Automating the Semantic Labeling of Stream Data

Konstantinos Kotis
Dept. of Cultural Technology and Communication
University of the Aegean,
Mytilene, Greece
e-mail: kotis@aegean.gr

***Abstract*—The collection of a voluminous real-world stream data is achieved today through a large number of distributed and heterogeneous data sources. On the other hand, it is quite rare to discover and collect semantic models associated with this data, in order to be able to represent implicit meaning and specifying related uncovered concepts and relationships between them. Such semantic models, however, are the key to make the data easily available, understandable and interlinkable for its potential users and applications. Manually modeling the semantics of data requires significant effort and expertise. Most of the related work focuses on the semantic labeling/annotation of the data fields (source attributes), given that a semantic model is already provided. Constructing a semantic model that explicitly describes the relationships between the data attributes in addition to their semantic types is critical. Related works support the semantic annotation of data using existing ontologies, but there are only a few that automatically construct the ontology based on the real-world stream data that will eventually annotate (two-step process). More important, existing solutions require a manually-created training data set and its mapping to existing related ontologies/models, in order to assist in the process of learning the mapping function between the actual stream data and the related semantic model (usually via a supervised machine learning approach). This paper a) presents the problem and representative related work, and b) proposes design directions that are aligned to key requirements.**

Keywords-semantic label; ontology; stream data.

I. INTRODUCTION

In domains such as the IoT, sensor devices are used to obtain insights about the things that 'live' in the surrounding world, and to facilitate an intelligent interaction with them (sense, analyze, act). The increasing need of using the data produced by the sensor devices (stream data) inevitably leads to Big Data, which requires new scalable and efficient methods to structure and represent the underlying information and to make the data accessible, processable and interlinkable for the applications/services that use it. For instance, this is the case for accessing, integrating and reasoning with large volumes of stream data generated from moving entities (e.g., ships, airplanes), dynamic data such as weather conditions, as well as historical/static data, in order to recognize high-level critical events to support real-time decision and policy making.

Semantic technologies are used for the formal representation of the real-world sensor data, due to its advantage of conceptualizing and representing raw data in an easy but still formal and explicit way, making them machine interpretable and allowing their interlinkage to existing resources (e.g., Web, Linked Open Data cloud). For instance, representing raw numerical values of measurements of weather conditions that are measured and conceptualized by meteorological or AIS (automatic identification system) sensors, e.g., attributes, such as date, time, swell height (or height of swell), wind speed (or speed of wind), visibility, and their corresponding values such as "28/08/2017, 09.00, 1, 20, 10" and "28/08/2017, 22.00, 8, 90, 5", can be done by automatically discovering their corresponding semantics and assigning to them the appropriate semantic labels.

Typically, individual data table columns (e.g., CSV/Excel table) are mapped to ontological properties, a set of data table columns are mapped to ontological classes, and row data table rows are mapped to ontological individuals. For structured data/information such as relational databases (RDB) and Web tables, the aim is to semantically annotate/label the sources of structured data by mapping RDB and Web tables against an ontology. As recently reported [1], such a task can be decomposed into different subtasks such as table-to-class mapping, row-to-instance mapping, and column-to-property mapping.

Defining the problem of semantic labeling of a data source S with a semantic labeling function $\boldsymbol{\varphi}:<\{\boldsymbol{a}\},\{\boldsymbol{v_i}\}>\rightarrow \boldsymbol{l}$ is explained in the following lines.

A data source S is a collection of ordered pairs $<\{\boldsymbol{a}\},\{\boldsymbol{v_a}\}>$, where $\boldsymbol{a}$ denotes an attribute name (e.g., 'date', 'time', 'swell height', etc.) and $\{\boldsymbol{v_a}\}$ denotes the set of data values corresponding to the attribute $\boldsymbol{a}$ (e.g., for $\boldsymbol{a}$ equals to the 'date' attribute the set will have values such as '28/08/2017', '30-04-2018'). Different data sources can have attributes that have different names but map to the same semantic label (e.g., 'swell height' of S_1 and 'height of swell' in S_2 are both mapped to the same property i.e., 'swell_height' of a Weather ontology). Multiple data sources are often mapped to the same ontology in many practical scenarios. The goal is to automatically learn the semantic labeling function. To assign a semantic label to an attribute in a new data source, we take an ordered pair $<\{\boldsymbol{a}\},\{\boldsymbol{v_a}\}>$ and use a semantic labeling function, which has been learned from training data, to predict its semantic label.

In the IoT domain, one of the biggest challenges is to discover and establish mappings between raw data and its intended meaning, formalized explicitly into ontological concepts, properties and/relationships between them. Such a problem is usually referred as the symbol grounding problem [2], describing the fundamental challenge of defining concepts/properties from numerical sensor data that is not grounded in meaningful real-world information. Voluminous real-world stream data are usually recorded and collected as

 ISBN: 978-1-61208-736-8

numerical values that cannot easily be related to meaningful information without knowing the context, such as the observation time and the location of the recorded data. Such data can change over time or it can be depended on other external factors. For instance, 30 degrees Celsius in summer time can be a normal condition, however in winter time such a reading could possibly mean an error of the sensor functionality.

In contexts where real-world row data are generated in a streaming fashion by sensor devices or other data sources, the main problems related to their semantic labeling are:

- In the absence of a semantic model (ontology), how to automatically construct one by learning/uncovering the semantics 'hidden' in the data
- Given a (learned) semantic model, how to automatically, accurately and on-time compute the corresponding data-to-semantics mappings, in a continuous and iterative fashion.

While syntactic information about data sources such as attribute names (e.g., title, name, location) or attribute types (e.g., string, int, date) may provide some hints towards discovering their related meanings and form the corresponding semantic types, often this information is not sufficient for an accurate prediction. For instance, the field 'title' of a data source that records artworks is not by itself indicative of the intended meaning of its values (e.g., 'Zinnias'), i.e., it might by the case that values (titles) are meant to be related with book titles, with song titles or with artworks. This prediction can be even harder if the attribute names are used in abbreviated forms e.g., 'dob' (i.e., date of birth) instead of 'birthdate'. That is the main reason why related approaches focus on learning data semantics using the data values rather their attribute names.

The general idea behind existing related approaches is to learn a semantic labeling function from a training set of data that has been previously semantically labeled in a manual fashion. Then, when presented with a new data source of the same topic/domain, the learned semantic labeling function can automatically assign semantic types to each attribute of the new source. The training data consists of a set of semantic types and each semantic type has a set of data values and attribute names associated with it. Given a new set of data values from a new source, the goal is to predict the top-*k* candidate semantic types along with confidence scores using the training data.

In the example of Artworks, the semantic types are 'title' of 'Artwork', 'name' of 'Person', and 'label' of 'Museum'. By simply labeling the attributes with those types however, is not sufficient. Unless the relationships between the columns are explicitly specified ('museum', 'painter'), a precise model of the data cannot be obtained. For instance, a person could be the owner, the painter, or the sculptor of an artwork, but in the context of a specific example data for paints, only the semantic relation of 'painter' correctly interprets the intended relationship between an artwork and a person. Thus, to build a rich semantic model that fully represents the intended semantics of the data, another step that determines the relationships between the attributes of the data sources in terms of the properties in the ontology is necessary.

Moreover, in terms of solving the same problem presented above, but for stream (mainly numerical) data as input, one is facing with even more challenging issues. Due to the nature of stream data (data arrives so rapidly in large volumes, usually with no structure or metadata attached to it), techniques such as dimensionality reduction and time-windowing of data, as well as statistical/probability techniques for analyzing the distribution of numeric values corresponding to a semantic label as well as linking those labels to each other, are required.

This paper presents a) a number of recent related works that try to solve the abovementioned problems, and b) design directions aligned to technological requirements that must be satisfied towards the goal of automating the semantic labeling of stream data. It is structured as follows: Section 2 provides background knowledge of key technologies, Section 3 presents the related work, Section 4 presents the proposed design directions towards efficiently approaching the problem, and Section 5 concludes the paper, also stating future work plans.

II. BACKGROUND KNOWLEDGE

A. Semantic Labeling

Semantic labeling (or annotation), in the most common cases, is the process of attaching additional information to various concepts (e.g., people, things, places, organizations etc.) in a given text or any other unstructured or structured content (e.g., video, RDB, Web tables). When a document (or another piece of content, e.g., video) is semantically labeled it becomes a source of information that is easy to be interpreted, combined and reused by computers. As described by OntoText [3], to semantically annotate concepts in the sentence "Aristotle, the author of Politics, established the Lyceum", we need to identify Aristotle as a 'person' and Politics as a 'written work of political philosophy'. Then we can index, classify and interlink the identified concepts in a semantic graph database (e.g., GraphDB), in order to be able to add (link) other information about Aristotle such as his date of birth, his teachers, and his works. Politics can also be linked to its subject, to its date of creation etc. Given the semantics about the above sentence and its links to other (external or internal) formal knowledge, algorithms will be able to automatically answer questions such as: who tutored Alexander the Great, which of Plato's pupils established the Lyceum. In other words, semantic labeling/annotation enriches content with machine-processable information by linking background information to extracted concepts.

For structured data/information, such as relational databases (RDB) and Web tables, the aim is to semantically annotate sources of structured data by mapping RDB and Web tables against an ontology. Such a task can be decomposed into different subtasks such as table-to-class mapping, row-to-instance mapping, and column-to-property mapping [1]. There are many studies on mapping data sources to ontologies and several approaches have been proposed to generate semantic Web data from databases and

 ISBN: 978-1-61208-736-8

spreadsheets [4]. RDB schemas are easy to handle when computing 1-1 mappings (table-to-class and field-to-property correspondences). The D2RQ [5] and Ontop [6] Ontology-Based Data Access (OBDA) approaches, introduces custom mapping languages that enables users to define mapping rules between tables of relational databases and target ontologies in order to annotate and publish semantic data in RDF format. R2RML [7] is a W3C recommendation for expressing customized mappings from relational databases to RDF datasets. Writing the corresponding mapping rules by hand however, is a time-consuming task. The users need to have a good understanding of the way source tables can be most effectively mapped to the target ontology. They also need to learn the syntax of writing the mapping rules.

In recent years, some efforts were introduced towards automatically inferring the implicit semantics of tables. Polfliet and Ichise [8] use string similarity methods between column names and names of the ontological properties in order to discover the corresponding mappings. Wang et al. [9] use the header of Web tables along with the values of the rows to map the columns to the attributes of the corresponding entity that is represented in a rich and general purpose taxonomy of facts (built from a corpus of over one million Web pages and other data). This approach can only deal with the tables containing information of a single entity type. Limaye et al. [10] use YAGO ontology [11] to annotate Web tables and generate binary relationships using machine learning approaches. This approach is limited to the labels and relations defined in the YAGO ontology. Venetis et al. [12] presents a scalable approach to describe the semantics of tables on the Web, leveraging a database of class labels and relationships that are automatically extracted from the Web. Although these approaches are very useful in labeling and publishing semantic data from tables, they are limited in learning the semantics relations: they only infer individual binary relationships between pair of columns. They are not able to find the relation between two columns if there is no direct relationship between the values of those columns.

Moreover, other related work exploits the data available in the Linked Open Data (LOD) cloud to capture the semantics of the tables and publish their data as RDF. Munoz et al. [13] present an approach towards mining RDF triples from the Wikipedia tables by linking the cell values to the resources available in DBpedia [14]. This approach is limited to Wikipedia. In Mulwad et al. [15], the Wikitology [16] is used to link cells in a table to Wikipedia entities. Wikitology is an ontology which combines some existing manually-built knowledge systems such as DBpedia and Freebase [17]. They query the background LOD to generate initial lists of candidate classes for column headers and cell values and candidate properties for relations between columns. Then, they use a probabilistic graphical model to find the correlation between the column's headers, cell values, and relation assignments. The quality of the semantic data generated by this category of work is highly dependent to how well the data can be linked to the entities in LOD.

In Karma [18], a graph from learned semantic types and a domain ontology is built. Then the graph is used to map a data source to the ontology interactively. In this work, the system uses the knowledge from the pre-defined existing domain ontology to propose models to the user, who can correct them as needed. The system remembers the semantic type labels assigned by the user, however, it does not learn from the structure of previously modeled sources.

In terms of stream data, the aim is to automatically annotate real-world data flows, in real-time, with semantics that are already available before-hand (for the training dataset) [19]–[21] or not i.e., extract/uncover the real-world semantics on-the-fly, during the annotation process [22]. In a real-time stream processing and large-scale data analytics for IoT and Smart City applications' context, the semantic annotation process of heterogeneous data for automated discovery and knowledge-based processing is sometimes referred as data virtualization [23][24][25].

B. Ontology learning

Ontology learning concerns the process of constructing an ontology from data/information source(s), in an automatic or semi-automatic manner, to minimize or eliminate cost, effort and time-consuming human involvement [26]. The process extracts the concepts and relationships between them from a corpus of natural language text or other sources of data and information, and encodes them in an ontology language (e.g., OWL). As building ontologies manually is extremely labor-intensive and time-consuming, there is great motivation to fully automate the process in several application domains. A typical text-based process of ontology learning, starts by extracting terms and concepts from plain text using techniques, such as part-of-speech tagging and phrase chunking. Then, statistical or symbolic techniques are used to extract relation signatures, often based on pattern-based or definition-based hypernym/hyponym/meronym extraction techniques.

A representative work that learns and constructs ontologies from text documents is presented in Wang et al. [27], introducing an automatic learning approach to construct terminological ontologies based on different text documents. In Lin et al. [28], a learning approach that constructs an ontology automatically without the requiring training data is presented. Other related approaches include the learning of lightweight ontologies from query logs [29], aiming at the efficient retrieval of Semantic Web documents. In another related work [30], a new ontology is automatically constructed by utilizing representations of entities, their attributes and relations, learnt using unsupervised machine learning techniques on facts extracted from Wikipedia tables. Furthermore, a challenge in the automatic transformation of an RDB model into an ontology is how to label the relationships between concepts [31]–[33]. This challenge depends heavily on the correct extraction of the relationship types, since RDB models does not store the meaning of relationships between entities but it only indicates the existence of a link between them [33].

Several works have been conducted in providing sensor data with semantic annotations. In Sheth et al. [34] semantics are used to represent and structure real-world data, however, automatically transforming the raw data into the semantic

representation in this work remains an open issue. Dietze et al. [35] describes the problem of symbolic grounding and the semantic sensor Web, and introduces an approach that uses conceptual spaces to bridge the gap between sensor measurements and symbolic ontologies in an automatic manner. In Stocker et al. [36] a system to identify and classify different semantic types of road vehicles passing a street is presented, using vibration sensors and machine learning algorithms. In Ganz et al. [22], a knowledge acquisition method is proposed that processes real-world stream data to automatically create and evolve domain ontologies, based on concepts-labeling rules that are automatically extracted from external sources.

C. Ontology matching

In recent years, ontology matching has received much attention in the Semantic Web community [37]. Ontology matching finds the correspondence between semantically related entities of different ontologies. Semantic annotation can benefit from some of the techniques developed for ontology matching. For example, instance-based ontology matching exploits similarities between instances of ontologies in the matching process. A semantic labeling algorithm can adopt the same idea to map the data of a new source to the classes and properties of a target ontology. Such an algorithm computes the similarity (e.g. cosine similarity between TF/IDF vectors) between the data of the new source and the data of the sources whose semantic models are known. Most of the work on ontology matching only finds simple correspondences such as equivalence and subsumption between ontology classes and properties. Therefore, the explicit relationships within the data elements are often missed when aligning the source data to the target ontology.

D. Stream data mining

Stream data, e.g., encoded in JSON, is mostly numerical and often with no rich (or any) metadata attached to it. Data arrives in a stream or streams, and if it is not processed immediately (or stored), it is lost. Moreover, the data arrives so rapidly that it is not feasible to store it all in active storage (i.e., in a conventional database), and then interact with it at the time of your choosing. The algorithms for processing streams involve summarization of the data, to make a useful sample of it and to filter it in order to eliminate most of the "undesirable" elements (e.g., stop words, noise), before it is annotated. Then the number of different elements in a stream is estimated using much less storage than would be required if all the elements were listed.

Knowledge acquisition requires several processing steps. Due to the large volume of real-world stream data, techniques are required to lower the amount (or dimensions) of the data input to make it manageable for processing algorithms such as clustering and statistical methods. In the domain of time-series analysis there has been a number of dimensionality reduction techniques such as Fast-Fourier transformation (FFT), Discrete Wavelete Transformation (DWT), Piecewise Aggregate Approximation (PAA), and Symbolic Aggregate ApproXimation (SAX). The comparative study by Ding et al. [38] reveals that SAX performs best in preserving the data features by remaining high dimension reduction (data compression).

SAX transforms time-series data into aggregated words that can be used for pattern detection and indexing. Since SAX was not developed for small constrained devices, authors in Ganz et al. [22] introduce Sensor-SAX, a modified version that has less data transmission in times of low activity in the sensor signal that is processed. In order to group similar types of patterns and events, clustering mechanisms are used. Cluster mechanisms do not require training data and can be unsupervised. However, the clustering methods rely on distance functions that map the data samples to a comparable space. The k-means clustering method provides fast computation of the groups even in large datasets. However, the biggest drawback is that the number of clusters (i.e., *k*) is an input parameter, and therefore should be known beforehand. In order to learn the ontological properties, a rule-mining approach can be used, similar to the one proposed in Hu et al. [39]. The authors aim at creating ontologies automatically by learning the logical rules to construct the ontology. In Ganz et al. [22] a rule learning approach is used, similar to the one of Hu et al. [39] to label the unnamed concepts in the ontology.

III. RELATED WORK

In the context of the work of Gao and Lianli [20], a Semantic Annotation and Activity Recognition (SAAR) approach is presented, integrating semantic annotation with Support Vector Machine (SVM) techniques to automatically identify animal behaviors from 3D accelerometry data streams. It enables biologists to visualize and correlate 3D accelerometer data streams with associated video streams. It also enables domain experts to accurately annotate segments of tri-axial accelerometer data streams, with standardized terms extracted from an activity ontology. These annotated data streams can then be used to dynamically train a hierarchical SVM activity classification model, which can be applied to new accelerometer data streams to automatically recognize specific activities. The approach requires a) significant human involvement, b) the creation of a training data set, and c) the use of predefined domain-specific ontologies.

Related approaches map each data value individually, typically by learning a model based on features extracted from the data using supervised machine-learning techniques. In the approach of Ramnandan et al [19], the difference is that it considers a holistic view of the data values corresponding to a semantic label, and uses techniques that treat this data in a collective manner. This way, it is possible to capture characteristic properties of the values associated with a semantic label as a whole. It supports both textual and numeric data analysis and proposes the top-*k* semantic labels along with associated confidence scores. For textual data, the TF-IDF-based approach is used, and for numeric data, the Kolmogorov-Smirnov (KS) statistical hypothesis test respectively. The semantics for the semantic annotation of data are automatically discovered, however the approach

 ISBN: 978-1-61208-736-8

requires the existence of training sets, as well as of domain-specific ontologies that are used to label the training sets.

Taheriyan et al. [21] exploits external knowledge from specific domain ontologies and other semantic models learned from previously modeled sources (based on the idea that data sources in the same domain usually provide overlapping data) to automatically learn an expressive new semantic model for a new source. The new semantic model represents the semantics of the new data source in terms of the concepts and relationships defined by the exploited domain ontology/ies. The approach is based on training/sample data of the new data source against the mapped semantics of the domain ontology and the known semantic models. Although the approach can be used to learn rich semantic models from data, human involvement as well as the existence of external knowledge (domain ontology and other related semantic models) is needed. Also, the data used to evaluate the approach (museum domain) cannot be considered as the hard case of streaming data (numerical vs textual semantic labeling). Finally, supervised machine learning (training data sets) is used for the semantic labeling of the data.

Ganz et al. [22] introduces a knowledge acquisition method that processes real-world streaming data to automatically create and evolve domain ontologies, based on concept-labeling rules that are automatically extracted from external sources. They use an extended *k*-means clustering method and apply a statistic model (Markov chain model approach) to extract and link relevant concepts from the raw sensor data and represent them in the form of a domain ontology. A rule-based system is used to label the concepts and make them understandable for the human user or for the semantic analysis, reasoning tools and software. The approach is based on the abstraction of numerical values, creating higher-level concepts from the large amount of data produced by sensor devices. To do so, as in other related work [23], the symbolic aggregate approximation (SAX) dimensionality reduction mechanism [40] is used. The approach uses the extended version of the SAX algorithm, i.e., SensorSAX. The approach uses the SSN Ontology [41] as a starting point and extend it by extracting new insights from the raw sensor data to construct a topical ontology representing an extract of the observed domain.

IV. DESIGN DIRECTIONS

In this section we propose a set of design directions for future approaches, based on the specific techniques/methods of existing ones that stand-out as key choices towards achieving the highest positive impact in an automated semantic annotation framework for real-world voluminous stream (sensor) data. The aim is to design an approach that transforms raw sensor streaming data (e.g., "28/08/2017, 09.00, 1, 20, 10" or "28/08/2017, 22.00, 8, 90, 5") into meaningful semantics (e.g., "Calmness" or "Storm"), as automatically and accurately as possible, minimizing human involvement and the use of pre-defined existing domain-specific ontologies.

The focus of these design directions is towards automating (as much as possible) the transformation of raw stream data related to the continuous monitoring of moving entities (vehicles, ships, aircrafts), for instance, trajectories, weather conditions, and low-level events (e.g., start, stop, turn), to valuable annotations of higher-levels of abstraction such as: change of course (a change in the direction that vessels are moving), three-point turn (the act of turning a vessel around in a limited space by moving in a series of back and forward arcs), cold wave (a wave of unusually cold weather), calmness (an absence of strong winds or rain), atmospheric phenomenon (a physical phenomenon associated with the atmosphere). Moreover, the focus is towards investigating how these automatically generated abstractions may be used in a combined way to infer and model even more higher levels of abstractions and critical high-level events such as: trade route (a route followed by traders, usually in caravans), migration route (the geographic route along which populations of animals/humans customarily migrate), flight path (the path of a rocket or projectile or aircraft through the air), collision (an accident resulting from violent impact of a moving object), crash, wreck (a serious accident, usually involving one or more vehicles).

The hardest problem of a data-to-semantics approach that uses an unsupervised machine learning algorithm for leaning concepts from numerical data, is probably the problem of automatically labeling the learned unnamed classes and properties. In the absence of a trained data-to-semantics learning algorithm, a rule-based mechanism must be applied on clustered symbolized SAX patterns in order to automatically add names to the unlabeled concepts. Such rules can be manually defined (increasing however the undesired, in our case, human involvement), or to construct a mechanism that can automatically extract those rules. The aim is to develop such a mechanism in order to automate the process of constructing such naming rules, possible encoded in the Semantic Web Rule Language (SWRL), towards supporting the automated concept and property naming task. For instance, such a rule set in the maritime/safe-shipping application domain may look like the ones presented in Table 1.

TABLE 1. EXAMPLE RULE SET IN THE MARITIME/SAFE-SHIPPING DOMAIN

isAISdata(?ad) & isSimpleEvent (?se) & equal(?se, 'turn')	=> VesselInTurn
isAISdata(?ad) & isSimpleEvent (?se) & equal(?se, 'lost communication')	=> VesselInLostCommunication
isWeatherData(?wd) & (swell_height_m(?sh) & greaterThanOrEqual(?sh, 8)) & (wind_speed_kmph(?ws) & greaterThanOrEqual(?ws, 90)) & (visibility(?v) & lessThanOrEqual(?v, 5))	=> Storm
badWeatherConditions & vesselInTurn & ???	=> WeatherForcedChangeOfCourse (inferred knowledge)
badWeatherConditions & vesselInLostCommunication & ???	=> VesselInDanger (inferred knowledge)

Natural Language Processing (NLP) techniques and heuristic rules will be further incorporated in order to assist the process of ontology construction. For instance, the

Vessel-related concepts learned from the rule-based mechanism, VesselInTurn and VesselInDanger, can be classified under WordNet-extracted learned concept Vessel, Storm under WeatherConditions, and WeatherForcedChangeOfCourse under ChangeOfCourse. Furthermore, WordNet (open multilingual knowledge graph) semantic relations that hold between the extracted concepts can be further analyzed in order to introduce labels for the unnamed properties as well.

A two-step process is proposed and presented below in an abstract design level. The first step concerns the learning of the ontology from a specific time-window of the stream data (Figure 1) and the second concerns the semantic data annotation of the data stream (Figure 2) i.e., the use of the learned ontology for the computation and refinement of data-to-ontology mappings of the data stream. The input of the process is: a) streaming data, mainly numerical, and b) external generic semantic lexicons or knowledge graphs. The proposed abstract steps of the process are:

1. Ontology Learning (Figure 1)
   - 1.1 Pre-process stream data:
     - Transform into a specific working format (e.g., from CSV or JSON to RDF),
     - Distinguish data between textual D_{ST} (e.g., vessels' historical data) and numerical D_{SN} (e.g., AIS and weather data).
   - 1.2 Define a time window T for a subset D_S of the stream data D that will be used for the automated learning of the ontology.
   - 1.3 Analyze data for D_S using external knowledge (e.g., WordNet) and a data summarization method (e.g., SAX).
     - For textual data D_{ST}: Methods for indexing and searching of documents (TF-IDF-based cosine-similarity method). The labeling algorithm will use the cosine similarity between TF/IDF vectors of WordNet documents (focused subset of synsets) and the input document to predict candidate semantic types (WordNet senses)
     - For numerical data D_{SN}: combined analysis of a) data values (using SAX) and b) data attribute names (using lexical and semantic analysis with the aid of external semantic lexicon such as WordNet or BabelNet)
   - 1.4 Automatically construct the top-*k* candidate ontological models M_k, for D_S, using a rule-based entity naming method.
   - 1.5 Present user the candidate semantic models M_k and allow the selection and refinement of the preferred *k* model i.e., the final learned ontology.
2. Semantic data annotation (Figure 2)
   - 2.1 Repeat step 1.1 for the rest of the stream data.
   - 2.2 Repeat step 1.3 using also the learned ontology as input to the data analysis method.
   - 2.3 Automatically compute data-to-ontology mappings *m* for D_S.
   - 2.4 Based on user feedback allow the manual correction/refinement of one of more mappings of *m*.
   - 2.5 Automatically (re)compute the mappings, based on users' corrections/refinements.

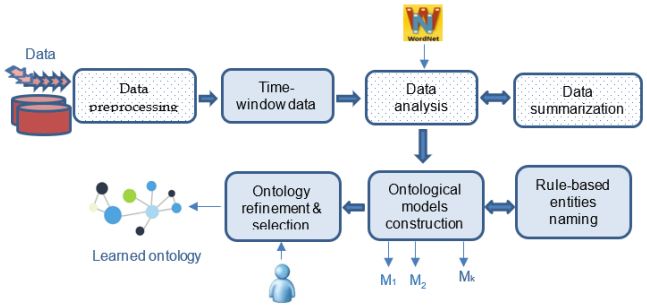

Figure 1. Ontology learning from time-window stream data

The output of the proposed process is: a) a learned-from-data domain ontology (e.g., encoded in OWL/RDF, and b) data-to-ontology mappings (e.g., encoded in R2RML).

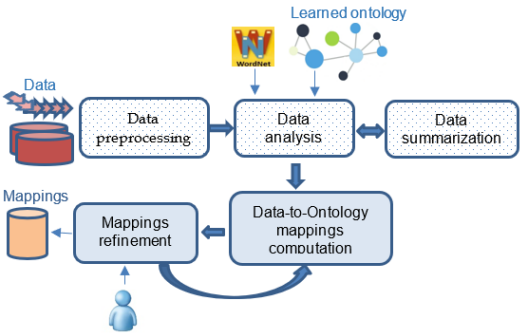

Figure 2. Semantic data annotation

The aim is to generate the learned-from-data domain ontology not just as a data-focused subset of the external lexicon source (e.g., WordNet, BabelNet), but a rich and expressive (as possible) lexicon-based ontology that reflects the intended meaning of analyzed data.

V. CONCLUSIONS AND RECCOMENDATIONS

The main problems related to the semantic annotation of stream data are: a) how to automatically construct a semantic model by learning the semantics 'hidden' in the data, b) given a semantic model, how to automatically, accurately and on-time compute the corresponding data-to-semantics mappings, in a continuous and iterative fashion.

We conjecture that there is a real need to develop approaches based on issues discussed in this paper, as well as on specific methods of existing approaches that stand-out as key choices towards achieving the highest positive impact in an automated semantic annotation framework for real-world voluminous stream (sensor) data. The need is to transform raw sensor streaming data into meaningful semantics, as automatically and accurately as possible, minimizing human involvement and use of pre-defined existing domain-specific ontologies.

 ISBN: 978-1-61208-736-8

The hardest problem, as identified in this paper, is related to the data-to-semantics approach that uses an unsupervised machine learning algorithm for learning concepts from numerical data: it is the problem of automatically labeling (adding names to) the learned unnamed classes and properties. In the absence of a trained data-to-semantics learning algorithm, a rule-based mechanism must be applied on clustered symbolized SAX patterns to automatically add names to the unlabeled learned concepts. Such rules can be manually defined (increasing however human involvement), or to construct a mechanism that can automatically extract them.

Based on the discussion and findings presented in this paper, the following key research actions are recommended to be integrated in related frameworks:

- Data synopses (summaries) from stream data sources, archival data, as well as detected and forecasted trajectories and events must be semantically annotated, transformed into a common form and be integrated. This task will exploit knowledge models and meta-data schemes that will be incorporated in the infrastructure, keeping them permanently up-to-date.
- Advance the related research towards automating, as much as possible, the transformation of raw stream data related to the continuous monitoring of moving entities (vehicles, ships, aircrafts), for instance, trajectories, weather conditions, and low-level events (e.g., start, stop, turn), to valuable annotations of higher-levels of abstraction.
- Develop a method for automatically learning a real-world domain-specific ontology that is needed for the semantic annotation of steam data related to moving objects' trajectories, weather conditions, and low-level events, minimizing human involvement and the usage of pre-defined external domain-specific semantics, as much as possible.
- Develop a set of novel NLP techniques and heuristic rules in order to assist the process of automated ontology construction.

REFERENCES

[1] D. Ritze and C. Bizer, "Matching web tables to DBpedia - a feature utility study," in Proceedings of the 20th International Conference on Extending Database Technology, 2017, pp 210-221.

[2] A. M. Cregan, "Symbol Grounding for the Semantic Web," in The Semantic Web: Research and Applications: 4th European Semantic Web Conference, ESWC 2007, Innsbruck, Austria, June 3-7, 2007. Proceedings, E. Franconi, M. Kifer, and W. May, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 429–442.

[3] OntoText, "Semantic Annotation example." [Online]. Available: http://ontotext.com/knowledgehub/fundamentals/semantic-annotation/. [Retrieved: Aug, 2017].

[4] S. Sahoo, et al., "A Survey of Current Approaches for Mapping of Relational Databases to RDF," W3C, 2009.

[5] C. Bizer and A. Seaborne, "D2RQ-treating non-RDF databases as virtual RDF graphs," in Proceedings of the 3rd International Semantic Web Conference (ISWC2004), 2004.

[6] D. Calvanese, et al., "Ontop: {A}nswering {SPARQL} {Q}ueries over {R}elational {D}atabases," Semant. Web J., vol. 8, no. 3, pp. 471–487, 2017.

[7] S. Das, S. Sundara, and R. Cyganiak, "R2RML: RDB to RDF Mapping Language." 2012.

[8] S. Polfliet and R. Ichise, "Automated Mapping Generation for Converting Databases into Linked Data," in Proceedings of the 2010 International Conference on Posters & Demonstrations Track - Volume 658, 2010, pp. 173–176.

[9] J. Wang, H. Wang, Z. Wang, and K. Q. Zhu, "Understanding Tables on the Web," in Conceptual Modeling: 31st International Conference ER 2012, Florence, Italy, October 15-18, 2012. Proceedings, P. Atzeni, D. Cheung, and S. Ram, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 141–155.

[10] G. Limaye, S. Sarawagi, and S. Chakrabarti, "Annotating and Searching Web Tables Using Entities, Types and Relationships," Proc. VLDB Endow., vol. 3, no. 1–2, pp. 1338–1347, 2010.

[11] Max Planck Institute for Informatics, "YAGO Ontology." [Online]. Available: http://www.mpi-inf.mpg.de/yago-naga/yago. [Retrieved: Aug, 2017].

[12] P. Venetis et al., "Recovering Semantics of Tables on the Web," Proc. VLDB Endow., vol. 4, no. 9, pp. 528–538, 2011.

[13] E. Muñoz, A. Hogan, and A. Mileo, "Triplifying wikipedia's tables," CEUR Workshop Proceedings, vol. 1057. 2013.

[14] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, "DBpedia: A nucleus for a Web of open data," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2007.

[15] V. Mulwad, T. W. Finin, and A. Joshi, "Semantic Message Passing for Generating Linked Data from Tables," in International Semantic Web Conference, 2013.

[16] Z. Syed and T. Finin, "Creating and Exploiting a Hybrid Knowledge Base for Linked Data," vol. 129. pp. 3–21, 2011.

[17] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, "Freebase: a collaboratively created graph database for structuring human knowledge," in In SIGMOD Conference, 2008, pp. 1247–1250.

[18] C. A. Knoblock et al., "Semi-automatically Mapping Structured Sources into the Semantic Web," in The Semantic Web: Research and Applications: 9th Extended Semantic Web Conference, ESWC 2012, Heraklion, Crete, Greece, May 27-31, 2012. Proceedings, E. Simperl, P. Cimiano, A. Polleres, O. Corcho, and V. Presutti, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 375–390.

[19] S. K. Ramnandan, A. Mittal, C. A. Knoblock, and P. Szekely, "Assigning Semantic Labels to Data Sources," in The Semantic Web. Latest Advances and New Domains: 12th European Semantic Web Conference, ESWC 2015, Portoroz, Slovenia, May 31 -- June 4, 2015. Proceedings, F. Gandon, M. Sabou, H. Sack, C. d'Amato, P. Cudré-Mauroux, and A. Zimmermann, Eds. Cham: Springer International Publishing, 2015, pp. 403–417.

[20] L. Gao and Lianli, "Semantic annotation and reasoning for sensor data streams," The University of Queensland, 2015.

[21] M. Taheriyan, C. A. Knoblock, P. Szekely, and J. L. Ambite, "Learning the Semantics of Structured Data Sources," Web Semant., vol. 37, no. C, pp. 152–169, 2016.

[22] F. Ganz, P. Barnaghi, and F. Carrez, "Automated Semantic Knowledge Acquisition From Sensor Data," IEEE Syst. J., vol. 10, no. 3, pp. 1214–1225, 2016.

 ISBN: 978-1-61208-736-8

[23] S. Kolozali, et al., “Semantic Data Stream Annotation for Automated Framework. D3.1 Report. Real-Time IoT Stream Processing and Large-scale Data Analytics for Smart City Applications, ‘City Pulse’ Collaborative Project, FP7-SMARTCITIES-2013, GRANT AGREEMENT No 609035,” 2013.

[24] S. Kolozali, M. Bermudez-Edo, D. Puschmann, F. Ganz, and P. Barnaghi, “A Knowledge-Based Approach for Real-Time IoT Data Stream Annotation and Processing,” in Proceedings of the 2014 IEEE International Conference on Internet of Things(iThings), and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom), 2014, pp. 215–222.

[25] C. Bizer, J. Volz, G. Kobilarov, and M. Gaedke, “Silk - A Link Discovery Framework for the Web of Data,” in 18th International World Wide Web Conference, 2009.

[26] G. Petasis, V. Karkaletsis, G. Paliouras, A. Krithara, and E. Zavitsanos, “Ontology Population and Enrichment: State of the Art,” in Knowledge-Driven Multimedia Information Extraction and Ontology Evolution: Bridging the Semantic Gap, G. Paliouras, C. D. Spyropoulos, and G. Tsatsaronis, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 134–166.

[27] W. Wang, P. Mamaani Barnaghi, and A. Bargiela, “Probabilistic Topic Models for Learning Terminological Ontologies,” IEEE Trans. Knowl. Data Eng., vol. 22, no. 7, pp. 1028–1040, 2010.

[28] Z. Lin, R. Lu, Y. Xiong, and Y. Zhu, “Learning Ontology Automatically Using Topic Model,” in Proceedings of the 2012 International Conference on Biomedical Engineering and Biotechnology, 2012, pp. 360–363.

[29] K. Kotis, A. Papasalouros, and M. Maragoudakis, “Mining query-logs towards learning useful kick-off ontologies: an incentive to semantic web content creation,” Int. J. Knowl. Eng. Data Min., vol. 1, pp. 303–330, 2011.

[30] C. Sekhar Bhagavatula, “Learning Semantics of WikiTables,” Department of Electrical Engineering and Computer Science, Northwestern University, 2013.

[31] K. Andrejs and B. Arkady, “Learning Ontology from Object-Relational Database,” Inf. Technol. Manag. Sci., vol. 18, no. 1, pp. 78–83, 2015.

[32] M. Pasha and A. Sattar, “Building Domain Ontologies From Relational Database Using Mapping Rules,” Int. J. Intell. Eng. Syst., vol. 5, 2012.

[33] B. El Idrissi, S. Baïna, and K. Baïna, “Ontology Learning from Relational Database: How to Label the Relationships Between Concepts?,” in Beyond Databases, Architectures and Structures: 11th International Conference, BDAS 2015, Ustro{ń}, Poland, May 26-29, 2015, Proceedings, S. Kozielski, D. Mrozek, P. Kasprowski, B. Małysiak-Mrozek, and D. Kostrzewa, Eds. Cham: Springer International Publishing, 2015, pp. 235–244.

[34] A. Sheth, C. Henson, and S. S. Sahoo, “Semantic Sensor Web,” IEEE Internet Comput., vol. 12, no. 4, pp. 78–83, 2008.

[35] S. Dietze and J. Domingue, “Bridging between sensor measurements and symbolic ontologies through conceptual spaces,” in 1st International Workshop on the Semantic Sensor Web (SemSensWeb 2009) at The 6th Annual European Semantic Web Conference (ESWC 2009), 2009.

[36] M. Stocker, M. Rönkkö, and M. Kolehmainen, “Making sense of sensor data using ontology: A discussion for road vehicle classification.” 2012.

[37] P. Shvaiko and J. Euzenat, “Ontology Matching: State of the Art and Future Challenges,” IEEE Trans. Knowl. Data Eng., vol. 25, pp. 158–176, 2013.

[38] H. Ding, G. Trajcevski, P. Scheuermann, X. Wang, and E. Keogh, “Querying and Mining of Time Series Data: Experimental Comparison of Representations and Distance Measures,” Proc. VLDB Endow., vol. 1, no. 2, pp. 1542–1552, 2008.

[39] W. Hu, J. Chen, H. Zhang, and Y. Qu, “Learning Complex Mappings Between Ontologies,” in Proceedings of the 2011 Joint International Conference on The Semantic Web, 2012, pp. 350–357.

[40] J. Lin, E. Keogh, S. Lonardi, and B. Chiu, “A Symbolic Representation of Time Series, with Implications for Streaming Algorithms,” in Proceedings of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, 2003, pp. 2–11.

[41] M. Compton et al., “The SSN ontology of the W3C semantic sensor network incubator group,” Web Semant. Sci. Serv. Agents World Wide Web, vol. 17, pp. 25–32, 2012.

 ISBN: 978-1-61208-736-8