ISSN XXXX XXXX © 2018 IJESC

Research Article Volume 8 Issue No.6

Integration of Data Tables Using Ontological and Terminological Resource

Vinayaka V M
Lecturer
Department of Computer Science
RJS Polytechnic, Koramangala, Bengaluru, Karnataka, India

Abstract:
In this work, a model for an Ontological and Terminological Resource (OTR) dedicated to the task of n-ary relations annotation in Web data tables has been proposed. This task relies on the identification of the symbolic concepts and the quantities, defined in the OTR, which are represented in the table's columns. It is proposed to guide the annotation by an OTR because it allows a separation between the terminological and conceptual components, and allows dealing with abbreviations and synonyms which could denote the same concept in a multilingual context. The OTR is composed of a generic part to represent the structure of the ontology dedicated to the task of n-ary relations annotation in data tables for any application and of a specific part to represent a particular domain of interest. The model of our OTR and its use for semantic annotation and querying of Web tables is presented.

Keywords: Ontology, Terminology, Fuzzy, Resource description Framework, ONDINE, data tables, @Web, MIEL++ and SPARQL.

I. INTRODUCTION

A. Preamble

The web contains not only a set of semi-structured documents interconnected via hyperlinks, but also contains a huge amount of technical and scientific documents including data tables. A software called Ontology-based Data INtEgration (ONDINE), using the semantic Web framework and language recommendations (XML, Resource Description Framework), to supplement existing local data sources with data tables which have been extracted from Web documents has been designed. ONDINE system relies on an Ontological and Terminological Resource (OTR) which is composed of two parts: on the one hand, a generic set of concepts dedicated to the data integration task and, on the other hand, a specific set of concepts and a terminology, dedicated to a given domain of application.

ONDINE system is composed of two subsystems:

1) @Web subsystem designed to load an XML/RDF data warehouse with data tables which have been extracted from Web documents and semantically annotated using concepts from the OTR.

2) MIEL++ subsystem designed to query simultaneously and uniformly the local data sources and the XML/RDF data warehouse using the OTR in order to retrieve approximate answers in a homogeneous way.

@Web subsystem has four steps. In the first step, relevant documents for the application domain described in the OTR are retrieved from the web and filtered by the human expert. In the second step, data tables are semi-automatically extracted from the documents. In the third step, the extracted data tables are semantically annotated using the OTR. This step generates fuzzy annotations, represented in a fuzzy extension of RDF, which are associated with the data tables represented in XML. In the fourth step, the end user has to validate the fuzzy RDF semantic annotations associated with the data tables before loading them in the XML/RDF data warehouse.

@Web subsystem does not pretend to annotate all data tables extracted from any web documents, but to annotate accurately target data tables extracted from documents identified as relevant for a given domain. Therefore the human intervention is required at each step to guarantee the accuracy of the approach. This paper focuses on the third step that is the semantic annotation method, of @Web subsystem. Its main originality is to produce fuzzy RDF annotations which allow:

1) The recognition and the representation of the imprecise numerical data appearing in the cells of a data table.

2) The computation and explicit representation of the semantic distance between terms in the cells of a data table and the terms of the OTR.

MIEL++ subsystem allows the fuzzy RDF annotations to be queried using SPARQL which is recommended by W3C to query RDF data sources. This subsystem is an extension of the MIEL flexible querying. The main originalities of our new flexible querying subsystem are:

1) To retrieve not only exact answers compared with the selection criteria but also semantically close answers.

2) To compare the selection criteria expressed as fuzzy sets representing preferences with the fuzzy annotations of data tables.

II. LITERATURE SURVEY

[1] In this paper, the authors present the design of ONDINE system which allows the loading and the querying of a data warehouse opened on the Web, guided by an Ontological and Terminological Resource (OTR). The data warehouse, composed of data tables extracted from Web documents, has been built to supplement existing local data sources. The main step of this semiautomatic method is to annotate data tables driven by an OTR. The output of this method is an XML/RDF data warehouse composed of XML documents representing data tables with their fuzzy RDF annotations. This paper presents a flexible querying system which allows the local data sources and the data warehouse to be simultaneously and

uniformly queried, using the OTR. This system relies on SPARQL and allows approximate answers to be retrieved by comparing preferences expressed as fuzzy sets with fuzzy RDF annotations. **[2]** This paper presents an ontology-driven workflow that feeds and queries a data warehouse opened on the Web. Data are extracted from data tables in Web documents. As web documents are very heterogeneous in nature, a key issue in this workflow is the ability to assess the reliability of retrieved data. The main step of this method is to annotate and query Web data tables driven by domain ontology. This paper adopts a method to assess Web data table reliability from a set of criteria by the means of evidence theory. Finally, it shows how to extend the workflow to integrate, the reliability assessment step. **[3]** Companies, governmental agencies and scientists produce a large amount of quantitative (research) data, consisting of measurements ranging from e.g. the surface temperatures of an ocean to the viscosity of a sample of mayonnaise. Such measurements are stored in tables for example spreadsheet samples and research reports. To integrate and reuse such data, it is necessary to have a semantic description of the data. However, the notation used is often ambiguous, making automatic interpretation and conversion to RDF or other suitable format difficult. For example, the table header cell "f (Hz)" refers to frequency measured in Hertz, but the symbol "f" can also refer to the unit farad or the quantities like force. Current annotation tools for this task mentioned in the paper either work on less ambiguous data or perform a more limited task. **[4]** In this paper the authors motivate why it is crucial to associate linguistic information with ontologies and why more expressive models, beyond the label systems implemented in RDF, OWL and SKOS, are needed to capture the relation between natural language constructs and ontological structures. They argue that in the light of tasks such as ontology-based information extraction (i.e., ontology population) from text, ontology learning from text, knowledge-based question answering and ontology verbalization, currently available models are not sufficient as they only allow us to associate literals as labels to ontology elements. Using literals as labels, however, does not allow us to capture additional linguistic structure or information which is definitely needed as they argue. In this paper they propose a model for linguistic grounding of ontologies called LexInfo. LexInfo allows us to associate linguistic information with respect to any level of linguistic description and expressivity to elements in ontology. LexInfo has been implemented as OWL ontology and is freely available together with an API. The authors have discussed the implementation of the LexInfo API, different tools that support the creation of LexInfo lexicons as well as some preliminary applications. **[5]** There are a large number of ontologies currently available on the Semantic Web. However, in order to exploit them within natural language processing applications, more linguistic information than can be represented in current Semantic Web standards is required. Further, there are a large number of lexical resources available representing a wealth of linguistic information, but this data exists in various formats and is difficult to link to ontologies and other resources. The authors present a model that they call lemon (Lexicon Model for Ontologies) that supports the sharing of terminological and lexicon resources on the Semantic Web as well as their linking to the existing semantic representations provided by ontologies. Lemon can succinctly represent existing lexical resources and in combination with standard NLP tools we can easily generate new lexica for domain ontologies according to the lemon model. They have demonstrated that, by combining generated and existing lexica we can collaboratively develop rich lexical descriptions of ontology entities.

[6] The authors present an algorithm, Nomen, for learning generalized names in text. Examples of these are names of diseases and infectious agents, such as bacteria and viruses. These names exhibit certain properties that make their identification more complex than that of regular proper names. Nomen uses a novel form of bootstrapping to grow sets of textual instances and of their contextual patterns. The algorithm makes use of competing evidence to boost the learning of several categories of names simultaneously.

III.ONTOLOGY TERMINOLOGY AND FUZZY SET

A. The Conceptual Component of the OTR

The conceptual component is the ontology of the OTR. It is composed of two main parts: a generic part, commonly called core ontology, which contains the structuring concepts of the data table semantic annotation task, and a specific part, commonly called domain ontology, which contains the concepts specific to the domain of interest. In order to understand the structure of the core ontology, let us detail the data table semantic annotation task. A data table is composed of columns, themselves composed of cells. A data table must be structured in a standardized way, otherwise preliminary transformations are applied on it using state-of-the-art tools like spreadsheets. The cells of a data table may contain terms or numerical values often followed by a measure unit. During the semantic annotation of a data table, cells content are semantically annotated in order to identify the symbolic concepts or quantities represented by its columns and finally the semantic n-ary relationships linking its columns. The core ontology is therefore composed of three kinds of generic concepts: 1)simple concepts which contain the symbolic concepts and the quantities, 2) unit concepts which contain the units used to characterize the quantities, and 3) relations which allow n-ary relationships to be represented between simple concepts.

B. The Terminological Component of the OTR

The terminological component represents the terminology of the OTR, it contains the terms set of the domain of interest. A term is defined as a sequence of words, in a language, and has a label. Terms are divided according to their source language. A term denotes a concept; it must denote at least one concept and it can denote several concepts. The OWL object property denotes, belonging to the core ontology, allows a term to denote a concept. The OWL functional data properties Label and Language, belonging to the core ontology, allow a term to be associated with its label and its language, which are represented as a string. The OTR presented above is at the heart of the ONDINE system which allows local data sources to be supplemented with annotated Web data tables. The semantic annotation of a data table is composed of two steps: 1) identifying which relations defined in the OTR are represented in the data table, 2) instantiating the identified relations, which consists in associating a set of fuzzy RDF annotation graphs with each row of the data table.

C. The Fuzzy Sets

The notion of fuzzy set is an extension of classical subsets. In the classical case, elements of a definition domain X which have some properties belong to a subset A and elements which do not have these properties belong to the complementary subset of A in X. In a fuzzy set, elementscan belong partially to the fuzzy set with a membership degree between 0 (element which is not part of the fuzzy set) and 1 (element which is completely part of the fuzzy set). A fuzzy set defined on a continuous definition domain is called continuous fuzzy set

(CFS) and on a discrete definition domain, discrete fuzzy set (DFS).

IV. PROPOSED SYSTEM

A. The Fuzzy Querying Method

In this section the querying subsystem, called MIEL++, of ONDINE system is presented. MIEL++ querying subsystem allows a uniform querying of two kinds of data sources: the local data sources and the XML/RDF data warehouse, which has been loaded with the data tables extracted from Web documents and semantically annotated. It relies on the OTR used to index the local data sources and to annotate the data tables. MIEL++ querying subsystem allows the end-user to express preferences in his/her query and to retrieve the nearest data stored in the two kinds of data sources corresponding to his/her selection criteria: the OTR—more precisely the hierarchical set of symbolic concepts—is used in order to assess which data can be considered as near to the selection criteria. The end-user asks his/her query to MIEL++ subsystem through a single graphical user interface (GUI), which relies on the OTR. The query is translated into a query comprehensible by each kind of data source, using two subsystems wrappers: an SQL query in the relational source and a SPARQL query in the XML/RDF data warehouse for a complete description of the SPARQL subsystem wrapper. The final answer to the query is the union of the local results retrieved from the two kinds of data sources, which are ordered according to their relevance to the query selection criteria. In this section, the extension of MIEL++ subsystem is presented. This subsystem allows the end user to query fuzzy RDF annotations of data tables, represented in XML documents, by means of SPARQL queries.

B. MIEL++ Query

A MIEL++ query is asked in a view which corresponds to a given relation of the OTR. A view is characterized by its set of queryable attributes and by its actual definition. Each queryable attribute corresponds to a simple concept of the relation represented by the view. The notion of view must be understood with the meaning of the relational database model. It allows the complexity of the querying into different data sources to be hidden to the end user. A MIEL++ query is an instantiation of a given view by the end user, by specifying, among the set of queryable attributes of the view, which are the selection attributes and their corresponding searched values, and which are the projection attributes. An important feature of a MIEL++ query is that searched values may be expressed as continuous or discretefuzzy sets. A fuzzy set allows the end user to express his/her preferences which will be taken into account to retrieve not only exact answers (corresponding to values associated with the kernel of the fuzzy set) but also answers which are semantically close (corresponding to values associated with the support of the fuzzy set). When a MIEL++ query is asked by the end user into the XML/RDF data warehouse which contains fuzzy RDF graphs generated by our annotation method to annotate XML data tables, the query processing has to deal with fuzzy values. More precisely, it has 1) to take into account the certainty score associated with the relations represented in the data tables and 2) to compare a fuzzy set expressing querying preferences to a fuzzy set, generated by our annotation method, having a semantic of similarity or imprecision. For the first point, the end user may specify a threshold which determines the minimum acceptable certainty score to retrieve the data. In a MIEL++ query, the end user can express preferences in his/her selection criteria as fuzzy sets. Since fuzzy sets are not supported in a standard SPARQL query, it is proposed to "defuzzify" the MIEL++ query before translating it into SPARQL. This allows any implementation of SPARQL to be used by our querying subsystem. The SPARQL query is automatically generated 1) from the signature of the relation represented by the view and associated with the MIEL++ query and 2) from the sets of projection and selection attributes of the MIEL++ query.

C. The Construction of a MIEL++ Answer

An answer to a MIEL++ query must 1) satisfy the minimal acceptable certainty score associated with the query; 2) satisfy all its selection criteria, and 3) associate a constant value with each of its projection attributes. An answer to a MIEL++ query into the XML/RDF data warehouse is computed in three steps. First, the corresponding SPARQL query is generated and executed into the XML/RDF data warehouse. Then, the values associated with the selection attributes in each fuzzy RDF answer graph are extracted in order to measure how the answer graph satisfies the selection criteria. Finally, the values associated with the projection attributes in each fuzzy RDF answer graph are extracted to be retrieved to the end user. Let us notice that the values extraction from an answer graph is performed through SPARQL queries which are defined for each selection and projection attributes of the MIEL++ query. To measure the satisfaction of a selection criteria, the two semantics—imprecision and similarity—associated with fuzzy values of the XML/RDF data warehouse must be considered. On the one hand, two classical measures have been proposed to compare a fuzzy set representing preferences to a fuzzy set having a semantic of imprecision: a possibility degree of matching denoted and a necessity degree of matching denoted N. On the other hand, to use the adequation degree to compare a fuzzy setrepresenting preferences to a fuzzy set having a semantic of similarity has been proposed.

D. @Web subsystem

@Web is a data warehouse opened on the Web centered on the integration of heterogeneous data tables extracted from Web documents. The focus has been put on Web tables for two reasons: (i) experimental data are often summarized in tables, (ii) table structured data are easier to integrate than, e.g., in text or plots. A central role in data integration in @Web is played by the domain ontology. This ontology describes the concepts, their relations and the associated terminology of a given application domain. @Web can therefore be instantiated for any application domains (e.g., food predictive microbiology, food chemical risks, aeronautics, automobile), provided a proper domain ontology is defined. Once the ontology is built, @Web workflow includes the different steps to integrate new data in the warehouse. Concepts found in a data table and semantic relations linking these concepts are automatically identified. Data tables are then annotated with the identified concepts, allowing users to interrogate and query the data warehouse in a homogeneous way.

i. @Web generic ontology

The current OWL ontology representation used in the @Web system is composed of two main parts: a generic part, called core ontology, which contains the structuring concepts of the Web table integration task, and a specific part, commonly called domain ontology, which contains the concepts specific to the considered domain. The core ontology is composed of symbolic concepts, numeric concepts and relations between these concepts. It is separated from the definition of the concepts and relations specific to a given domain, i.e., the domain ontology. All the ontology concepts are materialized

by OWL classes. For example, in the microbiological ontology, the respectively symbolic and numeric concepts Microorganism and pH are represented by OWL classes, respectively subclass of the generic classes Symbolic Concept and Numeric Concept.

ii. @Web work flow

The first three steps of @Web workflow are as follows. The first task consists in retrieving relevant Web documents for the application domain, using keywords extracted from the domain ontology. It does so by defining queries executed by different crawlers. Data tables are extracted from the retrieved documents and are semi-automatically translated into a generic XML format. The Web tables are then represented in a classical and generic way – i.e., a set of lines, each line being a set of cells. In the third task, the Web tables are semantically annotated according to the domain ontology. The semantic annotation process of a Web table consists in identifying which semantic relations of thedomain ontology can be recognized in each row of the Web table. This process generates RDF descriptions.

E. SPARQL querying of RDF graphs

In the XML/RDF data warehouse, the querying is done through MIEL++ queries. Briefly recall how MIEL++ queries are executed in the current version of @Web . A MIEL++ query is asked in a view that corresponds to a relation of the ontology. A MIEL++ query is an instantiation of a view by the end-user, who specify among the set of querryable attributes of the view what are the selection attributes and their searched values, and what are the projection attributes (with the meaning of the relational model). An important specificity of a MIEL++ query is that searched values may be expressed as fuzzy sets, whose use allows end-users to represent their preferences in a gradual way.

F. Ontology-based filtering and disambiguation

A useful ontology-based scoring technique is to use concepts related to the candidate concept. If these related concepts are detected in the text near to the candidate concept, this increases the likelihood that a candidate is correct. This technique is implemented for our domain through the relationship between units and their quantity listed in our ontology. If the values lie outside the unit's value range, the candidate is removed. This is not likely to work for large quantitative ontologies and varied datasets. For example, a temperature value of "-20" can only rule out the unit Kelvin (its scale starts from 0), but leaves Celsius and Fahrenheit as possible interpretations. In case we are dealing with a relative temperature, then "-20" can even not strike Kelvin from the list of candidates. Celsius and Fahrenheit can only be disambiguated by a few actual values, which are unlikely to appear in actual measurements. None of the techniques mentioned above addresses the problem of ambiguous compound concepts (e.g. m/s might refer to meter per second or mile per Siemens). A solution has to be developed that uses an ontology to determine whether the units together express a quantity that is defined in the ontology.

V. CONCLUSIONS

Asystem, called ONDINE, built, using the recommendations of the W3C, on a generic OTR expressed in OWL. ONDINE system allows XML data tables, which have been extracted from Web documents, to be annotated with fuzzy RDF descriptions and to be flexibly queried using SPARQL. Fuzzy RDF annotations are used to represent (1) the set of most similar symbolic concepts of the OTR which are automatically associated with the content of a cell belonging to a symbolic column, (2) imprecise values associated with a quantity expressed in one or several numerical columns, (3) a degree of certainty associated with each n-ary relation recognized in a data table. OTR, ONDINE and MIEL++ subsystems are built using the proposed algorithms. The other perspectives concern the improvement of ONDINE system by

1) Completing the cosine similarity measure used to compare terms with other syntactical and semantic techniques
2) Completing the semantic annotation of data tables in Web documents with the annotation of the text using the OTR.
3) Managing OTR evolution by taking into account annotation results and other ontology's.

VI. REFERENCES

[1]. Patrice Buche, Juliette Dibie-Barthelemy, Liliana Ibanescu, and LydieSoler , "Fuzzy Web Data Tables Integration Guided by an Ontological and Terminological Resource", IEEE Transactions on Knowledge and Data Engineering, vol.25, pp 805-819, April 2013.

[2].S´ebastienDestercke and Patrice Buche and Brigitte Charnomordic, "Data reliability assessment in a data warehouse opened on the Web", Proceedings of the 9th international conference on Flexible Query Answering Systems, Springer-Verlag Berlin, pp 174-185, 2011.

[3].Mark van Assem, HajoRijgersberg, Mari Wigham and Jan Top, "Converting and Annotating Quantitative Data Tables", Proceedings of the 9th international semantic web conference on The semantic web – Springer Volume PartI, pp 16-31, 2010 .

[4].P. Cimiano , P. Buitelaar , J. McCrae , M. Sintek, "LexInfo: A Declarative Model for the Lexicon-Ontology Interface", Journal of Web Semantics, vol. 9, no. 1, pp. 29-51, 2011.

[5].John McCrae, Dennis Spohr, and Philipp Cimiano, "Linking Lexical Resources and Ontologies on the Semantic Web with lemon", 8th Extended Semantic Web Conference, The Semantic Web: Research and Applications (ESWC), pp. 245-259, 2011.

[6].Roman Yangarber, Winston Lin, Ralph Grishman, "Unsupervised Learning of Generalized Names", Proceedings of the 19th International conference on Computational linguistics, vol.1 pp1-7, 2002 .