.746 %	90.840 %	90.860 %	90.805 %	90.789 %	89.633 %
	800	88.110 %	90.434 %	90.847 %	90.947 %	90.982 %	90.957 %	90.900 %	89.878 %
1-of-K	81,420	-	-	-	-	-	-	-	90.671 %

TABLE VII. ACCURACY FOR DEVELOPMENT DATASET OF IMDB DATASET.

Representation	Number of dimensions	NN-L0	NN-L1	NN-L2	NN-L3	NN-L4	NN-L5	NN-L6	LR
w2v	100	87.180 %	50.200 %	52.780 %	86.041 %	86.460 %	86.820 %	87.201 %	87.380 %
	200	87.761 %	86.520 %	57.820 %	86.920 %	86.780 %	87.080 %	87.840 %	88.100 %
	400	88.140 %	53.720 %	54.800 %	86.940 %	87.760 %	88.081 %	88.200 %	88.620 %
	800	88.780 %	50.580 %	52.200 %	88.260 %	87.520 %	88.240 %	88.300 %	89.000 %
	1600	88.599 %	60.460 %	50.000 %	87.620 %	87.600 %	87.920 %	88.300 %	89.040 %
1-of-K	35,309	88.780 %	86.520 %	57.820 %	88.260 %	87.760 %	88.240 %	88.300 %	89.040 %

representations for representing a word, but 1-of-K representations are difficult to use with many deep learners because they are vectors consisting of millions of dimensions. To reduce the number of dimensions of 1-of-K representations, we used distributed representations for words by using word2vec.

We conducted two experiments: (1) sentiment analysis using a small data set, the IMDB dataset, and (2) sentiment analysis using a large-scale data set, Rakuten Data. In the experiments, we observed that multi-layer neural networks did not work well for the small data set (i.e., neural networks without hidden layers achieved the best result), but multi-layer neural networks worked well for the large-scale data set. In the experiments using Rakuten Data, we tested the neural networks with $0-6$ hidden layers, and neural networks with three hidden layers achieved the best result. We think that these results partially support our hypothesis that extremely large datasets, such as Rakuten Data, enable neural networks to learn their hidden layers well in the task of sentiment analysis.

In the experiments for the IMDB dataset, we also compared 1-of-K representations with distributed representations. In the experiments, neural networks using distributed representations achieved better results than those using 1-of-K representations. We think that this may be because the size of the IMDB dataset was too small to learn the concepts of words in the neural networks. The tendency of how multi-layer neural networks are learned from 1-of-K representations can be seen more clearly if we could conduct experiments on the multi-layer neural networks with 1-of-K representations for Rakuten Data. We leave this for future work.

ACKNOWLEDGMENT

This work was supported by a JSPS KAKENHI Grant-in-Aid for Scientific Research (B) Grant Number 25280084.

TABLE VIII. ACCURACY FOR TEST DATASET OF IMDB DATASET.

Model	Accuracy
NN-L0(1-of-K)	85.040 %
NN-L1(1-of-K)	83.540 %
NN-L2(1-of-K)	76.840 %
NN-L3(1-of-K)	50.720 %
NN-L4(1-of-K)	50.000 %
NN-L5(1-of-K)	50.000 %
NN-L6(1-of-K)	50.000 %
NN-L0(w2v)	88.476 %
NN-L1(w2v)	50.920 %
NN-L2(w2v)	54.908 %
NN-L3(w2v)	86.504 %
NN-L4(w2v)	86.940 %
NN-L5(w2v)	86.984 %
NN-L6(w2v)	87.344 %
LR(1-of-K)	86.848 %
LR(w2v)	78.936 %

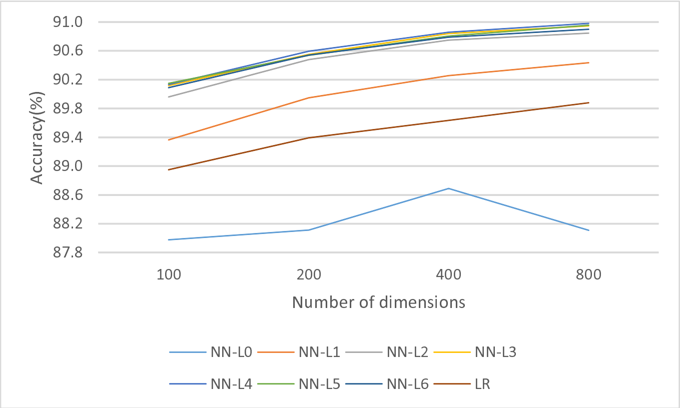


Figure 2. Accuracy of each model for development dataset of Rakuten Data

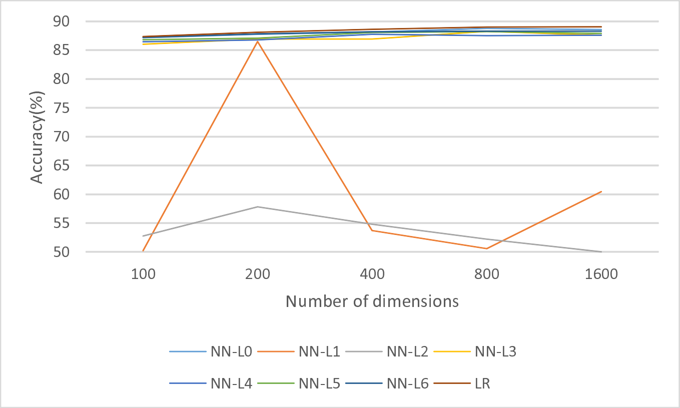


Figure 3. Accuracy of each model for development dataset of IMDB dataset

REFERENCES

[1] H. Kaiming, Z. Xiangyu, R. Shaoqing, and S. Jian, “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification,” 2015. [Online]. Available: http://arxiv.org/abs/1502.01852 retrived:05, 2015

[2] S. Christian, W. Liu, J. Yangqing, S. Pierre, R. Scott, A. Dragomir, E. Dumitru, V. Vincent, and R. Andrew, “Going deeper with convolutions,” 2014. [Online]. Available: http://arxiv.org/abs/1409.4842 retrived:05, 2015

[3] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” 2013. [Online]. Available: http://arxiv.org/abs/1301.3781 retrived:05, 2015

[4] “Rakuten dataset,” http://rit.rakuten.co.jp/opendataj.html retrived:05, 2015.

[5] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, “Learning word vectors for sentiment analysis,” in Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. Portland, Oregon, USA: Association for Computational Linguistics, June 2011, pp. 142–150. [Online]. Available: http://www.aclweb.org/anthology/P11-1015 retrived:05, 2015

[6] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, “Caffe: Convolutional architecture for fast feature embedding,” Proceedings of the ACM International Conference on Multimedia, 2014, pp. 675–678.

[7] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, “Liblinear: A library for large linear classification,” Journal of Machine Learning Research Volume 9, 2008, pp. 1871–1874.

[8] T. Kudo, K. Yamamoto, and Y. Matsumoto, “Applying conditional random fields to japanese morphological analysis, proceedings of the 2004 conference on empirical methods in natural language processing,” EMNLP-2004, 2004, pp. 230–237.

[9] Y. Tsuruoka, Y. Tateishi, J.-D. Kim, T. Ohta, J. McNaught, S. Ananiadou, and J. Tsujii, “Developing a robust part-of-speech tagger for biomedical text,” in Advances in Informatics, ser. Lecture Notes in Computer Science, P. Bozanis and E. Houstis, Eds. Springer Berlin Heidelberg, 2005, vol. 3746, pp. 382–392. [Online]. Available: http://link.springer.com/chapter/10.1007retrived:05, 2015

[10] R. Collobert and J. Weston, “A unified architecture for natural language processing: Deep neural networks with multitask learning,” ICML ’08 Proceedings of the 25th international conference on Machine learning, 2008, pp. 160–167.

[11] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, “Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion,” Journal of Machine Learning Research Volume 11, 2010, pp. 3371–3408.

Modelling of an Ontology for a Communication Platform

More safety at football events by improving the communication between stakeholders

Jürgen Moßgraber, Manfred Schenk, Desiree Hilbring
Fraunhofer Institute of Optronics, System Technologies and Image Exploitation IOSB
Karlsruhe, Germany
e-mail: {juergen.mossgraber, manfred.schenk, desiree.hilbring}@iosb.fraunhofer.de

***Abstract*— Safety and security in football events has been a heavily debated topic in the media for years. Especially communication processes between the police authorities, private security services, town councils, supporters and other spectators have often been neglected. Communication processes in the context of football matches can encounter technical limitations. Traditional communication channels are often characterized by a one-way communication structure, delayed forwarding of information or limited quality (e.g., stadium announcements). To address these problems, a new communication platform is explored. The semantic basis for this platform is formed by an ontology. The usage and benefits of the ontology are manifold: 1) The design process created a better understanding among the stakeholders. 2) The actors, roles and their relations are used for information filtering and access restrictions. 3) The relations of the ontology are used for structuring the information and navigating through it. 4) Heterogeneous information is fused into the platform from existing systems by annotating the data with concepts from the ontology. In this paper, the design of the ontology based on research of already existing ontologies is presented.**

Keywords-ontology design; communication platform; Web Ontology Language (OWL).

I. Introduction

In the football season 2013/2014, about 13 million people attended the matches of the German Bundesliga (first league) [1]. The games of the second, third and lower leagues were attended by several additional millions of spectators. On their journey to the stadium and back home, they travel through crowded urban regions and depend on using the local infrastructures.

In order to implement such big football events, a cooperation of police forces, local town councils, football clubs and private security services is necessary to provide a safe and secure environment. Together with spectators and supporter groups, these stakeholders strive for peaceful and positive sport events. By doing this, different perspectives on freedom, safety and security must be balanced. Following these presuppositions, the following research question emerges: How can the safety and security creation in the context of football games be optimized via communication?

The research project SiKomFan (Mehr Sicherheit im Fußball – Verbessern der Kommunikationsstrukturen und Optimieren des Fandialogs) [2], funded by the German Ministry of Education and Research (BMBF), therefore researches possible improvements of communication strategies, including technical solutions to support them. By using a broad perspective that involves 25 football locations in Germany's three professional football leagues the most relevant stakeholders are examined in order to contribute to a successful dialogue with supporters. So far, the inquiry revealed that there are different communication strategies in different locations leading to different results.

Examining the topic of communication processes in the context of football games can be done from different specialist perspectives and by using different methods. SiKomFan combines four disciplines, namely sociology, risk- and security management, law and computer science as well as their specific methods.

Communication processes in the context of football matches can encounter technical limitations. Traditional communication channels are often characterized by a one-way communication structure, delayed forwarding of information or limited quality (e.g., stadium announcements). New media, such as Twitter, provide opportunities for flexible, timely and rapid exchange of information, but have the disadvantages of information overflow [3] and uncertainty (the content is not reliable). To address these problems, a new communication platform is explored, implemented and tested in a demonstration scenario. All participating parties (e.g., police, private security services, supporters, etc.) can communicate with each other by means of the most appropriate way at football events. Using an app on a smartphone, football supporters and other stakeholders can access this platform in order to obtain relevant event information or to provide new information. The overall architecture of the SiKomFan system is described in [2].

The semantic base of this communication platform is formed by an ontology, which is presented in this paper.

The paper is structured as follows. The Section II starts with related work, followed by Section III describing the used methodology for designing the ontology. The following sections IV to VIII are structured along the lines of the methodology: Section IV describes the purpose of the ontology. In Sections V to VII the building of the ontology is shown. Section VIII is about evaluating the ontology. The paper ends with an outlook on future work and a conclusion in Section IX.

ISBN: 978-1-61208-420-6

II. Related Work

The research presented in this paper was influenced and partly based on several existing ontologies. Those ontologies can be clustered by forming the following groups: Generic base ontologies, geospatial ontologies, time related ontologies, event ontologies (in the meaning "something happening"), event ontologies (like concerts or sport events) and sport ontologies, especially for the topic football/soccer.

The "DOLCE+DnS Ultralite Ontology" (DUL) has been developed as a common base for ontologies in the field of context modelling, e.g., it provides concepts like persons, organizations and roles along with relations among them. Another widely used base ontology regarding the modelling of persons and their relations is the "Friend of a Friend" (FOAF) ontology. This ontology only covers a smaller spectrum of the needed concepts.

The *GeoNames ontology* does not only define the structure for modelling locations and relations between them, it also provides a large amount of actual data. The GeoNames ontology itself uses the "Basic Geo Vocabulary" for defining the structures.

The Basic Geo Vocabulary [8] is a vocabulary for representing latitude, longitude and altitude information in the WGS84 geodetic reference datum, defined by the W3C Semantic Web Interest Group.

Time modelling aspects are addressed in the "Ontology of Time for the Semantic Web" (OWL-TIME). It provides a vocabulary for durations, time intervals and instants of time.

The "Event-Model-F" is based on the DUL ontology and added support for representing time and space, objects, persons and relationships between events. In contrast to other event ontologies, it allows modelling of causality relationships and representing different interpretations of the same event.

Further relations are shown in Section VI, where the integration and reuse of existing ontologies is discussed.

III. Methodology

For the design process of the SiKomFan ontology, the methodology suggested by Uschold and King [9] was used, which is a general approach for ontology design with a focus on the informal aspects.

The methodology foresees several steps:

1) *Identify the purpose of the ontology:* It is important to be clear about why the ontology is being built, what its intended uses are. Furthermore, the stakeholders and their environment must be defined.

2) *Building the ontology,* splits up into several sub-steps:

a) *Ontology Capture:* Identification of the key concepts and relationships in the domain of interest. Production of precise unambiguous text definitions for concepts and relationships. Identification of terms to refer to such concepts and relationships. Agreeing on all of the above.

b) *Ontology Coding:* Explicit representation of the conceptualization captured in a formal language.

c) *Integrating Existing Ontologies:* Use existing ontologies, which are already in use and widely accepted in the research community.

3) *Evaluation:* make a technical judgment of the ontologies, their associated software environments, and documentation with respect to the frame of reference. The frame of reference may be the requirements specification, competency questions and/or the real world

4) *Documentation:* important for updating and re-using important assumptions for understanding (Meta models). Inadequate documentation is one of the main barriers for effective use of ontologies.

The following sections are structured by these steps of the methodology. Step 2b and 2c are swapped since it showed to make more sense to first research existing ontologies for reuse before starting to code.

IV. Purpose of the SiKomFan ontology

The application of the ontology in the communication platform is threefold:

1. It will store a model of the current situation during a football event. Information coming into the system will be annotated with elements from the ontology to create a common meaning. Having such a common meaning is especially helpful for data integration of already existing systems.
2. All actors, which are involved into an event, are modelled including their relations, tasks and roles.
3. The structure of the ontology (relations between concepts) is used to navigate the system and the mobile app.

V. Building the ontology – Ontology Capture

To identify the ontological classes and relations a workshop was held, which brought together the different stakeholders of the project with their different research and domain knowledge backgrounds. This group of about ten people was a good size for effective discussion and the interdisciplinary meeting brought many new insights into the domain.

As a side effect also caused by the formalized nature of an ontology, the understanding of the involved actors and their relations was hugely increased between the project partners during the necessary discussion. In addition, the requirement to be specific about the formalization brought up topics, which were so far not fully understood. These open issues created new research items and therefore influenced the research of the other work packages.

A. *Stakeholders*

Figure 1. displays the top-level of all stakeholders, which participate directly or indirectly in the event *football match* and therefore have specific information and communication needs. Each of these top-level stakeholders contains a tree of up to ten specific groups, which are not displayed to keep the figure small.

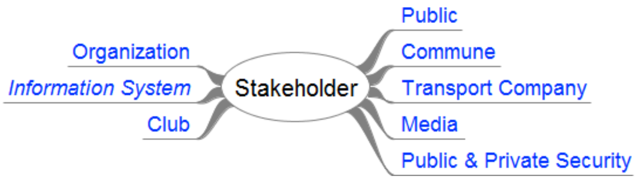


Figure 1. Top-level stakeholders of the event “football match”

A football *organization* is, e.g., an association like in Germany the Deutscher Fußball-Bund e.V. (DFB). They can send representatives to high-risk matches to check the safety precautions.

The stakeholder *Information System* is a special case since these are not human beings. Examples are existing systems for transport, police communication, information sites of clubs, etc. These systems provide information for the platform.

The *club* has several aspects: representatives of the guest and host club, the owner and manager of the stadium, the organizer, etc. The *public* is divided in visitors of the match and uninvolved people like abutters and affected travelers (train, car).

The *Transport Companies* are responsible for getting from and to the stadium, for example by train, bus or tram. Visitors are interested in delays and additional transport options for the match.

The *Media* are newspapers, radio, television, and Internet sites.

The *Public and Private Security* are emergency services like police, fire department, rescue services and private security companies. In Germany, there is the special situation that for one match there is police from different states involved and additionally also the *Bundespolizei* (federal police), which takes care about the railroad safety.

These stakeholders are the *actors* in the technical use-cases described in the following section.

B. Use-Case Scenarios

For inquiring communication processes between the different actors, a football event is split up into four scenarios.

The first one deals with the arrival and departure of spectators on the railway system of Deutsche Bahn AG (DB). In this sub-scenario, the (communicative) co-action of the federal police, the carriers and its services as well as the measures by football clubs are examined.

The second covers the travelling of spectators from a train station to the stadium. There are many different models in existence, for example the organized supporter march, the use of shuttle busses and the individual arrival by foot or with public transportation. The analysis in this context was focused on actions by state police and town councils but also accompanying measures by football clubs and public transport services.

The next scenario focuses on the interface of public area and the responsibility of the match-host: the entry controls. There, co-operation modalities between state police, town council and the club security commissioner as well as the private security service in the stadium are being included in the inquiry.

The fourth and last scenario examines the spectators’ visit in the stadium area. Here, actions by private security services in the stadium, the cooperation of different safety and security actors, for example in the safety and security operations headquarter, and the integration of the stadium announcer into communication concepts are inquired.

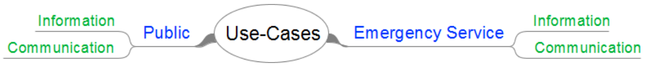


Figure 2. Top level Technical Use-Cases

From these scenarios, technical use-cases were created. Figure 2. depicts the top-level classification into the needs of the public and emergency services. Both of them are further divided into information (one-way) and communication (bi-directional). The complete mind map contains over 40 use-cases, therefore only some examples will be given:

- A father has lost his child in the crowded stadium and wonders whom to contact for support.
- A security officer wants to give an information to people in a certain area, e.g., “this entrance is overcrowded, please use entrance B”.
- A supporter wants to know which supporter items are allowed to bring into the stadium.

VI. Building the Ontology – Integrating Existing Ontologies

The evaluated ontologies (see Table I and also Section II) can be divided into the following categories:

- base ontologies (# 1 and 2)
- spatial ontologies (# 3 and 4)
- time related ontologies (# 5, 6, and 7)
- events (# 8 and 9), in the meaning of sport event or music event
- events (# 10 and 11), in the meaning that something happens and causes a reaction
- Sport and football (# 12, 13 and 14)

Additionally, ontologies from the domains of early warning and disaster would have been of interest but no suitable ontologies were found.

Based on our study we decided to pick "DOLCE+DnS Ultralite Ontology" (DUL) as the base ontology because it already covers the basic elements like people, organizations, roles, events and all relations between them.

VII. Building the ontology – Ontology Coding

Building on DUL all actors (supporters, transportation companies, clubs, unions, security organizations, police, press, etc.) and their roles were modelled. After that, time (for events and phases) and space (e.g., location of stadium) was added. Deciding on DUL set also the decision to use OWL (Web Ontology Language)/XML [10] as the coding language.

The decision to use automatic reasoning within the project poses an extra challenge for the combination of ontologies: The simple combination of different ontologies by defining equivalence relations between classes will lead to an inconsistent world model in most cases if not done carefully. This resulted in the definition of some criteria for the evaluation of other additional ontologies: The ontology should add substantial value to the base. It has to be combinable with DUL, i.e., it should have an OWL representation and defining classes from both ontologies as equivalent should not lead to inconsistencies. Furthermore, the integration of an ontology should only result in a minimal set of required dependencies since a larger number of involved ontologies will increment the potential of inconsistencies. In some cases, a consistent re-modelling of aspects from an existing ontology should be preferred over integrating the original ontology.

A. *Integration of Space and Time*

The “Geo-Pos” ontology [8] which was chosen for the spatial aspects is originally modelled as RDFS/XML and had to be transferred to OWL/XML.

The SiKomFan Ontology introduces the new class <#Position> (Fig. 3) as junction between the three ontologies DUL, Geo-Pos and SiKomFan. It is defined as subclass of <#SpaceRegion> from DUL since it is a spatial restriction in terms of the DUL terminology. The location attributes defined in Geo-Pos are added by declaring the Position class as EquivalentClass to <#Point> from Geo-Pos. With the help of this class, it is now possible to assign some absolute coordinates to entities.

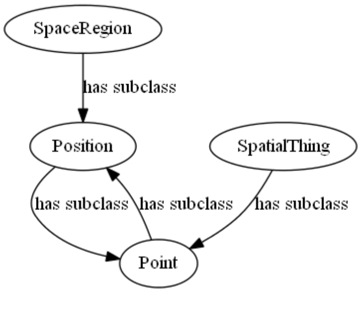


Figure 3. OWLClass Position

Some use cases of SiKomFan require the modelling of trajectories for persons and objects. This introduces the need for some time related attributes. To achieve this, the new class <#TemporalPosition> (Fig. 4) is defined that links <#Position> and <#TimeInterval>. It is defined by the restriction to have exactly one Position as location and to be observable at exactly one TimeInterval. The order of TemporalPositions within a trajectory is modelled with predecessor and successor relations between them.

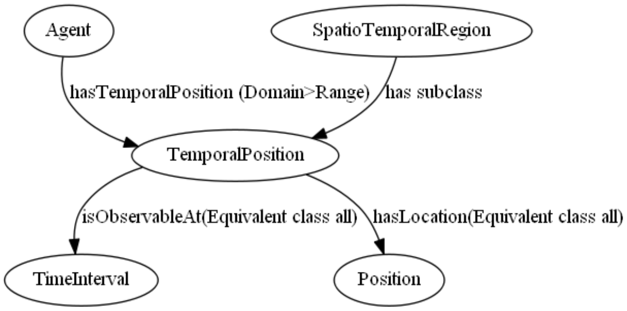

Agent
SpatioTemporalRegion
hasTemporalPosition (Domain>Range)
has subclass
TemporalPosition
isObservableAt(Equivalent class all)
hasLocation(Equivalent class all)
TimeInterval
Position

Figure 4. OWLClass TemporalPosition

B. *Extensions*

For the modelling of events (in the meaning of sport events), the decision was made to use a simple model based on the OWLClasses available in the DUL instead of importing a completely event-specific ontology. As an example, the football league is modeled as a new OWLClass <#League>, which is a subclass of <#Competition> which itself is a subclass of <#Event> from DUL. Other terms like arrival, match or season have been modeled in a similar way.

Persons acting as senders and receivers in the context of communication are important for the SiKomFan project. Based on research carried out by another work package in the project, some selected persons were modeled as part of the ontology. Therefore, the OWLClass <#SocialPerson> from DUL has been subdivided into several subhierarchies. The <#EmergencyServicesPerson> is the top-level class for all persons from emergency services, which are modelled in detail. In a similar way, the <#EventVisitor> is the top of another hierarchy for the persons visiting the event.

Besides the persons, IT systems can also be the origin of communication and have to be modeled also. In this case, <#System> is defined as the common base class for systems like Social Media, information pages, press portals and traffic information services. That class is defined as subclass of <#PhysicalAgent> from DUL.

The last major extension to the DUL is the subhierarchy beneath <#Organization> from DUL. This subhierarchy contains all the organizational aspects of emergency services as well as fan groups, companies (e.g., railway operators) and football clubs.

Table II shows some simple metrics of the three ontologies DUL, Geo-Pos and SiKomFan in comparison.

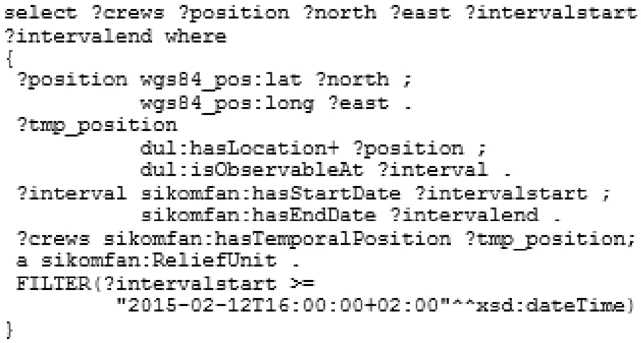
VIII. Evaluation of the Ontology in the Communication Platform

The system architecture of the communication platform has been derived from the use-case scenarios as well. Components of the sequence diagrams describing the technical use-cases have been grouped according to their functionalities: user interface (Apps and Desktop Applications), services (e.g., map or positioning services), data (e.g., information about the football event) and data sources (e.g., stakeholders or social media).

The main part of the system is the so-called Situation; it contains the connected information about the current football event. It can be visualized on a map showing various aspects for different actors and roles. For example, a visitor can see the location and availability of the transfer shuttle to the train station. The Situation is saved in an ontology store using OpenLink Virtuoso [7] facilitating the SiKomFan-ontology described above. The ontology store is connected to a web content management system (WCMS), which implements the additional functionality like visualization of a situation, role authorization, group notifications, etc.

The ontology was tested with queries and example data created from the SiKomFan use-cases.

Fig.5 shows a SPARQL [11] query, which returns positions of relief units at a specific time:



```
select ?crews ?position ?north ?east ?intervalstart
?intervalend where
{
 ?position wgs84_pos:lat ?north ;
           wgs84_pos:long ?east .
 ?tmp_position
           dul:hasLocation+ ?position ;
           dul:isObservableAt ?interval .
 ?interval sikomfan:hasStartDate ?intervalstart ;
           sikomfan:hasEndDate ?intervalend .
 ?crews sikomfan:hasTemporalPosition ?tmp_position;
 a sikomfan:ReliefUnit .
 FILTER(?intervalstart >=
         "2015-02-12T16:00:00+02:00"^^xsd:dateTime)
}
```

Figure 5. SPARQL query for evaluation

Several scenarios with up to 2000 police officers were simulated for the evaluation of the queries. Depending on the scenario size the query execution duration varied between 20ms and 3000ms.

IX. Conclusion and Future Work

In this paper, the first iteration of the SiKomFan ontology was presented, which addresses the manifold stakeholders and use-cases of the event football match. Using the methodology suggested by Uschold the purpose of the ontology was identified first and their content was defined at a stakeholder's workshop. After that, the ontology was designed building upon existing established ontologies. Finally, the ontology was evaluated by testing the SiKomFan use-cases.

Moreover, but not presented here, parallel sub-projects examine supporter cultures and their perspectives on safety and security as well as legal recommendations for a better information exchange and for optimizing the cognizance of public actors around football matches. These sub-projects therefore seek to deliver suggestions to optimize the communication strategies of the stakeholders, the communication processes between the stakeholders and especially to optimize the dialogue between supporters and the stakeholders. The new results of this interdisciplinary research will be taken into account and the ontology will be adapted accordingly in the second phase of the project.

Acknowledgment

The research project SiKomFan is funded by the German Ministry of Education and Research (BMBF). SiKomFan is currently in its second year of research and will conclude with final results in August 2016. For more information see: www.sikomfan.de.

References

[1] Zuschauergeschichte [Online] available from http://www.kicker.de/news/fussball/bundesliga/spieltag/1-bundesliga/zuschauer-geschichte.html 2015.05.20

[2] J. Moßgraber, T. Kubera, and A. Werner, "More Safety for Football Events: Improving the Communication of Stakeholders and the Dialogue with Supporters," Proceedings of the Future Security 2014, Berlin, Germany, 2014.

[3] Manuel, G., Gummadi, K., & Schoelkopf, B. (in press). Quantifying Information Overload in Social Media and its Impact on Social Contagions. In ICSWM '14. [Online] Available from http://hdl.handle.net/11858/00-001M-0000-0026-AE1B-4 2015.05.20

[4] J. R. Hobbs and F. Pan, "An Ontology of Time for the Semantic Web," ACM Transactions on Asian Language Processing (TALIP): Special issue on Temporal Information Processing, Vol. 3, No. 1, pp. 66-85, 2004.

[5] A. Scherp, T. Franz, C. Saathoff, and S. Staab, "F---A Model of Events based on the Foundational Ontology DOLCE+DnS Ultralight," International Conference on Knowledge Capturing (K-CAP), Redondo Beach, CA, USA, September, 2009.

[6] R. Troncy, B. Malocha, and A.Fialho, "Linking events with media," In Proceedings of the 6th International Conference on Semantic Systems (I-SEMANTICS '10), Adrian Paschke, Nicola Henze, and Tassilo Pellegrini (Eds.). ACM, New York, NY, USA, , Article 42 , 4 pages. DOI=10.1145/1839707.1839759 http://doi.acm.org/10.1145/1839707.1839759, 2010.

[7] O. Erling and I. Mikhailov, "Virtuoso: RDF Support in a Native RDBMS," Semantic Web Information Management, 2009, pp. 501-519.

[8] Geo-Pos Ontology [Online] available from http://www.w3.org/2003/01/geo/wgs84_pos# 2015.05.20

[9] M. Uschold and M. King, "Towards a Methodology for Building Ontologies," AIAI-TR-183, presented at Workshop on Basic Ontological Issuesin Knowledge Sharing; held in conjunction with IJCAI-95, 1995.

[10] G. Antoniou and F. van Harmelen, "Web Ontology Language: OWL," Handbook on Ontologies, International Handbooks on Information Systems 2004, pp 67-92, 2004.

[11] E. Prud'hommeaux and A. Seaborne, "Sparql query language for rdf," W3C Working Draft, http://www.w3.org/TR/rdf-sparql-query/, 2006.

TABLE I. RESEARCHED ONTOLOGIES FOR INTEGRATION INTO THE SIKOMFAN ONTOLOGY

#	URL	Coding	Language	License	Published	Comment
	Base ontologies					
1	http://www.ontologydesignpatterns.org/ont/dul/DUL.owl	OWL/XML	en, it			Organizations, Relations, Planning, Events, …
2	http://xmlns.com/foaf/0.1/	OWL/XML	en	CC BY 1.0	2014	Persons, Organizations and their relations
	Location ontologies					
3	http://www.geonames.org/ontology/documentation.html	OWL/XML	en, no, sv, bg, ru	CC BY 3.0	2012	Locations, Countries, Population, ZIP, …
4	http://www.w3.org/2003/01/geo/	RDFS/XML	en		2009	Geo locations
	Time related ontologies					
5	http://www.w3.org/2006/time#	OWL/XML	en		2006	Time
6	http://www.ontologydesignpatterns.org/cp/owl/timeindexedsituation.owl	OWL/XML	en		2011	Order of events
7	http://motools.sourceforge.net/timeline/timeline.html	OWL/XML	en		2007	Order of events, extends OWL-Time
	Events (in the meaning of sport event or music event) ontologies					
8	http://motools.sourceforge.net/event/event.html	OWL/N3	en	CC BY 3.0	2007	Rudimentary
9	http://linkedevents.org/ontology/	OWL/XML	en	CC BY-SA 3.0	2010	Rudimentary
	Events (something happens and causes a reaction) ontologies					
10	http://west.uni-koblenz.de/Research/ontologies/events/index_html	OWL/XML	en		2009	Time and space, objects, cause and impact; extends DUL
11	http://www.dcs.shef.ac.uk/~vita/files/ER-events.owl	OWL/XML	en		20xx	Diseases, fire protest, weather. No Properties.
	Sport events ontologies					
12	http://purl.org/ontology/sport/	OWL/XML	en		2011	Sport events, leagues, teams, etc.
13	http://www.r4isstatic.com/linkeddata/ontologies/football/football.owl	OWL/XML	en		2009	Leagues, teams, countries
	Ontologies about football (the game itself)					
14	http://www.lgi2p.ema.fr/~ranwezs/ontologies/soccerV2.0.daml	DAML/XML	en		2002	Events in the game

TABLE II. ONTOLOGY METRICS

Ontology	Class count	ObjectProperty count	DatatypeProperty count	Individual Count
DUL	75	104	5	-
Geo-Pos	2	1	3	-
SiKomFan (without DUL and Geo-Pos)	124	38	21	210

Text Document Clustering for Topic Discovery by Hypergraph Construction

Wei-San Lin

Graduate Institute of Biomedical Informatic
Taipei Medical University
Taipei, Taiwan 110
Email: g658102003@tmu.edu.tw

Charles Chih-Ho Liu

Cathay General Hospital
Taipei, Taiwan 106
Email: chliu@cgh.org.tw

I-Jen Chiang

Graduate Institute of Biomedical Informatic
Taipei Medical University
Taipei, Taiwan 110
Email: ijchiang@tmu.edu.tw

***Abstract*—The paper presents a hypergraph model and HYPERGRAPH DECOMPOSITION ALGORITHM for text document clustering. The experiments on three different data sets from news, Web, and medical literatures have shown our algorithm is significantly better than traditional clustering algorithms, such as K-MEANS, PRINCIPAL DIRECTION DIVISIVE PARTITIONING , AUTOCLASS and HIERACHICAL CLUSTERING.**

Keywords–document categorization/clustering; hypergraph;association rules; hypergraph components decomposition (HCD); hierarchical clustering (HCA); partition-based hypergraph algorithm.

I. Introduction

The exponential growth in the volume and the popularity of Web information, such as news, social media, scientific articles and discussion forums, induce a Big Data problem. Since massive amounts of contents have generated everyday, automatically discovering and organizing contextually relevant text information are very challenging.

How to get web sophisticated information mining strategies will be needed. Document clustering can deal with the diverse and large amount of Web information and particularly is used to discover latent concepts in a collection of Web documents, which is inherently useful in organizing, summarizing, disambiguating, and searching through large document collections [1].

Text document clustering is an unsupervised learning technique that has created a demand for a mechanism to discover topics from heterogeneous information. Document clustering aims to generate topic groups or clusters from a document collections. According to a single document, the content can mingle heterogeneous topics, the obtained topics from document clustering methods sometimes do not necessarily correspond to actual topics of interest and document clustering methods do not provide descriptions that summarize the clusters' contents [2]. Many clustering methods, such as k-means, hierarchical clustering (algorithms and non-negative matrix factorization (NMF) have been performed on the matrix to group the documents. However, these methods lack ability of interpretation to each document cluster [3].

In what follows, we start by briefly reviewing the related work in Section II and defining the frequent itemsets in a collection of documents in Section III, and generating a graph model of representing the concepts from the frequent itemsets in Section IV, then presenting the topic based clustering algorithm for partitioning documents into several semantic topics in Section V. In Section VI, you can see each of which represents a concept in the document collection, and documents can then be clustered based on the primitive concepts identified by this algorithm. The three different experimental data sets are also described in Section VII, and finally get into the conclusion in the last section, which showed a novel approach to document clustering which is compared with k-means, HCA, AutoClass or the Principal Direction Divisive Partitioning (PDDP). However, how to provide much more guarantee on precision, even for detailed queries is still an open research problem.

II. Related work

The frequent itemsets (undirected association rules) can demonstrate semantic topics and can be extracted from documents. A single item, i.e., word, does not carry much information about a document, yet a huge amount of items may nearly identify the document uniquely. Therefore, finding all meaningful frequent itemsets in a collection of textual documents presents a great interest and challenge.

Feldman and his colleagues [4], [5], [6] proposed the *KDT* and *FACT* system to discover association rules based on keywords labelling the documents, the background knowledge of keywords and relationships between them, but it is ineffective. Therefore, an automated approach that documents are labelled by the rules learned from labelled documents [7]. However, several association rules are constructed by a compound word (such as “Wall” and “Street” often co-occur) [8]. Feldman et al. [4], [9] further proposed term extraction modules to generate association rules by selected key words. It is beneficial for us to obtain meaningful results without the need to label documents by human experts. Association rule hypergraph partition was proposed in [10] to transform documents into a transactional database form, and then apply hypergraph partitioning [11] to find the item clusters. Holt and Chung [12] addressed Multipass-Apriori and Multipass-DHP algorithms to efficiently find association rules in text by modified the Apriori algorithm [13] and the DHP algorithm [14] respectively. Those methods did not consider to identify the importance of a word in a document. Hence, they addressed two clustering methods, CFWS and CFWM, to perform document clustering [15] by considering the sequential aspect of word occurrences.

III. Frequent Itemsets

Association rules was first introduced by Agrawal et al. [16] wherein two standard measures, called *support* and *confidence*, are often used. This paper only focuses on the support; a set of items that meets the support will be called the (undirected) association rules. The association rules are thereby for the use of finding co-occurring frequent terms in documents.

A. Feature Extraction

Feature extraction is to extract key terms from a collection of documents; And various methods such as association rules algorithms may be applied to determine relations between features.

This paper considers only noun entities, especially some representative entities. All NP chunkers extracted by *part-of-speech* (POS) tagger are weighted with respect to the documents after NP chunkers have been recognised and extracted. The simple and sophisticated weighted schema which is most common used in IR or IE is TFIDF indexing, i.e., $\mathrm{tf} \times \mathrm{idf}$ indexing [17], where tf denotes term frequency that appears in the document and idf denotes inverse document frequency [18] where document frequency is the number of documents which contain the NP chunkers. It takes effect on the commonly used NP chunker a relatively small $\mathrm{tf} \times \mathrm{idf}$ value. Moffat and Zobel [19] pointed out that $\mathrm{tf} \times \mathrm{idf}$ function demonstrates: (1) rare NP chunkers are no less important than frequent NP chunkers in according to their idf values; (2) multiple appearances of an NP chunker in a document are no less important than single appearances in according to their tf values. The $\mathrm{tf} \times \mathrm{idf}$ implies the significance of a term in a document, which can be defined as follows.

Definition 1: Let T_r be a collection of documents. The significance of a term, i.e., NP chunker t_i in a document d_j in T_r is its TFIDF value calculated by the function $\mathrm{tfidf}(t_i, d_j)$, which is equivalent to the value $\mathrm{tf(t_i, d_j)} \times \mathrm{idf(t_i, d_j)}$. It can be calculated as

$$\mathrm{tfidf(t_i, d_j) = tf(t_i, d_j) \log \frac{|T_r|}{|T_r(t_i)|}} \tag{1}$$

where $|T_r(t_i)|$ denotes the number of documents in T_r in which t_i occurs at least once, and

$$\mathrm{tf(t_i, d_j)} = \begin{cases} 1 + \log(N(t_i, d_j)) & \text{if } N(t_i, d_j) > 0 \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

where $N(t_i, d_j)$ denotes the frequency of terms t_i occurs in document d_j by counting all its nonstop words.

For the purpose of document clustering, we only need to consider when a set of terms that co-occur would become a concept. The metric *support* is used for defining the co-occurred term association. All the documents that are composed of those terms are able to organise a semantic cluster. Let t_A and t_B be two terms. The *support* defined for a collection of documents is as follows.

Definition 2: Support denotes to the specific significance of the documents in T_r that contains both term t_A and term t_B, that is,

$$\mathrm{Support(t_A, t_B) = \frac{tfidf(t_A, t_B, T_r)}{|T_r|}} \tag{3}$$

where

$$\mathrm{tfidf(t_A, t_B, T_r) = \frac{1}{|T_r|} \sum_{i=0}^{|T_r|} tfidf(t_A, t_B, d_i)} \tag{4}$$

$$\mathrm{tfidf(t_A, t_B, d_i) = tf(t_A, t_B, d_i) \log \frac{|T_r|}{|T_r(t_A, t_B)|}} \tag{5}$$

and $|T_r(t_A, t_B)|$ define number of documents contained both term t_A and term t_B.

The term frequency $\mathrm{tf(t_A, t_B, d_i)}$ of both chunkers t_A and t_B can be calculated as follows.

Definition 3:

$$\mathrm{tf(t_A, t_B, d_j)} = \begin{cases} 1 + \log(\min\{N(t_A, d_j), N(t_B, d_j)\}) \\ \text{if } N(t_A, d_j) > 0 \text{ and } N(t_B, d_j) > 0 \\ \\ 0 \\ \text{otherwise.} \end{cases} \tag{6}$$

A minimal support θ is given to filter the chunkers that their TFIDF values are less than θ. It helps us to eliminate the most common chunkers in a collection and the nonspecific chunkers in a document.

Suppose that with regard to a query term "network", the underlying graph is generated as shown in Figure 1. Each edge denotes the association between two terms is great than a given threshold and illustrates a semantic concept.

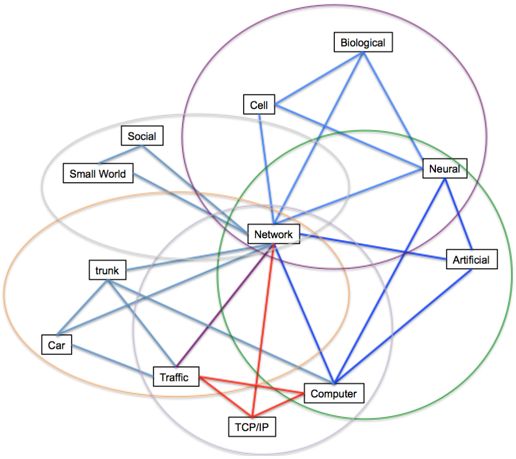


Figure 1. A graph structure generated by the query term "network."

IV. Graph Model of Frequent Itemsets

The set of all frequent itemsets of documents can form a hypergraph of NP chunkers, and this hypergraph can represent the totality of thoughts expressed in this collection of documents. A "simple component" of frequent itemsets organizes hypergraph that represents semantic concepts inside this collection of documents.

A. Preliminary

Let us briefly introduce hypergraphs and define some preliminaries for further descriptions.

Definition 4: A weighted *hypergraph* $G = (V, E, W)$ contains three distinct sets where (1) V is a finite set of vertices, called ground set, (2) $E = \{e_1, e_2, \cdots, e_m\}$ is a non-empty family of finite subsets of V, in which each subset is called a n*-hyperedge* (where $n + 1$ is the cardinality of the subset), and $W = \{w_1, w_2, \cdots, w_m\}$ is a weight set. Each hyperedge e_i is assigned a weight w_i.

Two vertices u and v are said to be *r-connected* in a hypergraph if either $u = v$ or there exists a path from u to v (a sequence of r-hyperedge, $(u_j, u_{(j+1)})$, $u_0 = u, \ldots, u_n = v$).

A r-connected hyperedge is called a *r-connected component* or *r-topic*.

B. Concept

For a collection of documents, we generate a hypergraph of frequent itemsets. Note that because of *Apriori* conditions, this hypergraph is closed. The goal of this paper is to establish the following belief.

Claim A connected component of a hypergraph represents a primitive *concept* in this collection of documents.

Hypergraphs are a perfect method to represent association rules. As seen in Figure 1, the vertex set V ={“network”, “artificial”, “biological”, “car”, $\cdots$ } that represents the set of key chunkers in a collection of documents, the edge set E that represents term association rules in the graph. In the graph, each circle represents a higer order association rules, which is a hyperedge. Each circle is also a complete subgraph that its support is bigger than a minimum support, so are all the non-empty subsets of it. In a hypergraph, the universe of vertices organizes 1-item frequent itemsets, the universe of 1-hyperedge represents all possible 1-item and 2-item frequent itemsets, and so on.

V. Topic-based Graph Model

This section will introduce the algorithm to find all frequent itemsets in documents that is generated from the co-occurring chunkers in a collection of documents.

A. Weighted Incident Matrix

The *weighted incident matrix* is
Definition 5:

$$a'_{ij} = \begin{cases} w_{ij} & \text{if } v_i \in e_j \\ 0 & \text{otherwise.} \end{cases} \quad (8)$$

where the weight w_{ij} denotes the *support* of an frequent itemset.

Each vertex in V represent a chunker that have been reserved (i.e., its support is greater that a given minimal support θ), and each hyperedge in E is undirected that identifies a support incident with an itemset. Each edge-connector denotes a topic, i.e., an undirected association rules. The number of chunker in an edge-connector defines the *rank* of a hyperedge. An edge-connector of a hyperedge with rank r is said to be a r-hyperedge or r-connected component. As seen in Figure 1, for instance, the set { “network”, “artificial”, “neural”, “computer” } is an edge-connector of a 4-hyperedge that could represents an “artificial neural network” topic.

B. Algorithm

A r-hyperedge denotes a r-topic, which is a r-frequent itemset. If we say a frequent itemset I_i identified by a hyperedge e_i is a subset of a frequent itemset I_j identified by e_j, it means that $e_i \subset e_j$. A hyperedge e_i is said to be a maximal topic if no other hyperedge $e_j \in E$ is the superset of e_i for $i \neq j$. Documents can be automatically clustered based all maximal topics. Considering an example in Figure 1, there are four maximal topics that both of them are 4-hyperedges in a hypergraph. One component is organized by the hyperedge { “network”, “artificial”, “neural”, “computer” }, and { “network”, “biological”, “neural”, “cell” } is another generated hyperedge. The boundary of a concept defines all possible term associations in a document collection. Both of them share a common concept that can be taken as a 1-hyperedge { “network”, “neural” }, which is an 2-item frequent itemset. Since all connected components are convex hulls, the intersection of connected components is nothing or a connected component.

Property 1: The intersection of concepts is nothing or a concept that is a maximal closed hyperedge belonging to all intersected concepts.

Since there is at most one maximal closed hyperedge in the intersection of more than one connected topics and the dimension or rank of the intersection is lower than all intersected hyperedges. It is convenient for us to design an efficient algorithm for documents clustering based on all maximal connected components in a hypergraph not needed to traverse all hyperedges. The algorithm for finding all maximal connected components is listed as follows.

```
Require: $V = \{t_1, t_2, \cdots, t_n\}$ be the vertex set of all reserved
    NP chunkers in a collection of documents.
Ensure: $\mathcal{E}$ is the set of all maximal connected components.
    Let $\theta$ be a given minimal support.
    $\mathcal{E} \Leftarrow \emptyset$
    Let $E_0 = \{e_i | e_i = \{t_i\} \forall t_i \in V\}$ be the 0-hyperedge set.
    $i \Leftarrow 0$
    while $E_i \neq \emptyset$ do
      while for all vertex $t_j \in V$ do
        $E_{(i+1)} \Leftarrow \emptyset$ be the $i+1$-hyperedge set.
        while for all element $e \in E_i$ do
          if $e' = e \bigcup \{t_j\}$ with $t_j \notin e$ whose support is no
          less than $\theta$ then
            add $e'$ in $E_{(i+1)}$
            remove $e$ from $E_i$
          end if
        end while
      end while
      $\mathcal{E} \Leftarrow \mathcal{E} \bigcup E_i$
      $i \Leftarrow i + 1$
    end while
```

All the hyperedges in $\mathcal{E}$ are maximal connected components. A hyperedge will be constructed by including all those co-occurring terms whose support is bigger than or equal to a given minimal support θ. An external vertex will be added into a hyperedge if the produced support is no less than θ. It is not necessary that the intersection of any two hyperedges in $\mathcal{E}$ is empty because the intersection can be taken as the common concept that both own as we have already stated. According to the Property 1, when a maximal connected component is found, all its subcomponents are also included in the hyperedge.

The documents can be decomposed into several categories based on its correspond concept that is represented by a hyperedge in $\mathcal{E}$. If a document consists in a concept, it means that document highly equates to such concept, thereby all the terms in a concept is also contained in this document. The document can be classified into the category identified with such concept. A document often consists of more than one concept and it can be classified into multi-categories.

VI. Experimental Results

Experimental results are conducted to evaluate the clustering algorithm, rather than analytic statements.

A. Data Sets

Three data sets are involved in making the validation and evaluating the performance of our model and algorithm. Effectiveness is the important criterion for the validity of clustering.

The first dataset is Web pages collected from Boley et al.[10]. 98 Web pages in four broad categories: business and finance, electronic communication and networking, labor and manufacturing are selected for the experiments. Each category is also divided into four subcategories.

The second dataset is the “Reuters-21578, Distribution 1” collection consisted of newswire articles, which is a multi-class, multi-labelled benchmark containing over 21000 newswires articles that are assigned 135 so-called topics. These topics refer to financial news related to different industries, countries and other categories. In our test 9494 documents are selected in which all multi-categorized documents were discarded and the categories with less than five documents have been removed.

The third dataset is 305 electronic medical literatures collected from the journals, *Transfusion*, *Transfusion Medicine*, *Transfusion Science*, *Journal of Pediatrics* and *Archives of Diseases in Childhood Fetal and Neonatal Edition*. Those articles are selected by searching from keywords, *transfusion*, *newborn*, *fetal* and *pediatrics*. The MeSH categories have the use of evaluating the effectiveness of our algorithm. It is best for us to make external validities on the concepts generated from our method by human experts.

B. Evaluation Criteria

The experimental evaluation of document clustering approaches usually measures their *effectiveness* rather than their *efficiency* [20], in the other words, the ability of an approach to make a *right* categorization.

TABLE I. THE CONTINGENCY TABLE FOR CATEGORY c_i.

Category c_i		Clustering Results	
		YES	**NO**
Expert Judgment	**YES**	$\mathrm{TP_i}$	$\mathrm{FN_i}$
	NO	$\mathrm{FP_i}$	$\mathrm{TN_i}$

Considering the contingency table for a category (Table 1), *recall*, *precision*, and F_β are three measures of the effectiveness of a clustering method. Precision and recall with respect to a category is defined as follows:

$$\mathrm{Precision_i} = \frac{\mathrm{TP_i}}{\mathrm{TP_i} + \mathrm{FP_i}} \tag{9}$$

$$\mathrm{Recall_i} = \frac{\mathrm{TP_i}}{\mathrm{TP_i} + \mathrm{FN_i}} \tag{10}$$

The F_β measure combined with precision and recall has introduced by van Rijsbergen in 1979 as the following formula:

$$F_\beta = \frac{(\beta^2+1) \times \mathrm{Precision_i} \times \mathrm{Recall_i}}{\beta^2 \times \mathrm{Precision_i} + \mathrm{Recall_i}} \tag{11}$$

F_1 measure is used in this paper, which is obtained when β is set to be 1 that means precision and recall are equally weighted for evaluating the performance of clustering. Because of many categories that will be generated and the comparison reasons, the overall precision and recall are calculated as the average of all precisions and recalls belonging to some categories, respectively. F_1 is calculated as the mean of individual results. It is a macro-average among categories.

In a non-overlapping scenario, each document belongs to exactly one cluster. Three validation metrics: precision, recall and F-measure, are proper to evaluate the performance of crisp clustering algorithms. The overlapping clustering schemes has been involved in a widely variety of application domains because many real problems are naturally overlapped. Information theoretic measures [21], [22], such as entropy and mutual information, hence have been used to estimate how much information is shared from the labelled instances in a cluster, especially, for a hierarchical clustering schemes [23]. In order to compare effectiveness with other methods, two different evaluation metrics, *normalized mutual information* [24], [21], [25] and *overall F-measure* [26], [27], were also used.

C. Results

The result of the first experiment is presented in Table II. The result of PDDP algorithm [10], is under consideration by all non-stop words, that is, the F1 database in their paper, with 16 clusters. The result of our algorithm, HCD, is under consideration by all non-stop words with the minimal support, 0.15 by comparing with four algorithms, HCD, PDDP, k-means and AutoClass. The PDDP algorithm splits the data into two subsets hierarchically. Based on the principal direction, i.e. principal component analysis, it also derives a linear discriminant function. Principal component analysis often hurts the results of classification if with sparse and high dimensional datasets, and induces a high false positive rate and false negative rate. Based on the average of the confidences of the frequent itemsets with the same items, PDDP generates the hyperedges. It is unfair that a possible concept would be withdrawn if a very small confidence of an itemset is existed from an implication direction.

TABLE II. THE PERFORMANCE COMPARISON ON THE FIRST DATASET.

Method	HCD	PDDP	k_means	AutoClass	HCA
Precision	68.3%	65.6%	56.7%	34.2%	35%
Recall	74.2%	68.4%	34.9%	23.6%	22.5%
F_1 measure	0.727	0.67	0.432	0.279	0.274

ISBN: 978-1-61208-420-6

In the first dataset, HCD generates 47 clusters, i.e., maximal connected components, as shown in Figure 2. It is larger than the original 16 clusters. After performing on decreasing the minimal support value to be 0.1, the number of clusters reduces to be 23 and its precision, recall, and F_1, become 63.7%, 77.3%, 0.698 respectively. The higher the minimal support value is, the lower the number of co-occurred terms in a hypergraph. Precision is worse than PDDP with lower minimal support because the clustering constraints generated from hyperedges are stronger to filter some documents that should be included, which makes a high false positive rate. Figure 3 demonstrates the performance on the first dataset of HCD.

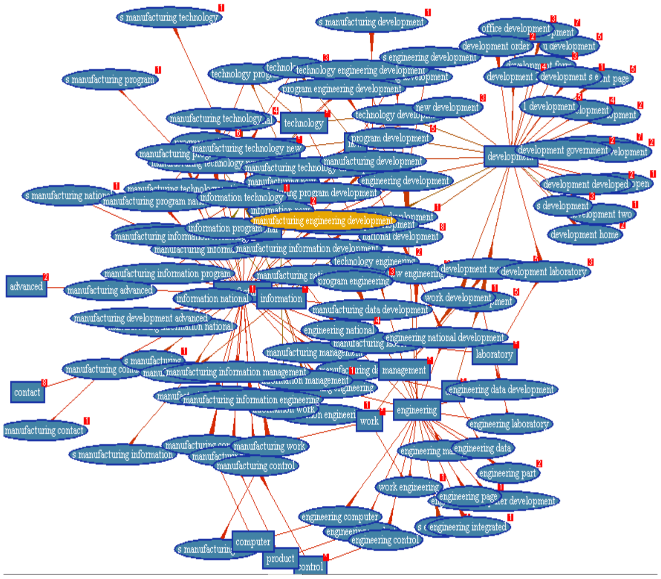


Figure 2. The hypergraph generated from the first dataset by using HCD.

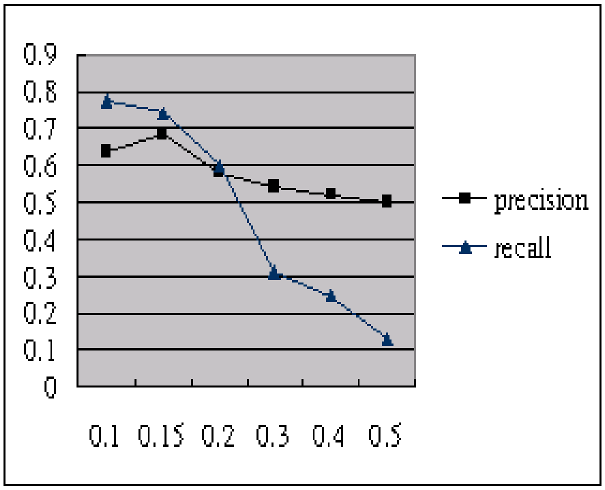


Figure 3. The effectiveness of HCD on the first dataset.

The evaluation was conducted for the cluster numbers ranging from 2 to 10 on the Reuters data set. For each given cluster number k, the performance scores were obtained by averaging those k randomly chosen clusters from the Reuters corpus in an one-run test. Some terms indicated a generic category in Reuters classifications are not designated the same category, so that the number of clusters is larger than the number of Reuters' categories. Table 3 indicates the evaluation results using the Reuters dataset in Figure 4.

TABLE III. THE PERFORMANCE OF REUTERS DATASET BY HCD.

HCD	k=2	k=3	k=4	k=5
Precision	93%	90.8%	93.8%	86.1%
Recall	68%	63.5%	77.9%	76.2%
F_1 measure	0.834	0.774	0.814	0.77

The MeSH categories (22 categories) have been taken to evaluate the effectiveness of HCD on each individual category of the third dataset. Document clustering is based on the MeSH terms related to "Transfusion" and "Pediatrics". The effectiveness of all categories is shown in Figure 5. The MeSH categories are a hierarchical structure that some categories are the subcategories of the other categories. Many concept categories are shared with the same terminology that induces a high false negative rate by HCD on document clustering. In this dataset, documents are not uniform distributed in all categories, some categories only contain a few documents that makes their latent concepts restricted by a few terms, for example, the *Anemia* and the *Surgery* categories whose precision are both below 70%.

VII. CONCLUSION

Concept identification from text documents is an open research problem. While *polysemy*, *phrases* and *chunker dependency* present additional challenges for search technology, single chunker are often insufficient to identify specific concepts in a document. Discriminating NP chunker associations naturally helps distinguish one topic from the others. A group of solid chunker associations can clearly identify a concept. While most methods, like *k-means*, *HCA*, *AutoClass* or *PDDP* classify/cluster documents from the matrix representation, matrix operations cannot discover all chunker associations. Hypergraphs allow a efficient way to find chunker associations in a collection of documents.

This paper presents a novel approach to document clustering based on hypergraph decomposition. An agglomerative method without the use of distance function is proposed. A hypergraph is constructed from the set of co-occurring frequent chunkers in the text documents. The r-hyperedges, i.e., r-topics, can represent basic concepts in the document collection. We presented a simple algorithm that can effectively discover the maximal connected components of co-occurring frequent chunkers. cluster documents. The proposed method is compared with traditional clustering methods, such as *k-means*, *AutoClass* and *HCA*, as well as the partition-based hypergraph algorithm, *PDDP*, on three data sets in our experiments. The hypergraph component decomposition algorithm demonstrated superior performance in document clustering. The results illustrate that hypergraphs are a perfect model to denote association rules in text and is very useful for automatic document clustering.

Our experiments also showed that the value of r is dependent on the given minimal support. The r-connected components represent the r-frequent itemsets with r different

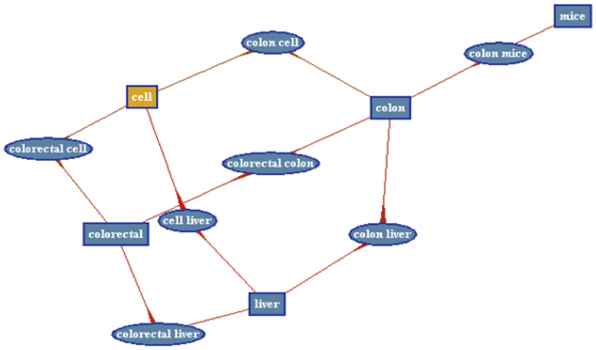


Figure 4. The hypergraph generated from the second dataset with minimal support, 0.1.

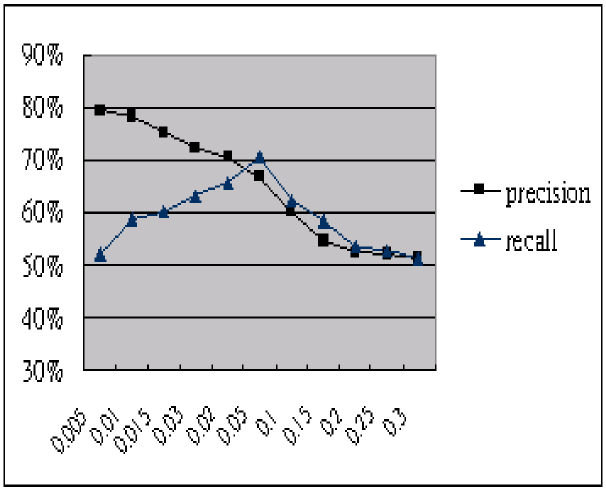


Figure 5. The effectiveness of HCD on the second dataset.

chunkers. The higher the minimal support value is, the lower the value of r is. That is, the number of co-occurring chunkers for organizing concepts in a collection of documents decreases with a higher minimal support value. In other words, the support for a more general concept is higher than the support of a more specific concept. That is, a general concept is less effective in classifying/clustering documents.

The strengths of our methods are: 1) an agglomerative Web document hierarchical clustering is addressed by using graph construction; 2) a hypergraph properly represents the concept organized by the associations of terms in a collection of documents; 3) considering the overlap of semantics between documents, our method can provide more comprehensible clustering results allowing concept overlap. However, as seen in Figure 1, the hyperedge neural, network in the hypergraph is an ambiguous concept. Not until the upper-leveled hyperedges, biological, cell, neural, network and computer, artificial, neural, network have been generated we could clearly identify these two distinct concepts. The weakness of our method is lack of considering uncertainties within documents. We will further consider to develop a fuzzy model on uncertainties.

REFERENCES

[1] R. Kosala and H. Blockeel, "Web mining research: A survey," *SIGKDD Explorations*, vol. 2, no. 1, pp. 1–15, 2000.

[2] H. Anaya-Sánchez, A. Pons-Porrata, and R. Berlanga-Llavori, "A document clustering algorithm for discovering and describing topics," *Pattern Recognition Letters*, vol. 31, pp. 502–510, 2010.

[3] D. Wang, S. Zhu, T. Li, Y. Chi, and Y. Gong, "Integrating document clustering and multidocument summarization," *ACM Transactions on Knowledge Discovery from Data*, vol. 5, no. 3, pp. 14–26, 2011.

[4] R. Feldman, Y. Aumann, A. Amir, W. Klósgen, and A. Zilberstien, "Text mining at the term level," in *Proceedings of 3rd International Conference on Knowledge Discovery, KDD-97*, Newport Beach, CA, 1998, pp. 167–172.

[5] R. Feldman, I. Dagan, and W. Klósgen, "Efficient algorithms for mining and manipulating associations in texts," in *Cybernetics and Systems, The 13th European Meeting on Cybernetics and Research*, vol. II, Vienna, Austria, April 1996.

[6] R. Feldman and H. Hirsh, "Mining associations in text in the presence of background knowledge," in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96)*, 1996, pp. 343–346.

[7] B. Lent, R. Agrawal, and R. Srikant, "Discovering trends in text databases," in *Proceedings of 3rd International Conference on Knowledge Discovery, KDD-97*, Newport Beach, CA, 1997, pp. 227–230.

[8] M. Rajman and R. Besanon, "Text mining: Natural language techniques and text mining applications," in *Proceedings of seventh IFIP* 2.6 *Working Conference on Database Semantics (DS-7)*, Leysin, Switzerland, 1997.

[9] R. Feldman, M. Fresko, H. Hirsh, Y. Aumann, O. Liphstat, Y. Schler, and M. Rajman, "Knowledge management: A text mining approach," in *Proceedings of 2nd International Conference on Practical Aspects of Knowledge Management*, Basel, Switzerland, 1998, pp. 29–30.

[10] D. Boley, M. Gini, R. Gross, E.-H. Han, K. Hastings, G. Karypis, V. Kumar, B. Mobasher, and J. Moore, "Document categorization and query generation on the world wide web using webace," *Artificial Intelligence Review*, vol. 13, no. 5-6, pp. 365–391, 1999.

[11] G. Karypis, R. Aggarwal, V. Kumar, and S. Shekhar, "Multilevel hypergraph partition application in vlsi domain," *Proceedings ACM/IEEE Design Automation Conference*, vol. 8, pp. 381–389, 1997.

[12] J. D. Holt and S. M. Chung, "Efficient mining of association rules in text databases," in *Proceedings of CIKM*, Kansas City, MO, 1999.

[13] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules," in *Proceedings of the 20th International Conference of Very Large Data Bases (VLDB)*, 1994, pp. 487–499.

[14] J. S. Park, M. S. Chen, and P. S. Yu, "Using a hash-based method with transaction trimming for mining association rules," *IEEE Transaction on Knowledge and Data Engineering*, vol. 9, no. 5, pp. 813–825, 1997.

[15] Y. Li, S. M. Chung, and J. D. Holt, "Text document clustering based on frequent word meaning sequences," *Data and Knowledge Engineering*, vol. 64, pp. 381–404, 2008.

[16] R. Agrawal, T. Imielinski, and A. Swami, "Mining association rules between sets of items in large databases," in *Proceedings of the ACM-SIGMOD 1993 International Conference on Management of Data*, May 1993, pp. 207–216.

[17] G. Salton and C. Buckley, "Term weighting approaches in automatic text retrieval," *Information Processing and Management*, vol. 24, no. 5, pp. 513–523, 1960.

[18] K. Sparck Jones, "A statistical interpretation of term specificity and its application in retrieval," *Journal of Documentation*, vol. 28, no. 1, pp. 11–21, 1972.

[19] A. Moffat and J. Zobel, "Compression and fast indexing for multi-gigabit text databases," *Australian Computing Journal*, vol. 26, no. 1, p. 19, 1994.

[20] F. Sebastiani, "Machine learning in automated text categorization," *ACM Computing Surveys*, pp. 1–47, 2002.

[21] M. Meilă, "Comparing clusterings-an information based distance," *Journal of Multivariate Analysis*, vol. 98, pp. 873–895, 2007.

[22] M. Sokolova and G. Lapalme, "A systematic analysis of performance

measures for classification tasks," *Information Processing and Management*, vol. 45, pp. 427–437, 2009.

[23] M. Aghagolzadeh, H. Soltanian-Zadeh, and B. N. Araabi, "Information theoretic hierarchical clustering," *Entropy*, vol. 13, pp. 450–465, 2011.

[24] T. Cao, H. Do, D. Hong, and T. Quan, "Fuzzy named entity-based document clustering," in *Proc. of the 17th IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2008)*, Hong Kong, 2008, pp. 2028–2034.

[25] W. Xu and Y. Gong, "Document clustering by concept factorization," in *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, Sheffield, United Kingdom, 2004, pp. 202–209.

[26] A. Dalli, "Adaptation of the f-measure to cluster based lexicon quality evaluation," in *Proceedings of the EACL 2003 Workshop on Evaluation Initiatives in Natural Language Processing: are evaluation methods, metrics and resources reusable?*, Budapest, Hungary, 2003, pp. 51–56.

[27] K. Kummamuru, R. Lotlikar, S. Roy, K. Singal, and R. Krishnapuram, "A hierarchical monothetic document clustering algorithm for summarization and browsing search results," in *Proceedings of the 13th international conference on World Wide Web*, New York, NY, 2004, pp. 658–665.

Ontology-Based Adaptive Information System Framework

Louis Bhérer, Luc Vouligny
Mohamed Gaha, Billel Redouane

Institut de Recherche d'Hydro-Québec, IREQ
Varennes, Québec, Canada
Email: Bherer.Louis2, Vouligny.Luc,
Gaha.Mohamed, Redouane.Billel @ireq.ca

Christian Desrosiers

École de Technologie Supérieure
Montréal, Québec, Canada
Email: christian.desrosiers@etsmtl.ca

***Abstract*—Software development does not usually end with the final release of the application. The software application have to be maintained throughout its useful lifetime in order to follow the users' needs. Most software applications are built around a rigid data models and modifications that must be performed on such data model impact the application, resulting in additional maintenance costs. The main focus of this work is to design and implement an ontology-based software framework for building information systems that can auto-adapt to evolving semantic data models. This framework has been used in the development of a client-server application as a proof of concept. This application can adapt dynamically to numerous changes that can be made in the model without recompilation of the client-side or the server-side of the application.**

Keywords–Adaptive Information System; Ontology; RDF; RDFS; OWL; Autonomic Computing.

I. INTRODUCTION

Most software applications are built around a rigid data model drawn from relational database (RDB) technologies. On one hand, RDB technologies are mature and performant when storing and accessing information. On the other hand, their data model are hard to change when modifications must be performed. The modification process of the software itself is rather time consuming as most of the changes in the data model will also require adjustments in the corresponding objects' model. The evolution usually requires a transitional program to transfer stored information to the new data model, the recompilation and the republishing of the application. Usually, when the application is on a client-server system in a large organization, all this work must be synchronized between different departments.

Ontologies can also be used to model information. They can be established and refined as new knowledge is acquired and needs evolve. Ontologies repository technologies such as triplestores can be used in software applications that can be built to take into account how the data model evolves. However, current programming languages, such as C, C#, Java, etc., usually require a compilation process in order to adapt to an evolving data model.

Staab et al. [1] "[...] recommend that the ontology engineer gathers changes to the ontology and initiates the switch-over to a new version of the ontology after thoroughly testing possible effects to the application[...]". We deduct that the ease of model modification in the ontology can be constrained by the applications' rigid development framework and resulting programs. Applications able to self-adapt to data models would certainly bring cost reductions on both development and maintenance processes.

The main focus of this work is to design and implement a framework for building an information system that can auto-adapt to evolving semantic data models. This framework has been used in the development of a client-server application as a proof of concept. This application can adapt dynamically to numerous changes that can be made in the model without recompilation of the client-side or the server-side of the application. The goal of this framework is to reduce the costs associated with application development, deployment and maintenance at Hydro-Québec, Québec's provincial utility that generates, transmits and distributes electricity. At IREQ, the research institute of Hydro-Québec, studies on the application of semantic technologies are currently underway as a mean to solve problems related to the increasing number of databases in the organization [2][3]. In addition, self-adapting technologies have already been applied with success [4].

The remainder of this article is organized as follows. The next section is a review of the previous works on system/framework with self-adaptive capacities. Section III presents the framework, its design and main functions, as well as the application built with it. Section IV presents the results of the project. Finally, Section V will cover the potential applications and advantages of this framework and future developments.

II. RELATED WORK

In 2001, IBM has proposed the Autonomic Computing initiative [5] with the objective to develop mechanisms that would allow systems and subsystems to self-adapt to unpredictable changes. Conferences, such as Software Engineering for Adaptive and Self-Managing Systems (SEAMS) [6] or Engineering of Autonomic and Autonomous Systems (EASe) [7], show that system and software self-adaptability is still an important research area, now scattered in a variety of subfields. Amongst them, one could include information system self-adaptability to an evolving data model.

As Dobson & al. stressed out in [8], the Autonomic Computing initiative did not achieve the promises announced in [9]. Many individual advances have brought some of those expected benefits, but there is no integrated solution resulting in an autonomous system. This is a task that some researchers have started working on, such as Bermejo-Alonso in [10] with her attempt to develop an ontology for the engineering of autonomous systems. The self-adaptability mechanisms of our framework could help in the development of self-aware or self-adjusting properties [11] leading to the development of autonomic components.

At Hydro-Québec, advances have been made in self-adapting applications with the Dynamic Information Modelling (MDI) development environment [4]. Some client-server applications built using this system have been put in production and are still in use today. Self-adaptation, even though it is only to the data model, have proven to be beneficial, especially when evolutionary prototyping is used as a development methodology [12]. In MDI, the proposed development library was not a client-server framework and was used as a private and closed semantic modeling system.

In [13], McGinnes and Kapros circumscribe the problem of non-adaptive applications as a conceptual dependence to the data model. They describe this dependence between the data model and the resulting application as an undesirable software coupling. The authors use the terms “Adaptive Information System” (AIS) for information systems that adapt to changes to the underlying data model. They conclude that most applications based on information systems used today are dependant on their domain model. Therefore, such systems must be maintained every time there is a modification on the data model and even the slightest change may result in costly and time consuming adaptations.

McGinnes and Kapros propose six principles to achieve conceptual independence over any data source (see Table 1). Using these principles, they show that it is possible to build an AIS based on an Extensible Markup Language (XML) mapping of a RDB data source [13]. Applying those principles to Resource Description Framework (RDF) based ontologies brings useful insights (see Table 1) on the use of those technologies in an AIS. One can argue that achieving conceptual independence using RDF-based technologies such as RDF, Resource Description Framework Schema (RDFS) and Web Ontology Language (OWL) seems more intuitive than using RDB data sources. RDF-based technologies have in fact many of the required properties inherently built in their design, thus reducing the complexity of achieving conceptual independence.

The proposed AIS framework based on semantic technology is presented in the next section.

III. Proposed AIS framework

Our AIS has been conceptualised and developed as a three-tier client-server framework: a triplestore, a generic server and a web interface.

The triplestore is used to hold the knowledge bases constituted by a conceptual model and its individuals. In the proposed AIS, two knowledge bases are used: one for the domain of expertise and one for the presentation of the information. The triplestore used in this framework is Oracle 12c RDF semantic Graph Triplestore.

The server-tier is coded using a standard Java J2EE technology. It is built as a web service server offering different generic functions with a REST client-server interface. These services are implemented using the library JENA to process the requests written in the SPARQL query language.

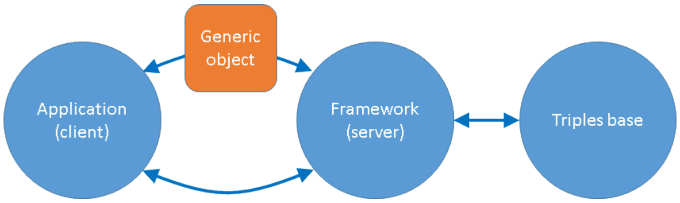


Figure 1. AIS framework.

The user interface is implemented in JavaScript with the Ext.js 4.2.2 library. It uses the REST interface to communicate with the server. Thus, it is independent to the server and could be coded using another technology.

We used the proposed framework to implement a decision support system application to be used at the Hydro-Québec research institute. The purpose of this application is to gather power transformer oil sampling data such as methanol and ethanol concentrations to monitor the health condition of the power transformers and provide suitable maintenance advice to the specialists. The application acts as a dashboard in which the users can add, update or delete entries and do simple searches. It also does automated calculations e.g., to calculate adjusted concentrations of some molecules depending on the temperature of the oil. The engineers will use the application to record their maintenance operations and measurements, to follow and compare the condition of the transformers and to test and refine parameters used in concentration adjustment equations.

The conceptual model of this application comprises six classes that will be used in the subsequent examples: PowerStation, PowerTransformer, Measurement, MaintenanceIntervention, ConversionParameter and PowerStationAndTransformerAssociation. Each of these classes has between two and twelve properties and comprise up to 7000 individuals. This application has been chosen to validate the framework since it requires a variety of functionalities that would be suitable for a wide range of applications.

In order to better understand the proposed AIS, Figure 1 presents a high-level view of the framework. At the initialization phase, the application requests the triplestore via a web service to show the initial presentation consisting of a tree view of the data model. The user uses this tree to select an individual of a class (e.g., a power transformer), represented by a leaf of the tree. When the user clicks on this leaf, the interface sends a request to the server through its web services. Upon reception of the request, the server dynamically gathers an undetermined number of classes, all of which have an association relation with the class of the selected individual. The server then gathers for each of these classes, the list of its properties and the list of its individuals related to the user’s selection. This information is transmitted to the client using a generic Java object and its corresponding JSON representation. This generic object is used to transfer the information to appear on the user’s interface and to request Create, Read, Update and Delete (CRUD) operations to the server.

A. *Application triplestore setting*

Most of the application data are stored in an enterprise RDB. A semantic meta-model (T-Box) has been designed to

TABLE I. CONCEPTUAL INDEPENDENCE PRINCIPLES AND APPLICATIONS.

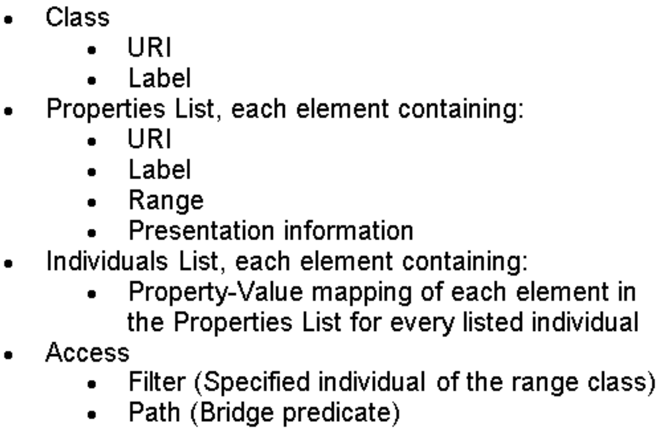
CONCEPTUAL INDEPENDENCE PRINCIPLES [13]	**APPLICATIONS OF THE CONCEPTUAL INDEPENDENCE PRINCIPLES WITH RDF-BASED TECHNOLOGIES**
1. Reusable functionality (structurally- appropriate behaviour): The AIS can support any conceptual model. Domain-dependent code and structures are avoided. Useful generic functionality is invoked at run time for each entity type. [13]	This principle applies similarly using a triplestore data source. Generics SPARQL requests will be obtained by exclusively hard-coding resources from the RDF, RDFS or OWL semantics, leaving the others resources soft-coded. The data model can be inspected at run time using generic SPARQL requests.
2. Known categories of data (semantically- appropriate behaviour): Each entity type is associated with one or more predefined generic categories. Category-specific functionality is invoked at run time for each entity type. [13]	All ontologies using RDFS or OWL languages contain *ipso facto* the same conceptual basis. The definition of those meta-entities are the semantics of RDF, RDFS and OWL. Employing those meta-entities as the most generic entities of the AIS allows the use of any RDF-based ontology. McGinnes and Kapros use archetypal categories taken from the field of psychology to classify entities according to the behaviours the AIS should adopt in their presence. This interesting idea will be considered later on in the development of this AIS, but is not currently essential.
3.Adaptive data management (schema evolution): The AIS can store and reconcile data with multiple definitions for each entity type (i.e.,multiple conceptual models), allowing the end user to make sense of the data. [13]	First, RDF technology uses what McGinnes and Kapros call soft-schemas: data models stored as data. Secondly, RDF technology allows individuals with different valued properties to coexist in the same class. Moreover, individuals can belong to more than one class. Axioms like *OWL:sameAs* or *OWL:equivalentClass* allow to reconcile data from distinctly described entities. Two previously distinct classes declared as equivalent will have, by inference, the same set of properties and then two individuals of this new class may have only different valued properties. Thus, this mechanism allows for reconciliation of data from different conceptual models. As the model evolves, data using different conceptual models remain available and is instantly accessible without any refactoring of the AIS.
4. Schema enforcement (domain and referential integrity) : Each item of stored data conforms to a particular entity type definition, which was enforced at the time of data entry (or last edit). [13]	In technologies such as OWL, domain integrity and referential integrity can be validated with reasoners. As for data types, literal data are usually associated with basic types upon entry in a semantic store.
5. Entity identification (entity integrity): The stored data relating to each entity are uniquely identified in a way which is invariant with respect to schema change. [13]	In the RDF technology, entity identification is provided by the URI mechanism, and is already invariant with respect to schema change.
6. Labelling (data management): The stored data relating to each entity are labelled such that the applicable conceptual models can be determined. [13]	Using the RDF technology, this principle would translate as: each individual needs to belong to a class. Then, is does not matter how much the class has change over time, because all of its individuals can have any number of valued or non-valued properties. However, human-readable labels are necessary to present the information to the users and it is mandatory to affect each entity with such labels.

model the required classes (PowerTransformer, PowerStation, etc.). Then, by using the D2RQ library, the data from the RDB have been converted into a RDF individuals graph (A-Box). The T-Box has been designed using RDFS semantics. It solely contains association relationships, and essentially describes the classes and the properties with their domain and range. Each class, property and individual have been labeled in order to be shown on the visual interface.

B. Dynamic visualization of the semantic data

Here are the main design elements for the dynamic visualization of the information.

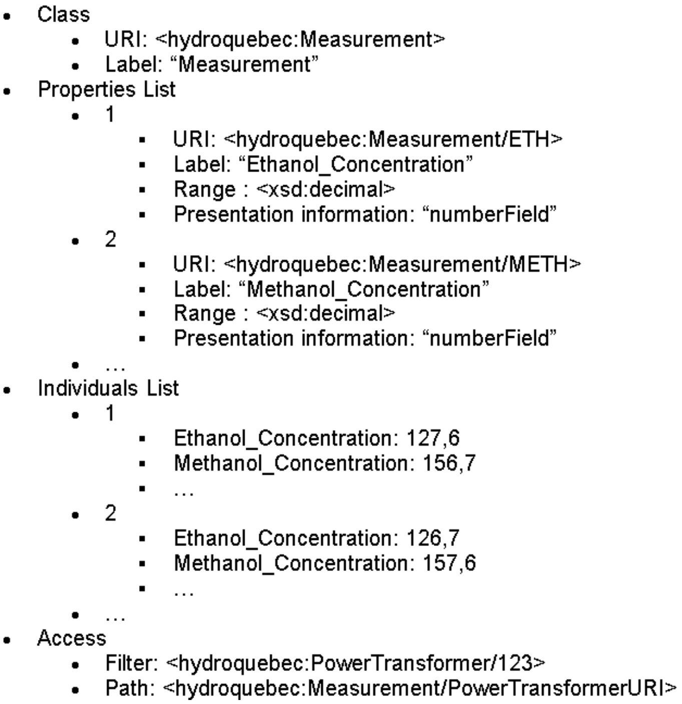
1) The generic object: A generic Java class (meta-class) was designed in order to allow dynamic recuperation of information from the semantic store. The resulting object is used to transfer information from the semantic store to the user's interface. A given object's instance is built from generic SPARQL requests using RDF and RDFS semantics. The object has fixed attributes used to hold information on the RDFS class, its properties and individuals. The generic object can also hold the path and filters used to select the individuals or the class itself. See Figure 2 for the definition of the object.



- Class
    - URI
    - Label
- Properties List, each element containing:
    - URI
    - Label
    - Range
    - Presentation information
- Individuals List, each element containing:
    - Property-Value mapping of each element in the Properties List for every listed individual
- Access
    - Filter (Specified individual of the range class)
    - Path (Bridge predicate)

Figure 2. Definition of the Java generic object.

Note here that each individual contains a property-value mapping for each property of the Properties List, and its corresponding value, if any. The access elements contain the



- Class
    - URI: <hydroquebec:Measurement>
    - Label: “Measurement”
- Properties List
    - 1
        - URI: <hydroquebec:Measurement/ETH>
        - Label: “Ethanol_Concentration”
        - Range : <xsd:decimal>
        - Presentation information: “numberField”
    - 2
        - URI: <hydroquebec:Measurement/METH>
        - Label: “Methanol_Concentration”
        - Range : <xsd:decimal>
        - Presentation information: “numberField”
    - …
- Individuals List
    - 1
        - Ethanol_Concentration: 127,6
        - Methanol_Concentration: 156,7
        - …
    - 2
        - Ethanol_Concentration: 126,7
        - Methanol_Concentration: 157,6
        - …
    - …
- Access
    - Filter: <hydroquebec:PowerTransformer/123>
    - Path: <hydroquebec:Measurement/PowerTransformerURI>

Figure 3. Example of a Java generic object.

path in the graph to get to the class (i.e., the property linking the individuals of the two classes) and the filter (i.e., an individual of the range class) used to select the individuals of the domain class. The term “Bridge predicate” will be used to refer to the property linking the domain class and the range class (i.e., the path) (See Figure 4).

In the developed application, selecting a power transformer in the tree will result in a request to find individuals linked to it from all classes having a property whose range is the Transformer class, i.e., individuals from the domain classes of the Transformer class. For each class found, a generic object will be created.

In order to better understand how generic objects are created, please refer to the example given in Figure 3. In this example, the user has selected the power transformer numbered 123. The framework then requested the model and found three classes having an associative relation with the PowerTransformer class: Measurement, MaintenanceIntervention and PowerStationAndTransformerAssociation. Those three classes are going to be fetched but this example presents only the Measurement class case. Its URI and label have been first retrieved, followed by the list of its properties and the list of its individuals. This second list contains a mapping for each individual, between every properties of the property list and its value for this individual, if any.

In the example in Figure 3, the filter is the specified individual of the range class, i.e., the power transformer numbered 123. It is considered a filter because it reduces the number of individuals retrieved. Here, the path is simply the Bridge predicate between the range and the domain classes. Further development should lead to the creation of more filter and path options, as well as sequences and aggregations of these options.

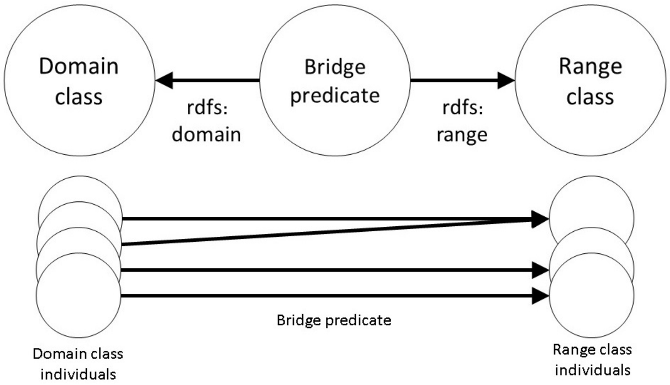


Figure 4. Graph representation of the range and domain classes in an associative relationship.

In our AIS, every time a power transformer is selected in the tree, the model is inspected dynamically to find all the domain classes of the PowerTransformer class and all their individuals linked to the selected power transformer. Hence, if a new domain class is added, the application will automatically present it to the users.

Due to the genericity of the functions, the changes made on the data model are immediately available to all the AIS users. From then on, every request will get individuals and classes from the new model, without any need to recompile the client nor the server. This is due to the fact that being written to adapt to any model, a request can then be used in run time to inspect the actual version of the model.

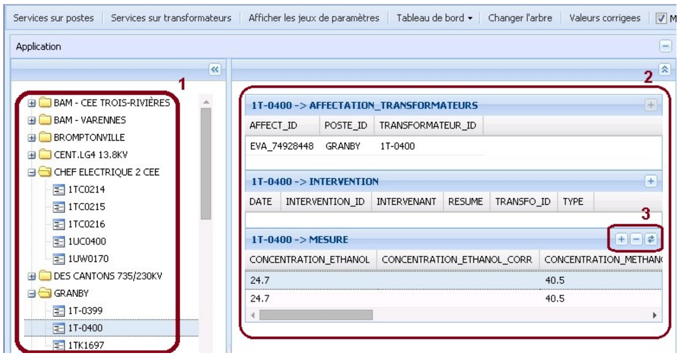
2) Visual representation: The application uses a tree to show the user a specific portion of the semantic graph (see Figure 5). In our case, the tree first shows all the power stations as folders that can be expanded to see the power transformers they contain.

When the user selects a node (e.g., the power transformer 123), the client user interface sends a request to the AIS server, using a generic process, to dynamically gather the domain classes (e.g., the Measurement class) in relation with the range class (e.g., the PowerTransformer class). For each of these classes, the properties will first be found, and then, all the individuals of the domain classes linked with the user selected individual will be retrieved. As a result, a list of generic Java objects will be generated where each object corresponds to a domain class.

These Java objects are then automatically converted to JSON, using the Jackson library [14] and sent to the user interface. The user interface will produce a bidimensional matrix for every class in the list (see Figure 5). These matrices show the information to the user using human-readable labels. The user can then request for CRUD operations on individuals represented in the matrices (see Figure 5).

The CRUD operations are programmed to retrieve the Java generic object and delete all the unselected individuals, so to keep only the selected individual. This individual is then modified according to the user’s needs and the resulting Java object is returned to the server.

In the current state of the framework implementation, if changes are made in the T-Box, either by modifying the



1T-0400 -> AFFECTATION_TRANSFORMATEURS

AFFECT_ID	POSTE_ID	TRANSFORMATEUR_ID
EVA_74928448	GRANBY	1T-0400

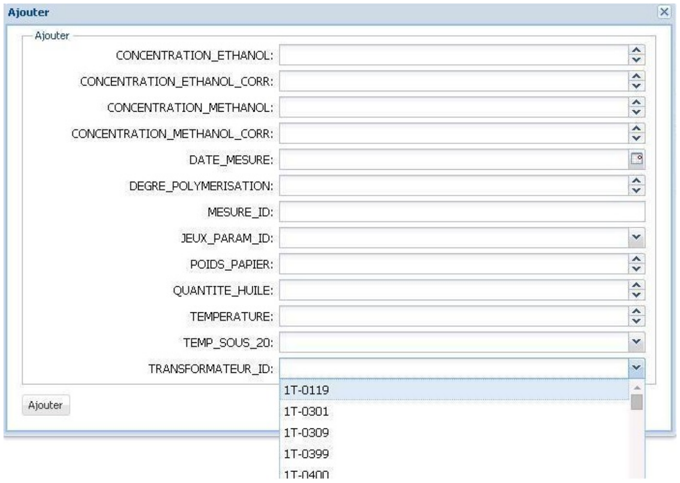
1T-0400 -> INTERVENTION

DATE	INTERVENTION_ID	INTERVENANT	RESUME	TRANSFO_ID	TYPE

1T-0400 -> MESURE

CONCENTRATION_ETHANOL	CONCENTRATION_ETHANOL_CORR	CONCENTRATION_METHAN
24.7		40.5
24.7		40.5

Figure 5. The tree view (1), matrices (2) and the buttons to request for CRUD operations (3).



CONCENTRATION_ETHANOL:
CONCENTRATION_ETHANOL_CORR:
CONCENTRATION_METHANOL:
CONCENTRATION_METHANOL_CORR:
DATE_MESURE:
DEGRE_POLYMERISATION:
MESURE_ID:
JEUX_PARAM_ID:
POIDS_PAPIER:
QUANTITE_HUILE:
TEMPERATURE:
TEMP_SOUS_20:
TRANSFORMATEUR_ID:

1T-0119
1T-0301
1T-0309
1T-0399
1T-0400

Figure 6. Individual CRUD form example.

properties of some domain classes or by adding a new domain class related to the class of the tree leaves, the users will instantly begin to navigate in the new conceptual model. Other changes are not yet possible.

The main presentation tree does not grant access to every class in the semantic graph. Therefore, the user interface has been given other access points from which the user can request directly for those previously inaccessible classes. The system uses a similar generic function to request this information except that it retrieves the class itself and all its individuals instead of using the previously presented domain classes mechanism. The same generic Java object is used, but does not have any access information. As the generic Java object is used, the same CRUD operations can still be performed on individuals.

3) The CRUD services: For the time being, the framework allows CRUD services on the individuals only, not on the classes and properties. Other means are used to edit the conceptual model. Further work will be made to allow modelization of the T-Box from the user interface. The CRUD services on the A-Box are done on the client-side using forms showing the properties of the class and their value for the selected individuals, if any (see Figure 6). These forms are created from the properties listed in the generic Java object.

In order to help the user and validate the input, a presentation knowledge base comprising the different presentation options has been established. This information is associated with every property of the domain knowledge base and is passed on by the Java generic object. It indicates how to build every entry fields of the forms. Those forms are constructed dynamically, adapting the user's interaction options on the values of properties according to the presentation knowledge base information.

In further developments, mechanisms will be designed to automatically link the domain ontology properties to the presentation ontology individuals. Some ontologies contain semantics, such as Enumeration, Sequence, or Bag, that can be used to predict the correct entry field's type for a certain property. Enumeration, for example, can be represented as a list of individuals from which the user will have to choose. In general, the range of a property is a good indicator of the required entry field's type. Finally, functions will be implemented to allow the user to change the type of the entry field in run time.

In the current state of the framework, four types of entry fields are implemented: numerical fields, text fields, list fields and date fields. Upon expansion, the list field requests for a service that finds all the existing values associated to this property. For the fields used to update literals, the range type of the property is used for validation. Cardinalities are present in the presentation knowledge base so the forms can indicates to the user the required fields, if any.

4) The graphics: Graphic classes and related properties have been added to the presentation knowledge base to represent graphic views such as histograms or clouds of points. Graphic properties are used to specify the association between the graphic elements such as the x-axis data, y-axis data, labelings, etc. The axis are linked to domain ontology properties. When these domain ontology properties are present in generic objects, the user interface could detect them and create a list of available graphics.

IV. Results

The framework has been used to create a client-server decision-support application. Thanks to the generic services of the AIS framework, one can modify the classes and properties in the conceptual model directly in the triplestore without affecting the application. The user interface will adjust its presentation automatically according to the latest update of the conceptual model, since the request interrogates the semantic graph dynamically. The proposed framework allows for all CRUD operations to be performed on individuals. Moreover, the framework will query the conceptual model in the semantic graph for each request, which is different to a standard application where the conceptual model is taken into consideration only at compilation time. The resulting application is ready to be put in production. Once in production, because it will be able to automatically adapt to conceptual model changes, it should easily evolve as the framework is extended.

The main limitation of this framework is how it explores the model at each request. While for now it only retrieved individuals from classes that are one associative relation away from a desired individual, further work is needed to find ways of expanding this exploration. The implementation of this mechanism will be crucial for the framework to be effective in large scale ontologies.

Tests still need to be run to determine performance differences between such an application and a non-dynamic one, and to observe the scaling potential. The resulting application from the proposed framework is not expected to be as performant as a similar application developed from a more conventional framework, but the difference in performance has yet to be established. Then, it will be possible to evaluate how much the cost reductions incurred during the development and the maintenance processes of AIS outweighs their performance aspect on the long run.

While implementing this proof of concept, we learned that many of the needed properties to achieve conceptual independence are inherent to the RDF technology. Exclusively hard-coding resources from the RDF, RDFS and OWL semantics in all the SPARQL requests and leaving all other resources soft-coded are necessary conditions to obtain this conceptual independence. Because the semantics of these three languages (RDF, RDFS and OWL) are shared across all RDF-based ontologies, they form a common conceptual basis to all the domains they can represent. Limiting the conceptual dependences to their semantics, the applications built can use any such ontology, regardless of its knowledge domain.

V. Conclusion and future work

As hypothesised, an AIS based on a triplestore seems easier to implement than an AIS using XML to dynamize functions on a RDB. Many artifices have to be considered when building an AIS from a RDB which are not required with semantic technologies, as described in Table 1. The use of a library to map the RDB into a triplestore appears judicious to easily and quickly gather the conceptual independence needed in an AIS.

With the use of a RDF representation to store the information, generic SPARQL requests that can search any semantic graph for both conceptual knowledge and individual information are easily devised. This leads the AIS to be able to adapt to the evolution of the conceptual model and to be used for different domains of application. The framework could also be used with evolutionary prototyping application development. At Hydro-Québec, other large scale client-server applications have already been successfully developed using evolutionary prototyping, highlighting the benefits of such technologies compared to standard development processes [12].

The construction of an application editor able to use the framework for developing new auto-adaptive applications seems to be the next logical step. Using the framework to build new applications will further test the approach and allow to complete the presentation knowledge base. In doing so, new functions will be developed leading eventually to a complete AIS. Ideally, the AIS should be able to take advantage of all the RDF, RDFS and OWL semantics.

The current application uses only RDFS semantics; adding OWL capabilities will allow for the use of inference reasoners. In the current release, only the individuals of the semantic domain can be edited by the user through forms. Editing possibilities on the meta-model will be authorized in the next iterations.

The framework and the application are the proof that an AIS can work easily and efficiently by capitalizing on the RDF technology and its inherent properties. Such systems can be useful in fast-evolving knowledge domains. They inscribe themselves well in the AGILE development philosophy, allowing the model data to evolve freely at each iteration. Those considerations allow to think that AIS and self-adapting applications could bring substantial cost reductions in application development and maintenance in the coming years.

Acknowledgment

The authors would like to thank Jérôme Côté for his insightful help.

References

[1] S. Staab, R. Studer, H.-P. Schnurr, and Y. Sure, “Knowledge processes and ontologies,” IEEE Intelligent Systems, vol. 16, no. 1, Jan. 2001, pp. 26–34.

[2] A. Zinflou, M. Gaha, A. Bouffard, L. Vouligny, C. Langheit, and M. Viau, “Application of an ontology-based and rule-based model in electric power utilities,” in 2013 IEEE Seventh International Conference on Semantic Computing, Irvine, CA, USA, September 16-18, 2013, 2013, pp. 405–411.

[3] M. Gaha, A. Zinflou, C. Langheit, A. Bouffard, M. Viau, and L. Vouligny, “An ontology-based reasoning approach for electric power utilities,” in Web Reasoning and Rule Systems - 7th International Conference, RR 2013, Mannheim, Germany, July 27-29, 2013. Proceedings, 2013, pp. 95–108.

[4] L. Vouligny and J.-M. Robert, “Online help system design based on the situated action theory,” in Proceedings of the 2005 Latin American Conference on Human-computer Interaction, ser. CLIHC ’05. New York, NY, USA: ACM, 2005, pp. 64–75.

[5] P. Horn, “Autonomic Computing: IBM’s Perspective on the State of Information Technology,” Tech. Rep., 2001.

[6] T. Vogel. Software engineering for self-adaptive systems. [retreived: 06, 2015]. [Online]. Available: https://www.hpi.uni-potsdam.de/giese/public/selfadapt/ (2015)

[7] I. T. C. on Software Engineering. Ieee ease 2014. [retreived: 06, 2015]. [Online]. Available: http://tab.computer.org/aas/ease/2014/index.html (2014)

[8] S. Dobson, R. Sterritt, P. Nixon, and M. Hinchey, “Fulfilling the vision of autonomic computing,” Computer, vol. 43, no. 1, 2010, pp. 35–41.

[9] J. O. Kephart and D. M. Chess, “The vision of autonomic computing,” Computer, vol. 36, no. 1, Jan. 2003, pp. 41–50.

[10] J. Bermejo-Alonso, R. Sanz, M. Rodríguez, and C. Hernández, “Ontology-based engineering of autonomous systems,” in Proceedings of the 2010 Sixth International Conference on Autonomic and Autonomous Systems, ser. ICAS ’10. Washington, DC, USA: IEEE Computer Society, 2010, pp. 47–51.

[11] R. Sterritt and M. Hinchey, “Spaace iv: Self-properties for an autonomous computing environment; part iv a newish hope,” in Engineering of Autonomic and Autonomous Systems (EASe), 2010 Seventh IEEE International Conference and Workshops on, March 2010, pp. 119–125.

[12] L. Vouligny, C. Hudon, and D. N. Nguyen, “Design of mida, a web-based diagnostic application for hydroelectric generators.” in COMPSAC (2), S. I. Ahamed, E. Bertino, C. K. Chang, V. Getov, L. L. 0001, H. Ming, and R. Subramanyan, Eds. IEEE Computer Society, 2009, pp. 166–171.

[13] S. McGinnes and E. Kapros, “Conceptual independence: A design principle for the construction of adaptive information systems,” Inf. Syst., vol. 47, 2015, pp. 33–50.

[14] Cowtowncoder. Jackson. [retreived: 06, 2015]. [Online]. Available: http://jackson.codehaus.org/ (2015)

Application of the Tensor-Based Recommendation Engine to Semantic Service Matchmaking

Andrzej Szwabe, Michal Ciesielczyk, Pawel Misiorek, Michal Blinkiewicz
Institute of Control and Information Engineering
Poznan University of Technology
Poznan, Poland
Email: {firstname.lastname}@put.poznan.pl

Abstract—**The paper presents a novel approach to semantic Web service matchmaking, which involves a use of multilinear data representation and processing. The proposed solution involves the use of a novel tensor data filtering method based on a set of covariance matrices derived from a hierarchical tensor structure. We provide results of experimental evaluation of the proposed solution conducted with the use of the Semantic Service Selection (S3) contest dataset. The evaluation has been done using the standard Information Retrieval methodology that assumes the methodologically correct partitioning of the dataset on mutually exclusive subsets: the training set and the testing set. The experimental evaluation results presented in the paper indicate superiority of the covariance-based tensor filtering method over other state-of-the-art tensor processing methods in terms of the matchmaking quality measured using mean average precision and Area Under the ROC curve (AUROC) measures.**

Keywords–Semantic Service Selection; tensor-based multirelational data modeling.

I. Introduction

Semantic Web Service (SWS) technologies are aimed at discovering and matching Web services using their functional and nonfunctional semantic representations. Due to practical importance of SWS solutions, in recent years the attention of scientific community has been paid on the development of methods which enable to embed the semantics into the discovery, matchmaking, and mediation processes [1]–[5]. In this paper, we investigate matchmaking of Web services described using the Semantic Annotations for Web Service Description Language (SAWSDL) standard which is based on enriching Web Services Description Language (WSDL) documents with semantic annotations in the form of references to ontologies. The presented research uses the widely-referenced [1]–[5] data collection – SAWSDL-TC [6] – developed for the purposes of the Semantic Service Selection (S3) contest [7].

A. Research Motivation

The most important part of each service matchmaker is its matching algorithm, which determines the means of the relevance measurement applied to a pair of Web services. The S3 contest editions have shown that the best results are achieved by the adaptive hybrid matchmakers, such as [2][4][7], which use a part of the test collection for optimization purposes. Hybrid matchmaker systems make use of several types of similarity algorithms for Web service descriptions (including logical and lexical similarity algorithms) and subsequently compute the overall similarity based on the importance weights of partial results optimized according to the cross-validation approach. One of the main goals of this paper is to investigate input data integration as an alternative to the widely-proposed integration at the level of final results provided by several hybridized subsystems (systems operating in accordance to different approaches to the Web service matchmaking task). In order to enable such a solution, we have used the tensor-based data representation which is suitable to integrate heterogeneous data. This approach results in no need for a further aggregation of all fragmentarily computed similarities.

The tensor-based data representation has been already recognized as a suitable tool for storing the multidimesional data in a compact way [8][9] that may be effectively used in many application areas related to machine learning [8][10]–[12]. We recognized the application of tensor data representation to the semantic service selection task as a promising approach, especially because the S3 task requires the need of retrieving the information from heterogeneous data sources. As a consequence, the experimental evaluation presented in this paper is focused on comparison of the proposed tensor-based data processing method with state-of-the-art methods.

B. Contribution

The main aim of this paper is to present the novel approach to the S3 task based on two-step processing consisting of the heterogeneous data integration step and the processing of the integrated data using the tensor model.

The important part of the paper contribution is related to evaluation methodology issues. In contrast to the methodology used in the S3 contest, the described experimental results are based on partitioning the dataset into a training and a testing set in such a way that the data used for testing the performance are not previously used to learn or tune the model. Alas, such an approach is not used in the S3 contest. According to the contest rules the participating matchmakers provide the recommendation results for the whole set of service requests described in the dataset. The S3 evaluation tool does not provide any additional set of reference matchings which may be used as a training set. For this reason, in this paper, we propose to consider the semantic service matchmaking task as a case of semi-supervised learning in which unlabeled data are used in conjunction with a small amount of labeled data [13]. Due to above-explained evaluation methodology differences, the provided performance evaluation does not contain a direct comparison of the proposed method's operation to the S3 contest results.

The rest of paper contribution includes: (i) the data integration framework, which enables the transformation of

SAWSDL and Web Ontology Language (OWL) documents into the set of n-tuples (or Resource Description Framework (RDF) statements) and then the aggregation of these data using the tensor-based data representation, (ii) the first tensor-based SWS matchmaking engine involving the use of the tensor-based data processing system based on a filtering method which applies the covariance data derived from a hierarchical structure of tensor flattenings, and (iii) the comprehensive comparison of several matchmaking algorithms including those based on the state-of-the-art tensor processing techniques (i.e., N-way Random Indexing [14] and Higher Order Singular Value Decomposition (SVD) [15]).

The paper is structured as follows. Section II provides a discussion on related work, which contains a brief presentation of state-of-the-art SWS matchmaking solutions, their limitations, as well as tensor-based data processing algorithms. The proposal of tensor-based semantic service recommendation system including the semantic data integration framework and tensor-based recommendation engine is given in Section III. Next, the tensor-based data representation and filtering method is provided in Section IV. Section V contains the description of the evaluation methodology and of the algorithms used for comparison. Section VI provides the experimental results and their analysis. Finally, the paper is concluded in Section VII.

II. Related Work

In this section, the advantages and limitations of leading state-of-the-art matchmaking solutions – in context of semantically annotated Web services – are discussed. Additionally, the tensor-based data processing assumptions and state-of-the-art algorithms are introduced.

A. *State-of-the-art Web Service Matchmakers*

The state-of-the-art Web service matchmakers make use of different knowledge representation formalisms and are usually referred to as *hybrid* solutions. They are known to achieve better results then logic-based only or non-logic-based only approaches in terms of the precision and recall measure [3]. Authors of the articles describing their hybrid matchmakers drew an attention to the problem of an aggregation of different matching results. Primarily weights of logical, text similarity and structural similarity matchings are set manually based on tests and analysis. It follows that any change of ontologies or services forces re-testing and re-analysis in order to select new appropriate weights [1]–[3][5]. Thus, a new *adaptive* approach has been proposed, which resolves this issue by letting the matchmaker learn what is the best adoption of weights. The main benefit of an adaptive approach is that a matchmaker settings are not dependent on a particular data collection. In order to adapt the system for a new dataset, it is sufficient to recalculate the weights in the off-line relearning process.

It should be stressed, however, that the evaluation procedure of the S3 contest [7] does not provide a separate set of reference matchings that may be used as a training set. Nevertheless, the contest participants' solutions based on the adaptive hybrid recommendations [1]–[5] use matchings from the test set in order to find the optimal set of weights in the procedure based on the k-fold cross-validation technique. In particular, the mentioned systems apply different machine learning techniques when determining the weights for particular strategies of the hybrid solution, including *logistic regression*, *simple linear regression* and *support vector regression* (SAWSDL-iMatcher [1]), *ordinary least squares* estimator (LOG4SWS.KOM [5]), and *support vector machine* (SAWSDL-MX2 [3]). Such an approach violates principles of recommendation systems evaluation [16]–[18] because it allows to learn from the information which is also the subject of testing. Another adaptive hybrid matchmaking system – URBE [2] – also assumes the system configuration phase in order to optimize the hybrid algorithm parameters, but, in this case, the authors have also conducted an evaluation assuming the partition of the set of reference matchings into mutually exclusive training and testing sets. However, this approach was applied only for the case of tests using dataset OWLS-TC of the S3 contest [7] track devoted to the OWL-S standard.

In contrast to the S3 contest evaluation methodology, the performance evaluation described in this paper assumes the explicit specification of the information about referential mappings used for training purposes and does not use these mappings in the testing phase. Such an approach is constitutional for the matchmaking system proposed in Section III, which assumes the application of input data integration instead of the integration done at the level of final results of using different strategies.

B. *Tensor-Based Data Representation and Processing*

The main goal of the paper is to propose a new approach to the S3 task involving the processing of the integrated data using the tensor model. Higher-order tensors are already used in many areas of research as a model for data representation [8]–[10][19], including the signal and image processing, higher-order statistic or scientific computing. At the same time, it may be observed that the tensor data model has been widely used for various information retrieval application, mainly by means of 3-rd order tensors used for multirelational data analysis, e.g., as presented in [11][12] for the case of processing RDF statements. It is well known that many problems in machine learning involve the processing of information with multiple aspects and high dimensionality. For such problems, the tensors are regarded as the most natural and compact representation for multidimensional data, however, they have to be accompanied by some low-rank approximation approach [8][9], e.g., based on tensor decomposition [19]–[21]. The semantic service matchmaking task, as an application scenario which involves the processing of multirelational and multidimensional data from heterogeneous sources (OWLs, SAWSDLs, textual data), seems to be another research area for which tensor data representation and processing methods may be efficiently applied.

It has to be admitted that the exponential grow of number of tensor elements observed with the increase of the number of tensor dimensions (usually referred to as tensor modes [10]) seems to be the main reason, why a significant part of the experimental research on tensor models is limited to the case of 3-rd order multidimensional structures [10][11][14][15][21].

In the context of multilinear data processing, the most widely known form of per-mode tensor filtering is the projection of tensor 'fibres' laying along the given tensor mode into a subspace spanned by the modes' principal components – the projection being the main tool of filtering based on Higher-order Singular Value Decomposition (HOSVD) [15] and Multilinear Principal Component Analysis [10]. However, the state-of-the-art solutions do not investigate the theoretical basis for optimality of the multilinear dimensionality reduction heuristics, as far as practical prediction quality, rather than some 'technical' criteria such as Frobenius norm preservation, is concerned [15][22]. Moreover, in order to be effective, the multilinear data modeling has to follow mathematical constrains derived from the area of statistics, algebra, or probability theory, [10][20][21][23]. The issues concerning the proper data centering necessary to provide the efficient multilinear principal component analysis are one of the examples of such a constrains [23]. In this paper, we present and experimentally evaluate the tensor filtering method involving the use of covariance matrices derived from different tensor structures, which addresses the above-mentioned issues.

III. Tensor-Based Semantic Service Recommendation System

The purpose of the proposed system is to provide accurate Web service recommendations – referred to as *offers* – for a given Web service description – referred to as *query*. Both *offers* and *queries* are assumed to be represented in the form of SAWSDL documents with references to objects described in OWL ontologies. The general architecture of the proposed system includes two main components:

1) The converter selecting essential information from SAWSDL descriptions, OWL documents, and reference matchings used to train the model and subsequently transforming them into a common representation.
2) The recommendation engine, described in Section IV, aimed at generating the high quality recommendations.

Thus, the quality of tuples, which are chosen as internal data representation of the system, generated by the converter is crucial for final recommendations accuracy. It should be also taken into account that the information from heterogeneous data is aggregated at the beginning of processing rather than, as in the case of the state-of-the-art solutions (discussed in Section II), as the last step.

As shown in Figure 1 an SAWSDL description is a WSDL document enhanced with semantic annotations linking various parts of the Web service description to corresponding OWL ontology classes.

The introduced framework processes every SAWSDL document along with other linked XML or OWL files, and transforms the acquired information into a common representation. Specifically, for each SAWSDL document the *portType* element, constituting the interface of the Web service, is parsed. The *portType* consists of a set of *operations* having exactly specified *input* and *output*, which in turn reference corresponding *messages*. Every *message* has a list of elements associated with a specific *types* expressed in the XML Schema (XSD) language, which in turn may reference to corresponding OWL ontology classes. As shown in Figure 1, the OWL classes are subsequently linked with the related instances from the ontology (super- and sub-classes). All of the human-readable names – appearing in SAWSDL documents as values of the schema fields pointed out in Figure 1 – are being tokenized and included into the common data representation.

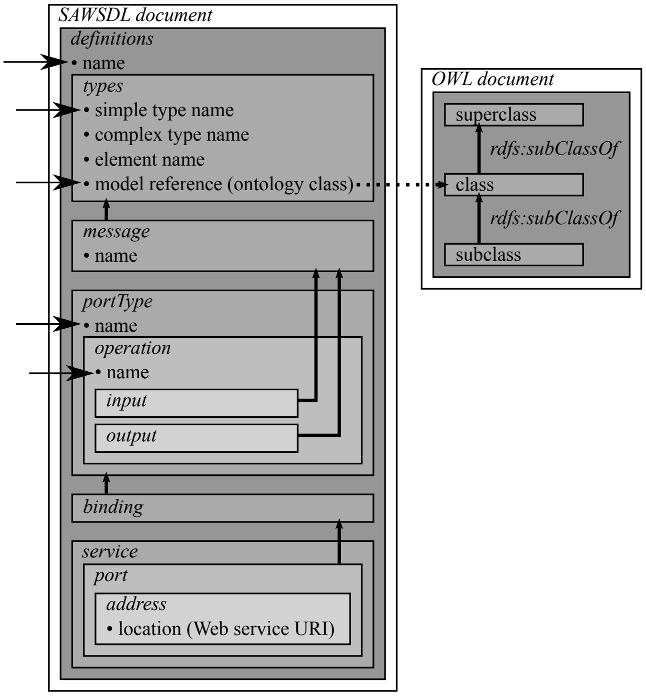


Figure 1. Data retrieved from SAWSDL and OWL documents in order to build the augmented representations of the matchings.

Finally, every matching used to train the model is augmented with the acquired semantic descriptions. Specifically, each matching consists of an information whether two corresponding Web services, depicted by an Uniform Resource Identifier (URI), are relevant or not. Subsequently, the augmented descriptions of every Web service used by our system have been built from the following attributes:

- tokenized Web service name or URI,
- tokenized *portType* name,
- tokenized *operation* names,
- XSD simple data type names,
- URIs pointing to corresponding OWL ontology classes.

The tensor-based recommendation engine operates on data provided in the form of n-tuples or its RDF equivalent.

A. Tuple-Based Representation of the Input Data

For the purposes of representing the tuple-based SAWSDL Web services descriptions we use a 3-mode tensor. The first tensor mode is used to model the information on service relevance (i.e., relevant/nonrelevant indicator). The second and the third modes represent the augmented semantic descriptions of a *request* and an *offer*, respectively, related to the matching specified by the first mode. Such a description is represented by a vector built as an L^1-normalized sum of index vectors (uniquely assigned random vectors) of related terms (i.e., the terms from the specific augmented Web service description).

An index vector is a uniquely assigned random vector for the object that it represents, as defined in [24].

Listing 1 shows a reduced (for the purpose of presentation) 3-tuple example. Line 1 indicates that the two considered Web services are relevant. Line 2 and 3 represent the terms from tokenized Web service request and offer descriptions, respectively.

```
1 ('relevant',
2  ['shopping mall', 'camera', 'price', 'Mid−level−
       ontology.owl#ShoppingMall', extendedCamera.
       owl#Camera'],
3  ['shopping mall', 'purchasable', 'item', 'Mid−
       level−ontology.owl#ShoppingMall', '
       extendedCamera.owl#PurchaseableItem'])
```

Listing 1. The 3-tuple example.

As specified formally in the next section, the tensor that represents the whole input data set is simply the sum of the individual rank-one tensors, while each of these tensors represents a single tuple form the data set. Thus, the elementary procedure of the input tensor construction process is the computation of the rank-one tensor representing a given tuple. Such a rank-one tensor is obtained as an outer product of the vectors representing all the consecutive elements of the given tuple.

Let us refer to the example presented in Listing 1 once again. As the first element of the tuple corresponds to the single token of 'relevant', the first argument of the tensor product is simply the vector representing this element in the vector space corresponding to all the first values of the tuples. In contrast to the simplest case, when the given tuple element is a set of elements, rather than a single element, the vector representing such a set of elements is built as a superposition of the vectors representing the elements. For example, when the second element of the tuple is a set of the following five elements the second argument of the tensor product is the normalized sum of the below-enlisted five vectors:

```
('shopping mall', 'purchasable', 'item', 'Mid−
     level−ontology.owl#ShoppingMall', '
     extendedCamera.owl#PurchaseableItem')
```

Listing 2. Elements of the second argument in the example 3-tuple.

It is worth noticing that the proposed approach does not require the model reference instances of any web service to be linked to classes from the same ontology. Any element of any given set (constituting the given tuple element) is treated simply as a regular token, i.e., exactly the same way as an ‘ordinary’ token (representing a word found in some text) is treated. Thus, there is no obstacles limiting the use of different ontologies in order to describe the inputs and the outputs of the same web service, not to mention the inputs and the outputs of different web services.

B. RDF-Based Representation of the Input Data

The RDF-based representation of the SAWSDL and OWL documents is obtained in a similar manner as the tuple-based representation except that the result is saved into RDF statements rather than tuples. First of all, the information whether two Web services are relevant is formed by the triple which subject is a *request* URI, object is an *offer* URI and predicate is one of the *isRelevant* or *isNonRelevant* properties. The augmented semantic description, derived from SAWSDL and OWL documents, is stored as triples with the subject being *request* or *offer* URI, predicate indicating the corresponding attribute property, and object containing associated information – such as a term or type (both in form of a literal) or an OWL class reference (in form of an URI). Thus, it should be also taken into account that, typically, one tuple is represented by more than one RDF statement. As an example, Listing 3 shows the same tuple as in Listing 1 in an RDF format (Turtle notation).

```
<shoppingmall_cameraprice.wsdl>
   :isRelevant <shoppingmall_purchaseableitemprice
      .wsdl> ;
   :terms "shopping mall", "camera", "price" ;
   :input_owl_uri_ref <Mid−level−ontology.owl#
      ShoppingMall> ;
   :output_owl_uri_ref <extendedCamera.owl#Camera>
      .
<shoppingmall_purchasableitemprice.wsdl>
   :terms "shopping mall", "purchasable", "item";
   :input_owl_uri_ref <Mid−level−ontology.owl#
      ShoppingMall> ;
   :output_owl_uri_ref <extendedCamera.owl#
      PurchaseableItem> .
```

Listing 3. The example tuple in an RDF format (Turtle notation, with URI prefixes removed).

Note that the statement form of a subject-predicate-object expression is also known as a triple – or equivalently as a 3-tuple – in the RDF terminology. Therefore, in this paper, as not to introduce confusion, RDF data is referred to only as statements, while the tuple-based representation refers solely to the representation described in Section III-A.

IV. Tensor-Based Recommendation Engine

The semantic Web service matchmaking algorithm presented in this paper is based on multilinear filtering framework proposed in [25]. In this section, the main features of this framework are presented. Moreover, all the settings and assumptions made in order to use this framework for the semantic service selection task have been provided.

A. Tensor-Based Representation of a Tuple Set

We assume that the heterogeneous data on Web services is transformed to the integrated set of n-tuples, where n is a number of attributes defining each event of relevance (or irrelevance) for a given pair of services. In order to describe events in a format which enables comparing them in quantitative way the weighed n-tuples have been chosen, which may be described as follows:

$$\Gamma = (n, \mathcal{V}^{(1)}, \ldots, \mathcal{V}^{(n)}, \Lambda, \psi), \tag{1}$$

where $\mathcal{V}^{(i)}$, $(i = 1, \ldots, n)$, is a set of values which may be used as the i-th element of an n-tuple, Λ is a set of n-tuples of the form $(v^{(1)}, \ldots, v^{(n)})$ where $v^{(i)} \in \mathcal{V}^{(i)}$, and $\psi : \mathcal{V}^{(1)} \times \cdots \times \mathcal{V}^{(n)} \to \mathbb{R}$ is a function used to assign the weight. To model the set of n-tuples as a multidimensional array (referred to as a tensor) one has to define the tensor space $\mathcal{T} = \mathcal{I}^{(1)} \otimes$

$\cdots \otimes \mathcal{I}^{(n)}$ where $\mathcal{I}^{(i)}$ is a basis of order $|\mathcal{V}^{(i)}| = n_i$ used to index elements of set $\mathcal{V}^{(i)}$. Finally, each set of n-tuples may be modeled as an element of $\mathcal{T}$.

In the presented framework we assume that $\psi : \mathcal{V}^{(1)} \times \cdots \times \mathcal{V}^{(n)} \to \{0, 1\}$ and $\psi(v^{(1)}, \dots, v^{(n)}) = 1$ if and only if $(v^{(1)}, \dots, v^{(n)}) \in \Lambda$. Then, input data may be modeled as tensor $T = [t_{i_1,\dots,i_n}]_{n_1 \times \cdots \times n_n}$ with binary entries. For the service matchmaking task based on S3 dataset [6], the set of used tuples contains events describing that a given service offer is relevant or irrelevant to a given service request.

Finally, it should be noted, that though the introduced function ψ herein returns binary values only, the model may be easily extended to use a weighted relevance information. In particular, $\psi'(v^{(1)}, \dots, v^{(n)}) = \beta$, where β indicates the weight assigned to an n-tuple. In such a case, $\beta = 0$ if $(v^{(1)}, \dots, v^{(n)}) \notin \Lambda$, and $\beta \in \mathbb{R}^+$ otherwise.

B. *Tensor-Based Processing*

The proposed multilinear filtering framework is based on tensor data modeling involving the use of so-called tensor-to-tensor transformations [25]. In general, tensor-to-tensor transformation is made according to the formula [25]:

$$\widetilde{T} = T \times_1 U^{(1)} \times_2 \cdots \times_n U^{(n)}, \tag{2}$$

where $T \times_i U^{(i)}$ is a tensor by matrix multiplication transforming tensor fibres of i-th mode of tensor T into new fibres in the corresponding mode of output tensor $\widetilde{T}$ in such a way that the entries of a new fibre are just inner products of the old fibre and columns of matrix $U^{(i)}$. The entries of the result tensor of each tensor-to-tensor transformation may be calculated as follows:

$$\widetilde{t}_{j_1,\dots,j_n} = \sum_{i_1 \in I^{(1)}} \cdots \sum_{i_n \in I^{(n)}} t_{i_1,\dots,i_n} u^{(1)}_{j_1,i_1} \dots u^{(n)}_{j_n,i_n}. \tag{3}$$

1) Transformation of input tensor into a state tensor of reduced size: Due to its multidimensional nature the input tensor suffers from its big size and high sparsity. In order to address these issues the proposed framework assumes the application of the preliminary dimensionality reduction similar to N-way Random Indexing (NRI) approach [14]. This step can be described as the tensor-to-tensor transformation using $n_i \times m_i$ matrices $U^{(i)}$ $(i = 1, \dots n)$, where n_i and m_i are the cardinalities of i-th mode of the tensor before and after transformation, respectively. Each row of the transformation matrix (i.e., $(u^{(i)}_{k,1}, \cdots, u^{(i)}_{k,m_i})$) forms the random vector of specified length and specified seed [24] – each entry of the vector is set to be equal to 0 or 1, and then the vector is normalized using L^1 norm. We denote the result of transforming the input data using the matrices $U^{(i)}$ described above as state tensor $X = [x_{i_1,\dots,i_n}]_{m_1 \times \cdots \times m_n}$.

The proposed model assumes that before being used for the processing and querying procedures the state tensor needs to be preprocessed according to two following steps (i) scaling in order to get the probability distribution done as follows

$$x_{i_1,i_2,\dots,i_n} := \frac{x_{i_1,i_2,\dots,i_n}}{\omega}, \tag{4}$$

where ω is the number of n-tuples used to build state tensor X, and (ii) preparing to be used in L^2-norm operations done by taking each entry square root value, i.e.:

$$x_{i_1,i_2,\dots,i_n} := (x_{i_1,i_2,\dots,i_n})^{1/2}. \tag{5}$$

2) Tensor querying: The tensor querying procedure is aimed at reconstructing the entries of the input tensor. In general, this procedure may be seen as a tensor-to-tensor transformation (reverse to the state tensor creation step), but due to practical reasons it is defined as a procedure of reconstructing the single entry of the input data tensor. For a given n-tuple $\gamma = (k_1, \dots, k_n)$ the query tensor $Q^\gamma = [q^\gamma_{i_1,\dots,i_n}]_{m_1 \times \cdots \times m_n}$ is constructed as a tensor of the same size as the state tensor. Its entries are calculated according to the formula:

$$q^\gamma_{i_1,\dots,i_n} = (u^{(1)}_{k_1,i_1})^{1/2} \cdot \dots \cdot (u^{(n)}_{k_n,i_n})^{1/2}. \tag{6}$$

Then, the result of the state tensor querying procedure is calculated as an inner product of preprocessed state tensor X (according to (4) and (5)) and query tensor Q^γ, as follows:

$$\widetilde{t}_\gamma = \sum_{1 \le i_1 \le m_1} \cdots \sum_{1 \le i_n \le m_n} x_{i_1,\dots,i_n} q^\gamma_{i_1,\dots,i_n}. \tag{7}$$

The same querying procedure is applied to the filtered state tensor which is constructed according to the procedure described in the next section.

C. *Covariance-Based Multilinear Filtering*

The proposed filtering algorithm is based on the application of the covariance data derived from a hierarchical structure of tensor flattenings. The proposed framework assumes the construction of filters for each tensor mode which are calculated as the linear combination of covariance matrices determined based on input state tensor X. The algorithm consists of steps described below. More details on the method may be found in [25].

1) Extracting covariance data from the tensor data: It has to be stressed, that different relations in data may be seen depending on the choice of attributes used to model tensor modes. The construction of different tensors modeling the dependencies between given mode elements may be done by building the most detailed tensor, i.e., the tensor involving the use of a maximum possible number of modes corresponding to the set of all event attributes provided in the input data, and then consecutive procedure of so-called tensor flattening (i.e., aggregating the tensor entries across the mode being flattened/hidden). Such a collection of different tensor structures is referred to as tensor network [8][9]. We denote the flattenings of tensor X as X_j, where j corresponds to the flattening code $(0 \le j \le 2^n - 1)$ defined in a way described in [25]. Each flattening except the totally flatten tensor (i.e., the tensor flatten to the scalar), and flattenings to one mode (i.e., to vectors), takes part in the procedure of filters' construction.

2) Overall centering: In order to provide the covariance data about elements of a given mode, each state tensor flattening has to be centered. The simplest way to provide the covariance matrix is to center across the tensor slices corresponding to the elements of this mode. The centering operation is provided by the subtraction of the mean of values in cells of a given tensor slice. However, this operation is not regarded as a most effective data centering [21][23]. Instead, so-called overall centering [21] should be used as the operation which leads to the minimum Frobenius norm of the covariance matrix. The overall centering may be done by consecutive centering of tensor fibres in each mode, i.e., for a given mode all fibres are centered and then this procedure is repeated for the next mode and so on. We denote the result of centering procedure applied for flattening X_j as X_j^c.

3) Generation of covariance matrices: Using the data collected in each centered tensor X_j^c we construct the matrices describing the relation among elements of the given mode, as follows: the unfolding matrix $X_j^{c,(i)} \in \mathbb{R}^{J_i \times (J_1 \times \cdots \times J_{i-1} \times J_{i+1} \times \ldots J_n)}$ is constructed, which collects i-th mode fibres of centered state tensor X_j^c as columns, and then, the symmetric matrix $A_j^{(i)} = [a_j^{(i)}]_{m_i \times m_i}$ such that:

$$A_j^{(i)} = X_j^{c,(i)} \left(X_j^{c,(i)} \right)^T \tag{8}$$

is obtained as a matrix representing the covariance between random dimensions used to enumerate the i-th mode. Finally, $A_j^{(i)}$ is the covariance matrix for elements of i-th mode constructed from the j-th flattening of state tensor X.

4) Constructing the filter based on covariance matrices: For mode i the optimal filter $F^{(i)}$ is constructed as a sum of an identity transformation and the average of matrices $A_j^{(i)}$. In particular, we have:

$$F^{(i)} = I_i + \frac{1}{k} \sum_j A_j^{(i)}, \tag{9}$$

where I_i is the identity matrix of size m_i, and k is a number of covariance matrices built for the i-th mode. We assume that before applying the filters the tensor X is centered according to overall centering [21] approach. The filters $F^{(i)}$ are used in order to transform centered tensor X^c into its filtered version $\widetilde{X}^c$ according to the formula: $\widetilde{X}^c = X^c \times_1 F^{(1)} \times_2 \cdots \times_n F^{(n)}$. At the next step the prediction tensor $\widetilde{X}$ is calculated as $\widetilde{X} = X - X^c + \widetilde{X}^c$. Finally, the tensor $\widetilde{X}$ is used for calculating the prediction results according to the querying procedure described by equations (6) and (7).

D. Complexity of the proposed method

Reducing the space and time consumption is a key issue for the tensor-based approach in which the complexity may grow exponentially with the number of tensor modes used in the model. Therefore, it is crucial to provide the dimensionality reduction step in the earliest phase of computing as possible, ideally, in the phase of data storing in the tensor structure. First of all, it has to be stressed that the tensor $\widetilde{X}$ used for prediction may be additionally transformed using the HOSVD approach [15] that leads to reduction of tensor size and, as consequence, shortens the time needed for querying. Furthermore, according to the research on existing state-of-the-art tensor-based data processing frameworks, including the incremental tensor analysis approach [26] and ALS-based tensor solutions [27], the space and time consumption for this kind of solutions may be efficiently reduced by using the approximation approach avoiding the diagonalization step, the fast approximation methods for finding principal components as well as random sampling techniques.

In particular, in the case of our approach, the space complexity of the proposed method is directly related to the size of a state tensor used to accumulate the data. In the case of the application scenario presented in this paper, we limit to 3-rd order tensors, so the space complexity is bounded by $O(m_1 m_2 m_3)$, where m_i is a cardinality of the i-th mode of the state tensor (as defined in Section IV-B). The computational cost of the method depends on the cost of state tensor construction based on accumulation of ω tuples ($O(\omega m_1 m_2 m_3)$), and the cost of the construction of covariance matrices from tensor unfoldings ($O(m_1^2 m_2 m_3 + m_1 m_2^2 m_3 + m_1 m_2 m_3^3) = O((m_1 + m_2 + m_3) m_1 m_2 m_3)$). In the case of the applying the additional dimensionality reduction step based on HOSVD, the additional cost of $O(h m_1 m_2 m_3)$ have to be taken into account, where h is the reduced number of dimensions. Since, in the application scenario presented in this paper, ω is greater than m_i for each i as well as than h, we have observed the biggest computational cost in the phase of state tensor construction. However, it has to be stressed, that the time of the state tensor accumulation may be efficiently shortened by taking into account that tensor structures corresponding to tuples are sparse. Nevertheless, due to the relatively small size of the state tensor used in the evaluation (see Section V-D), we have not applied such an optimization step.

V. Evaluation

It should be noted that a typical comparison of different matchmaking systems is not the main goal of the experimental research presented in this paper, as we focus on evaluating of several tensor processing methods in the experimental scenario of SWS matchmaking. We assume that each of the compared tensor processing methods is applied to process the same data obtained by means of data integration framework being a part of the system presented in Section III, given using the tuple-based internal data representation (n-tuples or RDF statements). The application of tensor-based data representation and processing methods have been already investigated in several domains for which – similarly as for the S3 task – the input data is multirelational or multidimensional.

A. SAWSDL-TC3 Dataset Use

The experimental evaluation presented in the paper is based on the use of the publicly available SAWSDL test collection – SAWSDL-TC3 [6]. The dataset provides 1080 semantic Web services written in SAWSDL (for WSDL 1.1) from 9 domains (education, medical care, food, travel, communication, economy, weapon, geography, simulation) and consists of both SAWSDL and OWL documents. The S3 SAWSDL-TC is divided into three main sets. The first and second set contain SAWSDL documents representing *queries* and potential *query* matches – *offers*, respectively. The third set consists of related OWL ontologies.

Additionally, the SAWSDL-TC3 contains the XML file `sawsdl-tc3.xml` describing the information on relevance between 4178 Web service pairs (i.e., *query* and *offer* pairs). The relevance information is provided using two independent relevance grades — *binary* and *4-graded*. For the purposes of the experiments presented herein, the *binary* relevance has been chosen. Nonetheless, it should be noted, that experiments could be easily extended – as briefly discussed in Section IV-A – to use the *4-graded* relevance information. For instance, one could set $\beta = 1.0$ if the grade was '*relevant*', $\beta = 0.5$ if '*potentially relevant*', and $\beta = 2.0$ if '*highly relevant*'. It should be kept in mind, however, that in order to adjust properly these weights for a specific dataset, a parameter optimization technique (such as cross-validation on the available training data) should be used.

B. Evaluation Scenarios

In order to experimentally evaluate the compared solutions we have used a part of the test collection, containing information about relevant and nonrelevant Web service matches, in the learning process. Moreover, as in other approaches the learning process itself is performed off-line (i.e., before the matchmaking). However, contrarily to the research presented in the literature, in our experiments we have tested how the training ratio tr – indicating the percentage of the entire test collection that is used to train the model – affects the quality of matchmaking. What is more, following the Information Retrieval experiment design practices [28][29], we did not use the full test collection during the evaluation (what is allowed in the case in S3 contest) but only the remaining part of the data (that was not used in the learning process) instead. For instance, for $tr = 0.2$ the remaining 80% of the test collection has been used to evaluate the predicted Web service matchings. Therefore, due to differences in the methodology, the final results are not directly comparable with the S3 contest results.

Such an evaluation methodology (i.e., based on both data sources) has been chosen as it does not assume that the textual and structural similarities between the items (here represented by semantic Web services) is directly correlated with the matching relation, and thus it may be considered as more comprehensive. In other words, the algorithm is expected to adapt to the specified task – as in a typical semi-supervised learning task – by inferring the meanings of the relations contained in the SAWSDL and OWL documents.

Apart from the hybrid scenario involving using both SWS descriptions and partial information about the relevant or nonrelevant matches, we additionally investigated a simplified scenario involving only the information about the Web service matches. Our motivation for such an approach is an attempt to show how much an algorithm is able to learn using sample mappings only, and how much the matching quality may increase by adding supplementary semantic information.

Finally, the quality of the generated Web service matches has been evaluated using typical Information Retrieval measures described in Section V-C. To compensate for the impact that the randomness of the dataset partitioning has on the results of the presented methods, all figures in this paper show series of values that represent the averaged results of 100 individual experiments. As a result, the standard error of each presented mean is less than 0.005.

C. Recommendation Accuracy Measures

Following other articles in the literature relevant to the Web service matchmaking, we have used the Mean Average Precision (MAP) measure:

$$MAP = \sum_{i=1}^{n} ap_i/n \tag{10}$$

where n is the number of requests tested and ap_i is the average precision for the i-th request. Particularly, the ap is defined as:

$$ap = \sum_{k=1}^{m} P(k)/min(m,r) \tag{11}$$

where r is the number of relevant matchings, m is the recommendation list length, and $P(k)$ denotes the precision at k-th prediction in the recommendation list. Specifically, the precision $P(k)$ is the ratio of correct matchings up to the position k over the k, and is equal to 0 when the k-th prediction is invalid.

Additionally, we have measured the Area Under the ROC curve (AUROC) as it directly allows to establish the probability of making correct or incorrect decisions by a system about whether a matching is relevant. According to [30], AUROC is equivalent to the probability of the system being able to choose properly between two items, one randomly selected from the set of relevant items, and one randomly selected from the set of non-relevant items. Hence, it allows one to abstract from any particular precision-recall proportion. Specifically, for an ordered list of predicted matchings R, AUROC is defined as:

$$AUROC = \frac{1}{|R|}\sum_{i=1}^{F}(s_i - i) \ , \tag{12}$$

where the probability score s_i is indicated by rank of the i-th true positive in R, and F is the number of false positives in R. In particular, if all relevant matchings appear before all nonrelevant matchings in the list, one will have a perfect ROC curve and $AUROC = 1$.

D. Recommendation Methods under Evaluation

We have compared the accuracy of our method to the accuracy of state-of-the-art tensor-based processing methods presented in the relevant literature. In order to perform such a comparison, we have developed our implementations of N-way random indexing (NRI) [14], HOSVD [15], joint feature mapping via tensor product [31], and a typical SVD-based matrix factorization [32]. The matrix factorization, herein referred to as '*MF (matchings)*', was performed on a matrix containing information about known Web service matchings, as it is not possible to unambiguously encode more relations in such a structure. As a consequence, we used *MF (matchings)* as a baseline method allowing to distinguish whether the use of tensor-based algorithms provides any significant benefit compared with classical matrix factorization.

We have also evaluated feature mapping via tensor product, herein referred to as '*Feature Mapping*', as it has been reported in [31] and followed in [33] that such a model allows to exploit not only the direct relations between individual objects but also the associated textual features. In this paper, in order to adapt the algorithm to the Web service matching scenario, in accordance with [33] we represent each pair of Web services (i.e., a request r and an offer o) as an outer product $r \otimes o$ of two corresponding vectors represented in a feature space of

Web service descriptions. Subsequently, the resultant tensor is defined as $\sum_i \sum_j (r_i \otimes o_j) m_{i,j}$, where $m_{i,j}$ indicates whether the matching between r_i and o_j is relevant ($m_{i,j} = 1$), nonrelevant ($m_{i,j} = -1$) or unknown ($m_{i,j} = 0$). Finally, Principal Component Analysis (PCA) is applied in order to reduce the noise and extract the most salient features.

The HOSVD algorithm has been performed on RDF data expressed in a form of a 3-rd order tensor, in which every predicate is represented as an adjacency matrix – forming the slice of the tensor – between subjects and objects. This model has been already used in the relevant literature [11][12] for the task of multirelational statistical learning. Contrarily, we were not able to obtain any meaningful results by applying solely HOSVD on a tensor model build from tuples (as presented in Section III-A), thus we omitted these results in the presented evaluation.

Finally, we have evaluated the effectiveness of the proposed covariance-based multilinear filtering (CMF) regardless of the underlying data representation model. For that reason we additionally present MAP and AUROC results of the experiments performed solely on a NRI-reduced vector space and the probabilistic state tensor introduced in Section IV-B, herein referred to as '*Probabilistic ST*'. By that means, the ability of CMF to process tensor spaces of reduced dimensionality is verified.

All the above described methods have been evaluated using the same data (correspondingly in the form of a n-tuple or RDF) as their input. The combinations of parameters (such as the k-cut or the core tensor size) that lead to the best recommendation quality (i.e., the highest AUROC value) were considered optimal, and used in experiments illustrated in this paper.

For purposes of evaluated methods we have used the framework introduced in Section III in order to construct both tuple-based and RDF-based internal representations of the input data. In particular, the n-tuple representation of the dataset is processed into the 3-rd order tensor structure of size $(2 \times 110 \times 615)$ in order to store the data concerning 4178 n-tuples. As described in Section III-A, the first tensor mode contains information about the relevance (i.e., relevant, non-relevant). Subsequently, the second tensor mode – concerning the *queries* – is constructed using vectors of length 110, while the third mode – concerning the *offers* – is built using vectors of length 615.

In the case of data given as RDF statements, the tensor of the size $(1084, 7, 1909)$ is constructed – according to Section III-B – to store the data on 15537 triples. In the presented experiments, we also investigate the standard collaborative filtering approach using the request-offer matrix of size $(42, 1043)$ containing only the data on service relevance modeled using 4178 non-zero values from the set $\{-1, 1\}$.

VI. Experimental Results

The results of our evaluation performed, using MAP and AUROC measures, are presented in Figures 2 and 3, respectively. The comparison has been performed with the use of different training ratios tr, ranging from 0.05 to 0.9. As it has been confirmed experimentally, the introduced algorithm – CMF – allows to achieve higher quality matchings, both in terms of MAP and AUROC and for all training ratios, as compared to other evaluated methods.

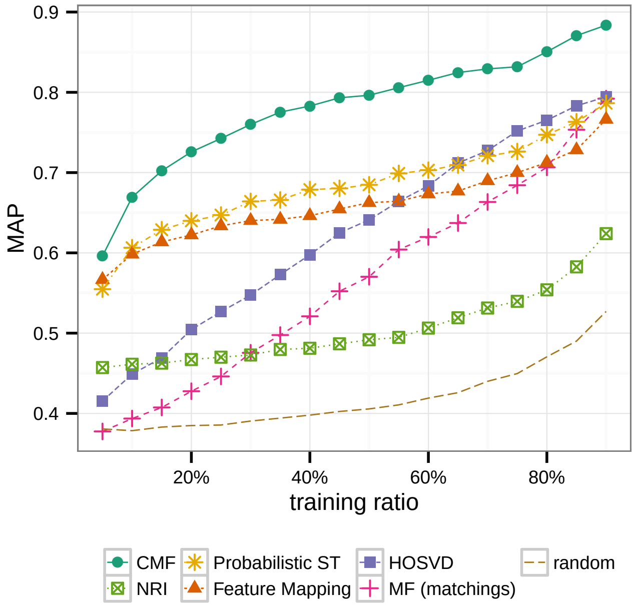


Figure 2. The MAP results.

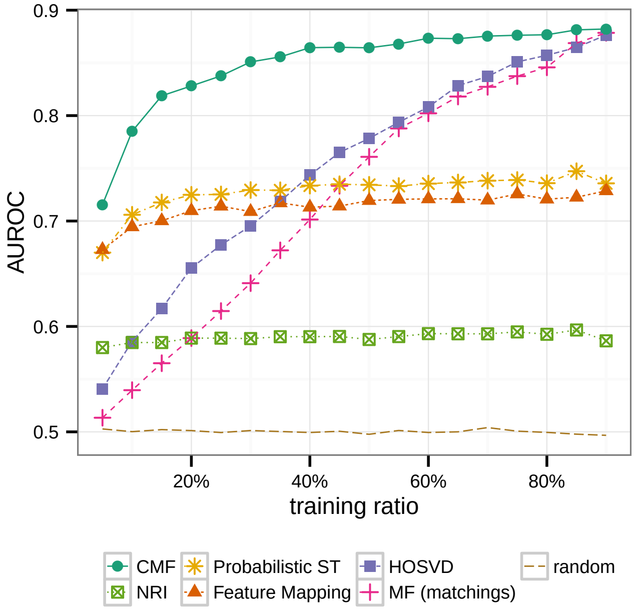


Figure 3. The AUROC results.

As it has been shown, the accuracy of *MF (matchings)* is almost linearly dependent on the training ratio. Particularly, for the smallest $tr = 0.05$ matrix factorization achieves the lowest score – similar to a random one, while for the highest $tr = 0.9$ the obtained results are comparable to other best performing methods (i.e., in terms of AUROC).

An algorithm generating random recommendations obtained $AUROC = 0.5$ for all tr – as expected due to its probabilistic interpretation, what additionally confirms the reliability of the AUROC measure. On the other hand, due to correspondingly smaller test set – more precisely smaller number of relevant matchings for every offer – for higher tr the MAP values for a random algorithm are also respectively higher. Therefore, for comparison, we included the random method in the presented evaluation.

It may be also observed that the HOSVD algorithm, performed on a 3-rd order tensor build from RDF statements, enabled us to obtain slightly higher results – although still statistically significantly higher – than *MF (matchings)*. Nevertheless, the results of HOSVD, even with optimally adjusted size of the core tensor, are still heavily dependent on tr. Particularly, for $tr < 0.1$ HOSVD performed only slightly better than the random method despite processing all of the semantic information extracted from SAWSDL files.

The algorithm based on feature mapping via tensor product, compared to the baseline *MF matchings*, clearly enabled to obtain higher AUROC and MAP values for smaller training ratios (i.e., $tr < 0.4$). In opposition, for denser train set the results are not so apparently conclusive. Specifically, *Feature Mapping* achieves higher quality results in terms of MAP than the baseline method for almost all tested tr. At the same time, in the case of the AUROC measure its performance is almost constant (with only relatively small gains for higher tr) and significantly inferior to a simple matrix factorization. Such a finding may be caused by the fact that AUROC probabilistically reflects the system's performance (see Section V-C) – which is rather independent from the amount of behavioral data (herein – known matchings) in case of content-based methods. On the other hand, the MAP measure takes into account the number of relevant matchings in the test set (as it has been shown on the case of random matchings).

Although the main purpose of NRI is to reduce the dimensionality of the input tensor, and not multiple factor analysis, we included this algorithm in the evaluation as the introduced CMF method is partially based on the NRI concept. As shown in Figure 3, the ability of NRI to provide high quality recommendations is independent of the number of input training matches – probably due to the fact that it merely reflects the co-occurrences of the terms in the requests and in the offers. It should be also noted that although the addition of scaling and L^2-normalization – in *Probabilistic ST* method – enabled to significantly improve the performance of tensor-based processing, such an algorithm still does not allow one to provide higher quality results than HOSVD or even *MF (matchings)* in case of higher tr. Additionally, we have performed experiments using a 3-rd order tensor build from RDF statements and processing methods such as NRI, *Probabilistic ST* and CMF. However, due to definitively lower quality of the provided recommendations we have omitted these results from the final evaluation so as not to obscure the presented results.

Therefore, it may be stated that in the application scenario investigated in this paper, the tuple-based probabilistic tensor modeling combined with covariance-based multilinear filtering enables to outperform other tensor-based methods, regardless of the amount of known matchings (herein depicted by tr).

VII. Conclusion and Future Work

The experimental evaluation results presented in the paper are expressed in terms of AUROC and MAP results. It is worth stressing that the evaluation has been done using the standard Information Retrieval methodology that assumes partitioning of the dataset on the training and testing sets in such a way that the data used for testing the performance cannot be used to learn or tune the model. Quite surprisingly, such a methodologically correct approach differs from the evaluation methodology used by the authors that have taken part in the S3 contest, as they frequently use the same data on matchings between services for both for the matchmaker system parameters tuning (e.g., by means of the cross-validation approach) and for the final performance evaluation. The results presented in the paper indicate the superiority of the proposed combination of the tuple-based probabilistic tensor modeling and the covariance-based multilinear filtering over other tensor-based methods, including NRI and HOSVD-based RDF processing – the superiority that is clearly visible regardless of the amount of matchings included in the training set.

Finally, it has to be stressed that contrarily to the state-of-the-art algorithms such as [3]–[5] the proposed Semantic Service Recommendation System does not rely on any kind of predefined rules customized for SAWSDL matchmaking. As introduced in Section IV, the recommendation engine is virtually unconstrained regarding any data structure, and thus it may be easily applied in other domains, as already shown in [25]. For that reason, for future work we plan to extend our research to address other semantic matchmaking tasks. Another potential directions of the further research would be an extended use of the referenced ontologies, and conducting the experiments involving the 4-graded relevance information, in addition to the presented herein binary relevance.

Acknowledgement

This work is supported by the Polish National Science Centre, grant DEC-2011/01/D/ST6/06788, and by the Polish National Centre for Research and Development, grant PBS1/B3/14/2012.

References

[1] T. Wang, D. Wei, J. Wang, and A. Bernstein, "SAWSDL-iMatcher: A Customizable and Effective Semantic Web Service Matchmaker," Web Semantics: Science, Services and Agents on the World Wide Web, vol. 9, no. 4, 2012, pp. 402–417. [Online]. Available: http://www.websemanticsjournal.org/index.php/ps/article/view/239

[2] P. Plebani and B. Pernici, "URBE: Web Service Retrieval Based on Similarity Evaluation," Knowledge and Data Engineering, IEEE Transactions on, vol. 21, no. 11, nov. 2009, pp. 1629 –1642.

[3] M. Klusch, P. Kapahnke, and I. Zinnikus, "Adaptive Hybrid Semantic Selection of SAWSDL Services with SAWSDL-MX2." Int. J. Semantic Web Inf. Syst., 2010, pp. 1–26.

[4] M. Klusch and P. Kapahnke, "The iSeM matchmaker: A flexible approach for adaptive hybrid semantic service selection," Web Semantics: Science, Services and Agents on the World Wide Web, vol. 15, 2012, pp. 1–14.

[5] S. Schulte, U. Lampe, J. Eckert, and R. Steinmetz, "LOG4SWS.KOM: Self-Adapting Semantic Web Service Discovery for SAWSDL," in Proceedings of the 2010 6th World Congress on Services, ser. SERVICES '10. Washington, DC, USA: IEEE Computer Society, 2010, pp. 511–518. [Online]. Available: http://dx.doi.org/10.1109/SERVICES.2010.40

[6] Semantic Service Selection Contest SAWSDL track dataset (SAWSDL-TC). http://projects.semwebcentral.org/projects/sawsdl-tc/. [retrieved: February, 2015]

[7] Semantic Service Selection Contest webpage. http://www-ags.dfki.uni-sb.de/~klusch/s3/index.html. [retrieved: February, 2015]

[8] N. Vervliet, O. Debals, L. Sorber, and L. De Lathauwer, "Breaking the curse of dimensionality using decompositions of incomplete tensors: Tensor-based scientific computing in big data analysis," Signal Processing Magazine, IEEE, vol. 31, no. 5, Sept 2014, pp. 71–79.

[9] A. Cichocki, "Era of big data processing: A new approach via tensor networks and tensor decompositions," CoRR, vol. abs/1403.2048, 2014. [Online]. Available: http://arxiv.org/abs/1403.2048

[10] H. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "Multilinear principal component analysis of tensor objects for recognition," in 18th International Conference on Pattern Recognition, ICPR 2006., vol. 2, 2006, pp. 776–779.

[11] M. Nickel and V. Tresp, "An Analysis of Tensor Models for Learning on Structured Data," in Machine Learning and Knowledge Discovery in Databases, ser. Lecture Notes in Computer Science, H. e. Blockeel, Ed. Springer Berlin Heidelberg, 2013, vol. 8189, pp. 272–287. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-40991-2_18

[12] T. Franz, A. Schultz, S. Sizov, and S. Staab, "Triplerank: Ranking semantic web data by tensor decomposition," in The Semantic Web - ISWC 2009, ser. Lecture Notes in Computer Science, A. Bernstein, Ed. Springer Berlin Heidelberg, 2009, vol. 5823, pp. 213–228.

[13] O. Chapelle, B. Scholkopf, and A. Zien, "Introduction to semi-supervised learning," in Semi-Supervised Learning, O. Chapelle, B. Scholkopf, and A. Zien, Eds. The MIT Press, 2006, pp. 1–8. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/21243728

[14] F. Sandin, B. Emruli, and M. Sahlgren, "Incremental dimension reduction of tensors with random index," CoRR, Mar. 2011, pp. 240–56. [Online]. Available: http://arxiv.org/abs/1103.3585

[15] L. De Lathauwer, B. De Moor, and J. Vandewalle, "A multilinear singular value decomposition," SIAM J. Matrix Anal. Appl, vol. 21, 2000, pp. 1253–1278.

[16] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, "Evaluating collaborative filtering recommender systems," ACM Trans. Inf. Syst., vol. 22, no. 1, Jan. 2004, pp. 5–53. [Online]. Available: http://doi.acm.org/10.1145/963770.963772

[17] C. D. Manning, P. Raghavan, and H. Schtze, Introduction to information retrieval. Cambridge University Press, NY, USA, 2008.

[18] T. Mitchell, Machine Learning. McGraw-Hill, New York, NY, USA, 1997.

[19] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," SIAM Review, vol. 51, no. 3, 2009, pp. 455–500. [Online]. Available: http://dx.doi.org/10.1137/07070111X

[20] L. R. Tucker, "Some mathematical notes on three-mode factor analysis," Psychometrika, vol. 31, no. 3, 1966, pp. 279–311. [Online]. Available: http://dx.doi.org/10.1007/BF02289464

[21] P. M. Kroonenberg, Three-mode principal component analysis: Theory and applications. DSWO press; three-mode.leidenuniv.nl, 1983, vol. 2.

[22] L. Grasedyck, D. Kressner, and C. Tobler, "A literature survey of low-rank tensor approximation techniques," ArXiv e-prints, Feb. 2013, pp. 53–78.

[23] R. Bro and A. K. Smilde, "Centering and scaling in component analysis," Journal of Chemometrics, vol. 17, no. 1, 2003, pp. 16–33. [Online]. Available: http://dx.doi.org/10.1002/cem.773

[24] T. Cohen, R. Schvaneveldt, and D. Widdows, "Reflective Random Indexing and indirect inference: a scalable method for discovery of implicit connections." Journal of biomedical informatics, vol. 43, no. 2, Apr. 2010, pp. 240–56. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/19761870

[25] A. Szwabe, P. Misiorek, and M. Ciesielczyk, "Multilinear Filtering Based on a Hierarchical Structure of Covariance Matrices," Schedae Informaticae, vol. 24, 2015, in press. [Online]. Available: http://ncn6788.cie.put.poznan.pl/images/ncn6788-tfml2015.pdf

[26] J. Sun, D. Tao, S. Papadimitriou, P. S. Yu, and C. Faloutsos, "Incremental tensor analysis: Theory and applications," ACM Trans. Knowl. Discov. Data, vol. 2, no. 3, Oct. 2008, pp. 11:1–11:37. [Online]. Available: http://doi.acm.org/10.1145/1409620.1409621

[27] I. Pilászy, D. Zibriczky, and D. Tikk, "Fast als-based matrix factorization for explicit and implicit feedback datasets," in Proceedings of the Fourth ACM Conference on Recommender Systems, ser. RecSys '10. New York, NY, USA: ACM, 2010, pp. 71–78. [Online]. Available: http://doi.acm.org/10.1145/1864708.1864726

[28] R. A. Bailey, Design of Comparative Experiments, ser. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2008.

[29] C. D. Manning, P. Raghavan, and H. Schutze, Introduction to information retrieval. New York, NY, USA: Cambridge University Press, 2008, no. c. [Online]. Available: http://www.langtoninfo.com/web_content/9780521865715_frontmatter.pdf

[30] T. Fawcett, "An introduction to roc analysis," Pattern Recogn. Lett., vol. 27, no. 8, Jun. 2006, pp. 861–874. [Online]. Available: http://dx.doi.org/10.1016/j.patrec.2005.10.010

[31] J. Basilico and T. Hofmann, "Unifying collaborative and content-based filtering," in Proceedings of the Twenty-first International Conference on Machine Learning, ser. ICML '04. New York, NY, USA: ACM, 2004, pp. 9–. [Online]. Available: http://doi.acm.org/10.1145/1015330.1015394

[32] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," Computer, 2009, pp. 42–49. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5197422

[33] S.-T. Park and W. Chu, "Pairwise Preference Regression for Cold-start Recommendation," in Proceedings of the Third ACM Conference on Recommender Systems, ser. RecSys '09. New York, NY, USA: ACM, 2009, pp. 21–28. [Online]. Available: http://doi.acm.org/10.1145/1639714.1639720

On Establishing Behaviorally Adoptive Semantic Narratives

Jason Bryant, Gregory Hasseler, Matthiew Paulini
Air Force Research Laboratory/RI
Rome, NY 13411
{Jason.Bryant8, Gregory.Hesseler, Mathiew.Paulini}
@us.af.mil

Noor Ahmed
Dept. of Computer Science
Purdue University and AFRL/RI
West Lafayette, IN 4906
ahmed24@purdue.edu

***Abstract*— Providing personalized consumer contents can be both empowered and simplified through adapting analytics modeling and results with semantic representations. Analytics representations tend to be unique to their proprietary technological solutions, growing silos of non-interoperable, non-shareable results. Our approach to overcoming these obstacles is to pair our analytical modeling solution, Direct Qualification, with middleware integrated algorithms for graph, content, identity, and behavioral-based analytics. This abstraction layer of semantically represented analytics enabled multiple best-fit analytics engines to be deployed in parallel while providing a common query front-end for analytics observations, provenance, and trends. We introduce the establishment and the adaptation of a Behavior Ontology (BO) and Behavior Analytics (BA) modeling. We describe the integration of such behavior modeling with the semantic modeling of analytics and state management for an effective consumer content personalization system. We illustrate our prototype with publish and subscribe middleware and show the preliminary results. These components will be integrated into a holistic semantic analytics solution with autonomous functions for behavior optimizations, pluggable algorithm components, and end-to-end, machine learned personalization for information consumers and producers.**

Keywords-semantic modeling; consumer behavioral modeling; direct qualification; transactional pattern matching.

I. Introduction

Modern information management systems perform an increasingly expansive catalog of operations with greater complexity and with more feature expectations, including personalized consumption of information that requires analytics-based solutions. The desired feature gain from these efforts include more in depth data analysis, business processing autonomy, provenance traceability, trusted information sharing, and enhanced query options, all while optimizing performance. The current landscape of analytics generates a multitude of unintended negative consequences. Analytics engines are generally not interoperable, resulting in multiple silos of analytic results which cannot be simply joined, queried, or introspected for quality. Analytics engines rely on multiple proprietary standards with completely different paradigms of information, including content parsing, graph traversals, rule engine deductions, keyword or vector modeling for frequency, image & video learning, etc. Additionally, analytic technologies tend to take one out of three common approaches, including deductive, inductive, or behavioral. Our effort seeks to empower middleware solutions by combining all three approaches into a single queryable semantic service.

Modern information middleware consists of data models (Deductive Capabilities) and data analysis (Inductive Capabilities), with a limited notion of identity management for authorization and authentication, and services that orchestrate these capabilities according to consumer needs. Generally, OWL and RDF solutions approach problems within the web domain, which is distinct from an Information Management (IM) system in several ways. Search engines and web sites have autonomous feedback mechanisms that can capture rough estimations of information quality by observation of consumer link selection. Alternatively, IM systems have a more difficult time assessing quality from solicited consumers due to a lack of knowledge about information interactions once results have been returned, however IM systems have other advantages. Unlike semantic web-domain solutions, IM systems have access to identity management provenance, information analytics, information sourcing, as well as broad information access across multiple information dimensionality, including roles, formats, types, and access to deployed workflow or process models.

In our approach we seek to develop a middleware IM system that leverages the internal model, service, identity management, and data components to make more advanced information analytics and personalization possible. The effects of these enhancements can be positive facilitators for IM system, participant, and information trust, as well as assessments that can score and compare information quality, information impact, algorithm effectiveness, or model / ontology quality.

A key part of our approach is to seed the information process within the IM system with several algorithms that, when integrated, offer critical capabilities to autonomously learn information domains (topic modeling), personal information preferences (affinity clustering), and are informed by models prescribing the general workflows of participating information roles. In this manner the application narratives can be established

ISBN: 978-1-61208-420-6

and adapted through the dynamic topic models, the consumer behaviors can be observed and analyzed partially through models (behavior ontology) and partially through data-driven induction (affinity clustering), while all values and analytic results are represented via OWL and RDF, providing a huge advantage over existing middleware systems that require complex orchestrations or combinational queries that span multiple deployed analytics engines.

The middleware system has some modeling and infrastructure components complete, such as the analytics representation into OWL and RDF (direct qualification), and a set of pre-loaded algorithms (VSM, PageRank, HITS, topic modeling), others are still under development, including the behavioral ontology and the collaborative filtering algorithm that blends the consumer behaviors with feedback from the affinity scoring. Our main contributions can be summarized as follows:

- We outline a holistic semantic system that can support the abstraction of multiple analytics engine, eliminating concerns for proprietary analytics silos, complex combinational queries, and incompatible result representation.
- We introduce a simple consumer behavior ontology model that enables semantic representations of IM system transactions.
- Efficient scheme of integrating consumer behavior models, semantic modeling, and transactional pattern matching for content personalization.
- We apply semantic web applications to IM system domain with distinct information and modeling challenges.
- We project multiple analytic paradigms into a common semantic representation to enable more powerful queries and analytical tooling.

We have organized the paper as flows: we first give a brief background of the subject and the motivation behind our work in section II. Our consumer behavior modeling techniques and planning are discussed in section III, followed by the adaptation and the integration of these models into the semantic modeling and analytics in section IV. We discuss our prototype design and implementation in section V, and the preliminary experiments in section VI. In section VII and VIII covers the related work and the conclusion respectively.

II. BACKGROUND

The essence of information personalization is to enable the pairing of predictive analytics within adaptive content capabilities. Accomplishing this within an Information Management Systems requires following a foundational capabilities. These include:

- OWL Representation of Analytic Provenance and Results (Direct Qualification).
- Prescribed Workflow Activity Models (Constrained by Behavior Ontology & Planned Machine Learning Component).
- OWL Representation of Transactional Behavior History (Behavior Ontology)
- IM System Supported Analytics Engines (Jung, Lucene, and Mallet)
- Dynamic Application/Consumer Narrative Algorithm (Topic Modeling via Mallet)
- Personalized Query Results Feedback via Data Affinity Algorithms (Affinity Cluster Scoring via Jung’s PageRank and Lucene’s VSM)

Supporting the semantic modeling and execution of analytics, we created approaches including Direct Qualification (DQ) and State Management (SM) for OWL in our previous work [1]. In this work, we seek to extend these models to support the introspection of behaviors and interactions between consumers and the IM system (middleware), and thereby measure and validate the effectiveness of the behavioral analytic approaches and models themselves.

By embracing a data perspective that combines relationships for unstructured knowledge representation with structured, document-centric relationships, the process of determining, modeling, and expressing personalized information relevance with semantic technologies can be performed. Our approach seeks to solve a combination of challenges within Information Management (IM), Semantic Information Modeling, Data to Information (D2I), and Quality of Service (QoS) Enabled Dissemination (QED).

III. BEHAVIORAL ONTOLOGY AND ANALYTICS MODELING

Consumer and producer behaviors leave fingerprints at the informational layer that can be discovered, tracked, analyzed, correlated, and mined. Analytics engine complex event processing can utilize behavioral metadata, information content, analytics results, and historical trends to correlate information with emerging common consumer narratives over time. Additionally, linking these narratives to data-driven analytics assessment can monitor narrative changes, behavioral anomalies, determine critical personalized information, adapt to trends, and identify where information may be insufficient for consumer needs.

Behavior will be a distinctly different set of transactions, dependent upon the system upon which it is modeled. For our efforts, the IM system is a RESTful system allowing producer publications with attached metadata tags for type, format, and identity, and consumers that submit XPATH, SPARQL, or keyword queries. For a simplistic transactional IM system like this, we have started with a similarly simplistic behavioral

ISBN: 978-1-61208-420-6

ontology consisting of entities for Consumer, Producer, Query, QueryExpression, Publication, QueryType, ResultSet, Result, Document, Type, Format, Role, and Identity. The object properties and data properties associated are simply possessives of these entities (e.g. hasResultSet, performedQuery, publishedDocument, etc.). Capturing these relationships over time provides URIs that can act as hooks for analytic results, such as affinity scores across identity, format, type, document, or consumer dimensions.

The general process by which behaviors are associated with either published or consumed information is illustrated in Figure 1. After behaviors are collected within transactional provenance, either the published document or set of results is scored for affinity and related to consumer/producer identities.

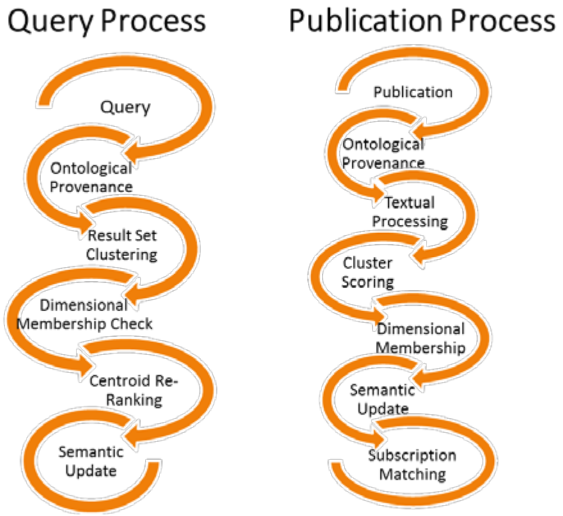


Figure 1- Process Overview

IV. SEMANTIC NARRATIVE INTEGRATION

Creating topic models, effectively producing dynamic information domain ontologies, on the fly is effective only if done utilizing sound techniques and dictionary to ontology classifications. WordNet and Mallet have built in tools that perform these features adequately, utilizing LDA to produce dynamic lists of topic models.

In our approach topic models are filtered according to configured quality metrics, and then assigned dimensional information categories. After being categorized as a topic model, the 2D grid of analytic scores for single dimensions such as role, query purpose, format, geography, or temporality. These can be subdivided into a 3-dimensional information centroid by utilizing cross-cutting attribution and grouping, most effectively according to identity.

Utilizing identity, role, and tasking metadata attributes to track information system transactions can support real-time content adaptation to information consumers by measuring trust and metadata-based information hotspots. Topic models cataloged via analytics can establish on the fly information domain ontologies, and paired with semantically modeled analytics, can result in advanced combinational forms of information and analytics queries. These customized data services enable topically modeled semantic narratives, behavioral adaptation, and content personalization via low-cost, reality-based solutions, rather than high-cost, prescribed, model-based solutions. Mallet is leveraged to perform LDA upon each received document, and when a minimally necessary set is received within an information dimension (format, type, role, identity, activity, etc.), a set of 30-40 dynamic topics are extracted across each identity, and then re-oriented around new centroids over time as the personalization features adapt.

A. STATE MANAGEMENT

Maintaining a historical record by creating a new instance of a resource and its present relationship states can generate mass duplication and waste memory and processing resources. Over time, particularly if an event has relationships that change quite often, numerous of versioned instances of the same event could be created, making queries overly complex and resulting in a high degree of overhead due to duplication for relationships may or may not be static.

Some attributes of an asset may be occurrent (e.g. name, identity, asset type, etc.), while others are continuant (e.g. fuel level, latitude, longitude, role, etc.). Semantics, even when using instances of an entity, treat all relationships as occurrent, although there is allowance for limiting their cardinality. OWL, SPARQL, and most ontologies do not have a built in mechanism to support the distinction between occurrent and continuant relationships. In order to retrieve changes of state for a data or object attribute of an instance, that attribute must be explicitly defined within an ontology or an additional, customized layer of abstraction.

Traditional semantic data model approaches fall short when confronting the challenge of state-based relationships. They focus on static knowledge representation, extractions of static data properties, or enabling of information management features via rule engines and inferencing. Managing states for data and object properties are applicable to all stateful semantic resources. Managing the state of semantic relationships is significant in reducing the computation time of semantic queries, the load on semantic DBs, and eliminating wasteful property and instance duplications. The key relationship used for this is the `specializationOf` predicate of the Provenance Ontology (Prov-O) W3C recommendation. It is intended to apply state-based relationships to any Entity, Agent, or Activity, auspices under which any semantic asset instance should fall.

In our experiments we apply this approach by requiring all occurrent relationships to be related to the singular instance URI of a specific entity, while all relationships involving state changes are related through Specialization deltas, such as a Consumer entity having a temporary role relationship.

B. DIRECT QUALIFICATION

Reification, an intrinsic complexity of the semantic standards, is the consequence of attempting to simplify all relationships into Subject-Predicate-Object sets. It is normally implemented when a semantically modeled instance is seeking to express either the qualification or provenance of a relationship. These two cases can be mitigated without resorting to reification, however. Adopting a quad-based perspective of semantic relationships can achieve a basic form of provenance by allowing traceability to the source named graph's unique URI. The Prov-O ontology expands the set of provenance support and supplies some generalized predicates for qualification. This effectively solves the non-probabilistic subset of analytics use cases. However, even with pairing both of these approaches there is a failure to solve the qualification of probabilistic analytics, such as the results of Vector Space Modeling, PageRank, HITS, or other Natural Language Processing (NLP) analytics. Our approach towards supplying these capabilities is the Direct Qualification of probabilistic relationships with a supporting relevancy ontology.

The primary steps for enabling Direct Qualification are outlined as follows:

1. Support persistence for raw documents and semantic quad-based relationships, ideally by using the semantic URI of the document's named graph as the unique key for the raw document retrieval.
2. Strictly enforce the separation of the semantic models for class instances from the events affecting their state relationships.
3. Support graph-based processing of analytics over semantic edges and vertices.
4. Support event-based scoring triggers for analytics, such as SPARQL queries, XPath queries, semantic reasoning, or keyword searches of raw text.
5. Determine the appropriate Direct Qualification Model based upon tests for occurrence, continuance, and the monotonicity of the entities involved in the applied analytic.
6. Express the scoring of documents through the pairing of a provenance ontology with an analytics/relevancy ontology.
7. Persist the DQ results within the quad-store.

An example of DQ is illustrated in the following figures, with a general purpose (Figure 2) set of analytics relationships from the ontology, followed by more concrete examples utilizing PageRank (Figure 3) and VSM (Figure 4).

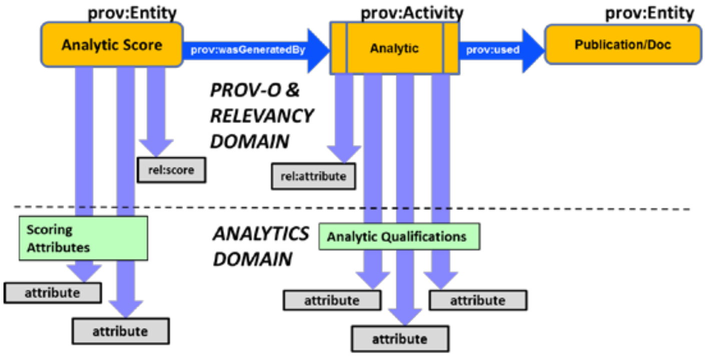


Figure 2 – General Direct Qualification Application

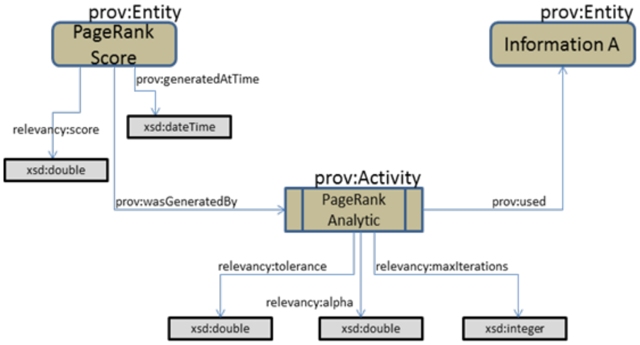


Figure 3 – PageRank Direct Qualification Use Case

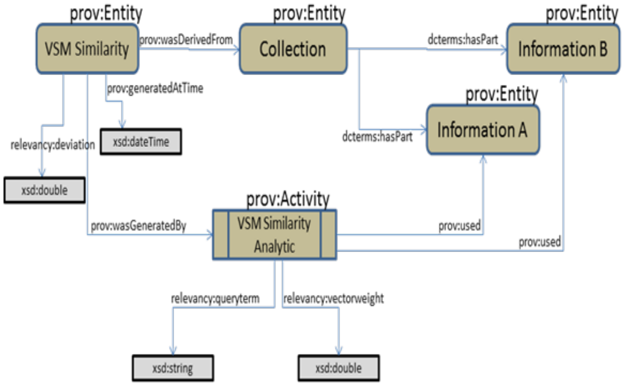


Figure 4 – VSM Direct Qualification Use Case

V. SYSTEM DESIGN AND IMPLEMENTATION

After stages for pre-processing, format determination (Aperture) and semantic extraction completes, the execution of the analytics, DQ, narrative creation, analytic scoring, and affinity scoring are executed as part of the publication and query processes. The system utilizes the following:

- Pre-requisite: Establishing a common representation for high level application narratives, research provenance, and analytic domain concepts.
- Pre-requisite: Establishing an ontology with entities and relationships supporting affinity, clustering, topic modeling, and role, format, type and identity-oriented membership groups.
- Pre-requisite: Create topic model classifiers for each primary information dimension of the metadata tags, including behavior, role, information format, geolocation, and identity.

1. Score each new publication according to a similarity / affinity vector within each primary information dimension.
2. Recalculate cluster centroids and dynamic topic model relations after every n publications.
3. Evaluate thresholds for information grouping inclusion / exclusion when thresholds of affinity are reached for each measured information dimension.
4. Score the information for aggregate personalization for cross-information domain grouping, according to existing queries.
5. Adapt the baseline metric for personalized relevancy thresholds after an initial n publications.

- Post-Operations: Measure the transition of information centroids / clusters over time as classifiers improve.
- Post-Operations: Compare results of the trend-based relevancy metric to a prescribed workflow template in order to validate models.

The functional composition of the system is illustrated below (Figure 5) by showing the internal resources and capabilities established and utilized in order to bridge the gap between raw information, behavior, validated narratives, and personalized content.

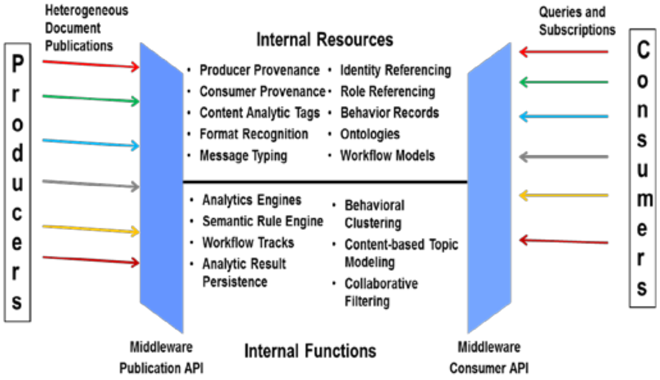


Figure 5 – Mapping of Middleware Resources/Functions

VI. EXPERIMENTS

The test scenarios include a compilation of 10000+ research papers of disparate fields and a 15 GB set of imagery from Flickr. The semantic relationships created were produced by means of the extraction framework we created in our previous ICCRTS research (Bryant, 2014), with added support for GeoSPARQL location extractions.

The results of semantic processing is an semantic document represented via RDF/OWL relating internal values for details involving times, locations, narratives, cataloged topic models, points of contact, metadata, etc.

Ontology support includes common solutions for time, geospatial (GeoSPARQL), common elements (U-Core SL), and custom ontologies for information management, and relevancy. Format and XML type determination is performed in a pre-processing stage prior to semantic extraction, if applicable. The Aperture source project is adopted to provide the majority of the format and type determination solution.

The results of the experiments include semantically represented cross-dimensional domain membership of all published data, according to the determined applicable information domains, and the centroid k-means scores of each information and identity-based data dimension. This enables the determination of domain-based relevancy measures, and in particular, cross-cutting identity-oriented information relevancy measures.

VII. RELATED WORK

This research addresses trust, affinity, collaborative filtering, and cross-cutting analytics engine abstractions for interoperable queries. This supports the modeling and probabilistic analytics that assesses and qualitatively relates information, enhances query options, and detect information anomalies or model workflow outliers. These capabilities provides potentially critical advancements towards autonomously determining whether information should be excluded from an information result set based upon its determined value, historical precedence, and personalized interest. While this allows queries to alter from a pre-defined domain-based set of operations, it also eliminates extensive modeling and domain ontology costs.

As semantic standards mature and applications expand into new domains, research regarding semantic management of stateful relationships is beginning to be explored more fully. Current research has been tangential, at best, while missing many of the niche problem areas of semantics. Approaches in this area have focused on inferencing through the use of join sequences [5] or resolving models with conflicting states [4]. So far, approaches involving applied analytics for state management have attempted to do so during the extraction phase of data [6], rather than utilizing semantic technologies or ontology models.

While there are ongoing efforts towards document analysis using analytics such as PageRank (Ding, 2002), VSM, and HITS, the focus of those efforts has been on ontology matching [2] or temporal/geospatial query enhancement [8]. Our approach differs in that it stays confined to semantic technologies with special emphasis on event-based information sharing, modeling, data mining, and retrieval, all combined. Furthermore, one of the key differentiator of some existing semantic models with DQ approach is that they adopt a constantly "present" based view that updates the instance with relationships reflecting any changes in its state. Thereby, the state changes' value can never be considered truly distinct from its identity URI.

While some approaches were discovered that sought to model analytics similar to the direct qualification and semantic state management techniques, none were found that sought to provide OWL and RDF abstractions and system support to facilitate analytic engine interoperability and freedom from multiple proprietary query dependencies.

VIII. CONCLUSION

Applying these models, ontologies and approaches to a new type of information set can make that information, and its relevancy score results, more discoverable and of higher quality. The most critical takeaway from this work is that the advantages of semantic standards and reasoning can be leveraged upon analytic provenance and results, providing a common query representation, eliminating the need for proprietary or complex combinational queries that span multiple analytical data silos. Also critically, behavioral observations expressed with DQ can be leveraged with dynamically adaptive consumer usage narratives to build powerful semantic functionality that augments traditional SPARQL queries for simplistic data extractions that were ignorant of behavior, analytics, or consumer information affinities, with new features that, essentially, enable an autonomous collaborative filtering process that is represented 100% via semantic standards.

Autonomous collaborative filtering would itself be a powerful feature, but leveraged with semantic technologies that bridge extensible pluggable ontologies, while simultaneously abstracting analytics-engine queries and personalizing information to consumer needs, could enable novel research, new information and web functionality, and act as a unifying analytics front-end.

In our future work, we will explore semantic state traceability paired with diverse analytics. Reasoning over stateful trends within segmented time periods can demonstrate possible advanced uses of semantics for stochastic, graph-based, boolean-based, or other analytics algorithms, thus producing support for personalized prioritization, query result set ordering, and provenance modeling of analytics. Additionally, the enhancements from this work could enable determinations of efficiency for different analytics, and have the potential to combine analytic-based queries with semantic queries.

REFERENCEs

[1] Bryant, J., Paulini, M., Hasseler, G., Lebo, T., "Enhancing Information Awareness through Directed Qualification of Semantic Relevance Scoring Operations", ICCRTS, 2014

[2] Tous, R., Delgado, J., "A Vector Space Model for Semantic Similarity Calculation and OWL", IN DEXA, 2006.

[3] Ding, C., He, X., Husbands, P., Zha, H., Horst, D., "PageRank, HITS, and a Unified Framework for Link Analysis", 25th SIGIR Proceedings, 2002

[4] Zhang T, Xu D, Chen J. Application-oriented purely semantic precision and recall for ontology mapping evaluation. Knowl-Based Syst 2009;21(8):794–799.

[5] Kai Zeng , Jiacheng Yang , Haixun Wang , Bin Shao , Zhongyuan Wang, A distributed graph engine for web scale RDF data, Proceedings of the VLDB Endowment, v.6 n.4, p.265-276, February 2013.

[6] P. D. Turney and P. Pantel, "From Frequency to Meaning: Vector Space Models of Semantics", Volume 37, pages 141-188, 2010.

[7] Das, SR. Bunescu, R. Mooney. "Collective Information Extraction with Relational Markov Networks". 42nd Annual Meeting of the Association for Computational Linguistics, July, 2004

[8] Perry, M., Sheth, A., Arpinar, I., "Geospatial and Temporal Semantic Analytics", Encyclopedia of Geoinformatics, 2009.

[9] Bryant, J., Paulini, M., "Making Semantic Information Work Effectively for Degraded Environments", ICCRTS, 2013.

[10] K. Sudo, S. Sekine, R. Grishman. "An Improved Extraction Patterns Representation Model for Automatic IE Pattern Acquisition". 41st Annual Meeting of the Association for Computational Linguistics, July, 2003.

[11] Lebo, T., Graves, A., McGuinness, D., "Content-Preserving Graphics", Consuming Linked Data Conference, 2013.

[12] J. Hobbs, D. Appelt, J. Bear, D. Israel, M. Kameyama, M. Stickel, M. Tyson. "FASTUS: A Cascaded Finite-state Transducer for Extracting Information for Natural-Language Text". Finite-State Language Processing. MIT Press, Cambridge, MA. 1997

ISBN: 978-1-61208-420-6