IMMM 2018

The Eighth International Conference on Advances in Information Mining and Management

ISBN: 978-1-61208-654-5

July 22 - 26, 2018

Barcelona, Spain

IMMM 2018 Editors

Dirk Labudde, Hochschule Mittweida, University of Applied Sciences, Germany

IMMM 2018

Foreword

The Eighth International Conference on Advances in Information Mining and Management (IMMM 2018), held between July 22 - 26, 2018- Barcelona, Spain, continued a series of academic and industrial events focusing on advances in all aspects related to information mining, management, and use.

The amount of information and its complexity makes it difficult for our society to take advantage of the distributed knowledge value. Knowledge, text, speech, picture, data, opinion, and other forms of information representation, as well as the large spectrum of different potential sources (sensors, bio, geographic, health, etc.) led to the development of special mining techniques, mechanisms support, applications and enabling tools. However, the variety of information semantics, the dynamic of information update and the rapid change in user needs are challenging aspects when gathering and analyzing information.

We take here the opportunity to warmly thank all the members of the IMMM 2018 Technical Program Committee, as well as the numerous reviewers. The creation of such a broad and high quality conference program would not have been possible without their involvement. We also kindly thank all the authors who dedicated much of their time and efforts to contribute to IMMM 2018. 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 are grateful to the members of the IMMM 2018 organizing committee for their help in handling the logistics and for their work to make this professional meeting a success.

We hope that IMMM 2018 was a successful international forum for the exchange of ideas and results between academia and industry and for the promotion of progress in the field of information mining and management.

We are convinced that the participants found the event useful and communications very open. We hope that Barcelona provided a pleasant environment during the conference and everyone saved some time to enjoy the charm of the city.

IMMM 2018 Chairs:

IMMM Steering Committee

Nitin Agarwal, University of Arkansas at Little Rock, USA
Andreas Schmidt, Karlsruhe University of Applied Sciences, Germany
Michele Melchiori, Università degli Studi di Brescia, Italy
Bernhard Bauer, University of Augsburg, Germany
Mehmed Kantardzic, University of Louisville, USA
Daniel Thalmann, Institute for Media Innovation (IMI) | Nanyang Technological University, Singapore
Verena Kantere, University of Geneva, Switzerland
Duarte Trigueiros, ISCTE - University Institute of Lisbon, Portugal, Portugal

IMMM 2018

COMMITTEE

IMMM Steering Committee

Nitin Agarwal, University of Arkansas at Little Rock, USA
Andreas Schmidt, Karlsruhe University of Applied Sciences, Germany
Michele Melchiori, Università degli Studi di Brescia, Italy
Bernhard Bauer, University of Augsburg, Germany
Mehmed Kantardzic, University of Louisville, USA
Daniel Thalmann, Institute for Media Innovation (IMI) | Nanyang Technological University, Singapore
Verena Kantere, University of Geneva, Switzerland
Duarte Trigueiros, ISCTE - University Institute of Lisbon, Portugal, Portugal

IMMM Industry/Research Advisory Committee

Dirk Labudde, Hochschule Mittweida, Germany
Adrienn Skrop, University of Pannonia, Hungary
Qing Liu, Data61 | CSIRO, Australia
Stefan Brüggemann, Airbus Defence and Space, Germany
Xuanwen Luo, Sandvik Mining, USA
Maria Luisa Villani, Italian National Agency for New Technologies, Energy and Sustainable Economic Development (ENEA), Italy
Emir Muñoz, Fujitsu Ireland Ltd / INSIGHT Centre at NUI Galway, Ireland
Xiang Ji, Bloomberg LP, USA

IMMM 2018 Technical Program Committee

Nitin Agarwal, University of Arkansas at Little Rock, USA
Akhlaq Ahmad, College of Engineering and Islamic Architecture - Umm Al Qura University, Saudi Arabia
Zaher Al Aghbari, University of Sharjah, UAE
Aletéia Araújo, University of Brasília, Brazil
Liliana Ibeth Barbosa-Santillan, University of Guadalajara, Mexico
Cristina Barros, University of Alicante, Spain
Bernhard Bauer, University of Augsburg, Germany
Stefan Brüggemann, Airbus Defence and Space, Germany
Erik Cambria, Nanyang Technological University, Singapore
Mirko Cesarini, University of Milan Bicocca, Italy
Nadezda Chalupova, Mendel University in Brno, Czech Republic
Zhiyong Cheng, School of Computing | National University of Singapore, Singapore
Pascal Cuxac, INIST-CNRS, Nancy, France
Mirela Danubianu, "Stefan cel Mare" University of Suceava, Romania
František Dařena, Mendel University Brno, Czech Republic
Ke Deng, RMIT University, Melbourne, Australia
Qin Ding, East Carolina University, USA
Ionut Cosmin Duta, University of Trento, Italy

Copyright Information

For your reference, this is the text governing the copyright release for material published by IARIA.

The copyright release is a transfer of publication rights, which allows IARIA and its partners to drive the dissemination of the published material. This allows IARIA to give articles increased visibility via distribution, inclusion in libraries, and arrangements for submission to indexes.

I, the undersigned, declare that the article is original, and that I represent the authors of this article in the copyright release matters. If this work has been done as work-for-hire, I have obtained all necessary clearances to execute a copyright release. I hereby irrevocably transfer exclusive copyright for this material to IARIA. I give IARIA permission or reproduce the work in any media format such as, but not limited to, print, digital, or electronic. I give IARIA permission to distribute the materials without restriction to any institutions or individuals. I give IARIA permission to submit the work for inclusion in article repositories as IARIA sees fit.

I, the undersigned, declare that to the best of my knowledge, the article is does not contain libelous or otherwise unlawful contents or invading the right of privacy or infringing on a proprietary right.

Following the copyright release, any circulated version of the article must bear the copyright notice and any header and footer information that IARIA applies to the published article.

IARIA grants royalty-free permission to the authors to disseminate the work, under the above provisions, for any academic, commercial, or industrial use. IARIA grants royalty-free permission to any individuals or institutions to make the article available electronically, online, or in print.

IARIA acknowledges that rights to any algorithm, process, procedure, apparatus, or articles of manufacture remain with the authors and their employers.

I, the undersigned, understand that IARIA will not be liable, in contract, tort (including, without limitation, negligence), pre-contract or other representations (other than fraudulent misrepresentations) or otherwise in connection with the publication of my work.

Exception to the above is made for work-for-hire performed while employed by the government. In that case, copyright to the material remains with the said government. The rightful owners (authors and government entity) grant unlimited and unrestricted permission to IARIA, IARIA's contractors, and IARIA's partners to further distribute the work.

Table of Contents

Customer Demands of FinTech in Taiwan

Chih-Chuan Chen
Interdisciplinary Program of Green and Information Technology
National Taitung University
Taitung, Taiwan, R.O.C.
e-mail: ccchen@nttu.edu.tw

Hui-Chi Chuang
Institute of Information Management
National Cheng Kung University
Tainan City, Taiwan, R.O.C.
e-mail: huichi613@gmail.com

Ming-Hui Pai
Department of International Business Management
Tainan University of Technology
Tainan City, Taiwan, R.O.C.
e-mail: joyhappy432000@yahoo.com.tw

Yi-Chung Cheng
Department of International Business Management
Tainan University of Technology
Tainan City, Taiwan, R.O.C.
e-mail: t20042@mail.tut.edu.tw

Sheng-Chun Pai
Taiwan Certificate Authority, Inc.
Taipei, Taiwan, R.O.C.
e-mail: a365373@hotmail.com

***Abstract*—In recent year, FinTech advances have been rapidly transforming every aspect of our daily life through financial services. Advancing technologies, evolving customer expectation, and changing regulatory frameworks are opening doors to disruptive innovation in financial services. In Taiwan, although the government has announced 2015 as the year of FinTech, the progress of FinTech development is still slow as compared to its neighboring countries, such as Hong Kong, Singapore and China. This study aims to identify the critical success factors for FinTech's development in Taiwan based on customers' acceptance of FinTechs and innovative financial services. A modified Delphi method was adopted in order to obtain the most reliable opinion consensus of a group of 28 experts by subjecting them to a series of questionnaires, which was designed to extract customers' FinTech acceptance with respect to the six core functions of financial services, namely, payments, insurance, deposits & lending, capital raising, investment management, and market provisioning. The results showed that FinTechs of payment are more accepted by the customers in Taiwan. The top four payment options are highly approved by the customers with scores higher than 4.0. These four payment options are very popular in China. Therefore, the experts all agree that they are promising in Taiwan.**

Keywords- fintech; modified Delphi method; taiwan fintech industry

I. INTRODUCTION

In the recent years, Financial Technology (FinTech) has become a popular term in a wide range of operations for enterprises or organizations thanks to the advances in technology and business innovation, as well as the growing market expectations, cost-saving requirement and customer demands. Most competitive financial firms are considering FinTech as one of their major investments [1]. The expanding of the scope of FinTech applications has resulted in great challenges in adoptions and planning. Could higher economic growth happen? Will the new FinTechs make the institutions and markets more efficient and effective? In 2015, also known as the year of FinTech in Taiwan, the Financial Supervisory Commission in Taiwan (TFSB) proposed the "Creation of digitized financial environment 3.0" and it is started the year of FinTech in Taiwan. The Taiwanese government also setup a grand office of financial technology to promote the FinTech transforming action of the finance institutions, aiming to loosen restrictions on online banking and its applications [2][3]. However, the FinTech development is still slower as compared to its neighboring countries, such as Hong Kong, Singapore and China [4]. Recently, the Taiwanese government has been putting efforts to speed up the FinTech development. For example, securities investment consulting enterprises providing a Robo-Advisor service are allowed to execute automated "re-balance transactions" for their clients through their computer systems under specific conditions [5]. The regulations on electronic payment have been relaxed [6]. However, it is still long way to go to fulfill the goals of the blueprint of the Financial Supervisory Commission. It is an urgent task for the government, industry and academy to explore these factors and this leads to the motivation of this study.

The key factors to FinTech success are yet to be discovered and their influences on FinTech application need to be thoroughly understood. The goal of this study is to identify the critical success factors for FinTech's development in Taiwan based on customers' acceptance of FinTechs and innovative financial services. A modified Delphi method is adopted in order to obtain the most reliable opinion of experts in this field.

In the final report of the 2015 World Economic Forum about the future of the financial services, they have

 ISBN: 978-1-61208-654-5

structured a framework of six financial services, namely, Payments, Insurance, Market Provisioning, Deposits & Lending, Investment Management, and Capital Raising [7]. The Taiwanese Financial Supervisory Committee also includes the six financial functions in the strategic framework [6]. Therefore, this study aims to identify the critical key factors of FinTech based on these six functions, with the help of experts in finance.

A modified Delphi method was adopted in order to understand, from the point view of the experts, how the Fintechs are acceptable by the customers with respect to the six financial functions. The results include the key factors to FinTech success and the acceptability of the six financial functions, respectively.

II. LITERATURE REVIEW

In Section 2, we defined what is "Financial technology (FinTech)" and the latest development status in Taiwan. In addition, modified Delphi Method is explained and described in detail.

A. FinTech

Financial technology (FinTech) is defined as the new technology and innovation that aims to compete with traditional financial methods in the delivery of financial services [8]. FinTech has emerged in both the developed and developing world [9], and it has unleashed a new era of competition, innovation and job creating productivity in our economy, and very worthy of encouragement. However, in addition to established competitors, FinTech companies often face doubts from financial regulators like issuing banks and the governments [10]. Another concern is data security. The threats of hacking as well as the needs to protect sensitive consumer and corporate financial data are unavoidable issues. It poses challenges for regulators, as well as for market participants, in balancing the potential benefits of innovation with the possible risks.

FinTech companies and industries are leveraging new technology to create new and better financial services for both consumers and businesses, which operate in personal financial management, insurance, payment, asset management, etc. Thanks to the increasing popularity of digital wallet in online and In-App payment, as well as at brick-and-mortar stores, customers' expectations are getting higher, that says, more technologies and applications are added to the financial services, such as payable on demand, smart assistant, virtual reality. Based on Ventures Scanner's report [11], FinTech sector is organized 16 categories. They are, Banking Infrastructure, Business Lending, Consumer and Commercial Banking, Consumer Lending, Consumer Payments, Crowdfunding, Equity Financing, Financial Research and Data, Financial Transaction Security, Institutional Investing, International Money Transfer, Payments Backend and Infrastructure, Personal Finance, Point of Sale Payments, Retail Investing, and Small and Medium Business Tools.

Also, as mentioned earlier, in the final report of the 2015 World Economic Forum about the future of the financial services, they have structured a framework of six financial services, namely, Payments, Insurance, Market Provisioning, Deposits & Lending, Investment Management, and Capital Raising [7].

B. FinTech in Taiwan

In Taiwan, the banking sector is highly supported yet tightly regulated by the government. In September 2015, Taiwan's Financial Supervisory Commission (FSC) officially announced the establishment of its FinTech Office, a platform responsible for the planning and promotion of FinTech developments, which particularly focuses on topics including digitalization of the financial environment, mobile payment, third party payment, Internet financing (peer-to-peer lending), online investment, and the Internet-of-Things, among others [2][3]. The FSC declared 2015 as the year of FinTech. A series of actions were taken to promote FinTech development in Taiwan, such as requesting all domestic banks to offer online financial services, initiating big data application projects for the banking industry [12].

A FinTech Development Strategy White Paper released in 2016 by the FSC detailed the authority's strategy for FinTech development. The white paper followed World Economic Forum and planned the major development dimensions of six financial services, namely, Payments, Insurance, Market Provisioning, Deposits & Lending, Investment Management, and Capital Raising [7]. Also, it noted a number of objectives for the years to come:

1. Double the ratio of e-payment within five years from the present ratio of 26% via public promotion and private sector participation;
2. Promote blockchain technology and the establishment of a special task force in the Bankers' Association for research on applications of the technology;
3. Support finance innovation efforts of startups through the Financial Technology Development Fund and provide coaching;
4. Create a world-class incubation center for fintech innovations;
5. Allow financial institutions to invest in 100% shares of fintech companies through reinvestment
6. Encourage the use of tokenization technology for virtual and physical cards;
7. Raise the percentage of e-orders to 70% and promote e-service of securities firms and robo-advisors;
8. Encourage insurtech development;
9. Develop physical and virtual branches of financial institutions to achieve the diversification of service providers and multiple access points of their services, as well as improve the existing facilities of financial institutions;
10. Create an integrated, secure online ID verification mechanism.

C. Modified Dephi Method

The modified Delphi method is a kind of expert prediction method, and the purpose is to integrate the knowledge and experience of experts in the field for a specific issue [13][14]. It makes the results to achieve consensus among many experts through specific procedures

 ISBN: 978-1-61208-654-5

and steps. In the past, the traditional Delphi method is often time-consuming and labor-intensive due to the multiple round-trips of the questionnaire. It causes the members to reduce motivation to finish the questionnaires and leads to low recovery rate of questionnaires. In summary, these reasons will make the results lose authenticity and get little significance. Therefore, the modified Delphi method, which captures the features and advantages of Delphi method has been developed [13][14]. It simplifies the process of complicated questionnaires. The common way is to omit the first round of open ended consultation questionnaire to acquire expert opinions. Then, formulate the questions based on the relevant research results or the experience of researchers. Then, the experts are asked to express their personal opinions according to the proposed project. This correction method can avoid the difficulty in answering the open questionnaire and reduce the influence of low questionnaire recovery rate [15][16].

III. METHODOLOGY

In the beginning, we proposed some questions which are on the basis of existing literatures and domain experts' suggestions. The questionnaire is to ask the customers' acceptance of FinTechs and it includes six dimensions which are Payments, Insurance, Market Provisioning, Deposits & Lending, Investment Management, and Capital Raising. For example, "What do you think about that people accept the mobile Payment?", "What do you think about that people accept the third party Payment?", "What do you think about that people accept the online insurance?", etc. The key items of questionnaire are showed on the next section.

The first round of the traditional Delphi method is based on the experience of experts to answer open-ended questionnaires to provide information related to the research topics as the basis for the second round of questionnaire design. It is time-consuming and labor-intensive, therefore, the modified Delphi method is adopted to skip open-ended questionnaires provided by expert's experience and then replacing by relevant research reviews. In order to improve the efficiency of the research, the modified Delphi method is used in this paper. The modified Delphi method process is showed in Figure 1.

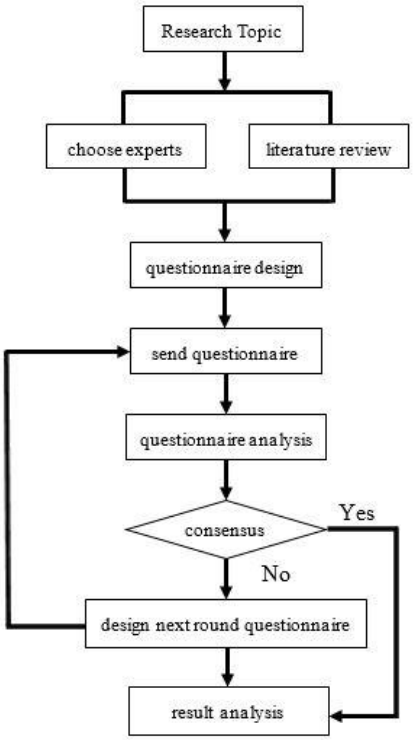


Figure 1. The modified Delphi method process

Delbecq et al. [17] indicate that ten to fifteen people could be sufficient if the background of the modified Delphi subjects is high homogeneous. In contrast, if various reference groups are involved in the modified Delphi method, we need to invite more people to join. If there are more than 13 experts in modified Delphi process, the reliability of result will be higher than 80% [18]. Based on the principles of expert selection, we invited four expert group which included 28 scholars and experts related to innovation in financial science and technology. The selected experts included senior financial experts, law & legal experts, senior corporate executives, scholars. After explaining the purpose of this study to the group of experts, they help fill out of the modified Delphi questionnaire.

IV. QUESTIONNAIRE ANALYSIS

This research aims to explore the customer's acceptance of FinTech from financial experts' point of view by adopting a modified Delphi approach. The modified Delphi method is an appropriate method that allows gathering consensual subjective judgement from a group of experts [19]. For this study, a panel consists of 28 selected FinTech experts from four distinct categories, namely, senior executives in finance, legal experts, senior executives in general, and academic experts. The questionnaire is designed based on the six dimensions of Payments, Insurance, Market Provisioning, Deposits & Lending, Investment Management, and Capital Raising.

The modified Delphi study composes three rounds. During each round, the researchers clearly explained to the

 ISBN: 978-1-61208-654-5

experts one by one about the purpose of this study. Quartile Deviation is used to determine if the consensus of all of the 28 experts is reached.

The data were examined thematically and the themes were ranked by their frequencies. The results of the analysis are shown in Table I. The acceptance score is the average score of all experts and the ranking is the sorting according to acceptance score. One can see that the experts agree that in Taiwan, customers are mostly familiar to the FinTech applications with respect to payment. The most popular FinTech application is using cellphone for payment transfer. It makes sense since mobile transfer payment is one of the services provided by all of the banks in Taiwan. The number two application is WebATM, which is also a common service in Taiwan. Number three is mobile payment, such as App, QR code, digital wallet. Mobile payment is very popular in China, but it is still at the development stage in Taiwan. Thanks to its success in China, all experts are confident in its acceptance in Taiwan. Two FinTech applications tie at rank number four. They are third party payment and digital wallet, such as Apple Pay. All the results are shown in Table I.

TABLE I. RANKING OF THE CUSTOMER ACCEPTANCE OF FINTECH

No.	FinTech Application	Acceptance	Ranking
Payment			
1	Mobile Payment	4.1	3
2	Third Party Payment	4.0	4
3	Biormetric Banking	3.9	5
4	Mobile Transfer	4.6	1
5	Virtual Card	4.0	4
6	Cashless Life	3.8	6
7	WebATM	4.3	2
8	Virtual Currency	2.9	12
Insurance			
1	Online insurance	3.6	8
2	P2P Friendsurance	3.4	10
Deposits & Lending,			
1	P2P Internet Lending	3.3	11
2	P2P Internet Exchange	3.6	8
3	VTM (Virtual Teller Machine)	3.6	8
4	Cloud Account	3.6	8
5	Self-ServiceBanking	3.5	9
6	Mobile loan	3.6	8
Capital Raising			
1	Online fundraising	3.3	11
Investment Management			
1	Online Invest Management	3.9	5
2	Robo-advisor	3.7	7
Market Provisioning			
1	Big Data and AI for Searching Market Information	3.9	5

V. DISCUSSION AND CONCLUSIONS

This study aimed to explore the FinTech acceptance of customers based on the opinions of domain experts. A three round modified Delphi approach was adopted with a panel of 28 FinTech experts from four different categories including senior executives in finance, legal experts, senior executives in general, and academic experts. The questionnaire is designed according to six FinTech dimensions of Payments, Insurance, Market Provisioning, Deposits & Lending, Investment Management, and Capital Raising.

The results showed that all the experts agreed that the top four of the most popular FinTech applications are mobile transfer, WebATM, mobile payment, third party payment and virtual card. All top four applications have acceptance scores greater than 4.0, which indicates high acceptance by the Taiwanese customers. The results reflect the current FinTech environment in Taiwan, which can be used for FinTech industry and the government to make strategy for FinTech development in Taiwan.

This study is still at tentative. In the future, more FinTech applications will be included, and more experts will be interviewed and inquired.

ACKNOWLEDGMENT

We would like to appreciate those who provided insight and expertise to greatly assist this research.

REFERENCES

[1] R. Wigglesworth, Fintech: Search for a super-algo. Financial Times, 20, 2016.

[2] L. L. Wang, T. M. Huang, M. C. Lee, Digital Financial Environment Building Program Fully Activated. In D. o. Planning (Ed.), Financial Output Monthly. Taipei, Taiwan: Financial Supervisory Commission, Taiwan, pp. 5-8, 2015a.

[3] L. L. Wang, T. M. Huang, M. C. Lee, FSC establishes the Financial Technology Office to promote the development of financial technology. In D. o. Planning (Ed.), Financial Output Monthly. Taipei, Taiwan: Financial Supervisory Commission, Taiwan, pp. 5-8, 2015b.

[4] M. C. Chen, S. S. Chen, H. M. Yeh, W. G. Tsaur, "The key factors influencing internet finances services satisfaction: An empirical study in Taiwan," American Journal of Industrial and Business Management, vol. 6, no. 6, pp. 748, 2016.

[5] L. L. Wang, T. M. Huang and M. C. Lee, Securities investment consulting enterprises providing a Robo-Advisor service are allowed to execute automated "re-balance transactions" for their clients through their computer systems under specific conditions. In D. o. Planning (Ed.), Financial Output Monthly. Taipei, Taiwan: Financial Supervisory Commission, Taiwan, pp. 5-8, 2017.

[6] Financial Supervisory Commission R.O.C (Taiwan), FinTech Development Strategy White Paper. Taipei, Taiwan: [Online]. Available from:https://www.fsc.gov.tw/ch/home.jsp?id=517&parentpath=0,7, 478. 06, 2018.

[7] World Economic Forum, The Future of Financial Services: How disruptive innovations are reshaping the way financial services are structured, provisioned and consumed. Ginebra: FEM. Consultado en: http://www3. weforum. org/docs/WEF_The_future__of_financial_services. pdf. 06, 2018.

[8] T. C. Lin, "Infinite financial intermediation," Wake Forest L. Rev., vol. 50, pp. 643, 2015.

[9] D. W. Arner, J. Barberis and R. P. Buckley, "The evolution of Fintech: A new post-crisis paradigm," Geo. J. Int'l L., vol. 47, pp. 1271, 2015.

[10] Business Insider, India, Groundbreaking FinTech Innovations: Threat for banks or opportunity of a lifetime? [Online]. Available from: https://www.businessinsider.in/Ground breaking-FinTech-Innovations-Threat-for-banks-or-opportunity-of-a-lifetime/articleshow/61682406.cms, 06, 2018.

[11] Venture Scanner, Financial Technology Market Overview—Q4, 2016.

 ISBN: 978-1-61208-654-5

[12] J. L. Hung and B. Luo, "FinTech in Taiwan: a case study of a Bank's strategic planning for an investment in a FinTech company," Financial Innovation, vol. 2, no. 1, pp. 15, 2016.

[13] B. H. Eubank, N. G. Mohtadi, M. R. Lafave, J. P. Wiley, A. J. Bois, R. S. Boorman and D. M. Sheps, "Using the modified Delphi method to establish clinical consensus for the diagnosis and treatment of patients with rotator cuff pathology," BMC Medical Research Methodology, 2016. http://doi.org/10.1186/s12874-016-0165-8

[14] D. Khodyakov, S. Grant, C. E. Barber, D. A. Marshall, J. M. Esdaile and D. Lacaille, "Acceptability of an online modified Delphi panel approach for developing health services performance measures: results from 3 panels on arthritis research," Journal of evaluation in clinical practice, vol. 23, no. 2, pp. 354-360, 2017.

[15] R. L. Custer, J. A. Scarcella and B. R. Stewart, "The modified Delphi technique-A rotational modification," Journal of Career and Technical Education, vol. 15, no. 2, pp. 50-58, 1999.

[16] C. C. Hsu and B. A. Sandford, "The Delphi technique: making sense of consensus," Practical assessment, research & evaluation, vol. 12, no. 10, pp. 1-8, 2007.

[17] A. Debecq, A. H. Van de Ven and D. H. Gustafson, Group techniques for program planning. Glenview, Illinois: Scott, Foresman and Company, 1975.

[18] N. C. Dalkey, The Delphi method: An experimental study of group opinion. Santa Monica, CA: RAND Corporation, 1969.

[19] V. Mahajan, "The Delphi method: Techniques and applications," JMR, Journal of Marketing Research, vol. 13, no. 3, pp. 317, 1976.

 ISBN: 978-1-61208-654-5

A Study of Exploring the Factors on Women's Choice of Dessert

A-Mei Lin Hsu
Ph.D. Program of Technology Management
Chung Hua University
Hsinchu City, Taiwan, R.O.C.
e-mail: t90094@mail.tut.edu.tw

Hui-Chi Chuang
Institute of Information Management
National Cheng Kung University
Tainan City, Taiwan, R.O.C.
e-mail: huichi613@gmail.com

Chiu-Chi Wei
Dept. of Industrial Management
Chung Hua University
Hsinchu City, Taiwan, R.O.C.
e-mail: a0824809@gmail.com

Yi-Chung Cheng
Department of International Business Management
Tainan University of Technology
Tainan City, Taiwan, R.O.C.
e-mail: t20042@mail.tut.edu.tw

***Abstract*—The feeling of happiness can affect people's choices. In order to raise the purchasing power, it is a common way to advertise and promote the product with happiness. For example, parent-child wedding and car advertisement pursue the goal of family happiness. Café uses the feelings of happiness as marketing theme to attract in-love couples. However, happiness is a subjective and emotional which is affected by temporal focus. There are two primary experiences of happiness, namely, high awakening (excitement) happiness and low awakening (peace) happiness. This study aims to investigate how the choice of dessert of a couple is affected by the loving atmosphere with temporal focus as well as happiness. It uses two dimensions, namely, temporal focus (present, future) and loving atmosphere (passionate love, not passionate love) to conduct experiments with questionnaires. Experimenters are randomly assigned to four experiments for verifying research hypotheses. The experimental result shows that loving atmosphere affects temporal focus, temporal focus affects happiness, and happiness affects choice of dessert.**

Keywords- love atmosphere; time focus; happiness

I. INTRODUCTION

It has been shown that consumer behaviors and psychology are highly related. The decisions of consumers for purchasing products are not only based on the services and the qualities, but also affected by their feelings and emotions. In other words, times, environment, atmosphere and location will cause various purchasing motivations [1]. There is a relationship between happiness and consuming behaviors, which gives chances for enterprises to make advertisement by taking advantages of happy images, such as advertisement of cars, real estate, furniture, coffee, even TV commercial for oden, a Japanese one-pot winter dish. Happiness changes along with the shifts of temporal focus. Barrett [2] and Russell and Barrett [3] defined happiness as high awakening (excitement) happiness and low awakening (peace) happiness, and here both are positive emotions. When happiness of a person is in different level of awareness, his or her purchasing motivations are different. Happiness also affects the feelings of loving atmosphere. Nowadays, young people prefer instant love which is called "fast food love". We can see couples post their relationship status on Facebook, such as "keep in love", "celebrate keeping in love for one week", "celebrate keeping in love for two weeks" or "celebrate keeping in love for a month" and so on. Soon after, we may see someone post the relationship status as "single" on Facebook. Therefore, the happiness of loving atmosphere is affected by temporal focus. However, how the length of keeping in love affects the temporal focus is an important issue in this research. First, we consider the number of weeks that young people had been falling in love and then we use the median of the numbers as the cut point of high awakening (excitement) happiness and low awakening (peace) happiness. Will women in high awakening (excitement) happiness purchase exciting dessert and in low awakening (peace) happiness purchase peaceful dessert? In the second study, we choose the top 10 desserts from the Internet and we invite young people to rate each dessert from 1 to 5. The dessert which got the highest score is marked as "exciting dessert", and on the other hand, the dessert with the lowest score is marked as "peaceful dessert".

The study takes two dimensions, namely, temporal focus (present, future), and loving atmosphere (passionate love, not passionate love) to conduct experiments and questionnaires. Experimenters are randomly assigned to four experiments. First of all, couples show their recent photos and share the stories of the photos, such as shooting location or travel memories in order to recall truly loving atmosphere. The experimenters were asked two independent questions and two test questions to confirm if the loving atmospheres were successfully reminded. After that, they filled out the form of questionnaires about happiness and choice of dessert.

The study includes 200 women in loving atmosphere as the experimenters. Questionnaires are based on Likert 5-point scale and the questions are:

 ISBN: 978-1-61208-654-5

1. Is happiness affected by temporal focus and loving atmosphere?

2. How does happiness affect the choice of dessert?

II. LITERATURE REVIEW

Consumer behaviors are closely related to psychology. It is well known that consumers' behaviors and purchasing motivation are inseparable. However, it is rarely discussed if in-love happiness influences purchasing motivations and behaviors. In their study, Trope and Liberman [4][5] mentioned that time focus affects one's view and action, such as indulge desire [6] and purchase decision [7]. In summary, there is no literature that discusses the loving atmosphere and happiness, how to cast influence on purchasing behavior, and the interaction between temporal focus and loving atmosphere. Therefore, this research is motivated to address these issues by investigating how the choice of dessert of a couple is affected by the loving atmosphere with temporal focus as well as happiness.

The word happiness has many meanings. It can be considered as a broad term for feeling good. Myers and Diener [8] defined happiness as "a healthy, happy and satisfied state, which is also a pleasant or satisfying experience." Myers and Diener [8] and Layard [9] claimed that the meaning of happiness is the same for everyone, which means happiness is unity; however, other researchers believe happiness is subjective. Happiness does not mean the same to each other and happiness is different [10]. Furthermore, Barrett [2], Russell and Barrett [3] separate happiness in two types: high awakening and low awakening. The definition of the former is excited, joyful, elated and passionate; the meaning of the latter is calm, quiet, peaceful and ordinary. Loving atmosphere refers to the feelings of happiness of the couple who are dating. Therefore, the happiness found in loving atmosphere could be an experience, a status, or a reaction.

The sense of happiness is a subjective enjoyment, feeling and experience, but it is an objective positive emotion or mood. Carstensen et al. [11] proposed that age is a potential psychological factor for happiness and is defined as temporal focus. Young people have longer time to create their own future, but when growing older, they do not pay more attention to the future and gradually focus on the moment. Couples who are in love would also have different subjective feelings about happiness based on the time when they interact with each other, and thus show different objective behaviors for happiness. Is it because young people who experience a loving atmosphere can focus more on the future? Will young people, who have been in a long-term loving atmosphere, tend to focus on the present? This study will define the temporal focus in love by designated experiment.

Argyle suggested that the sense of happiness includes two levels of emotions and cognition [12], and in the study it verifies that positive emotions indirectly influence consumers' choices by affecting individual cognition. In the field of marketing, it is broadly accepted that happiness will affect consumers' behavior. Therefore, many advertisements take advantage of the sense of happiness to attract the consumers, for example, advertising for cars, real estate, coffee, and even oden, the traditional Japanese food. All of these demonstrate that consumers' happiness will affect their choices of products. In the research of Isen and Patrick, they found that positive emotions would affect many choices of life with optimism [13], which proves that when people are in positive emotions, they would make healthy choices. That is, people with positive emotions would be more optimistic. With such attitude, people focus more on what are expected to happen in the future, but not pay attention to the immediate concerns.

Couples in loving atmosphere are affected by unique and distinct happiness, and often have various positive emotions. Different positive emotions are accompanied with different senses of happiness, and moreover, positive emotions affect choices [13]. Therefore, we can conclude that happiness plays an intermediate role in the relationship between love and dessert selection.

III. RESEARCH METHOD

In this section, we illustrate and display the research framework in detail, which is shown in Figure 1.

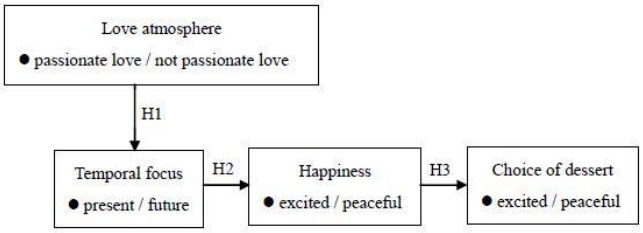


Figure 1. Research hypothesis framework.

A. *The research hypotheses are as followings:*

H1: loving atmosphere affects temporal focus

H2: temporal focus affects happiness

H3: happiness affects choice of dessert

B. *Research variables and operational definition*

1) Loving atmosphere

It should distinguish "passionate love" and "not passionate love" in order to understand the strength of loving atmosphere.

2) Temporal focus

This research focuses on the definition of the individual's subjective view of present and future. In the study, we adopt the item to measure the time by Mogilner et al. [14]. The item "I think the future is more important than now" uses a 5-point scale. "1 means strongly disagree" and "2 means disagree", "3 represents normal", "4 represents agree", and "5 represents strongly agree". The answer strongly disagree or disagree represents temporal focuses on "now". The answer "strongly agree" or "agree" represents temporal focuses on the "future". If the answer is 3, it means that the questionnaire is invalid because it cannot be used to distinguish the focus of time and provide valuable information.

 ISBN: 978-1-61208-654-5

3) Happiness

The definition of happiness in this study is reflecting two primary experiences of happiness, namely, excited (high awakening) and peace (low awakening). Mogilner et al. [14] proposed the definition of happiness, the item entitled "I feel that a colorful life is happier than a calm and stable life", using a 5-point scale.

4) Dessert

Desserts are rated from exciting to peaceful. First, we picked 10 different kinds of dessert from the internet. The questionnaire is "This dessert makes me excited and happy", with the answer of Likert 5-point scale.

C. Research method

In order to distinguish the strength of the loving atmosphere, in the beginning, we investigate the length of time in-love of the experimenters. According to the data, the median number of weeks is adopted as the cutting point to distinguish "passionate love" and "not passionate love ".

The 10 desserts are selected from internet, which are familiar and favorable desserts to the local people (Table I). Every experimenter was asked the question, "Does this dessert make me excited and happy"; this can help us define the most exciting dessert and the most peaceful dessert.

We adopt different temporal focuses and loving atmospheres to discuss the impact of happiness on the choice of desserts. The experiment takes two dimensions, namely temporal focus (present, future) and loving atmosphere (passionate love, not passionate love) to conduct the experiments. Four questionnaires are designed. Respondents are randomly assigned to four experiments for verifying research hypotheses in this study, four different questions are used to test experimenters to evoke the loving atmosphere, and to confirm that the loving atmosphere is valid. Finally, experimenters were required to answer the questions about their status of happiness and choice of desserts.

IV. EXPERIMENT AND DISCUSSION

In order to distinguish the intensity of loving atmosphere, in the study, we investigated the experimenter's during the length of their love relationship. Then the median of the length is located and used as cutting point of the loving atmospheres of "passionate love" and "not passionate love". In the beginning, we did the pre-testing which is focused on young people aged 20-30, and a total of 200 questionnaires were collected, including 29 invalid questionnaires and 171 valid questionnaires. According to the study, the average age is 24 years old. The average in-love occurrence is 2, the average endurance of in-love relationship is 48.57 weeks, and the median is 32 weeks. Therefore, the loving atmosphere with respect to the length in time of in-love relationship defined as "passionate love" for those more than or equal to 32 weeks, and "not passionate love" for those less than 32 weeks.

TABLE I. TEN SELLECTED DESSERTS

Dessert name	Picture	Dessert name	Picture
I soufflé		VI Red bean purple rice tangyuan	
II Macaron		VII Almond tofu	
III Sesame paste		VIII mille-feuille	
IV white fungus with crystal sugar		IX Crème brûlée	
V Mille Crepe Cake		X Longan walnut cake	

The study uses Likert 5-point scale to calculate the rank for desserts; the highest scores represent exciting dessert, and the lowest scores represent peaceful dessert. According to the survey results, mille crepe cake with the highest average score of 4 is an exciting dessert, and longan walnut cake with the lowest average score of 3 is a peaceful dessert.

In this study, we adopt different temporal focuses and loving atmosphere to evaluate the impact of happiness on the choice of desserts. Two dimensions, namely, loving atmosphere (passionate love, not passionate love) and temporal focus (present, future) were used to conduct the experiments. There was a total of four questionnaires and the subjects were randomly assigned to four experiments for verifying hypotheses. The loving atmosphere is based on how they share their loving photos and stories,

From Table II, the number of young people who are passionately in love and focusing on the present is 10, and the number of those who are passionately in love and focusing on the future is 16. It shows that young people who are passionately in love put their focus on the future. The number of young people who are not passionately in love but focusing on the present is 19, and the number of those who are not passionately love but focusing on the future is 15. It shows that young people who are not passionately in love put their temporal focus on the present. Therefore, the hypothesis H1: loving atmosphere affects temporal focus, is supported.

TABLE II. LOVING ATMOSPHERE V.S. TEMPORAL FOCUS

love atmosphere / time focus	Passionate love	Not passionate love	total
present	10	19	29
future	16	15	31
total	26	34	60

Table III shows that 19 young people whose temporal focus is on the present and happiness are peaceful, and 10 young people whose temporal focus is on the present and happiness are excited. Therefore, these young people put their temporal focus on the present, and their happiness is peaceful. There are 9 young people whose temporal focus is on the future and their happiness is peaceful, while there are

 ISBN: 978-1-61208-654-5

22 young people whose temporal focus is on the future and their happiness is excited. We adopt Chi-square test to measure the goodness and fitness of the model. The statistic result is significant (p<0.05), and it means the model has high fitness to verify the analysis result. Therefore, the hypothesis H2: temporal focus affects Happiness, is supported.

TABLE III. TEMPORAL FOCUS VS. HAPPINESS

temporalfocus / happiness	present	Future	total
peaceful	19	9	28
excited	10	22	32
total	29	31	60

Table IV shows that the number of young people whose happiness is peaceful choosing a peaceful dessert is 18, and the number of those whose happiness is peaceful choosing an excited dessert is 10. It indicates that low awareness (peaceful) of the happiness of consumers positively affects the possibility that they would choose peaceful desserts. The number of young people whose happiness is excited and choosing a peaceful dessert is 7. The number of those whose happiness is excited choosing an excited dessert is 25. Then, the Chi-square test is used to evaluate the model fitness. The result is significant (p<0.05), and it means the analysis result is evidential. It indicates that high awareness of the happiness of consumers positively affects the possibility that they would choose excited desserts. Therefore, H3: happiness affects choice of dessert, is supported.

TABLE IV. DESSERT CHOICE VS.. HAPPINESS

Happiness / Dessert	Peaceful	excited	total
Peaceful (Longan walnut cake)	18	7	25
Excited(Mille Crepe Cake)	10	25	35
total	28	32	60

V. CONCLUSION

Based on the above experimental results, we obtained the following conclusion. (1) Loving atmosphere affects temporal focus. (2) Temporal focus is affected by happiness. (3) Happiness affects the choice of desserts. This result proves that people who have optimistic attitudes will make healthy choices, and they tend to focus more on the future instead of paying attention to immediate worries [13]. This is a preliminary study. In the future, we will consider more factors which can influence the decision of choice and extend the scope of experiment and increase the number of questionnaire.

ACKNOWLEDGMENT

We would like to appreciate all experimenters who provided insights and opinions to greatly assist this research.

REFERENCES

[1] E. M. Tauber, “Why do people shop?,” Journal of Marketing, vol. 36, no. 4, pp. 46-49, 1972.

[2] L. F. Barrett, “Discrete Emotions or Dimensions? The Role of Valence Focus and Arousal Focus,” Cognition and Emotion, vol. 12, no. 4, pp. 579-599, 1998.

[3] J. Russell and L. Barrett, “Core Affect, Prototypical Emotional Episodes, and Other Things Called Emotion: Dissecting the Elephant,” Journal of Personality and Social Psychology, vol. 76, no. 5, pp. 805-819, 1999.

[4] Y. Trope and N. Liberman, “Time construal and time-dependent changes in preference,” Journal of personality and social psychology, vol. 79, no. 6, pp. 876, 2000.

[5] Y. Trope and N. Liberman, “Time construal,” Psychological review, vol. 110, no. 3, pp. 403, 2003.

[6] D. Read, G. Loewenstein, S. Kalyanaraman, “Mixing virtue and vice: Combining the immediacy effect and the diversification heuristic,” Journal of Behavioral Decision Making, vol. 12, no. 4, pp. 257, 1999.

[7] J. G. Lynch Jr and G. Zauberman, “When do you want it? Time, decisions, and public policy,” Journal of Public Policy & Marketing, vol. 25, no. 1, pp. 67-78, 2006.

[8] D. G. Myers and E. Diener, “Who is happy?,” Psychological science, vol. 6, no. 1, pp. 10-19, 1995.

[9] R. Layard, Happiness: Lessons from a New Science. New York: Penguim, 2005.

[10] D. T. Gilbert, Stumbling on Happines. New York: Knopf, 2006.

[11] L. L. Carstensen, D. M. Isaacowitz, S. T. Charles, “Taking time seriously: A theory of socioemotional selectivity,” American psychologist, vol. 54, no. 3, pp. 165, 1999.

[12] M. Argyle, The psychology of happiness. Routledge, 2013.

[13] A. M. Isen and R. Patrick, “The effect of positive feelings on risk taking: When the chips are down,” Organizational behavior and human performance, vol. 31, no. 2, pp. 194-202, 1983.

[14] C. Mogilner, J. Aaker, S. D. Kamvar, “The Shifting Meaning of Happiness,” Social Psychological and Personality Science, vol. 2, no. 4, pp. 395-402, 2011.

 ISBN: 978-1-61208-654-5

Automating Semantic Analysis of Website Structures for Ontology-based Benchmarking

Conceptual Model and Implementation in Retail Banking

Nikola Vlahović, Mirjana Pejić Bach, Andrija Brljak
Faculty of Economics and Business
University of Zagreb
Zagreb, Croatia
e-mail: nvlahovic@efzg.hr

Abstract— **Companies use benchmarking to improve the efficiency of their business processes, organizational structures and response to changes in their business environment. Benchmarking incites additional effort and drain on company resources. In this paper, we present an approach that may offer new outlook in making benchmarking less costly and time consuming. The goal of this paper is to present a conceptual model of the system based on grounded theory that uses current information retrieval methods, natural language processing and available web resources to create semantic ontology of best practices in structuring web-based information content. The developed model that is the result of the described approach can be used as a benchmarking model and tool for various purposes that we will illustrate using a banking web sites case study.**

Keywords- semantic annotations; ontologies; automatic information retrieval; business applications; web mining; natural language processing.

I. INTRODUCTION

Organizations rely on business intelligence in order to better understand data they store in their information systems and data that is being generated outside the organization. Valuable information generated through deep analysis of data may be used to improve their operational efficiency, organizational structure, quality of decision making, competitive potential and overall performance. Innovative approaches in discovering new information from databases have been used over the past few decades, with substantial success. Still, new approaches are implemented in order to retain comparative advantage over their competitors. Semantic networks and ontologies have been rarely in the focus of business intelligence, even though for a long time the potential is recognized [2]. In this paper, we propose a concept of an ontology-based methodology that can be used to create semantic model of a particular problem domain or area of interest that can be used by the company to conduct comparative analysis and determine its advantage and disadvantages in relation to good practices learned and inferred from available data sources outside organization.

The goal of this paper is to present an approach to analyzing web site structure in order to create a structural semantic model for particular type of web sites that can capture current best practice in web information organization. Different organizations organize their information in different ways and discovering the most prevalent approach may indicate best practice in presentation of information. The inferred model is an ontology created through inductive process and it serves as a benchmarking tool. The model consists of nodes that are interlinked as in any semantic network. These nodes represent typical web pages that are encountered in a particular class of web pages. They are described by their topic using keywords. The relations between nodes represent the hierarchy of web pages within the web site. These structures and position of each node in the structure is determined by the inductive learning processes that take information from a collection of existing web pages as will be described in this paper. The evaluation of sample web site structures heavily relies on pattern matching, natural language processing and available English language corpuses. The main contribution of this work is the implementation of semantic web as a benchmarking tool in business intelligence of an organization while preserving acceptable level of cost, time requirements and engagement of other resources.

The rest of this paper is organized as follows. Section II describes the background on implementations of web and text mining, ontologies and semantics in managerial decision-making processes, specifically benchmarking as an important tool for managerial decision making. Section III describes the conceptual model of automated information retrieval, analysis, contextualization and creation of benchmarking model based on semantic content analysis. Section IV addresses the implementation based on revealing best practices in structuring banking web sites and describes the developed semantic model. Section V presents the discussion of presented work, points out main conclusions and presents further steps in the development of applications for the described model in various practical areas of economics and business.

II. BACKGROUND

In this Section we will explain the role of benchmarking in Business Intelligence as well as most recent developments regarding benchmarking analysis in current

 ISBN: 978-1-61208-654-5

literature with special attention to semantic web and ontologies.

A. Benchmarking and Business Intelligence

Benchmarking is the process of analyzing business processes by comparing their performance and other properties with current best practice or industry standards for particular business domain. Companies have been applying this approach as an important part of decision support to make their processes more efficient [4]. Various quantitative and qualitative methods have been used to analyze available information about current best practices, such as knowledge management, knowledge-based systems, simulation modelling, datamining, etc. The main disadvantage of the currently used and proposed methodologies is that they are either time-consuming, costly or require overwhelming amount of resources. Therefore, the processes in decision support and business intelligence aim to automate various steps of implemented methodologies such as information retrieval, modelling, or analysis. Improvement in automation of these processes can be achieved with sufficient organization of information. This is one of the main reasons why semantic networks and ontologies have been identified as technologies that would greatly benefit business intelligence [2].

B. Related work

Over the past couple of decades there have been very few implementations of semantic web and ontologies in decision making [1]. Available implementations usually pertain to general web information and web services [6] published in traditional web sites with the purpose of adding a level of semantics in order to enhance search procedures [7]. Other implementations are concerned with the organization of specific expertise knowledge in various fields such as genetics [8], molecular biology [9], but also disaster management [1], e-governance and public data [10], project management [11] and social networking [12]. Special importance is given to the development of ontologies that are dedicated to syntactical information for various languages, such as lexical databases of English, German and other languages [13]. Lexical databases are often incorporated in other (semi)-automated systems for information retrieval as they can add the syntax layer to the retrieval procedure improving natural language processing significantly. Most of these ontologies are manually created to ensure correctness of concepts [14]. Manually created ontologies suffer from similar disadvantages as other methodologies implemented in business intelligence domain (high cost, time consuming, labor intensive, etc). On the other side there are informal ontologies that can be created in semi-automatic way either through volunteered information retrieval by virtual communities or by programmable information retrieval and analysis. Just in the last five years some implementations that are based on ontologies and semantic web have been proposed and developed [4] [5] that use semi-automated procedures.

In recent years, several tools have also been developed for construction of ontological databases, semantic networks and taxonomies. Some of the examples include Ontolearn [15], OntoLT [16], SOAT [17] and TextOntoEx [18]. These tools can be used to create ontologies from natural language texts and serve as a link between linguistics and ontology engineering. SOAT is created to use Chinese language corpus while the rest of the tools use English corpus. Neither of these tools, though, include functionalities specific for benchmarking, i.e., created models do not allow for visualizations and comparative exploration tools that is typical for business intelligence tools. In order to increase the variety of possible applications and problem domains, and to redefine ontological models to serve as benchmarking tools for decision making in business organizations, we will propose a specific methodology. This methodology is based on ontologies and several other technologies that can be implemented in business environment and improve decision making and managerial planning tasks. Specifically, we will concentrate on automated information retrieval from web sources and analysis based on the iterative creation of benchmarking model that can provide insight in current state of the organization of web information in particular industry or business domain.

III. AUTOMATED INFORMATION RETRIEVAL, ANALYSIS AND ONTOLOGY MODELLING

In this Section we will describe conceptual model of automated information retrieval, analysis, contextualization and creation of benchmarking model based on semantic content analysis. Key aspects of using existing lexical databases will be described as well and the algorithm of evaluating and improving the structure of the semantic model during the learning phase of the model creation process.

A. Best Practice Ontological Model

The ontological model is based on inductive synthesis approach. This is an approach originally implemented in automatic software application development based on second order logic [3]. In the case of ontology induction, the set of software specifications is replaced by a set of individual structures, so the goal of the algorithm is to create a generic ontological model that can describe all of the most typical, prevalent and semantically justified properties contained within each instance of the set. For the purpose of this paper, the structures used refer to individual web sites and the organization and structuring of their individual web pages based on the information content published. In order to automate this task, a complex algorithm was developed that goes through several different phases (Figure 1).

 ISBN: 978-1-61208-654-5

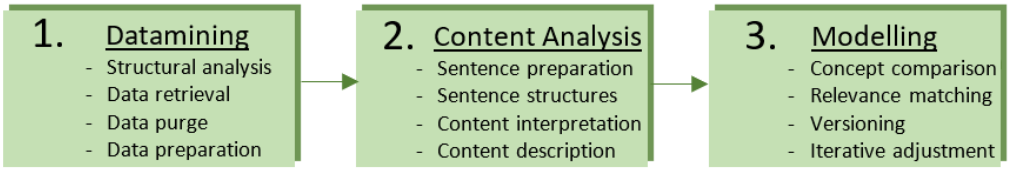


Figure 1. Overview of phases and steps in automated informationretrieval, analysis and ontology modeling

First is the stage of datamining activities Initially, information retrieval is conducted for each resource. Based on the initial structure hierarchy of constituent elements is analyzed. In case of web sites, retrieved navigation is analyzed so that each subpage within the web site structure can be accessed. Then, the rest of the content for each structure can be retrieved. Depending on content retrieved, irrelevant information or code is removed during data purge. In this case most of the HTML code that does not pertain to content itself is removed. Finally, data is prepared for the next stage of the analysis that involves Natural Language Processing (NLP) methods and procedures.

While it is necessary to perform first stage of the process online, second stage can be performed offline since all of the prepared data is stored in local database. In this stage the system still has to access online resources that include language lexicons in order to perform language analysis of the content. Firstly, content of each page is divided into sentences. The structure of each sentence is determined during syntax analysis (described in next section). Depending on the structure and syntactical role of words, a set of potential keywords is determined – token words. For each token word meaning and role is determined using lexical databases, in order to assess the interpretation of content. Here, it is important to consult language thesaurus, determine the definite meaning of potential keywords and create set of keywords that best describe the content of each page. Meanings that are most common to the set of token words are used to resolve disambiguation of any particular token word that has more than one acceptable meaning.

Finally, in the third stage of the process each page is represented by its content description that is then compared to existing concepts in the ontological model. Depending on the result of this analysis further steps in changing the ontological model are determined depending on the relevance of the page to the current model. Before any change is committed to the model versioning of the model is stored in order to provide insight in the evolution of the model or to manually select the most appropriate model for benchmarking. If there are new concepts introduced into the model additional tuning of the model may be performed at the last step of this activity. This process is repeated for each analyzed structure that will shape ontological model,

Based on this process a set of tools and appropriate interfaces were developed. The architecture of the completed system is given in Figure 2.

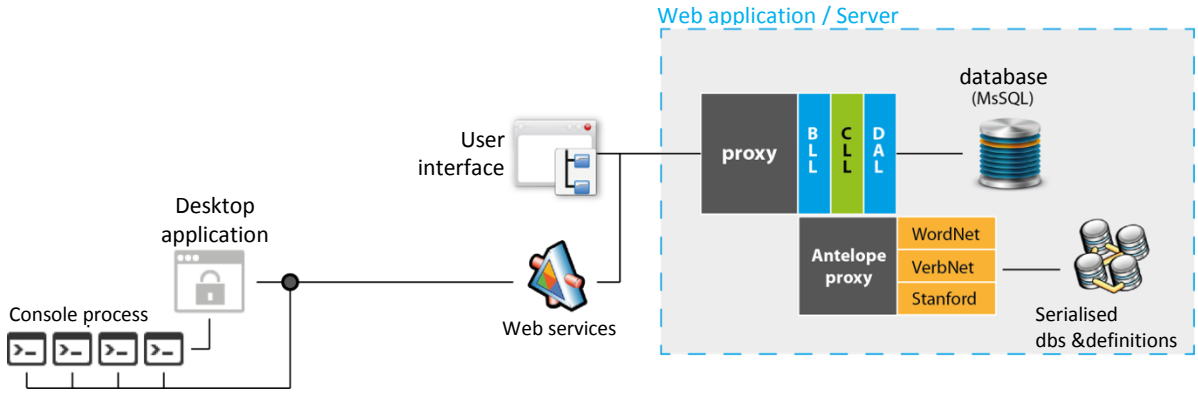


Figure 2. Overview of system components required for ontology modeling

 ISBN: 978-1-61208-654-5

The system is composed of multiple tiers. It consists of several components that include web information retrieval agent (as console process), parser and analyzer (as part of desktop application). Key elements of the system include the use of lexical databases and basic NLP procedures, as well as heuristics specifically developed to effectively determine keywords for content description. These features will be explained in more detailed in the rest of this Section.

B. *Natural Language Processing and role of Lexical Databases*

Natural Language Processing (NLP) methods that are used in this example deal with determining the meaning of text given in various sources. Meaning itself will be represented by a set of keywords that are best suited for particular content, i.e., web page. Keywords will determine position of each web page within the ontological model instead of using only title for each page. In order to adequately estimate meaning of the page content, it is important to consider the language of the text presented to the NLP procedures. There are particular properties of text specific to each language. For this purpose, only web sites and web pages written in English language are taken into account. NLP procedures can take advantage of existing language ontologies and lexicons that are available and accessible online.

There are several lexical databases available. WordNet is one of the first and most comprehensive lexicons for English language. Subsequently, other languages also developed their lexicons using WordNet organization model. Nouns, verbs, adjectives, adverbs are all grouped into cognitive synonym groups called SynSets. Each SynSet represents a clearly defined concept that fosters conceptual-semantic relations or lexical relations to other SynSets. There are 117000 SynSets, each with its own short description and example of usage in sentences.

In order to use lexical database, it is important to determine sentence structure and role of each word in each example. Stanford parser is a probabilistic parser used to determine grammatical structure of a sentence and group words into phrases and determine their function. It plays an important role in segmentation of sentences as it presents a sentence as Stanford Dependency (SD). SDs are hierarchical graphical representations of relations between words in sentence each described as a triplet: relation name, governing term and the dependent term.

Finally, modular solution that serves both as a language lexicon and parser is Proxem Antelope project and currently commercially available software tool Proxem Studio. This tool connects previously described lexicons and parsers and serves as an interface to procedures required for the analysis given in Figure 3.

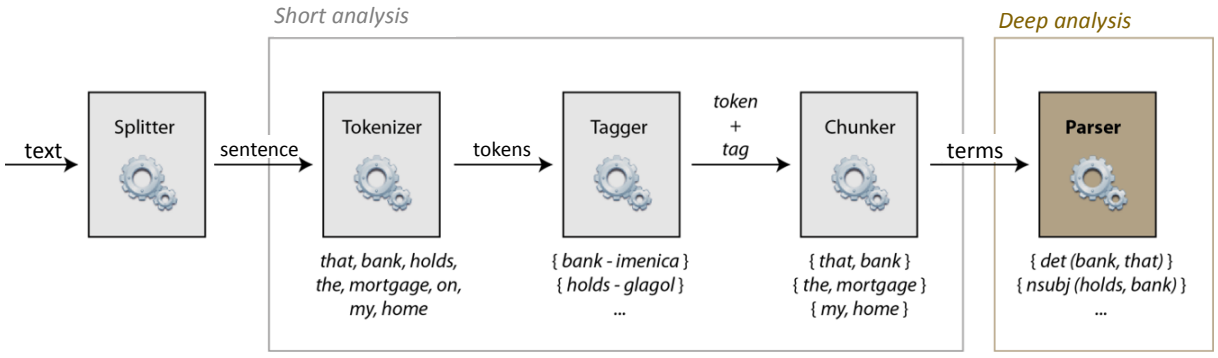


Figure 3. Overview of activities performed during syntax analysis

Content analysis goes through three steps: splitting text into sentences, performing short analysis and then performing deep analysis. During short analysis each sentence is disassembled into words or tokens. Each token is then tagged with the appropriate role in the sentence. For each tagged token, meaning is provided from the language lexicon and a set of relevant terms is created. These terms are then subjected to deep analysis by the parser that uses SDs to extract most probable keywords for the initial text.

C. *Created Heuristics in Analysing Web Page Context and Creating Ontological Model*

Now that each element of the hierarchy (in this case Web page) has a list of relevant key words describing the content, this structure is used to advance the ontological model (Figure 4). Comparative analysis between the current ontological scheme and prepared web page is performed. Based on the quantitative heuristics web page elements will be transformed into nodes of the ontological scheme. There are several options available: node will be added to the ontological model, removed from the structure or repositioned.

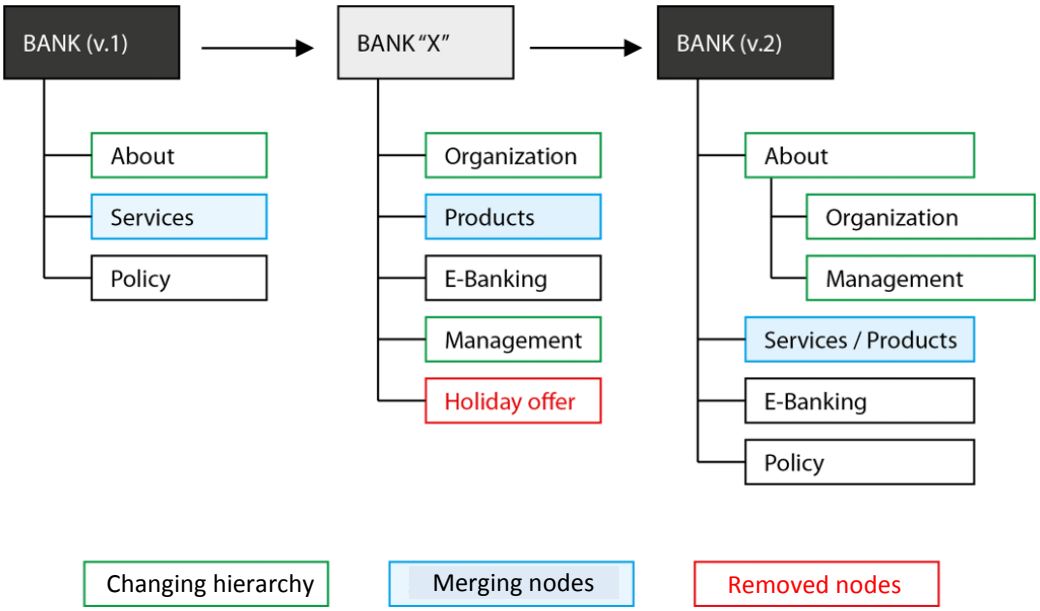


Figure 4. Example of operations over the ontology scheme during creation phase

Comparative analysis is performed for each node of the new web page (Bank X) and current version of ontological model (Bank v.1). Here, the heuristic list of keywords is chosen for each node and it is compared to each keyword set for nodes of the current ontological model. Distances between keywords are calculated in three steps.

In the first step, a set of potential elements, SE, is determined as (1)

$$SE = \arg\max_{x \le 10} f(x) = \{K_S \cap K_E\}, \quad |K_S \cap K_E| \ge m, x \in N, m > 0 \tag{1}$$

where maximum number of potential elements that are used for comparison x is ten. If K_S is a set of keywords describing the web page and K_E is a set of keywords describing elements of the ontological scheme, m is a heuristics parameter describing the minimum number of matching keywords both in K_S and K_E. Next weights for each matched keyword in SE is calculated as (2)

$$w_{SE} = \sum |K_{SE} \cap K_E|, \; K_{SE} \subseteq SE \tag{2}$$

w_{SE} is calculated as a sum of repetition of keywords contained in SE. Finally, distance between two sets K_S and K_E are calculated as (3)

$$M = \arg\max_{x=1} f(x) = \left\{\sum \frac{SE}{w_{SE}}\right\}, |SE| > 0, w_{SE} > 0 \tag{3}$$

where M is the sum of weighted keywords in SE. Since the goal is to find the smallest distance between the two sets, i.e., most similar pair of nodes from potential web page keywords and ontological model inverse of w_{SE} is used.

After ontological model is adjusted with information from each subsequent web page, additional *ex post* tuning using the same procedure is performed since new information changes the content of the set of potential elements SE, enabling additional corrections between nodes of the ontological model.

 ISBN: 978-1-61208-654-5

As we can see in Figure 4 depending on the comparison analysis, each node from new web site instance will influence the ontological model with three possible outcomes: (1) if the matching threshold is not passed the node will not be included in the model, (2) if there is matching between two or more nodes of the web page with one node in the model, these nodes will become sub-nodes of the ontological model increasing the hierarchical depth of the model and (3) if there is only one node of the web site that matches with only one of the nodes in the model these nodes will merge.

In initiating phases of the development of ontological model new nodes can be added to the model manually to help determine the most important features of the problem domain.

IV. BEST PRACTICE SEMANTICS: BANKING WEB SITES

In order to analyze the presented conceptual model a prototype system was developed as proof of concept. The goal of the prototype was to determine best practices in structuring web information on banking web sites in East European countries. Architecture of the developed system is given in Figure 5.

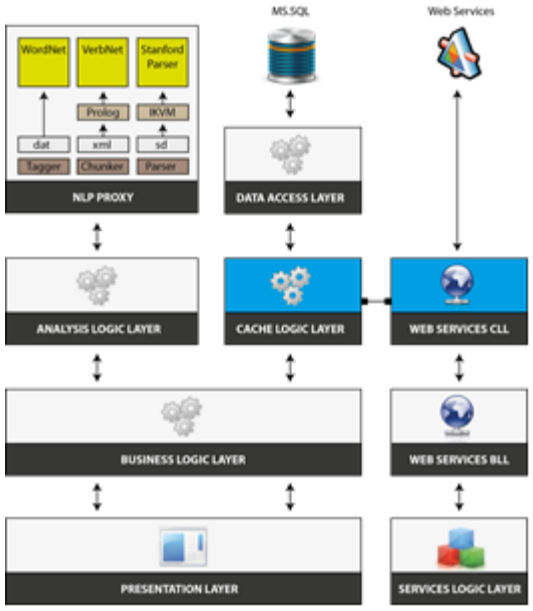


Figure 5. System architecutre for the developed prototype of the system

The system was given a database of URL addresses for a set of 42 banking web sites that have published information in English language. Total number of webpages included in the research was 2600. Retrieval of website structure and webpage documents was automated using a dedicated software agent. Software agent accessed each web site and parsed the content to search for web map or web navigation. Once located it was able to retrieve hyperlinks to each page in web map, access these pages and retrieve their content. All of the content was further prepared and stored into a local database that was used to initiate content analysis and iterative generation of the ontological benchmarking model of banking web sites. The induction process of the ontological model was conducted iteratively as described in earlier Section. Basic Statistics of the developed ontological scheme model are given in Table 1.

TABLE I. BASIS STATISTICS FOR DEVELOPED ONTOLOGY SCHEME

No. of iterations	No. of ontology elements	No. of matched pages
30	142	813
Total no. of used keywords	**Average no. of keywords per element**	**No. of matched pages per element**
2.393	16 (σ = 9,1777)	5 (σ = 7,9491)

Total number of iterations of the heuristic algorithm that calculated distances of each new node and its position in the model was 30. Final structure of the model included 143 content elements or pages in hierarchy. Each category generated a list of keywords from a total pool of generated keywords for the model 3993. Finally, Average number of keywords associated with each element is 16 with rather high variance of over 9. Average number of pages associated with each element of the ontological model is 5.

Part of the developed model can be seen in Figure 6.

 ISBN: 978-1-61208-654-5

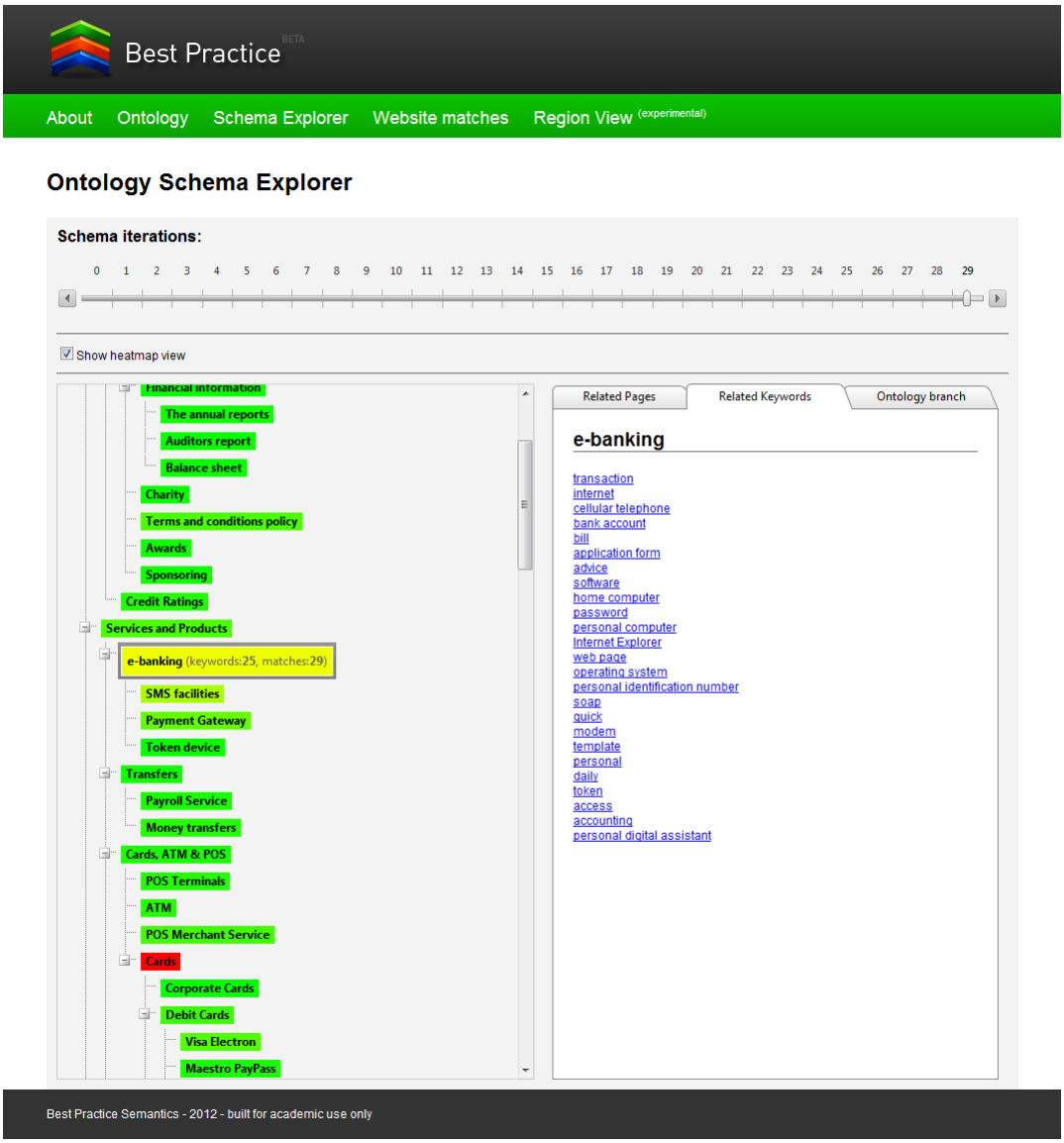


Figure 6. User interface of the developed prototype with portion of ontology model

The Ontology scheme Explorer shown in Figure 6 can also be used to view different versions of the scheme for each iteration, but it can also provide the list of keywords associated with each node presented in the model.

Some of the most common elements in bank website structure are given in Table 2 with respective list of keywords.

Keywords associated with particular node/web page category can also provide additional insight into the content of each page. Here we can see some web sites provide multiple pages with similar information, where the ontological model grouped these pages as same node. List of keywords can be used for further analysis and visualization. For example, presentation of the model benefits from color coded information, where green color is used to show categories that are present in most web sites, while less common categories are represented in yellow or gray color.

Additionally, Ontology Schema Explorer can provide comparison tool and show parallelly the scheme with best practices and structure of chosen web site with additional information about the similarities and differences between a web site and benchmarking structure (Figure 7).

 ISBN: 978-1-61208-654-5

TABLE II. MOST FREQUENT WEBSITE CATHEGORIES AND ASSOCIATED KEYWORDS

#	Name of Node	Number of pages	Most common keywords associated with the element
1.	MasterCard Standard	62	account, bank card, card game, credit card, customer, debit card, hotel, interest rate, internet, issue, logo, merchant, opportunity, personal identification number, swipe, transaction...
2.	Cards	35	application form, bank account, bank card, business card, calling card, credit card, debit card, internet, kind, larceny, merchant, overdraft, phone number, regular payment, seller...
3.	Depository Services	32	bank account, booklet, credit, depositor, depository, entrepreneur, interesting, investor, issuer, labour contract, mediator, quarterly, redemption, time deposit...
4.	International payments	31	bill, call mark, check, documentary, duty, European Union, foreign country, foreign exchange, franchise, futures contract, giro account, International, rate of exchange, savings, transaction...
5.	e-banking	29	access, accounting, application form, bank account, bill, cellular telephone, daily, home computer, internet, Internet Explorer, modem, operating system, password, personal digital assistant, personal identification number, quick, software, template, token, transaction, web page...
6.	Current Account	21	account, application form, bank account, check book, debit, foreign exchange, guardianship, income, interest rate, legal status, national, old-age pension, overdraft, pension, savings, Social Security, standing order, transaction, wage...
7.	SMS facilities	21	bank account, cellular telephone, customer, daily, debit, foreign exchange, password, phone number, stand-in, transaction, transfer payment...
8.	SME Financing	20	business activity, capital, cash flow, collateral, distribution channel, documentary, employee, entity, equipment, expertness, export, factorization, financing, guarantor, investment, letter of credit, mortgage, overdraft, postponement, production cost, real property, savings, subcontractor, suiting, supplier, wage...
9.	Individual customers	19	advisory, applicant, check, customer, employee, entity, financing, free, giro account, income, interest rate, memo, memorabilia, net income, pension, poor people, precious metal, private, safe-deposit, savings, Social Security, valuable, wage...
10.	Frequently Asked Questions (FAQ)	17	advice, alias, anti-virus program, at home, call, card, consent, database, encoding, income, letter of credit, overdraft, password, personal computer, personal identification number, phone number, prerequisite, procurement, purchase price, rate of exchange, small letter, smart card, software...

V. DISCUSSION AND CONCLUSIONS

As we can see from the developed prototype the presented concept can be implemented using currently available technologies in order to provide additional information about current status of published web information in various problem areas. It is important to stress that currently language and lexical databases are main constraining factor in further implementations of this approach with regards to assessing information in languages other than English. Actual implementation of the system may prove to require additional costs for the development in comparison to more standardized business intelligence tools used in benchmarking, but this approach does offer a new and fresh look at the available data that is rarely covered by current business intelligence or decision support tools.

If we take a closer look at the ontological model for banking web sites for the South Eastern Europe, we see that there is a high difference in web site quality between banks both in structuring web site and content. Many of the banks are part

 ISBN: 978-1-61208-654-5

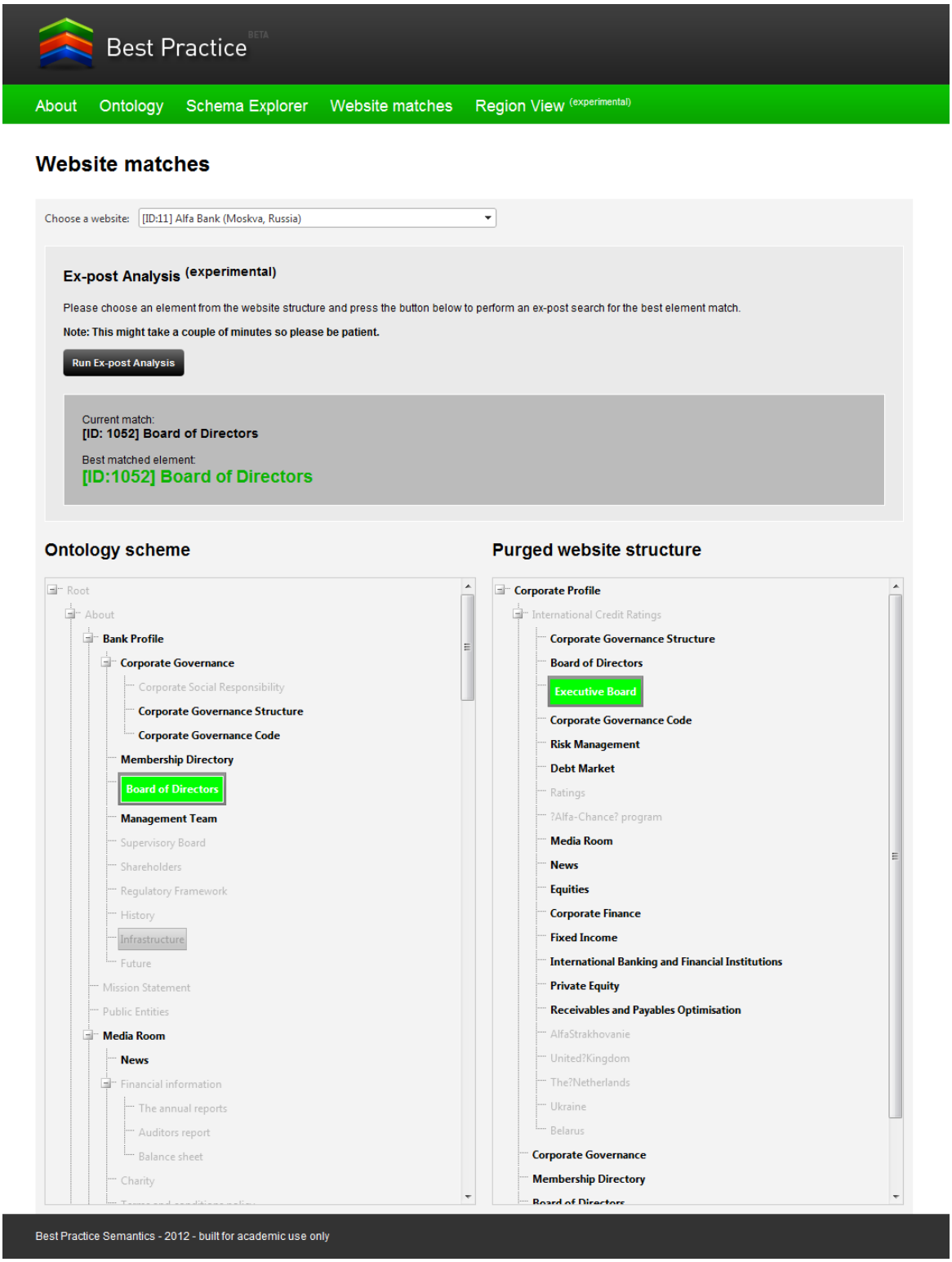


Figure 7. Comparative analysis of web site structure with (a) ontology scheme on the left side and (b) specific website structure on the right

 ISBN: 978-1-61208-654-5

of larger group of international banks that implement web content management solutions of the parent bank, so the specific local customization is not implemented consistently or adequately. The ontological model recognized this by grouping several pages of some of the same banks into the same category of the ontological model, while excluding several retrieved pages that did not pass the relevant threshold of the model. This is very important information that may be used to improve shortcomings of web sites of commercial banks in this region.

In conclusion, we see that there is still adequate potential in developing new approaches to automated and semi-automated tools that can help with Web information retrieval and generation of decision assistive models for decision making. Ontologies implemented in the area of benchmarking analysis in business can provide new means of analyzing publicly available data about markets and competition allowing companies to improve their processes and strategies.

In this paper we presented ontology-based benchmarking tool that may provide support to managers while providing semi-automated assistive decision models. This tool can be further improved in several different directions. Firstly, implementing additional analysis of created ontological model would show metrics for each node, explanation of the hierarchical position of a node within the model, etc. Secondly, adding the analysis of recovered data and visualizations in terms of color coding various content indicators, such as significance of content topics based on number of web sites including these topics, optional position of particular nodes in the structure, etc. This information may be used to guide decision makers during planning of their web site structure, ultimately improving their customers' experience. Finally, third possible improvement of the proposed model is further sophistication and automation of the ontology construction procedure so that it requires less manual intervention and corrections.

Future work includes developing approaches to automate additional steps during the creation of the benchmarking semantic model as it currently requires expert input, especially in the initial stages of the creation process. Another improvement of the model is the possibility to use other languages that have well defined language corpuses available.

REFERENCES

[1] C.-H. Chou, F. M. Zahedi, and H. Zhao, "Ontology for Developing Web Sites for Natural Disaster Management: Methodology and Implementation", IEEE Transactions on Systems, Man, and Cybernetics—part a: systems and humans, vol. 41, no. 1, January 2011, pp. 50-62.

[2] S. Drew, "From knowledge to action: the impact of benchmarking on organizational performance", Long Range Planning, Vol. 30 No. 3, 1997, pp. 427-441.

[3] S. Itzhaky, S. Gulwani, N. Immerman, and M. Sagiv, "A Simple Inductive Synthesis Methodology and its Applications," Proceedings of the ACM international conference on Object oriented programming systems languages and applications, Reno/Tahoe, Nevada, USA — October 17 - 21, 2010, ACM, 2010, pp. 36–46.

[4] F. Teuteberg, M. Kluth, F. Ahlemann, and S. Smolnik, "Semantic process benchmarking to improve process performance", Benchmarking: An International Journal Vol. 20 No. 4, Emerald Group Publishing Limited, 2013, pp. 484-511.

[5] K. N. Vavliakis, A. L. Symeonidis, G. T. Karagiannis, and P. A. Mitkas, "An integrated framework for enhancing the semantic transformation, editing and querying of relational databases", Expert Systems with Applications, Vol. 38, Issue 4, April 2011, pp 3844-3856.

[6] Auhood Alfaries, David Bell, and Mark Lycett, "Motivating service re‐use with a web service ontology learning", International Journal of Web Information Systems, Vol. 9 Issue: 3, 2013, pp.219-241.

[7] M. Calaresu, and Ali Shiri, "Understanding Semantic Web: a conceptual model", Library Review, Vol. 64 Issue: 1/2, 2015, pp.82-100.

[8] M. Singleton, "Combining Phenotype and Genotype For Discovery and Diagnosis of Genetic Disease", Doctoral Dissertation, August 2015.

[9] C. Stanley Funk, "Recognition and Normalization of Terminology from Large Biomedical Ontologies and Their Application For Pharmacogene and Protein Function Prediction", Doctoral Dissertation, 2015.

[10] Bhaskar Sinha, Somnath Chandra, and Megha Garg, "Development of ontology from Indian agricultural e-governance data using IndoWordNet: a semantic web approach", Journal of Knowledge Management, Vol. 19 Issue: 1, 2015, pp.25-44.

[11] D. Ruikar, C.J. Anumba, A. Duke, P.M. Carrillo, and N.M. Bouchlaghem, "Using the semantic web for project information management", Facilities, Vol. 25 Issue: 13/14, 2007, pp.507-524.

[12] George Macgregor, "Knowledge Representation in the Social Semantic Web", Library Review, Vol. 60 Issue: 8, 2011, pp.723-735.

[13] A. A. Krizhanovsky and A. V. Smirnov, "An Approach to Automated Construction of a General□Purpose Lexical Ontology Based on Wiktionary", Journal of Computer and Systems Sciences International, Vol. 52, No. 2, 2013, pp. 215–225.

[14] N. A. Astrakhantsev and D. Yu. Turdakov, "Automatic Construction and Enrichment of Informal Ontologies: A Survey", Programming and Computer Software, Vol. 39, No. 1, 2013, pp. 34–42.

[15] R. Navigli, P., Velardi, and A. Gangemi, " Ontology learning and its application to automated terminology translation". IEEE Intelligent Systems, 18(1), 2003.

[16] M. Sintek, P. Buitelaar, and D. Olejnik. " A protege plug-in for ontology extraction from text based on linguistic analysis". In: Proceedings of the 1st European semantic web symposium (ESWS), 2004.

[17] S. H. Wu, and W. L. Hsu. " SOAT: a semi-automatic domain ontology acquisition tool from Chinese Corpus". In: 19th international conference on computational linguistics Howard international house and Academia Sinica, Taipei, Taiwan, 2002.

[18] M. Y. Dahab, H. A. Hassan, and A. Rafea. "TextOntoEx: Automatic ontology construction from natural English text", Expert Systems with Applications 34, 2008, pp. 1474–1480.

 ISBN: 978-1-61208-654-5

ECF-means – Ensemble Clustering Fuzzification Means

A novel algorithm for clustering aggregation, fuzzification, and optimization

Gaetano Zazzaro, Angelo Martone
CIRA
Italian Aerospace Research Centre
Capua (CE), Italy
e-mail: {g.zazzaro, a.martone}@cira.it

***Abstract*—This paper describes a clustering optimization algorithm for Data Mining, called Ensemble Clustering Fuzzification (ECF) means, which combines many different clustering results in ensemble, achieved by *N* different runs of a chosen algorithm, into a single final clustering configuration. Furthermore, ECF is a simple procedure to fuzzify a clustering algorithm because each point in the original dataset is assigned to each cluster with a degree of membership. Moreover, a novel clustering validation index, called Threshold Index (TI), is also here defined. The proposed approach is applied to the well-known *k*-means clustering algorithm by using its Weka implementation and an ad-hoc developed software application. Two case studies are also here reported; the first one in the meteorological domain and the second one concerns the famous Iris dataset. All the outcomes are compared with the results of the simple *k*-means algorithm against which ECF seems to provide more effective and usable final configurations.**

Keywords-Clustering Optimization; Data Mining; Ensemble Clustering; Fuzzy Clustering; k-means; Weka.

I. INTRODUCTION

Clustering (or cluster analysis) is an unsupervised Machine Learning technique of finding patterns in the data. It is widely used [1] for Data Mining tasks, because it can be easily applied to understand, explore, prepare, and model data. It plays an outstanding role in many applications, such as scientific data exploration, information retrieval and text mining, web analysis, bioinformatics, and many others.

In the literature, there are many categories of algorithms for clustering: Heuristic-based, Model-based, Density-based [2]. Their common goal is to create clusters so that objects in the same cluster should be as similar as possible, whereas objects in one cluster should be as dissimilar as possible from objects in the other clusters. Usually, it is not easy to choose the most useful algorithmic approach, the most satisfying result, and therefore the most usable configuration. In fact, the different models for clustering may produce groupings that are very different from one another. Anyone applying a clustering algorithm immediately realizes how difficult it is to choose the final cluster configuration. We may have different results because we choose different algorithms, or different parameters of the fixed algorithm. Furthermore, the numerous available evaluation metrics often do not facilitate this choice because they lead to very discordant results.

In spite of the availability of a large number of validation criteria, the ability to truly test the quality of a final configuration remains vague and hard to achieve. Specific domain knowledge is not an aid because it is often hard to translate it into operating rules, neither the domain expert has a real target class for evaluating and comparing the results. So, why do not consider all the obtained configurations? That is, why do not find a method that summarizes all the results of clusterings? Meta-learning ensemble methods may be an answer. The idea is that no single model or criterion truly captures the optimal clustering, but a cooperation of models could provide a more robust solution. Cluster Ensemble, or Aggregation Clustering, or Multiview Clustering, aims to find a single clustering from multi-source basic clusterings on the same group of data objects [3]. However, these ensemble methods, such as voting-based clustering [4], consensus clustering [5], or clustering aggregation [6] do not assign a level of membership to every point in clusters.

In order to overcome the limits mentioned above, in this paper we present a strategy for cluster analysis. As will be evident, this simple method can be included within ensemble procedures. It is also an *a posteriori* criterion for optimization of the obtained groupings. This procedure takes in input any partitioning clustering algorithm for which it is possible to initially choose the *k* number of clusters to be determined and a seed for the random choice of the initial *k* centroids.

The *k*-means algorithm is one of the clustering algorithms that checks all the conditions listed. So, it is considered as a reference clustering algorithm. In Weka implementation of *k*-means [7] [8], the name of the algorithm is *SimpleKMeans*; in this version the seed parameter is *s*, that is the initialization value for the random number generator. Using the same seed value will always result in the same initial centroids then. Exploiting this seed parameter, many different configurations are evaluated and compared, and also used in our meta-algorithm for ensemble final configuration.

Finally, a "soft" interpretation of the clustering is presented, in order to better explore and understand the results, to find possible outliers in the dataset, and to fix the best parameters.

A. Structure of the paper

In Section II, we present some Cluster Analysis general outlines, including main definitions, its scope and its role in

 ISBN: 978-1-61208-654-5

Data Mining. Furthermore, some concepts regarding Ensemble Clustering, soft and hard clustering are mentioned.

In Section III, the original k -means algorithm is synthesized, exposing its pros and cons.

In Section IV, the Ensemble Clustering Fuzzification Means (ECF-means) is presented, including some main definitions, validation measures, and clustering validity indexes.

In Section V, the ECF-means SW application is explained.

In Sections VI and VII, we show how the implemented tool has been used in two different applications, underlining how it helped us to explore datasets, discover new knowledge and to group objects in order to train custom models.

Finally, in Section VIII, we show our general considerations and future works.

II. CLUSTER ANALYSIS

Clustering, or Cluster analysis, methods belong to intersection of Statistics, Machine Learning, and Pattern Recognition. It is a very useful method for discovery pattern in large amount of data. It is a technique to group a set of objects into subsets or clusters. Usually, objects are described by attributes, also called features. It has become one of the most widespread unsupervised techniques of Data Mining. It has multiple real applications, above all for the simplicity of the algorithms and their readings.

A. Introduction, Definitions and Scope

A Clustering algorithm produces a partition on an unlabeled data set, such that no cluster is empty, no two clusters intersect, and the union of all clusters is the data set itself.

The goal is to create clusters that are coherent internally, but substantially different from each other. In a nutshell, objects in the same cluster should be as similar as possible, whereas objects in one cluster should be as dissimilar as possible from objects in the other clusters.

Similarity between objects that belong to a cluster is usually measured by a metrics d. Two objects x and y are similar if the value of $d(x, y)$ is small; what "small" means depends on the context of the problem. d is defined by some distance measure. Typically, the Euclidean Distance (or simply the squared Euclidean Distance) is widely used in many applications (it is also used in the ECF-means) for the computation of similarities: $ED^2(x, y) = \sum_{i=1}^{n}(x_i - y_i)^2$.

It is important to underline that, also depending on the type of data, other many metrics are possible.

Numerous clustering algorithms are available in the literature and there are several points of view for examining clustering techniques; a very good landscape of Clustering algorithms can be retrieved in [2], and an in-depth and complete study of clustering techniques, algorithms and applications can be retrieved in [9].

B. Ensemble Clustering

Different clustering approaches or different views of the data can lead to different solutions to the clustering problem. Indeed, also initial settings of a fixed algorithm may produce clusters that are very different from one another. This evidence is closely related to the theory of Ensemble Clustering (or Multiview Clustering), that studies this issue from a broader perspective [3] [10]. Therefore, instead of running the risk of picking an unsuitable clustering algorithm, a cluster ensemble can be used in order to get a "better" clustering configuration. The idea is that no single model or criterion truly captures the optimal clustering, but a collective of models will provide a more robust final solution.

Most ensemble models use the following three steps to discovery the final clusters configuration:

1. Generate *N* different clusterings, by using different approaches, or different data selection, different settings of the same algorithm, or different clusterings provided by different runs of the same algorithm. These represent the ensemble components.
2. Combine the results into a single and more robust clustering, by using a rule or a set of rules (called meta-rule).
3. Evaluate the ensemble clustering result and compare it with the results of the *N* components.

As already mentioned, the ensemble components can be selected in a wide variety of ways. Some strategies for building clustering ensemble components follow:

1. By using different subsets of features. Each clustering configuration is found by means of overlapping or disjoint subsets of the original features set.
2. By selecting different subsets of the data, via random sampling.
3. The different components can be selected combining a variety of models and algorithms such as partitioning, hierarchical or density-based methods, random or deterministic algorithms, and so on.
4. The different components can correspond to different settings of the same algorithm.
5. The different components could be obtained from a single algorithm, randomizing the initial choice of the clusters centroids. Of course, an example is k-means; thus, the ensemble can be formed as the result of *N* different runs of the algorithm.

After the individual components have been obtained, it is often a challenge to find a meta-rule able to combine the results from these different solutions in order to create a unified ensemble clustering.

C. Hard and Soft Clustering

Clustering algorithms can also be classified into hard and soft algorithms. A hard clustering algorithm leads to a partition of crisp sets. In a crisp set, an element is either a member of the set or not. On the other hand, a soft clustering algorithm leads to fuzzy clusters. Fuzzy sets allow elements to be partially in a set. Each element is given a degree of membership in a set.

One of the most famous fuzzy clustering algorithms is Fuzzy C-means [11], which allows an object to belong to two or more clusters with a membership degree between zero (not an element of the set) and one (a member of the set). It has been widely used in many real-world application domains where well-separated clusters are typically not available.

 ISBN: 978-1-61208-654-5

The method presented in this article leads to a fuzzy partitioning of the starting dataset, by repeatedly applying the results of the k-means algorithm.

III. THE k-MEANS ALGORITHM

k-means is a simple clustering algorithm whose main goal is to find k non-overlapping clusters. Each final cluster is represented by its centroid that is typically the mean of the points in that cluster.

A. Introduction, scope and procedure

k-means is one of the oldest and still widely used algorithms for cluster analysis. Without any doubt, it represents the archetype of the clustering partitioning algorithms. Because of its mathematical simplicity, it is also the most studied unsupervised learning technique [12], and over the years, many of its variations and extensions have been implemented (for High-Dimensional Data, for Data Streams, Time Series, for Data with noise, and so on).

Its basic algorithmic structure is shown in the Figure 1.

k-means Clustering Algorithm

Input: S set of instances; k number of clusters
Output: set of k clusters with k centroids

1. Randomly initialize k cluster centers (centroids)
2. **While** termination condition is not satisfied {
3. Assign instances to the closest cluster center
4. Update cluster centers using the instances assignment
5. }

Figure 1 – k-means Algorithm.

The condition of termination of the process is satisfied when no point changes clusters.

B. Pros and Cons

The algorithm has been very successful thanks to its simplicity and also for its linear time complexity $O(knl)$, where n is the number of objects to be clustered and l is the number of iterations that the algorithm is performing.

Like most partitioning clustering algorithms, k-means has some disadvantages:

1. It is very sensitive to the presence of outliers and noise.
2. The number of clusters need to be specified by the user and often it's not simple to choose it.
3. It is not able to discover concave-shaped clusters.
4. Since the initial choice of k centroids is random, different selections can also lead to very different final partitions, especially for large datasets with many features.

The k-means algorithm always terminates, but it does not necessarily find the "best" set of clusters.

IV. ENSEMBLE CLUSTERING FUZZIFICATION MEANS

The initial selection of centroids can significantly affect the result of the k-means algorithm. To overcome this, the algorithm can be run several times for a fixed value of k, each time with a different choice of the initial k centroids.

In many software implementations of k-means, for example in its Weka version, it is possible to choose a seed parameter (s), useful for the random selection of the first initial centroids (s is the random number seed to be used). Using this parameter, it is possible to realize, as will be described in the following sections, a procedure able to optimize and reinforce the obtained partition.

A. Introduction and Definitions

Let $S \subseteq \mathbb{R}^m$ be a set of points. Let k be the desired number of clusters to be determined. Changing the seed (s) from 0 to $N-1$, N partitions of S can be generate by applying the k-means algorithm. Some of these partitions are exactly the same, considering or not the order of groupings. Others, however, differ for very few records, and others for many.

In the following $N \times k$ matrix, called Clustering Matrix C of S, each row is a partition of k clusters of S.

$$C = \begin{pmatrix} C_{1,1} & C_{1,2} & \dots & C_{1,k} \\ C_{2,1} & C_{2,2} & \dots & C_{2,k} \\ \dots & \dots & C_{i,j} & \dots \\ C_{N,1} & C_{N,2} & \dots & C_{N,k} \end{pmatrix}$$

$C_{i,j}$ is the j-th cluster obtained at the i-th iteration of the clustering algorithm, with $i = 1, \dots, N$ and $j = 1, \dots, k$.

It is possible associate a new $N \times k$ matrix to C, called MU matrix, that is the matrix of the centroids of the clusters:

$$C \to \begin{pmatrix} \mu(C_{1,1}) & \mu(C_{1,2}) & \dots & \mu(C_{1,k}) \\ \mu(C_{2,1}) & \mu(C_{2,2}) & \dots & \mu(C_{2,k}) \\ \dots & \dots & \mu(C_{i,j}) & \dots \\ \mu(C_{N,1}) & \mu(C_{N,2}) & \dots & \mu(C_{N,k}) \end{pmatrix} = MU$$

$\mu(C_{i,j})$ is the arithmetic mean of the j-th cluster of the i-th iteration of the algorithm, with $i = 1, \dots, N$ and $j = 1, \dots, k$.

B. Clusters Sort Algorithm

The algorithm in Figure 2 is useful for sorting the clusters partitions of C matrix. This step is essential because k-means can produce different orders of clusters in different runs, even if the partitioning results can be the same.

Please note it is possible that the average of some elements of the second row C_2 in Algorithm 1 of Figure 2 has a minimum distance from two or more averages of elements of the first row C_1. In this case, the minimum value of the minimum values is chosen.

C. The ECF-means Algorithm

Let C be a Cluster Matrix of S, sorted by using the Algorithm 1. We define $\underline{C_j}$ as ***floor of*** $\boldsymbol{C_j}$: $\underline{C_j} = \bigcap_{i=1}^{N} C_{i,j}$, with $j = 1, \dots, k$. It is possible that $\underline{C_j} = \emptyset$ ($j = 1, \dots, k$). Moreover, $\underline{\boldsymbol{S}} = \bigcup_{j=1}^{k} \underline{C_j}$ is defined as the ***floor of*** $\boldsymbol{S}$.

Let x be an element of S; we can count the number of clusters of the first column of C where x is, the number of clusters of the second column of C where x is, and so on. In this way, we can associate a new numerical vector to x, called ***attitude of*** $\boldsymbol{x}$ ($att(x)$):

 ISBN: 978-1-61208-654-5

$$att(x) = \left(att_1(x), att_2(x), \dots, att_k(x)\right),$$

where $att_j(x)$ is the number of clusters in the j-th column of C where x is located. $att_j(x) = N \Leftrightarrow x \in \underline{C_j}$ and $\sum_{j=1}^{k} att_j(x) = N$. In this manner, we are defining a function att_j ($j = 1, \dots, k$ and $I = \{1,2, \dots, N\}$):

$$att_j : x \in \bigcup_I C_{i,j} \rightarrow att_j(x) = \left|\{i \in I : x \in C_{i,j}\}\right|$$

where, as usual, $|A|$ is the number of the elements of the set A.

Algorithm 1: Clusters Sort Algorithm

Input: two different rows of C:
$C_1 = \left(C_{1,1}, C_{1,2}, \dots, C_{1,k}\right)$ and $C_2 = \left(C_{2,1}, C_{2,2}, \dots, C_{2,k}\right)$
Output: a new order of the second row:
$\left(C'_{2,1}, C'_{2,2}, \dots, C'_{2,k}\right) = C'_2$

C_1 represents the reference row of the current sorting procedure (e.g., obtained by fixing $s = 0$ in the Weka k-means algorithm).

1. Calculate the $2 \times k$ matrix of clusters centroids:
   $MU = \begin{pmatrix} \mu(C_{1,1}), \mu(C_{1,2}), \dots, \mu(C_{1,k}) \\ \mu(C_{2,1}), \mu(C_{2,2}), \dots, \mu(C_{2,k}) \end{pmatrix}$
2. Compute the Euclidean Distances (ED) in MU. The following $k \times k$ matrix is the Δ matrix of the EDs:
   $\Delta = \begin{pmatrix} d_{1,1} & d_{1,2} & \dots & d_{1,k} \\ d_{2,1} & d_{2,2} & \dots & d_{2,k} \\ \dots & \dots & \dots & \dots \\ d_{k,1} & d_{k,2} & \dots & d_{k,k} \end{pmatrix}$
   Where:
   $d_{i,j} = ED\left(\mu(C_{1,i}), \mu(C_{2,j})\right)$, with $i, j = 1, \dots, k$.
3. Calculate the minimum value of each row of Δ.
   $\min\{d_{1,1}, d_{1,2}, \dots, d_{1,k}\} = d_{1,\overline{j_1}} = min_1$
   $\min\{d_{2,1}, d_{2,2}, \dots, d_{2,k}\} = d_{2,\overline{j_2}} = min_2$
   … … … … … …
   $\min\{d_{k,1}, d_{k,2}, \dots, d_{k,k}\} = d_{k,\overline{j_k}} = min_k$
4. The second row C'_2 is:
   $\left(C'_{2,1}, C'_{2,2}, \dots, C'_{2,k}\right) = \left(C_{2,\overline{j_1}}, C_{2,\overline{j_2}}, \dots, C_{2,\overline{j_k}}\right)$
   Where:
   $C'_{2,1} = C_{2,\overline{j_1}}$ is the cluster (in C_2) that has the centroid with the minimum distance from the centroid of the first element of C_1.
   … … …
   $C'_{2,k} = C_{2,\overline{j_k}}$ is the cluster (in C_2) that has the centroid with the minimum distance from the centroid of the k-th element of C_1.

Figure 2 – Clusters Sort Algorithm.

Finally, we can define the ***probability vector of x***, as:

$$p(x) = \left(\frac{att_1(x)}{N}, \frac{att_2(x)}{N}, \dots, \frac{att_k(x)}{N}\right)$$

Thanks to the simple mathematical notions of the current section, we are able to "soften" the "hard" k-means algorithm and we can have a new Fuzzy Clustering Algorithm. According to this approach, each element of the dataset belongs to each cluster with a different degree of membership, and the sum of these probabilities is equal to one.

Furthermore, the method can also be interpreted in a different way. Indeed, this "fuzzification" procedure can be used not only with k-means algorithm, but also for others partitional clustering algorithms for which it is possible to choose the number of clusters to be determined. In this way, the algorithm is part of the Ensemble algorithms. For these reasons, ECF-means is also a meta-algorithm because we reach a fuzzy partition of the dataset by using a multiple clustering algorithm schema.

Algorithm 2: ECF-means (Fuzzification of k-means)

Input: $S \subseteq \mathbb{R}^m$; number k of clusters to be determined; membership threshold t ($0 \leq t \leq 1$); number N of k-means iterations
Output: set of k clusters of level t; probability vector of each element x of S

1. **Apply** the k-means algorithm to S, fixing the random seed $s = 0$, obtaining the clusters $C_{0,1}, \dots, C_{0,k}$ ($C(0)$-configuration)
2. **foreach** $s = 1, \dots, N-1$
3. **Apply** the k-means algorithm to S, obtaining the clusters $C'_{s,1}, \dots, C'_{s,k}$ ($C'(s)$-configuration)
4. **Apply** the Clusters Sort Algorithm to $C'(s)$, considering $C(0)$ as reference, obtaining the clusters $C_{s,1}, \dots, C_{s,k}$ ($C(s)$-configuration)
5. **end**
6. **foreach** $j = 1, \dots, k$
7. **foreach** $x \in S$
8. **Calculate** $p_j(x) = att_j(x)/N$
9. **Fix** the cluster $C_j^t = \{x \in S |\, p_j(x) \geq t\}$
10. **end**
11. **end**

Figure 3 – ECF-means Algorithm.

The membership threshold t in the Algorithm 2 (Figure 3) is fixed by the user and it is very useful to change the "level" to clusters final configuration. If $t = 1$, then $C_j^1 = \{x \in S | p_j(x) = 1\} = \underline{C_j}$. Additionally, $\underline{S} = \bigcup_{j=1}^{k} C_j^1$. If $t = 0$, then $C_j^0 = \{x \in S | p_j(x) \geq 0\}$ and $\bigcup_{j=1}^{k} C_j^0 = C_j^0 = S$.

Let $p(x)$ be the probability vector of x and let $M = \max att(x) = \max\{att_1(x), att_2(x), \dots, att_k(x)\}$ be the maximum of $att(x)$, if this exists. We can define the position of M in $att(x)$ as $PMA(x)$, if this exists.

An element $x \in S$ is an ***o-rank fuzzy outlier of S*** if $p_j(x) - p_l(x) \leq o$, where $p_j(x)$ and $p_l(x)$ are the first two highest value components of $p(x)$. Note that, in this contest, the word "outlier" takes on a different meaning than its scientific usual use; but the definition of *o-rank fuzzy outlier* helps us to treat these points as special points, that need to be observed more closely, because they belong at least to two different clusters.

D. Validation Measures for Fuzzy Clustering

Clustering validation has long been recognized as one of the critical issues essential to success of clustering applications [9].

Let $U = (u_{li})$ $(1 \leq l \leq k, 1 \leq i \leq n)$ be the membership's matrix of a fuzzy partition of a dataset S with n records, and k is the number of clusters.

The first validity index for fuzzy clustering is the Partition Coefficient Index (PC) [13]. PC is based on U and it is defined as: $PC = \frac{1}{n}\sum_{l=1}^{k}\sum_{i=1}^{n} u_{li}^2$.

$PC \in [1/k, 1]$. Furthermore, a PC value close to $1/k$ indicates that clustering is "very fuzzy"; the value $1/k$ is obtained when $u_{li} = 1/k$, for each l, i.

Another index is the Partition Entropy Coefficient (PE): $PE = -\frac{1}{n}\sum_{l=1}^{k}\sum_{i=1}^{n} u_{li} log_a(u_{li})$.

$PE \in [0, log_a k]$. Furthermore, a low PE value indicates that clustering is "not very fuzzy". PE values close to the upper limit indicate an absence of any clustering structure within the dataset or the inability of the algorithm to extract it.

The main disadvantage of PC and PE is their monotonic evolution tendency with respect to k. To avoid this, a modification of the PC index can reduce the monotonic tendency and was defined by: $MPC = 1 - \frac{k}{k-1}(1 - PC)$, where $0 \leq MPC \leq 1$.

Finally, let we define a novel validity index, that we call the Threshold Index TI, by the following formula:

$$TI = \frac{|\underline{S}|}{|S|}$$

V. ECF-MEANS TOOL

With the purpose of testing the ECF-means algorithm, a software application has been designed and developed. It has been carried on using a Client/Server architectural pattern, where the Server part consists of the algorithm and other support utilities, while the Client part is made by a browser-based application, responsible of the ECF-means result visualization.

A. Software Implementation

The ECF-means web application is built up of two main modules: the first one wraps the ECF-means Algorithm, that has been implemented in Java programming language, and it makes use of the Weka k-means algorithm (*SimpleKMeans*) [7] [14] as clustering algorithm implementation.

The second module consists of the web application client part, that has been implemented by using JavaScript libraries, such as D3.js, as visualization library, and jQuery for Ajax asynchronous data communication and Document Object Model (DOM) manipulation tasks.

B. GUI & Data Visualization

The implemented tool provides a user-friendly GUI, by which it is very easy to load datasets, fix the ECF-means parameters, and understand the algorithm results visually.

The GUI can be divided into three functional blocks, as highlighted by red numbered circle in Figure 4.

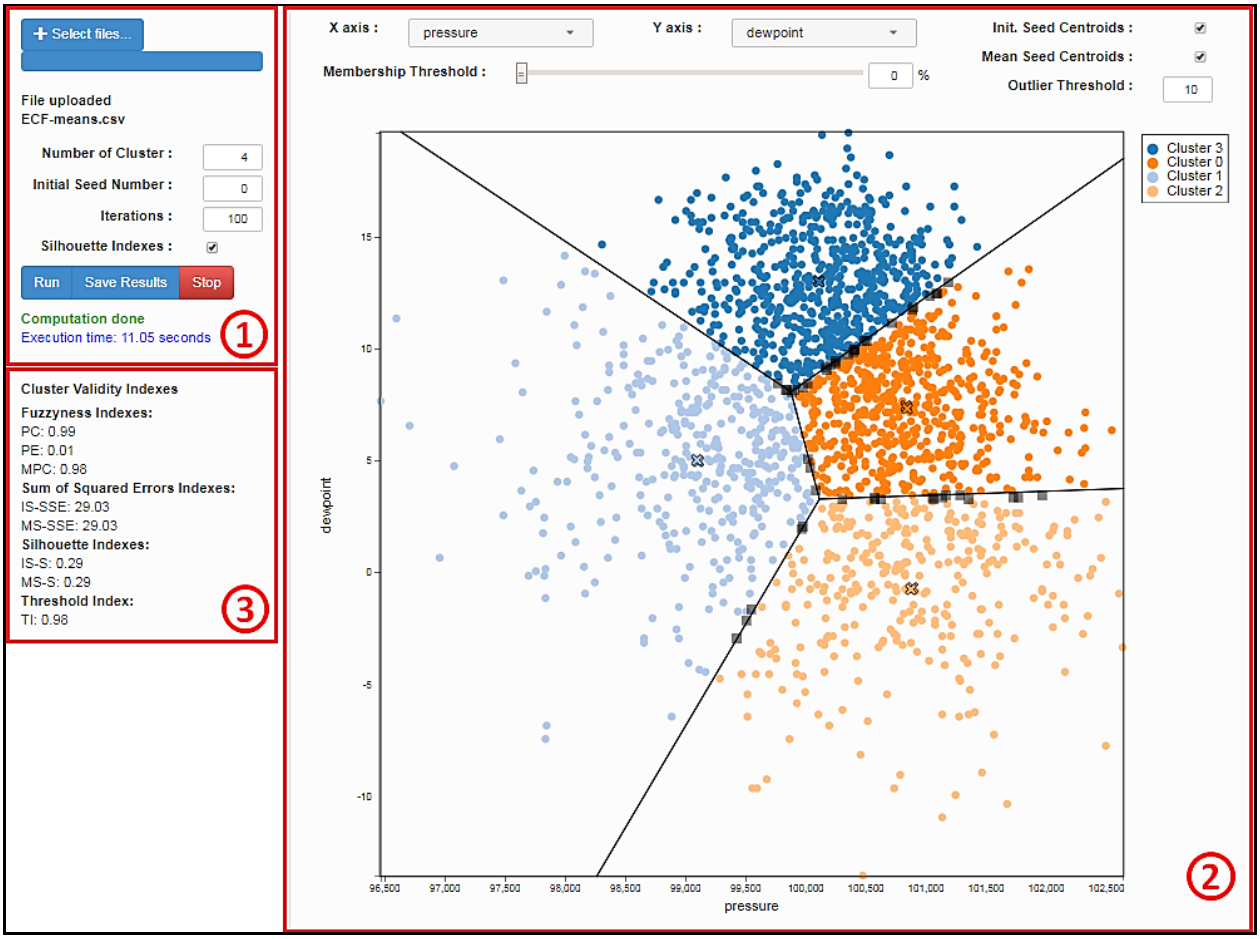


Figure 4 – Software GUI and Clustering Visualization.

Through the first functional block, user can upload a dataset from a local file system, in csv or arff formats; after that he can specify the number of clusters k (default is set to two), the initial seed number (default 0), and the number of iterations N to perform (default 100). Lastly, a set of buttons allow the following operations:

1. *Run*: runs the ECF-means algorithm and displays the results (clustering graphical visualization and validation measures output).
2. *Save Results*: saves results to an output csv file.

The second block is where clustering visualization takes shape: dataset points are displayed as circle with the color of the belonging cluster (resulting from the highest value of the probability membership vector) and with an opacity due to the degree of membership to the same cluster (stronger opacity means higher membership). If the attributes of the dataset are two, Voronoi lines (computed considering initial seed) are also displayed. In the top of the block, some input controls are used to affect data visualization. In particular, two combo boxes are used to allow the choosing of dataset's attributes that has to be displayed. Below this, a slider allows to set the degree of membership above which a point is displayed (Membership Threshold t).

Instead, rightmost input fields, in order from top to bottom, control, respectively:

1. The displayed of the Initial Seed Centroid and Mean Seed Centroid points (as a bold colored X).
2. The Outlier Threshold, that controls the o-rank fuzzy outliers, with the meaning that, if a data point has a difference between the two highest values of its probability membership vector less than this value, the point is displayed as a grey squared.

Lastly, in the third box the validation measures are displayed, as described in IV.D, such as PC, PE, MPC, and TI. In addition, Sum of Squared Errors (SSE) and Silhouette (S) measures have also been included; they are calculated considering Initial Seed (IS) and Mean Seed (MS), where MS is the mean value of the measure over all the N iterations, which lead to the definition of IS-SSE, MS-SSE, IS-S, and MS-S.

C. Output Results

The ECF application exports results in csv format, where each row of the output file represents a point x of the dataset. The application appends ECF-means algorithm results as additional columns to the attributes columns of the point x.

TABLE I. COLUMN NAMES MEANING

Column Names	Description
ISCDistance$_i$, with $i = 1, \dots, k$	Vector of Euclidean Distances between point x and Initial Seed Centroids
ISCMembership	Cluster membership derived from the position of the smallest value in ISCDistance vector
MSCDistance$_i$, with $i = 1, \dots, k$	Vector of Euclidean Distances between point x and Mean Seed Centroids
MSCMembership	Cluster membership derived from the position of the smallest value in MSCDistance vector
Membership$_i$, with $i = 1, \dots, k$	Probability vector of point x, $p(x)$
ECFMembership	Cluster membership derived from the $PMA(x)$

Table I shows these additional column names meaning, where Mean Seed Centroid (MSC) is the arithmetic mean value of all computed centroids in N iterations.

VI. CASE STUDY IN METEOROLOGICAL DOMAIN

In order to test the ECF-means algorithm and the validation measures, an historical dataset made up of 9200 meteorological observations has been collected. Data have been retrieved from ECMWF MARS Archive [15] containing the surface Synoptic observations (SYNOP) provided by 4 geographical sites: Charles De Gaulle (CDG) airport in Paris and Grazzanise, Milan, and Pantelleria airports in Italy.

TABLE II. LIST OF METEOROLOGICAL VARIABLES (FEATURES)

#	Name	#	Name
1	Pressure	6	cloud cover
2	three-hour pressure change	7	height of base of cloud
3	wind direction	8	Dewpoint
4	wind speed	9	Drybulb
5	Visibility	10	SITE

SYNOP observations are recorded every hour and the list of the meteorological variables [16] used for applying the ECF-means algorithm is reported in Table II. Each airport site has got 2300 records and the SITE attribute has 4 values.

A. k-means Application

By changing k value (number of clusters) from 2 to 7 and fixing the seed $s = 0$, the k-means algorithm has the silhouette measures of the Table III. These outcomes have been calculated by using the ECF-means application fixing *Initial Seed Number* = 0 and *Iterations* = 1. Considering this measure, the best clustering partition is obtained by selecting $k = 3$.

Fixing $k = 3$ and considering SITE attribute as Class attribute, Classes to Clusters (contingency table) is showed in Table IV. CDG and Milan have been inserted into the same cluster (Cluster 0) by the algorithm (4 sites in 3 clusters): it seems that the two sites have a lot in common! Thus, we try to merge these two sets, obtaining a new set called CDG+MIL.

TABLE III. k-MEANS RESULTS

k	Silhouette	k	Silhouette
2	0.34	5	0.38
3	0.49	6	0.36
4	0.37	7	0.31

TABLE IV. CLASSES TO CLUSTERS

0	1	2	Assigned to cluster
1593	410	297	CDG → Cluster 0
499	1260	541	Grazzanise → Cluster 1
1313	692	295	Milan → Cluster 0
387	589	1324	Pantelleria → Cluster 2
41%	32%	27%	

The incorrectly clustered instances are 3710 and represent 40.32% of the original dataset. k-means does not provide homogeneous clusters with respect to SITE attribute. From an intuitive point of view, the 3 sites (Grazzanise, Pantelleria and CDG+MIL) have an ambiguous meteorological nature and the

 ISBN: 978-1-61208-654-5

3710 unclustered instances are on the border between two or more sites. In other words, the datasets have overlapping areas, with "similar" meteorological conditions, and perhaps the sites are not so different, and they are not well-separated from each other.

If we expected the 3 sites to be able to determine (or clearly separate) even the 3 clusters, we are now disappointed. And nobody knows if this disappointment is due to k-means algorithm or to the fact that the sites are not completely different from each other from a weather (and consequently statistical) point of view.

B. ECF-means Application

The ECF-means algorithm tries to overcome the problem in which the k-means algorithm falls in this meteorological case study, and as we will see in next section, in part it succeeds, if only because it provides much more information on datasets, clusters, and on clustering results, thanks to which the data analyst can make more informed and useful choices.

An Ensemble combination of many runs of k-means by using ECF-means application showed other results and better performances than the single run of the previous section.

By fixing $k = 3$, the ECF-means algorithm provides the results of the Table V that shows how the metrics stabilize as the number I of iterations increases.

TABLE V. ECF-MEANS METRICS

I	MPC	TI	I	MPC	TI
2	0.98703	0.98054	150	0.97716	0.9762
5	0.9917	0.98054	200	0.97975	0.9762
10	0.99066	0.97828	250	0.98138	0.58293
25	0.98657	0.97828	300	0.98269	0.58293
50	0.9877	0.97828	**350**	**0.98339**	**0.58293**
75	0.97428	0.9762	400	0.98083	0.58293
100	0.97612	0.9762	500	0.98131	0.58293

TABLE VI. CLASSES TO CLUSTERS

0	1	2	Assigned to cluster	# of Records
2494	232	69	CDG+Milan → Cluster0	2795
67	1183	141	Grazzanise → Cluster1	1391
53	142	982	Pantelleria → Cluster2	1177
49%	29%	22%		tot. 5363

By selecting $I = 350$, ECF-means has the highest value of MPC $(= 0.98339)$ and the lowest value of TI $(= 0.58293)$. Thanks to the ECF-means application, we are able to select the floor $\underline{S}$ of whole dataset. It has got 5363 records that have the distributions in Table VI. The incorrectly clustered instances are 704 and represent 13.12% of $\underline{S}$.

C. Experimental Results

The results obtained lead to a clear improvement of the clustering: the clusters seem much more separate, if the contingency matrices are calculated starting from the floor set. ECF-means manages to break down the percentage of instances that are incorrectly clustered from 40.32% to 13.12%.

The elements belonging to $\underline{S}$ never fluctuate from one cluster to another (considering the 350 iterations) and constitute approximately 58.3% of the initial dataset. The elements of $S - \underline{S}$, on the other hand, have a more fuzzy nature and we found that 1369 points (about 15% of the initial dataset) have an *Outlier Threshold* (difference between the two highest values of his probability membership vector), less than 0.2.

These very fuzzy points can belong to more than one cluster and probably to more than one airport site (to overlapping areas).

VII. CASE STUDY 2: THE IRIS DATASET

The famous Iris dataset is a multivariate dataset that contains 3 classes of 50 instances each, where each class refers to a species of iris plant (Iris-setosa, Iris-virginica, and Iris-versicolor). Four features were measured from each sample: the length and the width of the sepals and petals, in centimeters. The use of this dataset is very common in classification and clustering tasks, where numerous results have been obtained.

The data set only contains two clusters with rather obvious separation: one of the clusters contains Iris setosa, while the other cluster contains both Iris virginica and Iris versicolor. This makes the dataset a good example for the ECF-means algorithm.

TABLE VII. 2-MEANS RESULTS

Index	Value	Index	Value
MPC	1.00	SSE	12.14
PE	0.00	TI	1.00

Moreover, fixing $k = 2$, the ensemble effect due to the random choice of the two initial centroids via s parameter seems to vanish, because all the iterations always lead to the same result; therefore the clustering validity indexes are (for each I) showed in Table VII.

A. ECF-means Application

Fixing $k = 3$, more interesting results are obtained by applying the ECF-means algorithm. The initial configuration ($s = 0$ and $I = 1$) has the contingency matrix of the Table VIII. The incorrectly clustered instances are 18 and represent the 12% of the original Iris dataset.

TABLE VIII. CLASSES TO CLUSTERS ($s = 0$ AND $I = 1$)

0	1	2	Assigned to cluster
0	0	50	Iris-setosa→ Cluster 2
40	10	0	Iris-versicolor → Cluster 0
8	42	0	Iris-virginica → Cluster 1
32%	35%	33%	

TABLE IX. ECF-MEANS VALIDITY INDEXES

Case	I	TI
1	2-31	0.91333
2	32-1500	0.50000

TABLE X. CLASSES TO CLUSTERS ($I = 31$)

0	1	2	Assigned to cluster
0	0	50	Iris-setosa→ Cluster 2
47	3	0	Iris-versicolor → Cluster 0
14	36	0	Iris-virginica → Cluster 1
41%	26%	33%	

 ISBN: 978-1-61208-654-5

TABLE XI. CLASSES TO CLUSTERS ($I = 31$ AND $\underline{S}$)

0	1	2	Assigned to cluster
0	0	50	Iris-setosa→ Cluster 2
40	3	0	Iris-versicolor → Cluster 0
8	36	0	Iris-virginica → Cluster 1
35%	28%	37%	

By changing the I parameter, mainly the TI index takes two values, as reported in Table IX. Considering case number 1, EFC-means provides the results of the Table X. The incorrectly clustered instances are 17 and represent the 11.33% of the original Iris dataset. Thanks to the ECF-means application, we are able to select the floor $\underline{S}$ of whole Iris dataset S. $\underline{S}$ has got 137 elements that have the distributions in Table XI. $\underline{S}$ has got 11 incorrected clustered instances that represent the 8% of $\underline{S}$. In conclusion, if $t = 1$, then $|C_0^1| = 48$, $|C_1^1| = 39$, and $|C_2^1| = 50$.

B. Experimental Results

Thanks to the obtained results, we can easily understand how the algorithm is able to optimize the partitioning of the data space with respect to the class that expresses the floral typology. The algorithm is able to find this partitioning in one fell swoop. By applying the simple k-means we may not be able to get the same partition. However, the most interesting result is that the algorithm is able to preserve the cluster with Iris-setosa label and to find the floating elements that are at the limits of the floral types. These "disturbing" elements can be analyzed separately in order to understand if they are, from some point of view, outliers or records that have undergone measurement errors.

Moreover, analyzing the floor of S, only a small fraction of Iris-virginica is mixed with Iris-versicolor and only the cluster 0 is modified by the procedure. Also in this case, the 13 elements of $S - \underline{S}$ can be analyzed separately in order to understand their fuzzy nature. These 13 elements have $att(x) = (21, 10, 0)$ and $p(x) = (0.68, 0.32, 0)$. Then they are 0.36-rank fuzzy outliers of S $(p_0(x) - p_1(x) \leq 0.36)$. Furthermore, 7 elements of $S - \underline{S}$ have Iris-versicolor label whilst 6 elements have Iris-virginica label, and all of them have the same maximum degree of membership that is equal to 0.68.

VIII. CONCLUSION AND FUTURE WORKS

In this work, we have presented an algorithm for Ensemble or Aggregation clustering that has, as a simple consequence, the fuzzy reinterpretation of the obtained groupings. We have applied the developed procedure to two different case studies by using a simple software application, which made us understand how this approach helps to explore the dataset, to optimize the results and to assign a degree of membership to each element of the original dataset.

We have had that all the most exciting results can be obtained by the active interaction with the software tool interface, thanks to which, by scrolling the sliders, changing parameters, and visualizing groupings, numerous properties of the dataset can be discovered. In future works, we are going to evaluate our method on other several data sets, for example Datasets from UCI ML Repository.

For the current application, we have chosen the simple k-means as the reference clustering algorithm. Furthermore, we can consider other algorithms in substitution or in addition to it, and this will surely be the next improvement of the tool.

ACKNOWLEDGMENT

The authors would mention the TECVOL II and Big Data Facility projects, both funded by the Italian PRORA, in which the meteorological database has been realized and the tool has been designed and developed.

REFERENCES

[1] P. Berkhin, "Survey of clustering data mining techniques," Technical report, Accrue Software, San Jose, CA, 2002.

[2] A. K. Jain, A. Topchy, M. H. C. Law, and J. M. Buhmann, "Landscape of Clustering Algorithms," IAPR International Conference on Pattern Recognition, vol. 1, pp. 260-263, Cambridge, UK, 2004.

[3] L. I. Kuncheva, "Combining Pattern Classifiers. Methods and Algorithms," Wiley-Interscience, A John Wiley & Sons, Inc., Publication, 2004.

[4] A. Strehl and J. Ghosh, "Cluster Ensembles – A Knowledge Reuse Framework for Combining Multiple Partitions," Journal of Machine Learning Research 3 (2002), pp. 583-617.

[5] S. Monti, P. Tamayo, J. Meisirov, and T. Golub, "Consensus Clustering: A Resampling-Based Method for Class Discovery and Visualization of Gene Expression Microarray Data," Machine Learning, 52, pp. 91–118, 2003, Kluwer Academic Publishers.

[6] A. Gionis, H. Mannila, and P. Tsaparas, "Clustering Aggregation," on 21st International Conference on Data Engineering (ICDE), pp. 341-352, 2005.

[7] R. R. Bouckaert et al., "WEKA Manual for Version 3-9-2," the University of Waikato, Hamilton, New Zealand, December 22, 2017.

[8] E. Frank, M. A. Hall, and I. H. Witten, "Data Mining: Practical Machine Learning Tools and Techniques," Morgan Kaufmann, 2016.

[9] G. Gan, C. Ma, and J. Wu, "Data Clustering. Theory, Algorithms, and Applications," ASA-SIAM Series on Statistics and Applied Probability, SIAM, Philadelphia, ASA, Alexandria, VA, 2007.

[10] C. C. Aggarwal, "Data Mining. The Textbook," Springer International Publishing, 2015.

[11] J. C. Bezdek, R. Ehrlich, and W. Full, "FCM: The Fuzzy c-Means Clustering Algorithm," Computers & Geosciences Vol. 10, No. 2-3, pp. 191-203, 1984.

[12] J. Wu, "Advances in k-means Clustering. A Data Mining Thinking," Springer, 2012.

[13] R. N. Dave, "Validating fuzzy partition obtained through c-shells clustering," Pattern Recognition Lett. 17, pp. 613– 623, 1996.

[14] M. Hall et al. (2009), "The WEKA Data Mining Software: An Update," SIGKDD Explorations, Volume 11, Issue 1.

[15] ECMWF, Mars User Guide. User Support. Operations Dep. 2013.

[16] G. Zazzaro, G. Romano, and P. Mercogliano, "Data Mining to Forecasting Fog Events and Comparing Geographical Sites. Designing a novel method for predictive models portability," Int. Journal on Advances in Nets and Services, vol. 10 no 3 & 4, pp. 160-171, 2017.

 ISBN: 978-1-61208-654-5

Analyzing Data Streams Using a Dynamic Compact Stream Pattern Algorithm

Ayodeji Oyewale
School of Computing, Science& Engineering
University of Salford
Salford, Manchester, United Kingdom
E-mail: a.oyewale@edu.salford.ac.uk

Chris Hughes
School of Computer, Science & Engineering
University of Salford
Salford, Manchester, United Kingdom
E-mail: c.j.hughes@salford.ac.uk

Mohammed Saraee
School of Computing, Science & Engineering
University of Salford
Salford, Manchester, United Kingdom
Email: m.saraee@salford.ac.uk

***Abstract*— In order to succeed in the global competition, organizations need to understand and monitor the rate of data influx. The acquisition of continuous data has been extremely outstretched as a concern in many fields. Recently, frequent patterns in data streams have been a challenging task in the field of data mining and knowledge discovery. Most of these datasets generated are in the form of a stream (stream data), thereby posing a challenge of being continuous. Therefore, the process of extracting knowledge structures from continuous rapid data records is termed as stream mining. This study conceptualizes the process of detecting outliers and responding to stream data. This is done by proposing a Compressed Stream Pattern algorithm, which dynamically generates a frequency descending prefix tree structure with only a single-pass over the data. We show that applying tree restructuring techniques can considerably minimize the mining time on various datasets.**

Keywords- Data Mining; Frequent Pattern (FP); Stream data; Compact Pattern Stream (CPS) & Interactive Mining, Path Adjustment Method (PAM), Branch Sort, Merge Sort Algorithm.

I. INTRODUCTION

Within the global business world, understanding data is crucial to success. A lot of research has been dedicated to the computation on data where most companies are able to collect enormous amounts of it with relative ease. Without doubt, many companies now have more data than they can handle, a vital portion of this data entails large unstructured data sets that amount up to 90 percent of an organization's data. The ability to mine and analyse data, in any form, from many sources, gives us deeper and richer insight into business patterns and trends, helping drive operational efficiencies and competitive advantage in manufacturing, marketing, security and IT at large [4]

In an ideal corporate world, competitiveness should be successfully and sustainably built on data; however, the reality is that obtaining good quality data for decision-making in this milieu is a big ordeal [1]. Detecting meaningful patterns in streaming applications is particularly challenging. The detector must process data and output a decision in real-time, rather than making many passes through batches of files. In most scenarios the number of streams is large and there is little opportunity for human, let alone expert intervention. As such, operating in an unsupervised, automated fashion (e.g., without manual parameter tweaking) is often a necessity.

Data streams are continuous, changing sequence of data that constantly arrive at a system and needs to be processed in near real-time. The dissemination of data stream phenomenon has necessitated the development of diverse range of stream mining algorithms. Several studies [2] describe the approaches currently being used to overcome the challenge of storing and processing fast, continuous and uninterrupted streams of data.

Subsequently, latter sections of this paper enumerates on the data models which is found in section two, the approach to exploit the CSP algorithm is introduced in section three. Section four explains the criteria through which the algorithm is evaluated. The procedures in analyzing a stream of data are explained in section five.

II. DATA MODELS

When weighed against data in traditional databases, data streams are unbounded and the number of transactions increases over time. As a result of these, different data models for effective processing have been suggested in different mining algorithms.

The adapted model is based on a sliding window. That is, despite the infinite arrival of the stream data, the frequent itemsets are derived based on the most recent data that is being captured within a stipulated sliding window where the present time signifies end point of that window.

One justification for such a sliding-window model is that due to temporal locality, the data in streams is bound to change with time, and many a times people are interested in the most recent array and patterns from the stream data(2).

Data streams differ from the conventional stored relation model in several ways:

i.) The data elements in the stream arrive online.

ii) The system has no control over the order in which data elements arrive to be processed, either within a data stream or across data streams.

iii) Data streams are potentially unbounded in size.

 ISBN: 978-1-61208-654-5

iv) the instance an element from a data stream has been processed it is discarded or archived — unless specifically stored in an external storage point it cannot be retrieved easily, which ideally is small relative to the size of the data streams. Frequently, information stream inquiries may perform joins between information streams and put away social information.

For the motivations behind this paper, we will accept that if put away relations are utilized, their substance stay static.

Therefore, we block any potential exchange preparing issues that may emerge from the nearness of updates to put away relations that happen simultaneously with information stream handling.

III. APPROACH TO CSP ALGORITHM

The algorithm uses data passed to it to construct a CSP tree, which is always in ready state to be mined. There are several techniques out there in building this kind of algorithm; most of them work by firstly constructing FP tree then restructuring the FP tree into CSP. At first, transactions in the data stream are inserted into the CSP-tree based on a predefined item order (e.g., lexicographical item order). This item order of the CSP-tree is maintained by a list, called the I-list, with the respective frequency count of each item. After inserting some transactions, if the item order of the I-list deviates significantly from the current frequency- descending item order, the CSP-tree is dynamically restructured by the current frequency-descending item order and the I-list updates the item order with the current one.

In contrast, the technique used in the CSP algorithm allows for direct development of frequency-descending item order list from data. This will save the algorithm from iterating over tree nodes several times during the creation and modification of tree. Sliding window is used in this work where data are captured in panes which are housed in windows. When new data is inserted, the window slides thereby removing some old pane and inserting new ones; depending on sliding window size. It constantly updates itself by extracting the expired transaction after each window slides. Ensuring that the tree does not contain unwanted data points .

Trans ID	Transactions	
A06	a, b, c, f, g	-Window1
A07	a, c, f, g	
A08	b, d, e, f	
A09	b, c, d, e	--Window2--
A10	a, d, f, g	
A11	a, b, c, d	

Figure 1 Transaction with Window

Each transaction is processed with the definition of window size and pane size as show the transaction table above. A stream of data of window *n* = size 2, pane size = 2 During sliding of the window, the data with transaction id from A06 to A09 are being processed in the first window, the reason for this is that a window is of size 2 which means a single window can have a minimum of two panes, and each pane in return holds two transactions, and each window therefore contains four transactions. The window slide is one, which makes the window to move one pane at a time, one pane contains two transactions.

A. Formatting of Data

The algorithm does not work on arbitrary data, it expects a dictionary with the item sets as the dictionary keys and the frequency as the value.

```
def
create_init_set(data_set):
        ret_dict={}
       for trans in data_set:
  if frozenset(trans) in ret_dict.keys():
    ret_dict[frozenset(trans)] += 1
      else:
       ret_dict[frozenset(trans)] = 1
return ret_dict
```

This code snippet creates new dictionary and fills it with the transaction, it is the dictionary that will be supplied to the algorithm

B. Creation of Window

In order to study the characteristics of dynamic flow, it is eminent to configure window size since the appropriate window size is a determinant in effectively carrying out an analysis on the datasets.

Creation of new window requires;

a) Size of window: maximum number of pane the window will contain
b) Size of pane, and
c) Sliding size

```
self.window = Window (self. windowSize, self.paneSize,
                      self.slideSize)
```

The code snippet above defines the new window that needs to be created after pan slide.

IV. ALGORITHM EVALUATION

Through theoretical and experimental analysis, frequent-item identifying algorithms are often evaluated based on three aspects:

- accuracy,
- runtime, and
- space usage

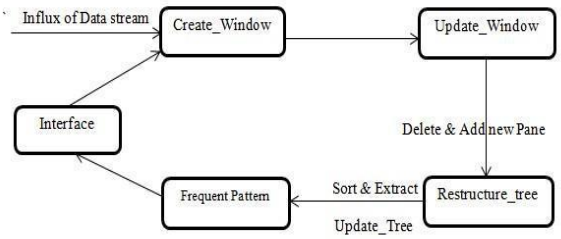


Figure 2: Data-Flow Architecture

A successful connection should be established which makes easy passage of the data. A given transaction is introduced into the current sliding window with its respective support value. The sliding window is used in the next update window module which takes the input from window creation module and deletes the old panes thereby adding new panes so as to mine the latest frequent patterns. Thereafter, the updated window is passed as input to the restructure module which in turn performs the extraction, sorting and reinsertion operation. The sorted, extracted data is also passed as input to the final module which mines all the transactions over dynamic data streams and finds the latest frequent patterns. The discovered patterns which are greater than threshold value are finally displayed as the set of latest frequent patterns over the dynamic data stream.

In order to restructure the CPS-tree, we use our proposed efficient tree restructuring mechanism, called Branch sorting method (BSM) [6], and the Path adjusting method proposed in [5]. As the CPS-tree is developed within the current window, a base up FP tree mining procedure is used to create exact set of most recent frequent patterns. The mining task is very proficient due to the recurrence plummeting tree structure.

V. PROCEDURES IN ANALYZING STREAM DATA

In order to develop and analyse an incessant influx of data, it is required to take into account vital information with respect to stream data.

1) Data preprocessing:

Information preprocessing in this examination work includes the utilization of a middleware which fills in as a working shrouded interpretation layer which empowers a correspondence and collaboration, and information administration between the working framework and the application running on it.

2) The middleware:

The middleware class is made to assemble skeleton of communication between the system and the information it is intended to be processed. The constructor of this class requires record name of the document that ought to be prepared, the thing column(s) that the calculation should process and the value-based section the thing ought to be coordinated against.

```
_  def  init  (self, file, transaction_field, item_field):
```

The middleware class has a function call *format_data* which is used to handle data that returns the algorithms specific data type. The middleware also declares an abstract method named *process_data* that must be implemented by every subclass of the class.

3) Sliding Window:

A sliding window algorithm places a buffer between the application program and the network data flow. For most applications, the buffer is typically in the operating system kernel, but this is more of an implementation detail than a hard-and-fast requirement. The sliding window technique inspects every time window at all scales and location over a time stamp which means our data will be classified according to the most recent item set provided. Typically, sliding window algorithms serve as a form of flow control for data transfers. Datasets in a sliding window are often described in a structure.

$$(Di) = \frac{\{So, S1 \ldots Si\} i \leq n-1}{\{Si, Si+1 \ldots . Sn + i - 1\} \geq n-1} \quad (1)$$

If the timespan (length, l) of a window is denoted with *n*

However; data at a point Ti in the window is denoted by

Where; Di: Dataset present in the sliding window

Si: Data values of the dataset at the point i

VI. RESULT ANALYSIS

An empirical evaluation of the performance of CSP implementation is made. A comparative analysis of CSP tree with FP tree algorithm is done using. All programs are done using Oython 3.0 and executed in WInsows 7 on a 2.66 GHz CPU with 1GB memory usage. And this is done using the BSM, Sales and Telecom datasets.

TABLE I: DATASET CHARACTERISTICS

Datasets	No of Transactions	Size(M)
BSM	515,597	2GB
Sales	88, 475	9.7M
Telecoms	9,683,900	10.6GB

The aforementioned Table I above shows the characteristics of the datasets used for the analysis. The BSM dataset contains several entries from an electronics retailer. The Sales dataset consists of retail Watson Analytics Sample Data of Sales Products and the telecoms datasets consists of the call detail record of customers. In this experiment, the average runtime of all active windows

 ISBN: 978-1-61208-654-5

are computed for the two algorithms on each data. At every window, mining is perfromed on the recent window. The pane size is dependent on the size of the dataset introduced.

TABLE II: RUNTIME ANALYSIS

Test Runtime	CSPTree			FPTree			Window Size	Pane Size
	Creation	Restruct	Mining	Creation	Restruct	Mining		
	1.87	2.97	0.00040	2.53	3.64	0.00021	2	10K
	6.17	6.76	0.00107	11.02	11.02	0.00073	4	10K
	7.90	9.23	0.00301	13.20	15.09	0.00221	5	10K

A runtime analysis for the three dataset is shown in the Table II which comprises of the tree creation, tree restructuring and how long it takes to mine a specific dataset present in the pane of a window.
It represents the measure of amount of time needed for the CSP algorithm to execute each stream of data presented to it in comparison with that of the FPTree.

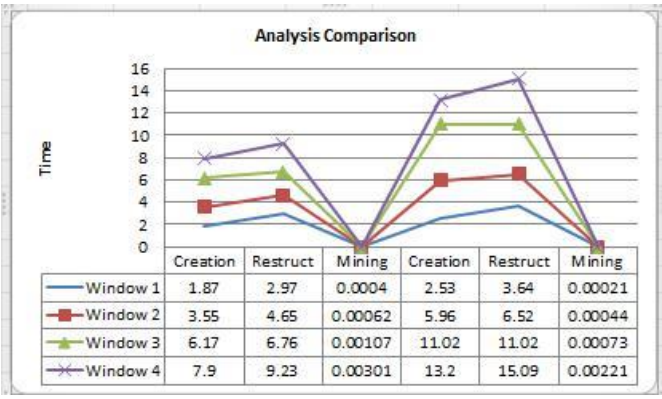


Figure 3: CSP AND FP Comparison on BSM Data

A comparative analysis for BMS dataset between CSP Tree and FP tree is shown in the Figure 3 above. The graph indicates precisely that at each level of data creation, restructuring and mining, CSP tends to outpace FP Tree.

TABLE III: RUNTIME ANALYSIS ON SALES DATA

Sales

Total Runtime	CSPTree			FPTree			Window Size	Pane Size
	Creation	Restruct	Mining	Creation	Restruct	Mining		
	0.00017	0.00303	5.09e-5	9.27e-5	0.00135	3.90e-5	2	150
	0.00014	0.00166	3.90e-5	0.00017	0.00170	2.63e-5	3	150
	0.00031	0.00193	4.35e-5	0.00017	0.00343	6.89e-5	4	150
	0.00021	0.00312	4.64e-5	0.00016	0.00305	3.15e-5	5	150

The runtime analysis on Sales dataset shown in a Table III above comprises of the time it takes for a tree to be created before restructuring can take place. Thereafter, the tree restructured based on the new set of incoming dataset. It also shows how long it takes to mine a specific dataset present in the pane of a predefined window.

TABLE IV: RUNTIME ANALYSIS ON TELECOMS DATASET

Total Runtime	CSPTree			FPTree			Window Size	Pane Size
	Creation	Restructure	Mining	Creation	Restructure	Mining		
	0.01738	2.95e -5	0.00026	0.01272	2.79e -5	4.19e -5	2	700
	0.02285	8.62e -6	0.00031	0.019948	8.62e -6	0.00025	3	700
	0.02567	9.44e -6	0.00043	0.02543	1.02591	0.00029	4	700
	0.02475	8.62e -6	0.00032	0.01855	9.03e -6	0.00026	5	700

The runtime analysis in Table IV is the Test runtime for the Telecoms dataset which comprises of the tree creation, tree restructuring and how long it takes to mine a given instance of the telecoms data present in the pane of a window. As can be seen, the mining rate of CSP is quite faster when compared to that of Fp tree.

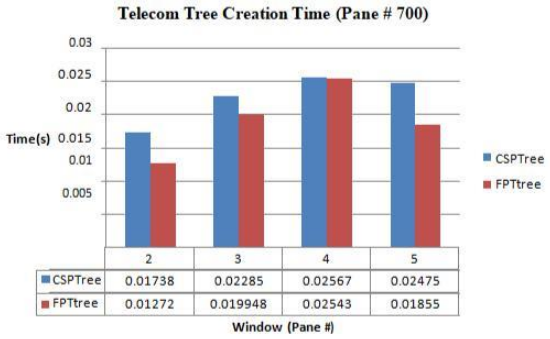


Figure 4: Tree Creation

The time it takes for the algorithm to create a tree for the telecoms between CSP Tree and FP tree is drawn up in the Figure 4 above. The resulting outcome explains that FP Tree in the first window has a faster tree creation rate compared to CSP Tree. And at each subsequent window the time it takes to create the tree gradually lowers till the necessary patterns are derived.

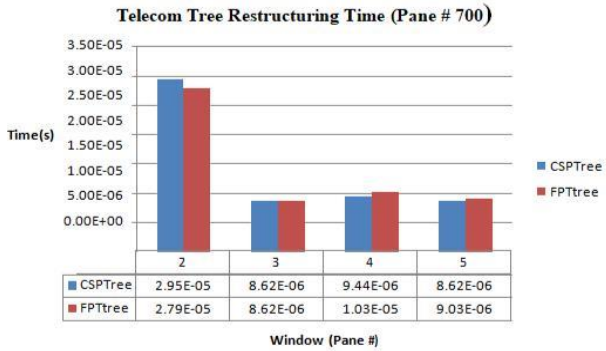


Figure 5: Tree Restructuring

 ISBN: 978-1-61208-654-5

Also shown above is the time it takes to restructure each incoming tree for the two algorithms. In the first window, FP tree restructures faster but in subsequent windows CSP Tree outperforms it. The following datasets are being analysed with respect to the time of tree creation, the time rate for restructuring and how long it takes to mine the data in conjunction with the varying time window. This juxtapose has been made to make a comparative analysis between CSP and FP algorithm. The time it takes to create a tree using CSP algorithm is shorter when compared to FP algorithm. It takes CSP 1.8sec to build a tree while it takes FP algorithm over 3 sec. The figure 5 below shows a graphical advantage of CSP over FP.

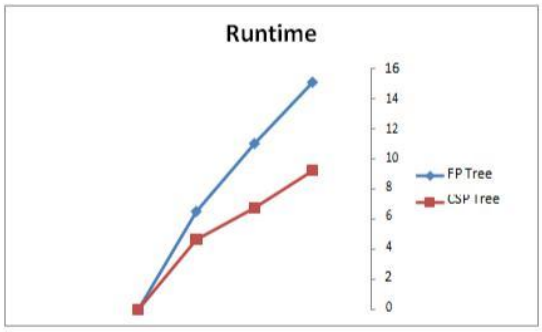


Figure 6: Graphical Advantage of CSP over FP

The above Figure 6 depicts the Tree restructuring phase of the all dataset. While restructuring the tree, we use the Path Adjustment method (PAM) which depends on the Degree of Displacement of two items and it swaps two nodes with Bubble sort method. And also, the branch sort algorithm makes use of the merge sort approach. Due to the high rate of displacement while using PAM, it is unsuitable for the algorithm hence BSM performs better during tree restructuring approach with merge algorithm. The merge sort method unlike quicksort makes use of Divide and Conquer algorithm. The most recent input array is intermittently divided in two halves; it sorts the two halves and then merges the two sorted halves.

VII. CONCLUSION

We have proposed CSP-tree that dynamically achieves frequency-descending prefix tree structure with a single-pass by applying tree restructuring technique and considerably reduces the mining time. We also adopted Branch sorting method using merge sort which is a new tree restructuring technique and presented guideline in choosing the values for tree restructuring parameters. It shows that despite additional insignificant tree restructuring cost, CSP-tree achieves a remarkable performance gain on overall runtime. The easy-to-maintain feature and property of constantly summarizing full data stream information in a highly compact fashion facilitate its efficient applicability in interactive, incremental and stream data.

REFERENCES

[1] E. Ascarza, P. Ebbes, O. Netzer and M. Danielson, Beyond the Target Customer: Social Effects of CRM Campaigns.2016.

[2] B. Babcock, S. Babu, M. Datar, R. Motwani, J. Widom, Models and issues in data stream systems.*In:Proceedings of 21st Acmigmod-Sigact-Sigart symposium on principles of database systems*(PODS' 02), pp 1–16. 2002.

[3] B. Borja, C. Bernardino, C. Alex, R. Gavald`a, M. David Manzano-Macho, The Architecture of a Churn Prediction System Based on Stream Mining. 2013.

[4] Intel IT, IT Best Practices Business IntelligenceRetrieved from http://www.intel.com/it/.Feb 2012.

[5] J.-L. Koh and S.-F. Shieh. An efficient approach for maintaining association rules based on adjusting FP-tree structures. In Lee Y-J, Li J, Whang K-Y, Lee D (eds) Proc. of DASFAA 2004. Springer-Verlag, Berlin Heidelberg New York, 417–424. , 2004

[6] S. K. Tanbeer, C. F. Ahmed, B. S. Jeong,, and Y.-K. Lee. CP-tree: A tree structure for single-pass frequent pattern mining. In Proc. of PAKDD, Lecture Notes Artif Int, 1022-1027. 2008

[7] T.A. Rashid, Convolutional Neural Networks based Method for Improving Facial Expression Recognition. In: Corchado Rodriguez J., Mitra S., Thampi S., El-Alfy ES. (eds) *Intelligent Systems Technologies and Applications 2016*. ISTA 2016. Advances in Intelligent Systems and Computing, vol 530. Springer, Cham. Oct 2016.

 ISBN: 978-1-61208-654-5

Opinion Leaders in Star-Like Social Networks: A simple Case?

Michael Spranger, Florian Heinke, Hanna Siewerts, Joshua Hampl and Dirk Labudde
University of Applied Sciences Mittweida
Mittweida, Germany
Email: {*name.surname*}@hs-mittweida.de

***Abstract*—In recent years, the automated, efficient and sensitive monitoring of social networks has become increasingly important for criminal investigations and crime prevention. Previously, we have shown that the detection of opinion leaders is of great interest in forensic applications. In the present study, it is argued that state of the art opinion leader detection methods have weaknesses if networks exhibit star-like social graph topology, whereas these topologies result from the interaction of users with similar interests. This is typically the case for Facebook pages of political organizations. In these cases, the underlying topologies are highly focused on one (or only a few) central actor(s) and lead to less meaningful results by classic measures of node centrality commonly used for leader detection. The presents study examines these aspects closer and exemplifies them with the help of data collected from the Facebook page of a German political party for five consecutive months. Furthermore, a quantitative indicator for describing star-like network topologies is introduced and discussed. This measure can be of great value in assessing the applicability of established leader detection methods. Finally, a modified LeaderRank score is proposed – the CompetenceRank – which aims to address discussed problems.**

Keywords–Forensic; Opinion Leader; Graph Theory.

I. INTRODUCTION

The detection of opinion leader has been discussed extensively in the past few years. Based on the work by Katz [7] many approaches have been presented. In this paper it will be shown that in some situations these approaches do not capture the core of the problem and as a result lead to an inaccurate assessment of opinion leadership. This section shall give a brief introduction to the field in which such situations occur as well as an overview of topology-based approaches and finishes with the scope and structure of the paper.

A. General Motivation

Analyzing social networks has become an important tool for investigators, intelligence services and decision makers of police services. The information gained this way can be used to solve crimes by searching for digital evidence that relates to the crime in the real world. Additionally, methods of predictive policing can help to organize police missions as was shown in [1]–[3]. The detection of opinion leaders in social networks is an important task for different reasons. On the one hand, owners of influential profiles are often also influential in the offline world. Knowing these people helps to determine the direction of an investigation or more concretely to target persons of interest. On the other hand, as was suggested in previous work [3], it might be of interest to contact these profiles by means of chatbots to gain access into closed groups in an effort to gather important information for intelligence services. Intuitively, opinion leaders, when considered as nodes with high structural importance, can be detected with the help of centrality measures. However, different kinds of influence in a network have to be distinguished. Nodes can have a great influence as corresponding actors are able to spread information fast and widely in a network, or they can have a great influence because they write something of importance, which attracts many other users in the network to respond.

B. Leader Detection by means of Network Centrality Measures

In the literature, one can mainly find centrality measures for the former type of influence. For example, highly active profiles can be recognized using degree centrality, meaning, the relative number of outgoing edges of a node. These profiles are represented by nodes with a high degree centrality which are especially useful to spread information in a network due to their high interconnectedness.

In this context, the closeness centrality – the inverse of the mean of the shortest path of a node to any other node in the network – is even more effective. It describes the efficiency of the dissemination of information of a certain node.

Furthermore, the betweenness centrality of a certain node, which is defined as the number of shortest paths between two nodes that cross this node, describes the importance of this node for the dissemination of information in a network. Therefore, the higher the betweenness centrality of a node, the greater its importance for the exchange of information in a network.

Moreover, the eigenvector centrality of a node is defined as the principle eigenvector of the adjacency matrix of a network. In contrast to the measures discussed beforehand, PageRank [4], as one of the best measures of node centrality, does not only consider the centrality of the node itself, yet also of its neighboring nodes.

As part of the opinion leader detection research, LeaderRank [5] was introduced as a further development of PageRank in order to find nodes that spread information further and faster. However, all of these centrality measures consider nodes that are involved in the dissemination of information mainly based on their activity. For the purpose of the intended usage, users who achieve high impact through what they have written are of much greater interest. Thus, similar to the citation of papers and books and its impact on the author's reputation, the importance of a node has to be higher when it reaches a high number of references and citations with low activity.

Interestingly, Li et al. considered the so-called node spreadability as the ground truth for quantifying node importance in a subsequent study [6]. Node spreadability is based on a straightforward Susceptible-Infected-Removed (SIR) infection

 ISBN: 978-1-61208-654-5

model from which the expected number of infected nodes upon initially infecting the node in question is estimated. However, this expected number can only be estimated from simulation, which furthermore is dependent on the parameterization of the SIR model. In this respect, all centrality measures can be considered as heuristic approximations of node spreadability.

C. Scope and Structure of the Paper

In this case study, we discuss problems that can arise when aiming to detect opinion leaders in social networks yielding highly central topologies similar to star graphs. Examples for such networks are group pages on Facebook or vk.com where user interactions and activities are mostly triggered by and focused on posts made by the page owner. In such cases, the page owner – a trivial leader in the sense of centrality measures discussed above – acts as a score aggregator and can thus lead to distorted scoring, which can eventually be adverse in the context of opinion leader detection. In this case, classic centrality measures can be considered inappropriate. Based on interactions of users of the political Facebook page "DIE LINKE" tracked for five consecutive months (January - May 2017), this problem is illustrated. We further introduce the LeaderRank skewness as a quantitative measure of aggregator-induced distorted LeaderRank scoring, which in experiments show to be superior to network entropy with respect to expressiveness. Finally, a modified LeaderRank score, we refer to as CompetenceRank, is introduced which is proposed to be suitable for opinion leader detection in such networks.

The paper is structured as follows: in Section II, a brief literature overview on the topic of opinion leader detection is given, followed by a summary of the LeaderRank algorithm. Issues of LeaderRank scoring in star-shaped network topologies are discussed in Section III, including the deduction and definition of LeaderRank skewness. In Section IV, the social network dataset in question is discussed. The CompetenceRank is introduced in Section V. We finally give a conclusion as well as an overview on future work in Section VI.

II. Detection of Opinion Leaders

Opinion leaders in the context of the intended analysis of social networks are individuals, who exert a significant amount of influence on the opinion and sentiment of other users of the network through their actions or by what they are communicating. In social sciences the term 'opinion leader' was introduced before 1957 by Katz and Lazarsfeld's research on diffusion theory [7]. Their proposed two-step flow model retains validity in the digital age, especially in the context of social media.

Katz et al. assume that information disseminated in the Social Network is received, strengthened and enriched by opinion leaders in their social environment. Each individual is influenced in his opinion by a variety of heterogeneous opinion leaders. This signifies, that the opinion of an individual is mostly formed by its social environment. In 1962, Rogers referenced these ideas and defined opinion leader as follows:

> "Opinion leadership is the degree to which an individual is able to influence informally other individuals' attitudes or overt behavior in a desired way with relative frequency." [8, p. 331]

For the present study, one important question to answer is what influence means, or rather how to identify an opinion leader or how the influencer can be distinguished from those being influenced. Katz defined the following features [7]:

1) personification of certain values,
2) competence,
3) strategic social location.

One approach to identify opinion-leaders is to extract and analyze the content of nodes and edges of networks to mine leadership features. For instance, the sentiment of communication pieces can be analyzed to detect the influence of their authors, as shown by Huang et. al., who aim to detect the most influential comments in a network this way [9]. Another strategy is to perform topic mining to categorize content and detect opinion leaders for each topic individually, as opinion leadership is context-dependent [7] [10]. For this purpose, Latent Dirichlet Allocation (LDA) [11] can be used, as seen in the work of [12].

In this study, we considered the implementation of content-based methods problematic, as texts in social networks mostly lack correct spelling and formal structure, which impairs such methods' performance. Additionally, leaders can be identified by analyzing the flow of information in a network. By monitoring how the interaction of actors evolves over time, one can identify patterns and individuals of significance within them. To achieve this, some model of information propagation is required, such as Markov processes employed by [13] and the probabilistic models proposed by [14]. These interaction-based methods consider both topological features and their dynamics over time.

We utilized methods, which are solely based on a network's topology, therefore, consider features, such as node degree, neighborhood distances and clusters, to identify opinion leaders. One implementation of this is the calculation of node centrality. The underlying assumption is that the more influence an individual gains, the more central it is in the network. Which centrality measure is most suitable is dependent on the application domain. We judged eigenvector centrality to be most adequate. One of the most popular algorithms is Google's PageRank algorithm [4]. The application of PageRank for the purposes of opinion leader detection has seen merely moderate success [15] [16]. With LeaderRank scores, Lü et al. advocate further development and optimization of this algorithm for social networks, and have achieved surprisingly good results [5]. Herein, users are considered as nodes and directed edges as relationships between opinion leaders and users. All users are also bidirectionally connected to a ground node, which ensures connectivity as well as score convergence. In short, the algorithm is an iterative multiplication of a vector comprised by per-node scores $s_i(t)$ at iteration step t with a weighted adjacency matrix until convergence is achieved according to some convergence criteria. Initially, at iteration step t_0, all vertex scores are set to $s(0) = 1$, except for the ground node score which is initialized as $s_g(0) = 0$. Equation (1) describes LeaderRank algorithm as a model of probability flow through the network, where $s_i(t)$ indicates the score of a node i at iteration step t.

$$s_i(t+1) = \sum_{j=1}^{N+1} \frac{a_{ji}}{k_j^{out}} s_j(t) \tag{1}$$

Depending on whether or not there exists a directed edge

from node i to node j, the value 1 respectively 0 is assigned to a_{ij}. k_i^{out} describes the number of outgoing edges of a node. The final score is obtained as the score of the respective node at the convergence step t_c and the obtained ground node score, as shown in (2). At t_c, equilibration of LeaderRank scores towards a steady state can be observed.

$$S_i = s_i(t_c) + \frac{s_g(t_c)}{N} \tag{2}$$

The advantage of this algorithm compared to PageRank is that the convergence is faster and above all that nodes, that spread information faster and further, can be found. In later work, for example, by introducing a weighting factor, as in [6] or [17], susceptibility to noisy data has been further reduced and the ability to find influential distributors (hubs) of information has been added.

III. Issues with LeaderRank

The LeaderRank algorithm can be understood as a reversion of a discrete model of diffusion. In that sense, the initialization $s_i(0) = 1$ at t_0 can be interpreted as assigning a uniform concentration distribution of some virtual compound which in the processes is re-distributed according to the model. In that respect, central actors showing the highest activity in star-like networks can induce score aggregation and migration towards their central nodes as well as their adjacent nodes, whereas nodes in the 'peripheral region' of the network become inadequately represented by their scores. One can thus hypothesize that ranked lists obtained from LeaderRank scores can not be considered meaningful if a given network in question exhibits star-topological topology.

Furthermore, the LeaderRank emphasizes the strategic social location of a user, whereas their competence is not considered. In star-shaped network topologies, high centralities of only a fraction of nodes leads to a heavily skewed LeaderRank score distribution.

In this case study, the network under investigation shows an even more extreme case of star topology in which the owner of the political Facebook page 'DIE LINKE' acts solely as the central actor (for more information see Section IV). In contrast, one could argue that someone is more important if any activity generates a high number of responses. Such a case is regularly given by political networks, which are dominated by the central node of the page owner. Thus, a straightforward modification of the LeaderRank score is proposed in Section V which addresses the imbalance the LeaderRank algorithm yields in such networks.

In the following paragraph a quantitative measure of LeaderRank distribution skewness is proposed, which could aid to ensure proper applicability of the LeaderRank algorithm for any given network. This measure is further compared to the classic measure of network entropy. Tests on simulated data show the LeaderRank skewness to be superior to network entropy with respect to topological changes.

A. Definition of LeaderRank Distribution Skewness

Let $LR = \{lr_1, ..., lr_i, ..., lr_N\}$ be the LeaderRank values of all nodes. Further, $\overline{lr}$ and sd_{LR} denote the arithmetic mean and standard deviation of LR. Based on the z-scaled LeaderRank values (3), the skewness ν of the LeaderRank distribution is calculated as shown in (4).

$$z(lr_i) = \frac{lr_i - \overline{lr}}{sd_{LR}} \tag{3}$$

$$\nu_{LR} = \left| \frac{1}{N} \sum_i z(lr_i)^3 \right| \tag{4}$$

As discussed above, score distribution skewness is correlated with network topology. Yet, normalization of computed skewness is required in order to make predications about the topology and whether a star-like topology is present. Thus upper and lower bounds, ν_{min} and ν_{max}, are needed. In this paragraph, derivation of both bounds are given.

Trivially, ν converges to the lower bound – the theoretical minimum ($\nu = 0$) – in almost regular graphs. Such graphs are regular graphs with one edge being removed. With N being sufficiently large, the supposition that $lr_i \approx lr_j$ of any pair of randomly selected vertices v_i and v_j holds true and a limit of $\lim_{sd_{LR} \to 0} \nu = 0$ can be assumed. In regular graphs however all LeaderRank scores are equal by definition, resulting to $sd_{LR} = 0$ and ν being undefined in this case.

In contrast, ν is equal to the theoretical maximum if the network graph exhibits a strictly star-shaped topology. Let lr_c be the value of the LeaderRank of the central vertex of such a network. The LeaderRank values of any randomly selected pair of vertices v_i and v_j with $v_i, v_j \neq v_c$ are then not distinguishable, i. e., $lr_i = lr_j$, according to the LeaderRank's definition. Furthermore, the total LeaderRank generally sums up to N, $LR_{tot} = \sum_{i=1}^{N} lr_i = N$ (which leads to $\overline{lr} = 1$ for every graph). Given the central node lr_c, each lr_i can thus be calculated as shown in (5).

$$lr_i = \frac{N - lr_c}{N - 1} \tag{5}$$

If lr_c is known, the set of LeaderRank values $\{lr_c, lr_2, ...lr_i, ...lr_N\}$ and the resulting ν_{max} can be derived. For star graphs of any size N, a linear correlation exists between lr_c and N ($lr_c \approx 0.20N + 0.66, R = 1.0$, data not shown). The upper skewness bound ν_{max} can thus be readily computed. Subsequently, for any irregular network graph LeaderRank skewness can be calculated and normalized subsequently using a min-max normalization as denoted in (6), whereas ν_{min} can be assumed as 0 as discussed above.

$$\hat{\nu} = \frac{\nu - \nu_{min}}{\nu_{max} - \nu_{min}} = \frac{\nu}{\nu_{max}} \tag{6}$$

B. Detection of star topology

LeaderRank skewness $\hat{\nu}$ can be utilized to indicate adverse leader ranking by means of LeaderRank scores. In this section, we compare ν to the classic measure of network entropy (denoted as H in the following text). In order to allow direct comparison to $\hat{\nu}$ as well as to entropies computed from other graphs, H is required to be normalized analogously to $\hat{\nu}$. In this subsection, we give a brief overview on how normalization can be conducted.

Let A be the adjacency matrix of a network with N vertices, where each element $a_{ij} := 1$, if there exists a directed edge e_{ij} between adjacent vertices v_i and v_j. Each element of

 ISBN: 978-1-61208-654-5

the principal diagonal a_{ii} is defined as $a_{ii} := \deg(v_i)$ and thus corresponds to the degree – the sum of the incoming and outgoing links – of vertex v_i. The trace of A is defined as the sum of all elements of the principal diagonal: $tr(A) = \sum_{i=1}^{N} a_{ii}$. The formalism for graph entropy used by Passerini and Severini $S(\rho) = -tr(\rho \log_2 \rho)$ [18] is based on the von Neumann entropy and can be adapted as follows:

$$\begin{aligned} S(\rho) &= -tr(\rho \log_2 \rho) \\ &= -\sum_{i=1}^{N} \rho_i \log_2 \rho_i \\ &= -\sum_{i=1}^{N} \frac{a_{ii}}{tr(A)} \log_2 \frac{a_{ii}}{tr(A)} \\ &= -\sum_{i=1}^{N} \frac{\deg(v_i)}{\sum_{j=1}^{N} \deg(v_j)} \log_2 \frac{\deg(v_i)}{\sum_{j=1}^{N} \deg(v_j)}. \end{aligned} \tag{7}$$

The matrix entropy describes the distribution of incoming and outgoing links in a graph. In a randomly generated graph one expects $\deg(v_i) \approx \deg(v_j)$. In this case the matrix entropy H approaches its maximum H_{max}. Graph entropy is thus only at a maximum if G is a regular graph where $\deg(v_i) = \deg(v_j) = D$. Because $\rho_i = D/DN = 1/N$ in a regular graph, one has H as shown in (8).

$$H = H_{max} = -\sum \rho_i \log_2 \rho_i = \log_2 N \tag{8}$$

In contrast, the minimum matrix entropy H_{min} is observable in networks showing star topology. The trace $tr(A)$ of such a graph corresponds to $2N-2$ and the degree of its central vertex is $\deg(v_c) = N-1$. Consequently, the entropy of the central vertex H_c is calculated as shown in (9).

$$H_c = -\frac{N-1}{2N-2} \log_2 \frac{N-1}{2N-2} = -\frac{1}{2} \log_2 \frac{1}{2} = 0.5. \tag{9}$$

The degree of any other vertex is $\deg(v_i) = 1$. Hence, the entropy of a graph constituted as a star is calculated as follows:

$$\begin{aligned} H &= H_{min} \\ &= 0.5 + \sum_{V \setminus v_c} -\frac{1}{2N-2} \log_2 \frac{1}{2N-2} \\ &= 0.5 + \frac{1}{2} \log_2 (2N-2) \\ &= 1 + \frac{1}{2} \log_2 (N-1). \end{aligned} \tag{10}$$

Normalized network entropy can be finally computed according to (11):

$$\hat{H} = \frac{H - H_{min}}{H_{max} - H_{min}}, \hat{H} \in [0,1] \tag{11}$$

In order to illustrate expressiveness of $\hat{H}$ and $\hat{\nu}$ with respect to the underlying network topology, a straightforward experiment was carried out in which synthetic networks exhibiting star topologies were continuously mutated over time, resulting in almost regular graphs after numerous generations. This simulated process thus yields a continuous change of the network topology for each graph. $\hat{H}$ and $\hat{\nu}$ were accordingly computed for every generation and tracked. The time series of both measures are shown in Figure 1. More precisely, simulations of

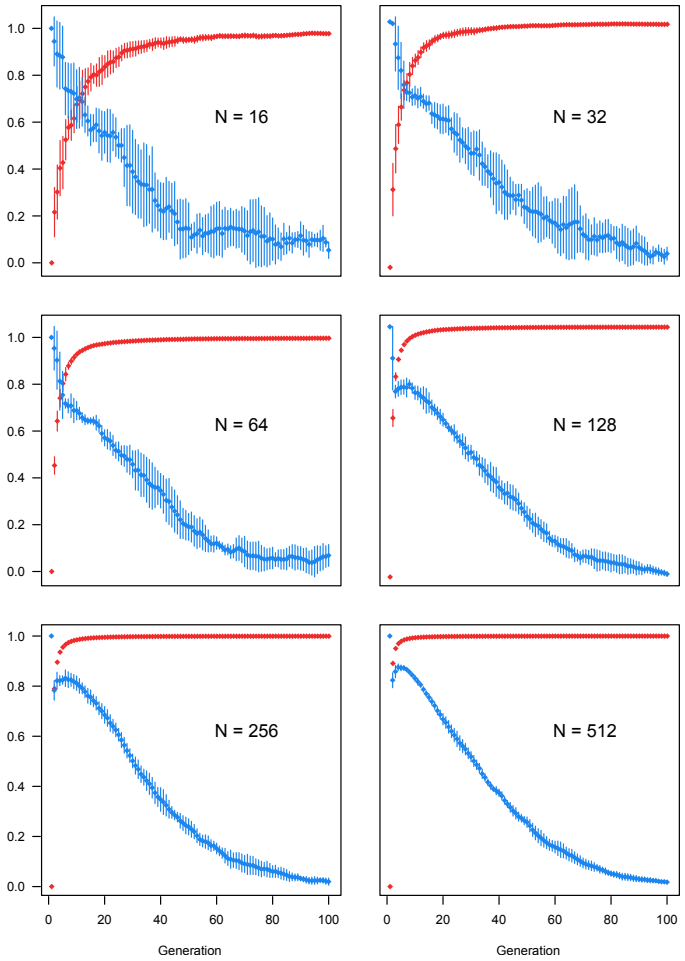


Figure 1. Simulation results of networks with various sizes N, whereas the red line represents $\hat{H}$, the blue line $\hat{\nu}$ and vertical bars indicate standard deviations.

topological change were conducted by starting with star graphs of fixed sizes ($N = 16, 32, 64, 128, 256$ and 512 vertices). In every generation, edges between every pair of vertices were randomly added and respectively removed. For each graph size, six runs were conducted in an effort to estimate variance.

As shown in Figure 1, both measures converged after 100 generations. All entropy trajectories show fast convergence compared to $\hat{\nu}$ trajectories, with the convergence time decreasing with increasing N. Although $\hat{\nu}$ yield larger variances (especially for $N \leq 32$), its slower convergence and qualitatively similar trajectories for all graph sizes N illustrates greater sensitivity to topological changes. In that respect matrix entropy loses significance with increasing graph size.

IV. DATASET

For this study, the structure of the Facebook page of the German party "DIE LINKE" was analyzed because it is a typical star-like topology with the page owner as a central node. This central node often has the highest activity, meaning the most in- and out-links. The communication on the page was explored over a period of five months, from January 2017 up until May 2017, whereas all posts, comments and replies were taken into account (see Table I).

 ISBN: 978-1-61208-654-5

TABLE I. SUMMARY OF THE DATA INCLUDING NORMALIZED ENTROPY AND SKEWNESS OF THE CONSIDERED NETWORKS.

month	actors	posts	comments	replies	$\hat{H}$	$\hat{\nu}_{LR}$
January	2,878	26	2,955	3,471	0.19	0.98
February	2,146	33	2,196	2,062	0.24	0.98
March	3,196	40	3,501	3,245	0.17	0.97
April	2,432	26	2,558	3,295	0.22	0.98
May	4,765	31	4,130	5,674	0.10	0.98
Epinions	75,879	n/a	n/a	n/a	0.65	0.07

During initial analysis of the dataset, it was observed that $12,031$ individuals were active during the five months. However, as shown in Figure 2, only 104 of these individuals were active in every single month. In general, it can be stated that users showed rather sparse and sporadic activity, with only a minority being recurrent users.

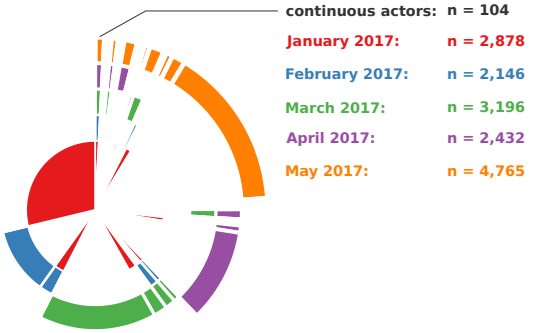


Figure 2. Sunburst chart of actor activity consisting of one radial segment for each user, whereas a user's segment in a time layer is left out if said user was observed to be inactive in that time period.

Figure 3 shows a comparison of two different network topologies. Each network represents the interaction of the users, in particular their communication, in a social network. The labels of the nodes of the users were anonymized using enumeration except of the central node in Figure 3a. This figure depicts the network of the Facebook page "DIE LINKE" from January 2017 as a graph in which the size of each node corresponds to the out-degree (number of out-links). As can be seen, the network is dominated by the central node of the page owner and thus is close to a star-shaped topology. In contrast, Figure 3b shows a part of the *Epinions* social network [19]. Due to the size of the network, it was necessary to limit the depiction by applying k-core ≥ 80, showing only the most active nodes. This network tends to be more decentralized, in other words, there is no node which dominates all others in terms of its degree.

Table I shows the normalized entropy and LeaderRank skewness of the "DIE LINKE" network, separately calculated for each month. It can be clearly seen, that obtained $\hat{H}$ values fluctuate over time, whereas LeaderRank skewness $\hat{\nu}_{LR}$ remains stable. For comparison, the *Epinions* social network [19] shows a considerably less skewed LeaderRank distribution, whereas normalized network entropy $\hat{H}$ is thus less expressive, as theoretically discussed in Section III.

V. COMPETENCERANK

To address the issues discussed in Section III, we present a competence-adjusted variant of the LeaderRank which down ranks nodes with a high amount of out-links in comparison to their in-links. The competence-adjusted LeaderRank, referred to as CompetenceRank, of a particular topic-specific opinion leader $CR(L_i)$ can be calculated as shown in (12).

$$CR(L_i) = \frac{LR(L_i)}{1 + \frac{k_i^{out}}{k_{total}^{out}} \cdot LR_{total}} \tag{12}$$

The CompetenceRank of a certain opinion-leader is calculated by dividing its original LeaderRank score LR_i by a fraction of cumulative sum of LeaderRank scores (which is equal to the number of users) defined by the node's share of network activity, and k^{out} being the number of outgoing links. By definition, LR_{total} – the sum of LeaderRank scores of all nodes in the social network graph – is equal to the number of nodes N. When considering regular graphs, one observes LeaderRank distribution skewness $\hat{\nu} = 0$ as well as $k_i^{out} = k_j^{out} = D$ for any pair of randomly chosen nodes v_i and v_j. Thus, $k_{total}^{out} = ND$. From this, the expression above can be conveniently rewritten as

$$CR(L_i) = \frac{LR(L_i)}{1 + \frac{D}{ND} \cdot N} = \frac{1}{2} LR(L_i). \tag{13}$$

We finally define the CompetenceRank based on the assumption that $LR(L_i) = CR_i$ in regular graphs, which is thus simply achieved by multiplying the expression in (13) by 2. Henceforth,

$$CR(L_i) = 2 \cdot \frac{LR(L_i)}{1 + \frac{k_i^{out}}{k_{total}^{out}} \cdot N}. \tag{14}$$

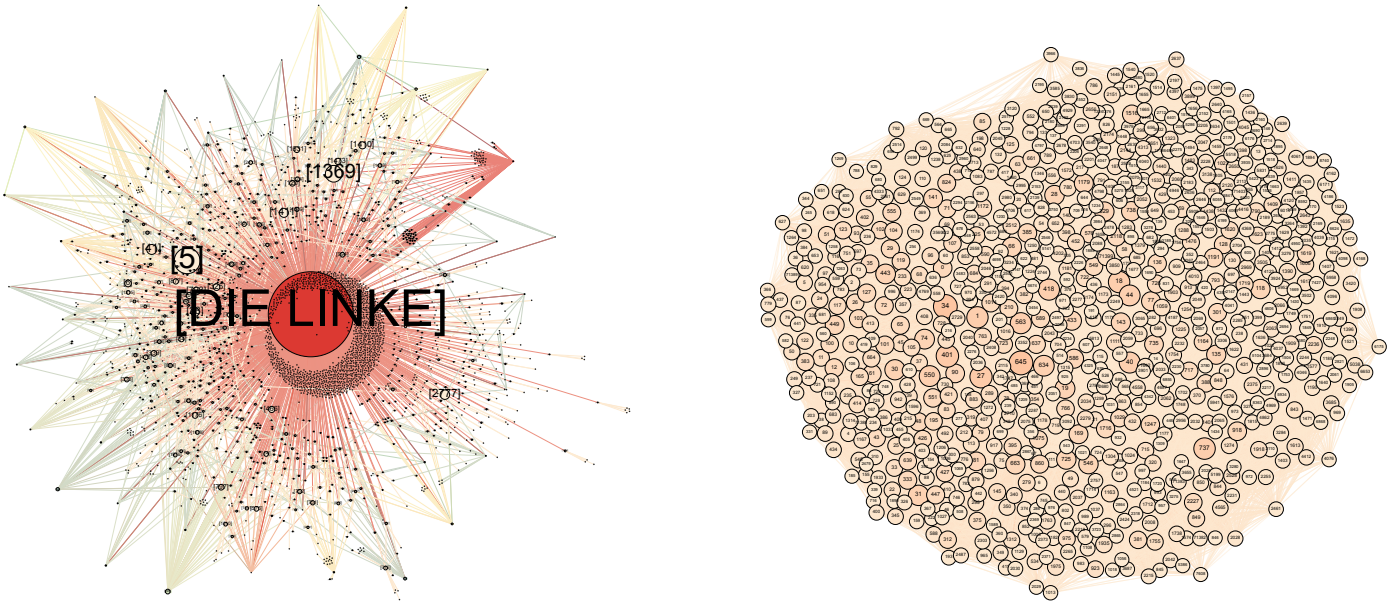
In turn one can interpret the cumulative discrepancy $\sum_i^N |CR(L_i) - LR(L_i)|$ as a function of network regularity.

VI. CONCLUSION AND FUTURE WORK

The analysis of social networks, and in particular the finding of influential and opinion-influencing profiles, is of great interest in forensic research for a variety of reasons. In the present study, it was shown that the usual centrality-based approaches, and in particular the LeaderRank, produce erronous results in star-like networks, such as Facebook pages of parties. Furthermore, LeaderRank skewness was presented as an appropriate measure to quantify the degree of distortion of a network or in other words its proximity to a star-shaped topology. Finally, the CompetenceRank was introduced as a measure that provided better results that the popular LeaderRank for the data used in the study.

In following studies, it would be interesting to analyze the observed phenomena in more fine-grained times pans as well as over a longer period of time. Additionally, it is necessary to take more and different network topologies into account. Furthermore, it was noticed that the texts in the data used were surprisingly well written. This provides an opportunity to conduct further textual analyses especially to answer the question whether there is a correlation between topics and opinion leaders and if so, how both develop over time.

 ISBN: 978-1-61208-654-5



(a) The network of the facebook page "DIE LINKE" of January 2017.

(b) Part of the *Epinions* social network [19] (filtered by k-core ≥ 80).

Figure 3. Comparison of two different network topologies.

REFERENCES

[1] M. Spranger, F. Heinke, S. Grunert, and D. Labudde, "Towards predictive policing: Knowledge-based monitoring of social networks," in The Fifth International Conference on Advances in Information Mining and Management (IMMM 2015), 2015, pp. 39 – 40.

[2] M. Spranger, H. Siewerts, J. Hampl, F. Heinke, and D. Labudde, "SoNA: A Knowledge-based Social Network Analysis Framework for Predictive Policing," International Journal On Advances in Intelligent Systems, vol. 10, no. 3 & 4, 2017, pp. 147 – 156.

[3] M. Spranger, S. Becker, F. Heinke, H. Siewerts, and D. Labudde, "The infiltration game: Artificial immune system for the exploitation of crime relevant information in social networks," in Proc. Seventh International Conference on Advances in Information Management and Mining (IMMM), IARIA. ThinkMind Library, 2017, pp. 24–27.

[4] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," Comput. Netw. ISDN Syst., vol. 30, no. 1-7, Apr. 1998, pp. 107–117.

[5] L. Lü, Y.-C. Zhang, C. H. Yeung, and T. Zhou, "Leaders in social networks, the delicious case," PloS one, vol. 6, no. 6, 2011, p. e21202.

[6] Q. Li, T. Zhou, L. Lü, and D. Chen, "Identifying influential spreaders by weighted leaderrank," Physica A: Statistical Mechanics and its Applications, vol. 404, no. Supplement C, 2014, pp. 47 – 55.

[7] E. Katz, "The two-step flow of communication: An up-to-date report on an hypothesis," Public Opinion Quarterly, vol. 21, no. 1, Anniversary Issue Devoted to Twenty Years of Public Opinion Research, 1957, p. 61.

[8] E. M. Rogers, Diffusion of innovations. New York: The Free Press, 1962.

[9] B. Huang, G. Yu, and H. R. Karimi, "The finding and dynamic detection of opinion leaders in social network," Mathematical Problems in Engineering, vol. 2014, 2014, pp. 1–7.

[10] P. Parau, C. Lemnaru, M. Dinsoreanu, and R. Potolea, "Opinion leader detection." in Sentiment analysis in social networks, F. A. Pozzi, E. Fersini, E. Messina, and B. Liu, Eds., 2016, pp. 157–170.

[11] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," J. Mach. Learn. Res., vol. 3, Mar. 2003, pp. 993–1022.

[12] X. Song, Y. Chi, K. Hino, and B. Tseng, "Identifying opinion leaders in the blogosphere," in Proceedings of the sixteenth ACM conference on Conference on information and knowledge management - CIKM '07, M. J. Silva, A. O. Falcão, A. A. F. Laender, R. Baeza-Yates, D. L. McGuinness, B. Olstad, and Ø. H. Olsen, Eds. New York, New York, USA: ACM Press, 2007, pp. 971 – 974.

[13] B. Amor et al., "Community detection and role identification in directed networks: Understanding the twitter network of the care.data debate," CoRR, vol. abs/1508.03165, 2015.

[14] M. Richardson and P. Domingos, "Mining knowledge-sharing sites for viral marketing," in Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, Za&iuml and O. R. ane, Eds. New York, NY: ACM, 2002, p. 61.

[15] C. Egger, "Identifying key opinion leaders in social networks: An approach to use instagram data to rate and identify key opinion leader for a specific business field," Master Thesis, TH Köln - University of Applied Sciences, Köln, 2016.

[16] M. Z. Shafiq, M. U. Ilyas, A. X. Liu, and H. Radha, "Identifying leaders and followers in online social networks," IEEE Journal on Selected Areas in Communications, vol. 31, no. 9, 2013, pp. 618–628.

[17] Z. H. Zhang, G. P. Jiang, Y. R. Song, L. L. Xia, and Q. Chen, "An improved weighted leaderrank algorithm for identifying influential spreaders in complex networks," in 2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), vol. 1, July 2017, pp. 748–751.

[18] F. Passerini and S. Severini, "Quantifying complexity in networks: The von neumann entropy," Int. J. Agent Technol. Syst., vol. 1, no. 4, Oct. 2009, pp. 58–67.

[19] M. Richardson, R. Agrawal, and P. Domingos, "Trust management for the semantic web," in The Semantic Web - ISWC 2003, D. Fensel, K. Sycara, and J. Mylopoulos, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 351–368.

 ISBN: 978-1-61208-654-5

A Mining Driven Decision Support System for Joining the European Monetary Union

Ray R. Hashemi[1], Omid M. Ardakani[2], Azita A. Bahrami[3], Jeffrey A. Young[1], and Rosina Campbell[2]
[1]Department of Computer Science
[2]Department of Economics
Georgia Southern University
Savannah, GA, USA

[3]IT Consultation
Savannah, GA, USA

***Abstract*—The European Monetary Union (EMU) is a result of an economic integration of European Union member states into a unified economic system. The literature is divided on whether the EMU members benefit from this monetary unification. Considering costs and benefits, a fiscal authority may ask whether it is a good decision to join the EMU. We introduce and develop a decision support system to answer the proposed question using a historical dataset of twelve Macroeconomic Outcomes (MOs) obtained for 31 European countries and for 18 years (1999-2016). The system meets the three-prong goal of: (1) identifying highly relevant MOs for a given year, y_i, using the data from years y_1 to y_i; (2) deriving decision of "join/not-join" the EMU along with its certainty factor using the relevant MOs for y_i; and (3) examining the accuracy of the derived decision using the data from y_{i+1} to y_{18}. The performance analysis of the system reveals that (a) the number of relevant MOs has declined nonlinearly over time, (b) the relevant MOs and decisions are significantly changed before and after the European debt crisis, and (c) the derived decisions by the system has 79% accuracy**.

Keywords- Mining Features; Mining-based Decision Support System; The European Monetary Union; Bayesian Theorem

I. INTRODUCTION

The European Monetary Union (EMU) is an agreement among the European countries to join together for creating one functioning monetary system with one currency. The idea of the EMU was given by the European Council in the Dutch city of Maastricht in December of 1991. Later the formation of this union was declared in the Treaty on European Union or as it is better known the Maastricht Treaty (1992). In January of 1999, eleven countries adopted the single currency—euro. For joining the European Monetary Union, a country must adhere to the following entry conditions: price stability, sound and sustainable public finances, durability of interest rate convergence, and exchange rate stability.

The advantages of joining the EMU have been enumerated by many sources including the European central bank that is in charge of stabilizing inflation by implementing a common monetary policy across the Eurozone [1]. Rogers [2] has shown that joining the EMU leads to more stable prices. Alesina and Barro [3] and also Frankel and Rose [4] have indicated that the trade costs among European countries have reduced and the credibility of monetary policy has enhanced. Kim *et al.* [5] and also Bernoth *et al.* [6] have claimed separately that the members have experienced a lower default risk premium. (Risk premium is defined as the spread between member's yield and the yield on the German Bund). Rose and Engle [7] have empirically shown that due to the trade volume, the volatility of exchange rate has declined in the Eurozone.

However, the disadvantages of joining the EMU have also been enumerated by many resources. Feldstein [8] expresses that the loss of monetary independence of the members is counted as one of the major disadvantages. Codongo *et al.* [9] have argued that credit risk has been greater in the euro area. After the European Sovereign Debt Crisis started in 2009, the disadvantages were more pronounced: Government debt increased due to massive tax cuts and increase in the government spending [10][11]. Banking crises became evident among some members that led to deep and prolonged asset market collapses associating with declines in total output measured by Gross Domestic Product (GDP) [12][13].

Considering the advantages and disadvantages of joining the EMU, a fiscal authority may ask: is it a good decision to join the EMU. We try to answer the proposed question using historical Macroeconomic Outcomes (MOs) obtained for 31 European countries and for 18 years [1999-2016]. The binary attribute of *membership* divides the 31 countries into two groups of member and non-member of the EMU. For a given year, y_i, the curtailed historical data will include all the records from year 1999 up to year y_i (y_i year data is not included.)

The goal of this research effort has three prongs: (1) Identifying the highly relevant MOs to the membership attribute for a given year, y_i, using the curtailed historical data for the year, (2) Deriving a decision of "join/not-join" the EMU along with a certainty factor, at the year y_i using the relevant attributes, and (3) Examining the accuracy of the derived decision by assessing the behavior of the MOs

 ISBN: 978-1-61208-654-5

of those countries that joined the EMU at year y_i for the years of y_i to y_{2016}.

The rest of the paper is organized as follows. The Previous Works is the subject of Section 2. The Methodology is presented in Section 3. The Empirical Results are covered in Section 4. The Conclusions and Future Research are covered in Section 5.

II. PREVIOUS WORKS

The literature presents a body of work for determining whether membership in the EMU is beneficial using MOs. Frankel and Rose have used trade volume and GDP (two MOs) to find the effects of monetary unification [4]. They applied a two-stage approach to the problem. In the first stage, trade between any two countries was estimated using a gravity model. In the second stage, the Ordinary Least Squares method was used to find how joining the EMU affects trade and GDP. Gomez-puig tested the existence of causal relationships between the bond yield and government debt and joining the EMU [14]. The Granger's causality test was used which is based on the concept of the causal ordering. The causal relationship was estimated using the first difference and lagged variables regression. Rose and Engle used a linear regression model to determine whether the MOs of openness, exchange rates, and price integration statistically changed within the EMU members compared to the non-EMU members [7]. Codongo et al. [9] and Bhatt *et al.* [15] separately and by different methodologies have determined the effects of joining the EMU on yield and yield spread as two MOs.

In all reported studies, the number of MOs used is limited and extremely small. However, we use a large set of MOs (twelve of them) and that is the major point of departure from the similar works reported in literature. As the second point of departure, we consider our dataset as a snapshot of the twelve MOs at a point in time and then we say knowing what we know from the snapshot, is it advisable to join the EMU. In addition we verify the accuracy of the advice.

III. METHODOLOGY

Let D be a dataset with N independent attributes of $A_1, \ldots, A_n$ and one dependent attribute of E. As a preprocessing step, we keep only one copy of the attributes that are correlated. The Pearson method is used to compute the correlation coefficients among every two attributes, Ai, A_j using formula (1)

$$\rho_{A_i,A_j} = \frac{cov(A_i,A_j)}{\sigma_{A_i},\sigma_{A_j}} \quad (1)$$

Where, ρ_{A_i,A_j} denotes the Pearson coefficient. $cov(A_i, A_j)$ is the covariance and σ_{A_i}, σ_{A_j} are the standard deviations. The correlation test calculates the S-statistics defined by

$$S = (n^3 - n)\frac{1-\rho_{A_i,A_j}}{6} \quad (2)$$

Where, n is the sample size. The S-statistics is compared to its critical value to reject the null hypothesis of no correlation between the two attributes.

The details of the methodology for meeting the three prongs of the goal are covered in the following three subsections

A. *Identifying the Most Relevant Attributes*

The first sub-goal is to identify the relevant attributes among the independent attributes of ($A_1 \ldots A_n$) that are indicative of E (the dependent variable in dataset D). We assume the possible values for E are $d_1, \ldots, d_g$. To meet this sub-goal, we use the Naïve Bayesian classification approach which is encapsulated as follows.

Let r be a new record with ($A_1 \ldots A_n$) attributes for which a predicted value of E is sought using D as a training set. The predicted value of E for r is determined by the highest probability amongst $P(E=d_j \mid r)$, for j = 1 to g. The $P(E=dj \mid r)$ is defined by formulas (3):

$$P(E = d_j \mid r) = \frac{p(E=d_j)\prod_{i=1}^{n} p(A_i=v \mid E=d_j)}{p(r)} \quad (3)$$

Since the denominator is the same for all probabilities of $P(E = d_j \mid r)$, we need to calculate only the numerator.

Two algorithms Core (Fig. 1) and Relevant (Fig. 2) are used for determining the *relevance degree* of each independent attribute in reference to the dependent attribute. The Core algorithm accepts a dataset Φ with k attributes of $A_1, \ldots, A_k$ as independent variables and the binary attribute of Z as dependent variable.

Algorithm Core (Φ, se, sp)
Given: Dataset Φ with independent attributes of ($A_1, \ldots, A_k$) and a binary dependent attribute of Z. S_1 and S_2 that are the set of all the record numbers in Φ with Z = 1 and Z =0, respectively.
Objective: Predict the Z value for every record in Φ and return sensitivity (se) and specificity (sp) for the overall prediction.
Method:
Step1- Repeat for each record, r_j, in Φ and in presence of the rest of the records
Step2- Treat r_j as the test set and Φ-r_j as the training set;
Step3- Apply the Naïve Bayesian classification to predict a Z value for r_j;
End;
Step4- W_1 and W_2 that are the set of all the record numbers in Φ with predicted values of Z = 1 and Z =0, respectively.
Step5- se = $|S_1 \cap W_1|/|S_1|$;
sp = $|S_2 \cap W_2|/|S_2|$;
Return (se, sp);
End;

Figure 1. The Algorithm Core

Core algorithm: (a) treats every record of the dataset Φ, individually, as a test set of one record while treating the remaining records as a training set (Steps 1 and 2), (b)

applies the Naïve Bayesian classifiers to predict the Z value for the record in the test set (Step3), (c) upon completion of predicting a Z value for every record in Φ, Core algorithm returns sensitivity and specificity of the classification as two parameters of se = $T^1/(T^1+F^0)$ and sp = $T^0/(T^0+F^1)$, where T^1 and T^0 are the number of records that truly predicted 1 and 0, respectively. F^1 and F^0 are the number of records that falsely predicted 1 and 0, respectively (Step5).

To explain the algorithm Relevant, the dataset D which is similar to dataset Φ and has n attributes of $A_1 \ldots A_n$ is given. Let us make n subsets out of dataset D, ($D_1, \ldots, D_n$), such that D_i is composed of the attribute A_i and a copy of the attribute Z (Steps 1 and 2.) Let us also apply Core algorithm on each subset, D_i, separately and calculate se_i and sp_i, (Step 3.) The prediction of Z values by A_i is as good as α_i = Min (se_i, sp_i), (Step4). Therefore, we consider α_i as the relevancy measure of A_i to Z (named the *relevance degree* of A_i in reference to Z.) The Relevant algorithm delivers a list of attributes and their corresponding relevance degrees such that each relevance degree in the list is greater than a given threshold (Step 5.) The list is the response to the first sub-goal.

Algorithm Relevant
Given: Dataset D with independent attributes of ($A_1 \ldots A_n$) and a binary dependent attribute of Z. A Relevance Degree matrix, RD, of the size n X 2. A threshold value of T_v.
Objective: Determining the most relevant independent attributes to Z.
Method:
Step1- Repeat for every attribute, A_i, in D.
Step2- D_i is the projection of D over attributes of (A_i, Z);
Step3- Invoke Core(D_i, se_i, sp_i);
Step4- α_i = Min (se_i, sp_i).
Step5- If ($\alpha_i > T_v$)
Then Insert the pair of (A_i, α_i) into RD.
End;
End;

Figure 2. The Algorithm Relevant

B. Deriving a Decision

Considering the dataset D given above and the outcome of the algorithm Relevant, deriving a decision is completed in three stages. In the first stage, a subset of D is chosen, D_r, that includes only the relevant attributes obtained from RD along with a copy of Z

In the second stage, the matrix RD is sorted in descending order of the relevance degree values and the attribute in the top row of matrix RD is selected as the *seed*, A_s. The attribute A_s is used as a root of a *search* tree with q-1 branches at the first level, where q is the number of the relevant attributes in D_r, as shown in Fig. 3.

To minimize the cost of building the search tree, we apply a filtering process at each level of the tree such that only the best leaf survives. Through the filtering process, "tie" cases may happen in selecting a node for expansion. Such cases are resolved by randomly choosing one node among the tied nodes.

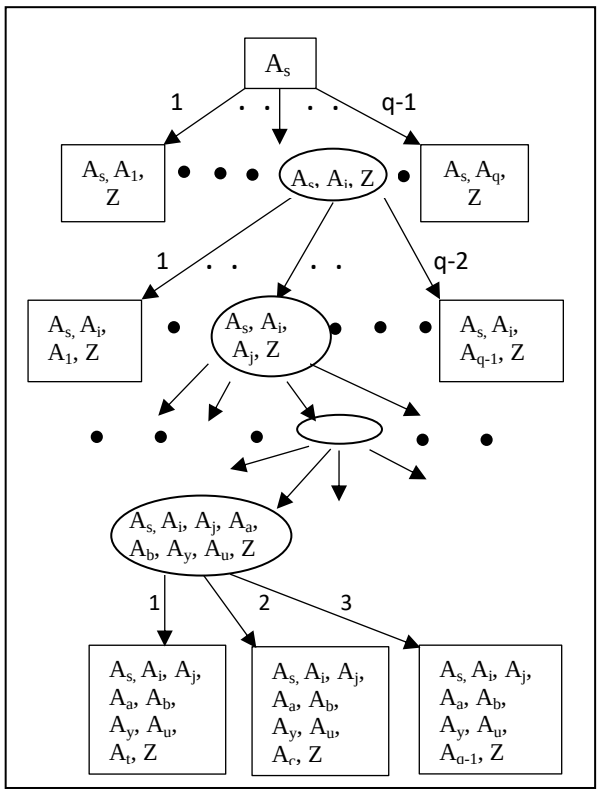


Figure 3. A search tree: The winner node at each level is designated by an oval shape and there is not any winner node at the last level

To explain it further, the i-th leaf of the first level is a projection of D_r over the Z, root of its subtree, and the attribute A_i (A_i cannot be the same as the root of its subtree.) Only one leaf from the first level is chosen as the winner and expanded (the condition for being a winner node are introduced shortly.) In the second level there are q-2 branches from the winner node of the first level. The j-th leaf of the second level is a projection of D_r over the Z, attributes in the path from the root to the winner node, and the attribute A_j (A_j is not the root of any subtree.) This process continues until the subtree cannot be expanded either because all the attributes in D_r are exhausted or there is not any winner node in the current level. Reader needs to be reminded that each node of the search tree has its own dataset which is a subset of the dataset D_r. Selecting a node as the winner of a given level is done by taking the following steps:

a. The Core algorithm is applied on each one of the node's datasets separately to obtain sensitivity and specificity, and relevance degree for the dataset.
b. The winner node is the one with the highest relevance degree among the nodes' datasets and it is greater than a set threshold.

 ISBN: 978-1-61208-654-5

The search tree delivers the most relevant attributes to the attribute Z which is considered feature extraction from D.

In the third stage, the last winner node of the search tree with sensitivity of se_l and specificity of sp_l is examined one more time. If ($se_l > sp_l$) it means the node's dataset more reflects those countries that are members of the EMU. Therefore, the suggested decision by the system is "join". Using the same argument, if ($se_l < sp_l$) then the suggested decision is "Not-Join". In the case that ($se_l = sp_l$), the weighted average (WA) for all the records in the winner node's dataset with Z =0 and Z =1 are calculated separately.

If ($WA_1 > WA_0$) Then decision is "join"
If ($WA_1 < WA_0$) Then decision is "not-join"
If ($WA_1 = WA_0$) Then no decision can be made.

A *certainty factor, CF,* ($0 \le CF \le 1$) is associated with any driven decision, which simply expresses the level of confidence that the decision support system has in the suggested decision. A higher value for CF means higher confidence in the decision. The CF is simply the average of sensitivity and specificity for last winner node.

C. Examining the Derived Decision

We examine the accuracy of the derived decision for a given year, y_i, by (a) identifying those countries that joined the EMU at year y_i, (b) assessing the behavior of their MOs for years of y_i to y_{18}, in reference to the average of their MOs behavior prior to joining the EMU, and (c) determining whether the assessment results support the derived decision. The assessing process is done by performing a trend analysis which is encapsulated as follows:

Let C = {C_a, . . ., C_p} be a set of countries that joined the EMU at year y_i and {A_1, . . ., A_n} be the set of attributes (MOs) that are the same for every country, C_j, in C. Let also G^{i-1} be the set of the average values for each one of the n attributes of C_j from year 1 to year y_{i-1}, G^{i-1} = {(g_1^{i-1}, . . ., g_n^{i-1}), that is used as the *baseline* during the trend analysis. In addition, let the values for the n attributes of C_j for the k^{th} year after y_i be (v_1^{i+k} , . . ., v_n^{i+k}.) The trend of attribute A_m for the year i+k is denoted by Trend(A_m^{i+k}) and it is computed by formula (4).

$$Trend(A_m^{i+k}) = \frac{v_m^{i+k} - g_m^{i-1}}{g_m^{i-1}} * 100 \quad (4)$$

The *overall trend* of A_m for the period of q years for C_j is:

$$ot(A_m^q)_{C_j} = \frac{\sum_i^q Trend(A_m^{i+q})}{q} \quad (5)$$

For the same period, the *overall trend* of A_m for the countries in C, OTC, is:

$$otc(A_m^q) = \frac{\sum_a^p ot(A_m^q)_{C_a}}{|C|} \quad (6)$$

The interpretation of the $otc(A_m^q)$ value depends on the nature of the attribute. For example, a negative overall trend value for the attribute "inflation" is considered an improvement whereas a negative trend value for GDP is considered deterioration. The same can be argued for a positive value. Table 1 is used to remove the duplicity of the interpretation. As a result, a positive/negative trend value is considered an improvement/deterioration trend, for any attribute, respectively.

The formula (6) delivers the overall trend for attribute A_m (m=1 to n) for a period of q years considering all countries in C. The *overall trend for the countries* in C given all the attributes and for the same period of q is calculated using formula (7).

$$Trend(C) = \frac{\sum_{m=1}^n OCT(A_m^q) * \beta_m}{n} \quad (7)$$

Where, $OCT(A_m^q)$ is the $oct(A_m^q)$ after its duplicity removed, β_m is the average of the relevance degrees of the attribute A_m for the period of q years and for all the countries in C.

TABLE I. ACTIONS FOR DUPLICITY REMOVAL

Trend Value: $otc(A_m^q)$	**Negative (Improvement)**	**Positive (Improvement)**
Negative	$otc(A_m^q)$*(-1)	$otc(A_m^q)$
Positive	$otc(A_m^q)$*(-1)	$otc(A_m^q)$

The architecture of the Mining Driven Decision Support System is summarized as a collection of three major modules: Feature Extractor, Decision Driver, and Trend Analyzer, Fig. 4. The dataset D goes to a pre-processing step to be cleaned and a subset of D is selected, Y, for years in the range of [1- y) which includes those countries that were adopted into the Eurozone in year y. The feature extractor module accepts Y as input. The module extracts a subset, K, of the attributes in D such that the K attributes are highly relevant to the dependent variable Z for the years of [1- y). The extraction is done by applying the algorithm Core, algorithm Relevant, and creation of search tree. The outcome of the module helps to make two different projections of D over the K attributes, Y' and Y", for the years [1-y) and [y+1, 18], respectively.

The decision Driver Module accepts the Y' as the input dataset and, in reference to Z, calculates the sensitivity (se) and specificity (sp) for Y' using a naïve Bayesian approach. A decision of "join" or "not-join" is suggested by the module using the values for se and sp.

The trend analyzer module accepts the dataset Y" and delivers an overall trend of the behavior of MOs for the countries (adopted into the EMU in year y) during the years of [y, 18]. The support of the overall trend for the suggested decision is used to confirm the validity of the suggested decision.

IV. EMPIRICAL RESULTS

Currently, the European Union consists of 31 members of which 19 countries have adopted the euro, Table 2. The Master Dataset used has the measurements of twelve MOs for every country in Table 2 and for the duration of 1999-2016. The short and expanded names of the MOs in the Master Dataset are given in Table 3.

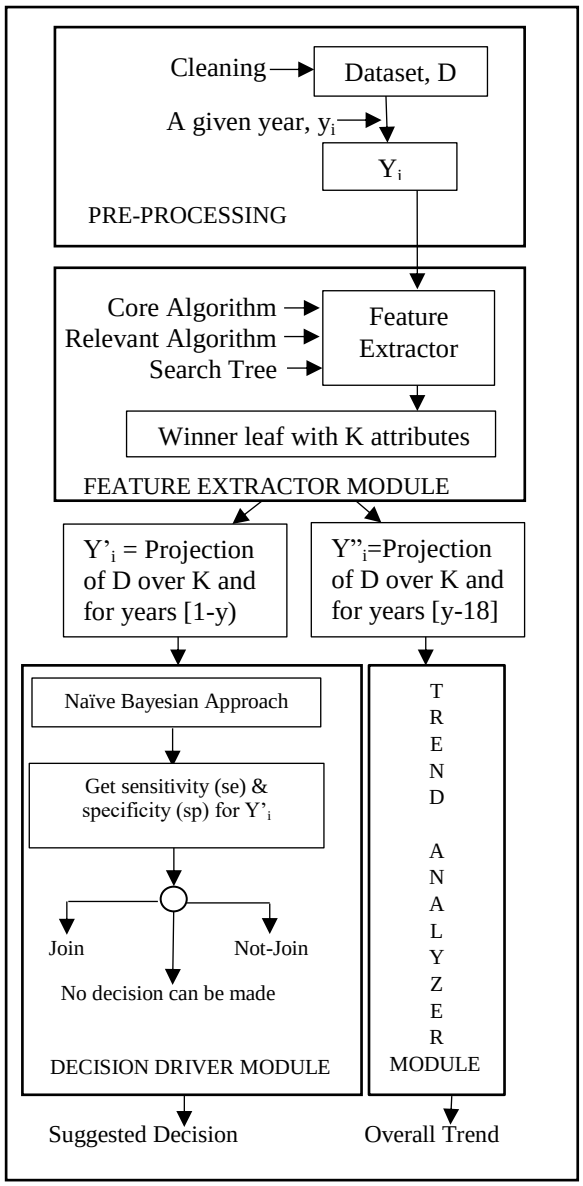


Figure 4. The architecture of the Mining Driven Decision Support System

Table 2 has eight unique adoption dates for the current members of the EMU (1999, 2001, 2007, 2008, 2009, 2011, 2014, and 2015.) We divided the Master Dataset into eight triplets of new datasets as follows (Y_{99}, Y'_{99}, Y''_{99}), (Y_{00}, Y'_{00}, Y''_{00}), (Y_{06}, Y'_{06}, Y''_{06}), (Y_{07}, Y'_{07}, Y''_{07}), (Y_{08}, Y'_{08}, Y''_{08}), (Y_{10}, Y'_{10}, Y''_{10}), (Y_{13}, Y'_{13}, Y''_{13}), and (Y_{14}, Y'_{14}, Y''_{14}.) The dataset Y_i ($i \neq 99$) held only those records of the Master Dataset for which the adopted year is i+1. The reason for choosing data up to one year prior to each one of the eight-unique adoption dates stems from the fact that we assume the decision of "join" or "not-join" is based on the historical data prior to the date of adoption. For the year 1999, however, there is no prior data and as a remedy we use the data for year 1999 itself as the historical data.

TABLE II. MEMBERS AND NON-MEMBERS OF THE EMU FROM 1999 TO 2016

EMU Members			
Country	***Adoption Date***	***Country***	***Adoption Date***
Austria	1999	Latvia	2014
Belgium	1999	Lithuania	2015
Cyprus	2008	Luxembourg	1999
Estonia	2011	Malta	2008
Finland	1999	Netherlands	1999
France	1999	Portugal	1999
Germany	1999	Slovakia	2009
Greece	2001	Slovenia	2007
Ireland	1999	Spain	1999
Italy	1999		
Non Members			
Bulgaria	Denmark	Romania	Iceland
Croatia	Hungary	Sweden	Norway
Czech Rep.	Poland	UK	Switzerland

In addition, two attributes of Treat and Year were added to the Master Dataset to reflect the membership of the country in the EMU and the year for which data was collected, respectively. The Treat was a binary attribute where the values of zero and one for a country meant "member" and "non-member" of the EMU, respectively. The Year values were 1 to 18, where value = 1 means year 1999 and value = 18 means year 2016. The total number of records was 31 * 18 = 558. The MOs were considered as independent variables and the Treat attribute was considered as the dependent attribute.

TABLE III. THE SHORT AND EXPANDED NAMES OF THE "MO"S

MO's Short Name	**MO's Expanded Name**
ldebt	First lag of the government-debt ratio
lgdpg	First lag of GDP growth
lrmg	First lag of real money growth
lpi	First lag of inflation
lopen	First lag of trade
lrer	First lag of real exchange rate
bndyld	10-year government bond yield
lspread	First lag of the spread between bndyld and average of French and German yields
bndvol	Volatility of the bond yield
gdpgvol	Volatility of GDP growth
pivol	Volatility of inflation
spreadvol	Volatility of the spread

The dataset Y'_i held only those records of Y_i that were non-members but became members in the year i+1, and Y''_i held only those records of the Master Dataset for which the year values ≥ i and country names were the same as the ones in Y'_i.

For a triplet of (Y_i, Y'_i, Y''_i), the dataset Y_i was used to determine the relevant attributes and derive the decision of "join/not-join" along with its certainty factor. The dataset Y'_i was used to provide a baseline for the trend analysis and Y''_i was used to study the trends in reference to the baseline.

 ISBN: 978-1-61208-654-5

The following process was repeated for each triplet of (Y_i, Y'_i, Y''_i). (This means the process repeated only for those years that one or more countries joined the EMU.):

Step1: Correlated attributes in Y_i were identified and from each group of correlated attributes only one attribute remained in Y_i (generating a cleaned Y_i.) The maximum calculated correlation among any two MOs was 0.3. Therefore, none of the MOs was dismissed.

Step2: The cleaned Y_i was used to identify the most relevant independent attributes of Y_i with regard to dependent attribute, Treat (using the Relevant algorithm) and the rest of the independent attributes were dismissed (generating a scrubbed Y_i.)

Step3: The independent attributes in Y'_i and Y''_i that were not found in the scrubbed Y_i were removed. A baseline was established using Y'_i by averaging the values for each attribute separately.

Step4: A decision of "join/not-join" was derived from scrubbed Y_i using the relevant attributes and the search tree outcome. The certainty factor for each decision was also calculated.

Step5: The derived decision for Yi was considered as the derived decision for the countries C = {Ca, . . ., Cp} joining the EMU at the year Y_{i+1}. The Y'_i was used as a baseline to verify the accuracy of the derived decision using overall trend for C in Y''_i.

In reference to the first sub-goal of the research, the list and rank of the relevant MOs produced by Step 2 of the process are shown in Table 4. (Rank of an MO is its importance to the derived decision and Rank equal to 1 is the highest rank.) The derived decisions along with the certainty factor produced by Step 4 of the process were also shown in Table 4.

TABLE IV. THE RELEVANT "MO"S AND THEIR RANKS ALONG WITH THE DERIVED DECISION AND THEIR CERTAINTY FACTOR FOR EACH ADOPTION DATE

Rank	**Y_{99}**	**Y_{00}**	**Y_{06}**	**Y_{07}**
(1)	spreadvol	spreadvol	lspread	lspread
(2)	lrmg	lrmg	lrer	lrer
(3)	ldebt	ldebt	bndyld	bndyld
(4)	bndyld	lspread	bndvol	
(5)	lrer	lrer		
(6)	lspread	bndyld		
(7)	gdpgvol	gdpgvol		
Derived Decision	*Join CF:0.93*	*Not-Join CF:0.98*	*Not-Join CF:0.84*	*Not-Join CF:0.79*
Rank	**Y_{08}**	**Y_{10}**	**Y_{13}**	**Y_{14}**
(1)	lopen	lopen	lopen	ldebt
(2)	lrer	bndvol	pivol	lopen
(3)		lrer	lrer	lrer
(4)		ldebt	ldebt	
Derived Decision	*Join CF:0.77*	*Join CF:0.79*	*Join CF:0.73*	*Join CF:0.69*

The trend analysis delivered by Step 5 of the process is shown in Table 5 in which the improvement/deterioration of the overall trend of the MOs were expressed by the notations of (+)/(–), respectively. The overall trend of (+) and the derived decision of "join" are in agreement and so the overall trend of (–) and derived decision of "not-join".

TABLE V. DERIVED DECISIONS AND OVERALL TRENDS FOR EACH UNIQUE ADOPTION DATE

Adoption Date	**No. of Countries**	**Derived Decision**	**Overall trend**
1999	11	Join	(+)
2001	1	Not-Join	(–)
2007	1	Not-Join	(+)
2008	2	Not-Join	(+)
2009	1	Join	(–)
2011	1	Join	(+)
2014	1	Join	(+)
2015	1	Join	(+)

VI. CONCLUSIONS AND FUTURE RESEARCH

The same procedure may be applied to only one of the countries for a given date of adoption as long as Y'_i and Y''_i include the past and ongoing economic performance of that one country. It is also true that the same procedure may be applied for any given year, but it is not necessary. To explain it further, the purpose is to compare the past and ongoing economic performance of a given country (captured in two datasets of Y'_i and Y''_i.) If the year i be any year then, the comparison of the two datasets Y'_i and Y''_i does not have any meaning. Therefore, framing the past and ongoing economic performance into the years of (1999-i) and (i+1- 2016) is relevant as well as adequate. As another point of clarification, the time(year)-lag effects are not the same over all MOs. We use the first lag for all MOs except bndyld, and volatility attributes (bndvol, gdpvol, pivol, and spreadvol.) The reason for using the first lag is that economic theories suggest that today's decisions by central banks are made with a one-period lag. This is not true for volatility variables and bndyld. Readers need to be reminded that our study is a longitudinal framework, which includes time (year) and unit (country).

The findings for our decision support system were presented in Tables 4 and 5. Table 4 indicates the relevant MOs for the years prior to the adoption dates and the derived decisions. The relevance of MOs has changed over time. A greater set of MOs determines whether the country must join the EMU for the years closer to the introduction of the euro, 1999. As time goes on, the relevant MOs set shrinks nonlinearly. For the years before the European sovereign debt crisis the MOs of, spreadvol, spread, lrmg, and lrer play significant roles in the decision making process; however, after the crisis other outcomes such as lopen, ldebt, bndvol, and pivol become pertinent.

The MO of lrer is the only one that was identified as the relevant attribute in all Y_i datasets. One possible reason is that the EMU members link their currency to the euro which protects them from currency fluctuations. This linkage makes the lrer a significant factor for all the samples. The MOs, bndyeld and lspread identified as relevant only for countries that joined the EMU prior to 2009. The reason stems from the fact that bond yield convergence is one of the goals of monetary unification and members had

 ISBN: 978-1-61208-654-5

benefited from joining the EMU through the bond yield channel [16]. However, the bond yields have reduced to nearly zero after the crisis due to expansionary monetary policy. Thus, the bond market attributes have become irrelevant in the post crisis era. In contrast, lopen was observed as a relevant MO for only countries that joined the EMU subsequent to 2009. The attempt of the EMU members to overcome the turmoil of the banking crisis by increasing trade within the monetary union was completely aligned with the observation of lopen as a relevant MO.

The derived decision is "join" for the year 1999 when 11 countries adopted the euro corroborating evidence that it is beneficial to join at the beginning of the European monetary union. However, the findings reveal that the decisions made by European countries to "join" the EMU in adoption dates of 2001, 2007, and 2008 (years prior to crisis), are not beneficial. In contrast, the findings provide evidence that joining would be beneficial after the crisis.

Table 5 shows the derived decisions and overall trends for each adoption date. Considering the last two columns of Table 5, four possible combinations of (Join, +), (Join, −), (Not-Join, +), and (Not-Join, −) may exist. Only the combinations of (Join, +) and (Not-Join, −) are evidence of the support for the derived decision by the actual trends in the behavior of the MOs. Based on Table 5, for five out of the eight groups of countries that joined the EMU on the unique adopted dates, the derived decision and trend analysis agreed. The five groups included 15 countries. To summarize, the accuracy of our decision support system was 15/19 = 0.79% with the false positive of 3/19 = 16% and false negative of 1/19 = 5%.

The certainty factor for the derived decision declines with time. One explanation for such decline is that the size of the $Y"_i$ datasets reduce with time. The reduced sample size is a potential threat to the validity.

It is worth mentioning that the proposed methodology can be applied to other policy making questions such as whether adopting a specific monetary policy strategy (e.g., Inflation Targeting) is effective.

The discussed mining driven decision support system is designed to suggest a decision of "join" or "not-join" the EMU to a fiscal authority. As a future research, we plan to investigate another popular/political question. The question is under what circumstances is it beneficial for a given country to leave the EMU? This issue has become significantly important for both policy makers and researchers after the recent talk of Brexit. Also, the development of a Bayesian belief system to support the casual relationships among MOs, if any, is in progress.

REFERENCES

[1] R. Beetsma and M. Giuliodori, "The Macroeconomic Costs and Benefits of the EMU and Other Monetary Unions: An overview of recent research", Journal of Economic Literature, vol. 48, no. 3, pp. 603–641, September 2010.

[2] J. H. Rogers, Monetary union, "price level convergence, and inflation: How close is Europe to the USA?", International Journal of Forecasting, no. 54, pp.785–796, April 2007.

[3] A. Alesina and R. J. Barro, "Currency unions", Quarterly Journal of Economics, vol. 117, no. 2, pp. 409–436, May 2002.

[4] J. Frankel and A. Rose, "An estimate of the effect of common currencies unions on trade and income", Quarterly Journal of Economics, no. 67, pp.437–466, May 2002.

[5] S. Kim, F. Moshirian, and E. Wu, "Evolution of international stock and bond market integration: influence of the European Monetary Union", Journal of Banking & Finance, vol. 30, no. 5, pp. 1507–1534, May 2004.

[6] K. Bernoth, J. V. Hagen, and L. Schuknecht, "Sovereign risk premia in the European government bond market", European Central Bank Working Paper Series, pp. 1–39, June 2004.

[7] A. K. Rose and C. Engel, "Currency unions and international integration", Journal of Money, Credit, and Banking, vol. 34, no. 4, pp. 1067–1089, November 2002.

[8] M. Feldstein, "The political economy of the European economic and monetary union: political sources of an economic liability", Journal of Economic Perspectives, vol. 11, no. 4, pp. 23–42, September 1997.

[9] L. Codongo, C Favero, and A. Missale, "Yield spreads on government bonds before and after EMU", Economic Policy, pp. 1–29, May 2002.

[10] P. R. Lane, "The European Sovereign Debt Crisis", Journal of Economic Perspectives, vol. 26, no. 3, pp. 49–68, August 2012.

[11] D. Sandri and A. Mody, "The Eurozone Crisis: how banks and sovereigns came to be joined at the hip", Economic Policy, pp. 199–230, March 2012.

[12] C. M. Reinhart and K. S. Rogoff, "The Aftermath of Financial Crises", American Economic Review, vol. 99, no. 2, pp. 466–472, April 2009.

[13] C. M. Reinhart and K. S. Rogoff, "Banking crises: an equal opportunity menace", Journal of Banking & Finance, vol. 37, no. 11, pp. 4557–4573, December 2013.

[14] M. Gomez-Puig, "The Immediate Effect of Monetary Union on EU-15 Sovereign Debt Yield Spreads", Applied Economics, vol. 41, no. 7, pp. 929–939, March 2009.

[15] V. Bhatt, N. K. Kishor, and J. Ma, "The Impact of EMU on Bond Yield Convergence: evidence from a time-varying dynamic factor model", Journal of Economic Dynamics and Control, no. 82, pp.206–222, September 2017.

[16] J. F. de Guevara, J. Maudos, and F. Perez, "Integration and Completion in the European Financial Markets", Journal of International Money and Finance, vol. 26, no. 1, pp. 26–45, February 2007.

 ISBN: 978-1-61208-654-5