Fuzzy Web Data Tables Integration Guided by an Ontological and Terminological Resource

Patrice Buche, Juliette Dibie-Barthelemy, Liliana Ibanescu, Lydie Soler

Fuzzy Web Data Tables Integration Guided by an Ontological and Terminological Resource

Patrice Buche, Juliette Dibie-Barthelemy, Liliana Ibanescu, Lydie Soler

***Abstract*—In this paper, we 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. First, we present the main steps of our semi-automatic method 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. We then present our 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.**

***Index Terms*—Knowledge and data engineering tools and techniques; XML/XSL/RDF; Uncertainty, "fuzzy," and probabilistic reasoning; Representations, data structures, and transforms; Knowledge modelling.**

I. Introduction

TODAY'S Web is not only a set of semi-structured documents interconnected via hyper-links. A huge amount of technical and scientific documents, available on the Web or the hidden Web (digital libraries, ...), include data tables. Those data tables can be seen as small relational databases even if they lack the explicit meta data associated with a database. They represent a very interesting potential external source for loading the data warehouse of a company dedicated to a given domain of application. They can be used to enrich local data sources. In order to integrate data, a preliminary step consists in harmonizing external data with local ones, i.e. external data must be expressed with the same vocabulary as the one used to index the local data. We have designed a software called ONDINE (ONtology based Data INtEgration), using the semantic Web framework[1] and language recommendations (XML, RDF, OWL, SPARQL), which implements the entire management system, presented in Figure 1, to supplement existing local data sources with data tables which have been extracted from Web documents.

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 as detailed in Figure 1. In the first step, relevant documents for the application domain described in the OTR are retrieved from the Web and filtered by a 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 data tables represented in XML. In the fourth and last step, the end-user has to validate the fuzzy RDF semantic annotations associated with data tables before loading them in the XML/RDF data warehouse. Let us notice that @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. The human intervention at each of its step is therefore required to guarantee the accuracy of the approach. In this paper, we focus on the third step, that is the semantic annotation method, of @Web subsystem. Its main originality is to produce fuzzy RDF annotations which allow: (i) the recognition and the representation of imprecise numerical data appearing in the cells of a data table; (ii) the computation and explicit representation of the semantic distance between terms in the cells of a data table and terms of the OTR. MIEL++ subsystem allows the

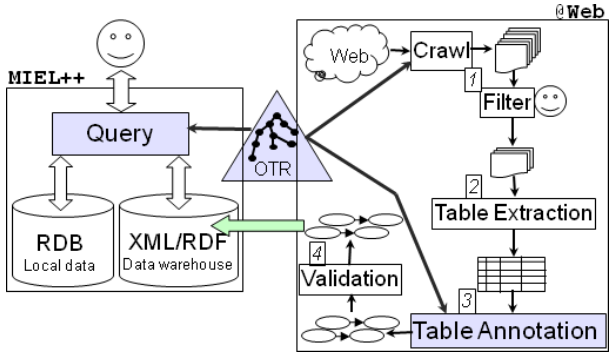

Fig. 1. ONDINE system.

P. Buche is with UMR INRA IATE and LIRMM, Montpellier, FRANCE. e-mail: buche@supagro.inra.fr.

J. Dibie-Barthelemy and L. Ibanescu are with INRA Metarisk & AgroParisTech, Paris, FRANCE. e-mail: {dibie, liliana.ibanescu}@agroparistech.fr.

L. Soler is with INRA Metarisk.e-mail: Lydie.Soler@paris.inra.fr

[1] http://www.w3.org/standards/semanticweb/

fuzzy RDF annotations to be queried using SPARQL[2] which is recommended by W3C to query RDF data sources. This subsystem is an extension of the MIEL flexible querying system proposed in [1] and [2]. The main originalities of our new flexible querying subsystem are: (i) to retrieve not only exact answers compared with the selection criteria but also semantically close answers; (ii) to compare the selection criteria expressed as fuzzy sets representing preferences with the fuzzy annotations of data tables. Some preliminary studies of this work have already been published in [3], [4] and [5]. This paper provides a synthetic overview of ONDINE system which relies on a new modelling of the OTR dedicated to the data integration task. The definition of this OTR, central in ONDINE system, was essential to consolidate the approach and ensure its sustainability and its future evolutions. @Web subsystem (previously presented in [3] and [4]) and MIEL++ subsystem (previously presented in [5]) have been revised to take into account this new OTR. In Section II, we present the new model of the OTR. The new @Web and MIEL++ subsystems are then presented in the three next sections. The semantic annotation method of @Web subsystem, which allows data tables, extracted from Web documents, to be fuzzy annotated using the OTR, is presented in two sections. In Section III, we present the method which allows one to identify which concepts of the OTR are represented in a data table. The instantiation of these concepts for each row of the annotated data table, relying on fuzzy RDF annotations, is presented in Section IV. In Section V, MIEL++ subsystem which allows a flexible querying of the fuzzy annotated data tables, stored in the XML/RDF data warehouse, using SPARQL is presented. Experimental results are given all along Sections III, IV and V. Our approach is compared with the state of the art in Section VI. We conclude and present the perspectives of this work in Section VII.

II. The Ontological and Terminological Resource

In [6]–[10] ontologies are associated with terminological and/or linguistic objects. In [6], authors motivate why it is crucial to associate linguistic information (part-of-speech, inflection, decomposition, etc.) with ontology elements (concepts, relations, individuals, etc.) and they introduce *LexInfo*, an ontology-lexicon model, implemented as an OWL[3] ontology. Adapting *LexInfo*, [7] presents a model called *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. The *CTL* model from [8] is a model for the integration of conceptual, terminological and linguistic objects in ontologies. In [9] a meta-model for ontological and terminological resources in OWL DL is presented, called an *Ontological and Terminological Resource (OTR)*, extended afterward in [11] in order to be used for ontology based information retrieval applied to automotive diagnosis.

The ontology we used in our previous works [3]–[5] was not designed to allow one to define the terminology and its variations (multi-lingual, synonyms, abbreviations, ...) denoting the concepts. We therefore propose to use an Ontological and Terminological Resource (OTR) [9] allowing joint representation of an ontology and its associated terminology. According to [9], three factors influence the OTR structuring: the task to realize, the domain of interest and the application. The OTR used in ONDINE system has been designed for the data table integration (annotation and querying) task. In this paper, the domain of interest is food safety but the OTR structure we propose is generic enough to be applied to many other domains. For example, in this paper, experimental results in aeronautics will be also presented. The application is the construction of a data warehouse opened on the Web.

Since ONDINE system allows local data sources to be supplemented with data tables which have been extracted from Web documents, the domain specific part of the OTR was manually built by ontologists taking into account (i) the vocabulary used in the preexisting local databases in order to index the data and (ii) the domain information available within the databases schema. Examples given in this paper concern the microbial risk domain. We present first, the conceptual component of the OTR and second, its terminological component, using the OWL2-DL model[4].

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.

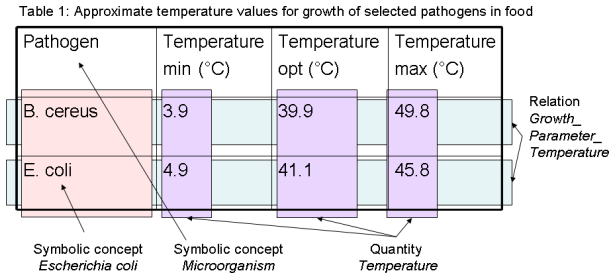

Fig. 2. Annotation of a table according to concepts defined in the OTR.

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 (which is included in the table extraction step in Figure 1). The cells of a data table may contain terms[5] 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

[2] http://www.w3.org/TR/rdf-sparql-query/
[3] http://www.w3.org/TR/2004/REC-owl-features-20040210/
[4] http://www.w3.org/TR/owl2-direct-semantics/
[5] A term is defined as a sequence of words.

its columns. For instance, in Figure 2, the cell content 'E.coli' is associated with the symbolic concept *Escherichia coli* by our annotation method (detailled in Sections III and IV), the content of the three cells 4.9, 41.1 and 45.8 are associated with the quantity *Temperature* and the entire content of the second row of the data table is considered as an instance of the n-ary relation *GrowthParameterTemperature* which associates a given microorganism (like Escherichia coli) with its temperature growing conditions in a food product.

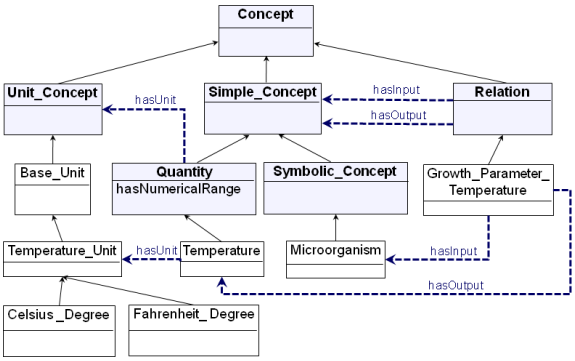

Fig. 3. An excerpt of the conceptual component of the OTR in microbial risk domain.

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.

The concepts belonging to the domain ontology, called *specific concepts*, appear in the OTR as sub concepts of the generic concepts. Figure 3 presents an excerpt of the conceptual component of the OTR in microbial risk domain. In OWL, all concepts are represented by classes which are pairwise disjoints and are hierarchically organized by the *subClassOf* relationship. The nodes represent the OWL classes, the solid arrows the "is-a" relationship between classes and the dashed arrows properties between classes. For instance, the property *hasUnit* links a quantity (e.g. a *Temperature*) with its units of measurement (e.g. *Celsius_Degree* and *Fahrenheit_Degree*). We detail below the three kinds of generic concepts and their sub specific concepts in microbial risk domain.

1) The unit concepts: Unit concepts allow the meaning of units to be represented. Our classification relies on the international system of units[6] which decomposes the units into base units and derived units. There exist several ontologies dedicated to quantities and associated units (OM[7], QUDT[8], QUOMOS, OBOE[9], ...). We learn from these ontologies to build ours, but they cannot contain all the required specific units for a given domain. For instance, in microbial risk domain, the ontologist has added some units such as ppm[10] or CFU/g[11]. Figure 4 presents an excerpt of the unit concepts in microbial risk domain.

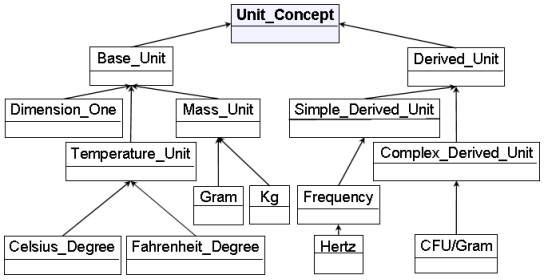

Fig. 4. An excerpt of the unit concepts in microbial risk domain.

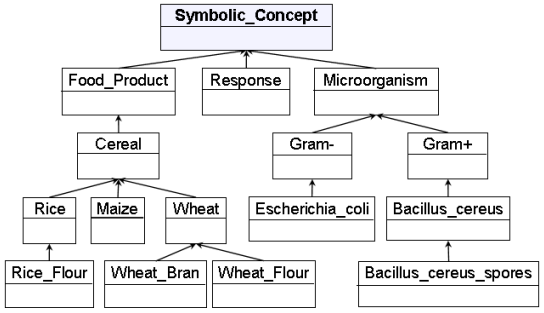

Fig. 5. An excerpt of the symbolic concepts in microbial risk domain.

2) The simple concepts: Symbolic concepts allow the meaning of terms to be represented. Symbolic concepts are hierarchically organized by the "is-a" relationship. Figure 5 presents an excerpt of the specific symbolic concepts in microbial risk domain. The microbial risk domain OTR contains three distinct sub hierarchies of specific symbolic concepts: the specific symbolic concept *Food_Product* with more than 400 sub concepts, the specific symbolic concept *Microorganism* with more than 150 sub concepts and the specific symbolic concept *Response* with three sub concepts: *growth*, *absence of growth* and *death*, which represent the possible responses of a microorganism to a treatment. These sub hierarchies have been defined by ontologists. We could not reuse pre-existing terminologies for food products such as AGROVOC[12] (from FAO - Food and Agriculture Organisation of the United Nations) or Gems-Food[13] (from WHO - World Health Organisation) because those terminologies are not specific enough compared with the one built from our corpus in microbial risk (respectively only 20% and 34% of common words).

Quantities allow the meaning of numerical values to be represented. A quantity is described by a set of units, which are sub concepts of the unit concept, and eventually a numerical range. Two properties *hasUnit* and *hasNumericalRange*, belonging to the core ontology, link respectively quantities to their associated units and numerical range. The OWL object property *hasUnit* allows a quantity to be described by one or several unit concepts. OWL2-DL datatype restrictions using facet spaces allow the numerical range of a quantity to be rep-

[6] http://www.bipm.org/en/si/
[7] http://www.wurvoc.org/vocabularies/om-1.8/
[8] http://www.qudt.org/
[9] http://marinemetadata.org/references/oboeontology
[10] parts per million. ppm is a unit of concentration often used when measuring levels of pollutants in air, water, body fluids, etc.
[11] colony-forming units per gram. Colony-forming units (CFU) is a measure of viable bacterial or fungal numbers in microbiology
[12] http://aims.fao.org/website/AGROVOC-Thesaurus
[13] http://www.who.int/foodsafety/chem/gems/en/

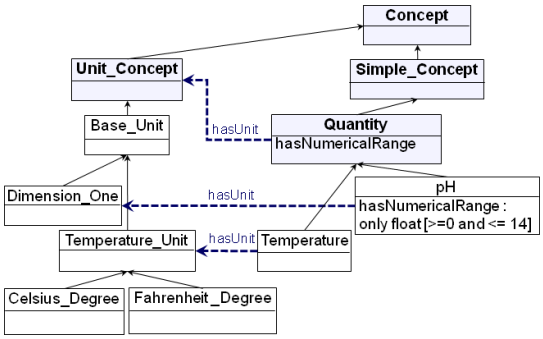

Fig. 6. An excerpt of the quantities associated with unit concepts in microbial risk domain.

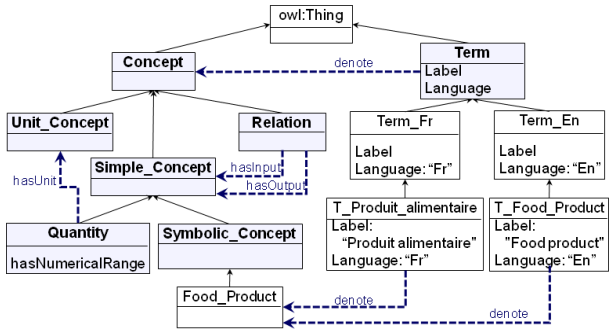

Fig. 7. An excerpt of the OTR in microbial risk domain.

resented in the OWL datatype property *hasNumericalRange*. Figure 6 presents an excerpt of the quantities in microbial risk domain. Eighteen specific quantities have been defined for the microbial risk domain. The specific quantity *Temperature* can be expressed using the unit °C (represented by the concept *Celsius_Degree*) or °F (represented by the concept *Fahrenheit_Degree*) and has no numerical range. The specific quantity *pH* is associated with the unit *Dimension_One* (i.e. with no unit) and is restricted to the numerical range $[0, 14]$.

3) The relations: Relations allow the meaning of n-ary relationships between simple concepts to be represented. A relation is defined by its signature, which is composed of several input simple concepts and one output simple concept. The input simple concepts represent the domain of the relation. A relation may have several input simple concepts. The output simple concept represent the range of the relation. The restriction of the range to only one output simple concept is justified by the fact that, in a data table, a relation often represents a semantic n-ary relationship between simple concepts with only one result, such as an experimental result which may have several entry factors. If a data table contains several result columns, it is then represented by as many relations as it has results. Two properties, belonging to the core ontology, called *hasInput* and *hasOutput*, link a relation to its domain and range. Since a relation represents in the OTR a n-ary relationship, we learned from W3C[14] which suggests to decompose a n-ary relationship into n binary relationships. Consequently, a n-ary relation is represented in OWL by a class associated with the simple concepts of its signature via the OWL object property *hasInput* or the OWL functional object property *hasOutput*. In Figure 3, for instance, the specific relation *Growth_Paramater_Temperature* has for input the specific symbolic concept *Microorganism* and for output the specific quantity *Temperature*. The microbial risk domain OTR contains sixteen relations.

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 *denote*, 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. Figure 7 presents an excerpt of the OTR in microbial risk domain. The specific terms *T_Produit_Alimentaire* and *T_Food_Product* both denote the specific symbolic concept *Food_Product*.

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 (see Figure 1). We present in the next two sections the semantic annotation method of @Web subsystem which allow data tables, extracted from Web documents, to be annotated thanks to the OTR, before being added to the XML/RDF data warehouse. The semantic annotation of a data table is composed of two steps: (i) identifying which relations defined in the OTR are represented in the data table, (ii) instantiating the identified relations, which consists in associating a set of fuzzy RDF annotation graphs with each row of the data table.

III. The Relations Identification in a Data Table

Given the OTR, described above, and given a data table extracted from a document found on the Web, we want to find which relations of the OTR are represented in this data table. An aggregation approach is used for that purpose, looking first at the contents of the cells, then identifying the simple concepts of the OTR represented in the columns and finally comparing the signature of the data table (the column concepts) with the signatures of the relations in the OTR. The main steps of the relations identification method are presented in Figure 8: first, symbolic and numerical columns are distinguished, using some of the knowledge described in the OTR (mainly the unit concepts; for better description of this step, please refer to [3] which is a preliminary version of this work); then, the simple concepts represented by the symbolic columns and by the numerical columns are identified; finally, the relations represented in the data table are identified.

We detail below the steps A, B, C and D from Figure 8. Each of these steps was experimented on three domains: microbial risk, chemical risk and aeronautics. Three OTR were build, their domain specific part being manually built

[14] http://www.w3.org/TR/swbp-n-aryRelations

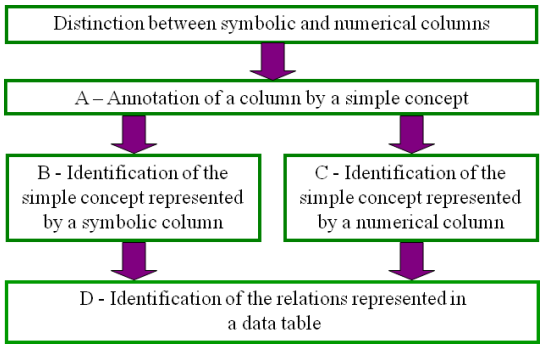

Fig. 8. The main steps of the relations identification method.

TABLE I
OTR SIZE FOR THREE DOMAINS.

domain	# U	# Q	# SC	# R	# T
microbial risk	33	16	689	18	1,492
chemical risk	12	8	2,732	6	5,413
aeronautics	45	28	121	26	516

TABLE II
CORPORA CHARACTERISTICS.

domain	# tables	# columns	# rows
microbial risk	60	342	700
chemical risk	12	87	533
aeronautics	18	160	128

by ontologists. The size of each set of concepts belonging to the three OTR is given in Table I, where U is the set of unit concepts, Q is the set of quantities, SC is the set of symbolic concepts, R is the set of relations and T is the set of terms. Characteristics of their associated corpora is presented in Table II. OTR and corpora are available on the Web[15].

A. The annotation of a column by a simple concept

In this step, we want to identify which simple concept of the OTR corresponds to a column classified as a symbolic column or as a numerical one. Only the simple concepts which appear in the signatures of the relations belonging to the OTR are considered. As a matter of fact, the main objective of the semantic annotation method is to identify which relations of the OTR are represented in a data table: those simple concepts are called in the following *simple target concepts*. In order to annotate a column col by a simple target concept, a score is computed for each simple target concept c of the OTR for the column col. This score, called *final score* and denoted $score_{final}(c, col)$, is computed as a combination of the score of the simple target concept c for the column col according to the column title, called *title score* and denoted $score_{title}(c, col)$, and the score of the simple target concept c for the column col according to the column content, called *content score* and denoted $score_{content}(c, col)$:

[15]http://www.paris.inra.fr/metarisk/research_unit/data_integration/softwares/ondine_corpora_and_otr_july_2011

$$score_{final}(c, col) = 1 - (1 - score_{title}(c, col))(1 - score_{content}(c, col)) \quad (1)$$

The final score is inspired from [12] where both combined scores reinforce each other. Nevertheless, at least one score must be high to have a high final score. Those scores rely, in particular, on the following term similarity measure,which is classically used in Information Retrieval: let a and b be two terms, represented as weighted vectors $(a_1, \ldots, a_n)$ and $(b_1, \ldots, b_n)$, the coordinate axis corresponding to the lemmatised words, the coordinate values corresponding to the weight of the word in the term (0 if the word is not part of the term); the term similarity measure between a and b, denoted $sim(a, b)$, is the cosine similarity measure [13]:

$$sim(a, b) = \frac{\sum_{i=1}^{n} a_i b_i}{\sqrt{\sum_{i=1}^{n} a_i^2 \times \sum_{i=1}^{n} b_i^2}} \quad (2)$$

Each word found in terms from the Web and in terms from the OTR is given a weight of 1 (except stopwords such as articles or prepositions as well as words which contain only one letter, which are given a weight of 0).

The title score of a simple target concept c for a column col, presented in Equation 3, is the maximum of the term similarities between the terms t_c^i denoting the concept c and the term t_{title} denoting the column title. Note that several terms can denote a concept.

$$score_{title}(c, col) = max_i \ sim(t_c^i, t_{title}) \quad (3)$$

The content score of a simple target concept for a column depends upon the column was categorized as a symbolic column or as a numerical one. We present the way this score is computed in the next two sub sections.

B. The identification of the simple concept represented by a symbolic column

The computation of the score of a symbolic target concept for a column according to the column content relies on term similarities between the terms found in the column and the terms defined in the OTR. To compute this score, we first explore each cell of the column (excluding the title). For each cell $cell$ of the column, the score of each symbolic target concept c of the OTR for the cell $cell$ is computed as the sum of the maximum of the term similarities between the term t_{cell} denoting the content of the cell and the terms $t_{c'}^i$ denoting the symbolic concept c', where c' is the symbolic target concept c or one of its sub concepts:

$$score_{cell}(c, cell) = \sum_{c' \in hierarchy(c)} (max_i \ sim(t_{c'}^i, t_{cell})) \quad (4)$$

The proportional advantage of the symbolic target concept having the best score is then computed for each cell $cell$ of the column: $advantage(best, cell) =$

$$= \frac{score_{cell}(best, cell) - score_{cell}(secondBest, cell)}{score_{cell}(best, cell)} \quad (5)$$

where $best$ is the symbolic target concept with the best score and $secondBest$ the symbolic target concept with the second best score.

The cell $cell$ is annotated with the symbolic target concept $best$ if its proportional advantage is higher than a specified threshold (for our experiments, this threshold was set to 10%). Otherwise, the cell is annotated with the generic concept *Symbolic_Concept*, which means that this cell is a symbolic one but we cannot say anything else.

Once all cells in the column have been annotated with a symbolic target concept, we compute the score of a symbolic target concept for the column according to the column content as the proportion of cells in the column which have been annotated with this symbolic target concept. Let col be a symbolic column with n_{col} the number of cells in the column (excluding the column title), c a symbolic target concept with $n(c, col)$ the number of cells in the column annotated with the symbolic target concept c, the content score of c for col is:

$$score_{content}(c, col) = \frac{n(c, col)}{n_{col}} \quad (6)$$

The content score is computed for each symbolic target concept c of the OTR. Considering the title score given in Equation 3, the final score of each symbolic target concept for the column is then computed according to Equation 1. The column is annotated by the symbolic target concept having the best final score for the column, assuming that its proportional advantage (replacing the score for a cell of Equation 5 by the final score for a column) is greater than a specified threshold (in our experiments, this threshold was set to 10%). Otherwise, the column is annotated by the concept *Symbolic_Concept*.

Computing the content score for a column has a complexity in # CELLS $\times$ # STCH $\times$ # T with # CELLS the number of cells in the column, # STCH the average number of concepts, sub concepts of a symbolic target concept of the OTR and # T the average number of terms denoting a simple concept of the OTR. This complexity may be expensive, especially for application domains with big tables and a large number of terms denoting symbolic concepts (see, for instance, the chemical risk corpus and OTR in Tables I and II). In @Web, it is possible to make this computation in batch mode, the user being alerted when it is finished.

Experimental results: Our annotation method was experimented using the three OTR presented in Table I and their associated corpora presented in Table II. In the microbial risk corpus, 81 columns were classified as symbolic. Those columns were manually annotated with one of the three symbolic target concepts of our microbial risk OTR: 46 columns were annotated with *Food_Product*, 16 columns were annotated with *Microorganism* and 1 column was annotated with *Response*; the rest of 18 columns were annotated with the symbolic concept *Other*. When applying our annotation method: i) 37 columns were annotated with *Food_Product* where 34 were true positive and 3 false positive; ii) 19 columns were annotated with *Microorganism* where 16 were true positive and 3 false positive; iii) 1 column was annotated with *Response*; iv) 24 columns were annotated with *Other* where 12 were true positive and 12 false positive. Experimental results are given in Table III, where precision and recall are calculated as the average of respectively precision and recall, which have been calculated for each target concept and for the concept *Other*. In order to assess the quality of our results, we compared our method with a machine learning classifier. To the best of our knowledge, there is no classifier which is dedicated to the classification of symbolic data using an OTR. Our method was compared with the SMO classifier [14] which is an optimized version of the well-known SVM. We therefore propose a comparison between two alternatives: SMO which uses no domain knowledge *but* uses learning and our annotation method which relies on domain knowledge *but* has no learning phase. For the SMO classifier, the following pre-treatment was used: each distinct lemmatized word present in a column results in an attribute; the value of this attribute for a given column is the frequency of the word in the column. The SMO classifier was evaluated using a leave-one-out cross-validation, with default parameters of the Weka[16] implementation.

TABLE III
CLASSIFICATION RESULTS ON 81 SYMBOLIC COLUMNS IN MICROBIAL RISK.

Target concept	# of columns	our method using the OTR			
		Food	Micro.	Resp.	Other
Food	46	**34**	0	0	12
Micro.	16	0	**16**	0	0
Response	1	0	0	**1**	0
Other	18	3	3	0	**12**

precision: **82%**; recall **85%**

Target concept	# of columns	SMO			
		Food	Micro.	Resp.	Other
Food	**46**	**45**	0	0	1
Micro.	16	4	**12**	0	0
Response	1	0	0	**0**	1
Other	18	7	0	0	**11**

precision: **88%**; recall **78%**

It appears that our method, which uses domain knowledge described in an OTR, but no learning phase, gives similar results to the learning classifier. We do not claim that it is easier to build an OTR than a learning set, but in our approach the OTR is supposed to exist and this step of symbolic columns annotation takes place in a more general system guided by an OTR. Besides, those good experimental scores can be explained by the fact that we work on restricted domain for which the OTR is significantly representative of its vocabulary and its variations. As a matter of fact, the ONDINE system does not pretend to annotate all data tables extracted from any Web documents, but target data tables extracted from documents identified as relevant for the studied domain.

Our annotation method was also experimented on two other corpora: chemical risk whose OTR contains 3 symbolic target concepts and aeronautics whose OTR contains 4 symbolic target concepts. Over the 22 columns of the chemical risk corpus classified as symbolic, our annotation method obtains a precision of 100% and a recall of 100%. Over the 46 columns of the aeronautics corpus classified as symbolic, our

[16] http://www.cs.waikato.ac.nz/ml/weka

annotation method obtains a precision of 82% and a recall of 100%. The good experimental scores the chemical risk corpus main be explained by (i) the richness of the terminological component (e.g. number of terms denoting symbolic concepts) of the chemical risk OTR (see I) and (ii) a good distinction in the terminology between the three symbolic concepts (Food Product, Contaminants and Countries).

C. The identification of the simple concept represented by a numerical column

As it has been done for the symbolic columns, we want to identify which target quantity of the OTR corresponds to a column classified as a numerical one. For that, the final score of each target quantity of the OTR for the column, and therefore its content score, must be computed.

The computation of the score of a target quantity for a column according to the column content relies on the units present in the column and its numerical values, which must be compatible with the numerical range of the target quantity. To compute this score, we first compute the score of a target quantity for each unit concept present in the column. A target quantity has a score for each unit concept which depends upon the number of quantities that can be expressed in this unit concept in the OTR: a unit concept is especially discriminant since less quantities can be expressed in this unit concept. Let u be a unit concept and C_u be the set of quantities of the OTR which can be expressed in this unit concept, the score of the target quantity c for the unit concept u is:

$$score_{unit}(c,u) = \begin{cases} \frac{1}{|C_u|} & \text{if } c \in C_u \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

A unit concept u is considered as present in a column col, denoted $u \in Unit_{col}$, if there exists a term in the column such that it is similar with at least one of the terms which denote the unit concept in the OTR. Let T_{col} be the set of terms present in the column col and T_u the set of terms which denote u:

$$u \in Unit_{col} \text{ if } \exists t \in T_{col}, \exists t_u \in T_u, sim(t_u, t_{col}) = 1 \quad (8)$$

Let us notice that if no unit concept from the OTR was identified in the column, then the column is considered as having the unit concept *Dimension_One* (i.e. no unit).

The most discriminant unit for a given quantity is favoured. Consequently, the content score of a target quantity c for the column col is computed as the maximum of the scores of the target quantity c for each unit concept present in the column:

$$score_{content}(c, col) = max_{u \in Unit_{col}} \ score_{unit}(c,u) \quad (9)$$

The content score is computed for each target quantity of the OTR. Considering the title score of Equation 3, the final score of each target quantity for the column is computed as follows:

- if all values in the column are compatible with the numerical range of the target quantity, then the final score of this target quantity for the column is computed according to Equation 1,
- else, the final score of the target quantity c for the column col is null: $score_{final}(c, col) = 0$.

Positive for Campylobacter (%)
0.07

Fig. 9. Example of a numerical column of a data table.

TABLE IV
QUANTITY IDENTIFICATION RESULTS IN NUMERICAL COLUMNS.

domain	# Q	# NC	precision	recall
microbial risk	18	261	96%	93%
chemical risk	5	65	100%	85%
aeronautics	29	114	96%	86%

The column is annotated by the target quantity having the best final score for the column, assuming that its proportional advantage (replacing the score for a cell of Equation 5 by the final score for a column) is greater than a specified threshold (in our experiments, this threshold was set to 10%). Otherwise, the column is annotated by the generic concept *Quantity*.

Example 1: In the numerical column presented in Figure 9, % was identified as the unit present in the column. There are 5 quantities which can be expressed in this unit: *NaCl*, *N2*, *O2*, *CO2* and *Sample Positive*. The score of each quantity c for the unit concept % is therefore: $score_{unit}(c, \%) = \frac{1}{5}$. Moreover, we have $score_{content}(c, col) = score_{unit}(c, \%)$ because only one unit has been identified in the column. Since only the quantity *Sample Positive*, with no numerical range, has a not null title score, then the column is annotated by this quantity with $score_{final}$(Sample Positive, col)=1-(1-0.5)(1-0.2)=0.6.

Experimental results: Our annotation method was experimented using the three OTR presented in Table I and their associated corpora presented in Table II. Columns extracted from data tables and automatically classified as numerical columns were manually annotated with one of the quantity target concepts of the considered OTR. In Table IV, Q is the set of quantities in an OTR and NC is the set of numerical columns manually annotated with a quantity from the OTR; precision and recall are given for our method. Those good experimental scores can mainly be explained by the combination of evidences used by our method (quantities names, units associated with quantities, numerical range of values), which give enough constraints to find the right identification in the three experimentations. Nevertheless, the scores may decrease if one of those evidences are less discriminant. For instance, units associated with quantities could be poorly discriminant in some applications and therefore lead to lower scores.

D. The identification of the relations represented in a table

Once all columns of a data table have been annotated with a simple concept of the OTR, we want to identify which relations of the OTR are represented in the data table. For that, Equation 1 is used to compute the final score of each relation of the OTR for the data table.

The title score of a relation for a data table is computed using Equation 3, where $title$ is the title of the data table.

TABLE V
EVALUATION OF RELATION IDENTIFICATION.

domain	# R	# RC	precision	recall
microbial risk	16	123	80%	97%
chemical risk	4	34	93%	79%
aeronautics	26	113	98%	88%

The content score of a relation r for the data table tab is the proportion of simple concepts in the signature of r which were represented by columns of tab. Let $Sign(r)$ be the set of simple concepts in the signature of r and $Sign(tab)$ the set of simple concepts represented by columns of tab, then:

$$score_{content}(r, tab) = \frac{\mid Sign(r) \cap Sign(tab) \mid}{\mid Sign(r) \mid} \quad (10)$$

If the output concept of the relation r was not represented by a column of the data table tab, then: $score_{final}(r, tab) = 0$. When the final scores of all relations of the OTR have been computed for the data table, we identify which relation(s) is(are) represented in the data table. A data table can represent several relations at a time: for example, if a data table gives the pH and the water activity of a food product, two separate relations are considered: *Food property: pH* and *Food property: water activity*. Two relations are called *concurrent relations* if they have the same output concept. If a relation has a non-zero final score for the data table and has no concurrent relation, this relation is considered as represented in the data table. If there are several concurrent relations with non-zero final scores for the data table, then we only keep the one with the highest final score. If several concurrent relations have the same highest final score, we keep them all.

Experimental results: Our annotation method was experimented using the three OTR presented in Table I and their associated corpora presented in Table II. Data tables were manually annotated with the relations of an OTR. In Table V, R is the set of relations in an OTR and RC is the set of relations manually annotated in the considered OTR; precision and recall are given for our method. Our method to identify relations depends on the identification of the symbolic concepts and quantities, which can be considered as a weakness. For this reason, our experimentation to automatically annotate the data tables with the relations of the considered OTR was applied without validating the intermediate steps. Consequently, even columns which were wrongly recognized were further annotated and used for the relation identification. The good experimental scores of Table V can mainly be explained by the good experimental scores of the identification of the symbolic concepts and quantities presented above. Moreover, the output concept identification constraint in the relation identification strengthens the precision scores. Let us notice that we are only interested in identifying n-ary relations which were already defined in the OTR, and not in discovering new n-ary relations in data tables, which is conform with our application objective, that is to supplement existing local sources with pertinent external data.

IV. THE INSTANTIATION OF THE RELATIONS IN ORDER TO ANNOTATE A DATA TABLE

In order to index a data table, we annotate each of its rows with an instance of each relation represented in the data table. An instance of a relation relies on instances of target symbolic concepts and quantities of its signature, represented by the columns of the data table. Instantiations generated for each row depend upon the data present in each cell of the row. These instantiations are fuzzy and allow one to take into account (i) the imprecision of the initial data found in the table (for instance an interval for a quantity), (ii) similarities between terms present in the data table and terms denoting the symbolic concepts of the OTR, and (iii) the certainty of the identification of a relation represented in the data table which relies on the quality of the columns' annotation. We first present briefly the theory of fuzzy sets used in the relations instantiation, then we detail the method for symbolic concept, quantity and relation.

A. The fuzzy sets

We use the definition of fuzzy sets given in [15] and [16]. 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, elements can 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). The membership degree of an element $x \in X$ for the fuzzy set A is denoted $\mu_A(x)$. The support of a fuzzy set A defined on a definition domain X is the set (in the classic definition) of elements $x \in X$ such that $\mu_A(x) > 0$. The kernel of a fuzzy set A defined on a definition domain X is the set (in the classic definition) of elements $x \in X$ such that $\mu_A(x) = 1$. A fuzzy set defined on a continuous definition domain is called *continuous fuzzy set* and on a discrete definition domain, *discrete fuzzy set*. A trapezoid fuzzy set TFS is a particular continuous fuzzy set which is described only by its support $sup = [min_{sup}, max_{sup}]$ and its kernel $ker = [min_{ker}, max_{ker}]$. The membership degree of an element x in this definition domain is then defined by:

- if $x \leq min_{sup}$ or $x \geq max_{sup}$ then $\mu_{TFS}(x) = 0$;
- if $min_{ker} \leq x \leq max_{ker}$ then $\mu_{TFS}(x) = 1$;
- if $min_{sup} \leq x \leq min_{ker}$ then $\mu_{TFS}(x) = \frac{x - min_{sup}}{min_{ker} - min_{sup}}$;
- if $max_{ker} \leq x \leq max_{sup}$ then $\mu_{TFS}(x) = \frac{x - max_{sup}}{max_{ker} - max_{sup}}$.

Several semantics for fuzzy sets are defined in [17]:

- the semantic of certainty or imprecision: there exists a "true" value for an element x of X, but as it is unknown, this "true" value is represented by a fuzzy set defined on X. The higher is the membership degree of a value in X, the more probable this value is close to the "true" value of x. This semantic is used in our annotation subsystem to represent (i) the imprecision of the initial data found in the data tables (in the instantiation of quantities), (ii) the certainty of the identification of a relation represented in the data table;

- the semantic of similarity: a new element is represented by its similarity with known elements in X. The higher is the membership degree of a known element x in X, the more it is similar to the new element. This semantic is used in our annotation subsystem to represent the similarity between a term from the Web and terms denoting symbolic concepts of the OTR (in the instantiation of symbolic concepts);
- the semantic of preferences: elements with the higher membership degrees are the preferred elements. This semantic is used in our querying subsystem to represent the end-users query preferences.

B. The instantiation of symbolic concepts

In order to instantiate, in a row of a data table, a symbolic target concept c which belongs to the signature of a relation represented in this data table, we construct a discrete fuzzy set A_c. The definition domain X of the fuzzy set A_c is a set of symbolic concepts which contains the symbolic target concept c and all of its sub concepts in the OTR. The definition domain X is thus hierarchically organized according to the "is-a" relationship. The membership degree of an element $x \in X$ in the fuzzy set A_c is computed as the maximum of the term similarities between the terms t_x^i denoting the symbolic concept x in the OTR and the term t_{cell} present in the cell which was annotated by the symbolic target concept c:

$$\mu_{A_c}(x) = max_i \ sim(t_x^i, t_{cell}) \quad (11)$$

Example 2: Figure 2 presents an example of a data table in which the relation *GrowthParameterTemperature* has been identified, with the input concept *Microorganism* and the output concept *Temperature*. For the first row of this data table, the input concept, *Microorganism*, is instantiated by a discrete fuzzy set which has a semantic of similarity: it indicates the list of symbolic concepts from the OTR, sub concepts of the symbolic target concept *Microorganism*, which are denoted by terms belonging to the OTR and having a close meaning to the data table term "B. cereus". Four terms were founded in the OTR: the terms "B. Cereus" and "Bacillus Cereus" which denote the symbolic concept *Bacillus_Cereus* and the terms "B. Cereus Spores" and "Bacillus Cereus Spores" which denote the symbolic concept *Bacillus_Cereus_Spores*. The symbolic concept *Bacillus_Cereus* (resp. *Bacillus_Cereus_Spores*) has a maximum term similarity of 1.0 (resp. 0.81) with the data table term "B. cereus". The input concept is therefore instantiated, in the first row, by the discrete fuzzy set $\{1.0/$*Bacillus_Cereus*, $0.81/$*Bacillus_Cereus_Spores*$\}$.

Experimental results: Our instantiation method was experimented using the microbial risk OTR presented in Table I and its associated corpus presented in Table II. In this corpus, 185 instances of food products, which are terms found in cells annotated by the symbolic target concept *Food Product*, were found. For each instance t of food product, we manually defined its "*best match*" in the OTR, i.e. the symbolic concept, sub concept of *Food Product*, which is denoted, in the OTR, by the most similar term to the term t. The evaluation is done by looking at the position of the "best match" obtained with our instantiation method, by order of descending membership degree. The position is evaluated at worse, i.e. if several symbolic concepts of the OTR have the same membership degree in the fuzzy set used for the instantiation, then the "best match" is always considered as being at the last position. This evaluation at worse comes from the need to manually validate the annotations: if the 5 symbolic concepts having the best membership degree are presented to the end-user such that he/she has to choose the best, we want to be sure that the "best match" will be among those 5. On the 185 terms found in data tables cells, 78% had a not null "best match", 46% had their "best match" in first position, while 66% had their "best match" among the five best positions. This validates the approach of keeping a fuzzy set for instantiating a symbolic target concept, instead of only keeping the symbolic concept which has the best membership degree defined in Equation 11.

C. The instantiation of quantities

In order to instantiate, in a row of a data table, a target quantity c which belongs to the signature of a relation represented in this data table, there are three possibilities:

a) There is one column in the data table (thus one cell in the row to annotate) which was annotated by the target quantity c. Numerical values in the cell are then used to instantiate the target quantity: it can be an isolated value, an enumeration of isolated values, an interval or a mean with a standard error. Intervals and means with standard errors are recognized using specific patterns; if those patterns are not recognized, then all numerical values in the cell are considered as isolated.
b) There are several columns in the data table which were annotated by the target quantity c. Relationships between these columns are then searched, looking for keywords in the columns' titles. A column can represent a minimum value, a maximum value or an optimum value (between the minimum and maximum values); it can also represent a mean value or a standard error.
c) There is no column in the data table which was annotated by the target quantity c. If the target quantity c can be expressed using unit concepts (different from the *Dimension_One* concept in the OTR), we search for occurrences of a numerical value followed by one of these unit concepts (using terms denoting these unit concepts in the OTR), in the data table's title or in the columns' titles: those occurrences are then considered as isolated values.

An instance of a target quantity is represented by a fuzzy set, for which the definition domain is the numerical range defined in the OTR for this target quantity. This fuzzy set is defined as the union of trapezoid fuzzy sets, expressed in the same unit concept, each of them being constructed as follows:

- when recognizing an isolated value x in the data table, a trapezoid fuzzy set is built with $sup = ker = [x, x]$;
- when recognizing an interval $[a, b]$ in the data table, either in one cell, or when a is the value in a cell of a column recognized as minimum and b is the value in a cell of a column recognized as maximum with no cell of a column

recognized as optimum, a trapezoid fuzzy set is built with $sup = ker = [a, b]$;
- when having a cell of a column recognized as minimum with the value min, a cell of a column recognized as maximum with the value max and a cell of a column recognized as optimum with its values included in $[a, b]$ where $min \leq a \leq b \leq max$, a trapezoid fuzzy set is built with $sup = [min, max]$ and $ker = [a, b]$;
- when having a cell of a column recognized as mean with the value m and a cell of a column recognized as standard error with the value e, a trapezoid fuzzy set is built with $sup = [m - e, m + e]$ and $ker = [m, m]$.

Example 3: For the first row of the data table of Figure 2, the output concept is instantiated by a continuous fuzzy set which has a trapezoidal form and a semantic of imprecision. It is expressed in the *Celsius_Degree* unit concept and indicates the possible growth limits ($[3.9, 49.8]$) and the possible optimal growth limits ($[39.9, 39.9]$) which are respectively represented as the support and the kernel of the continuous fuzzy set.

Experimental results: Our instantiation method was experimented using the microbial risk OTR presented in Table I and its associated corpora presented in Table II. In the microbial risk corpus, 119 relations were correctly recognized. The instantiation of target quantities was analyzed for the first row of each data table. We assume that the structure is enough homogeneous inside a data table, so that the instantiation of its first row can be considered as representative of what happens in the whole data table. On the 119 relations, there were 2 errors on the instantiation of target quantities (an error of concept recognition, an error of numerical value recognition). For 5 tables (corresponding to 13 relations), the target quantity *Temperature* was not instantiated because its value was not present in the data table but in its textual environment in the original publication. There were also 3 errors in interval reconstruction (values were considered as isolated while they represented an interval) and one error in the construction of a minimum/optimum/maximum trapezoid fuzzy set (values were considered as isolated). For all 100 remaining relations, all target quantities were correctly instantiated.

D. The instantiation of relations

Once all simple concepts of the signature of a relation, which are represented by the columns of a data table, have been instantiated for a row of the data table, this row can be annotated with an instance of the relation. This instance of a relation has a *certainty score*: the final score which was computed during the relation recognition phase (see Subsection III-D), and is related to the instances of target quantities and symbolic target concepts of the relation's signature.

Example 4: Figure 10 presents a part of the fuzzy RDF graph corresponding to the instantiation of the relation *GrowthParameterTermperature* in the first row of the data table of Figure 2. Let us notice that this instantiation is represented using instances of concepts of the OWL OTR presented in Figures 3 and 5. The first description of Figure 10 expresses that the first row (having the URI *uriRow1* in the XML document representing the data table) is annotated by a

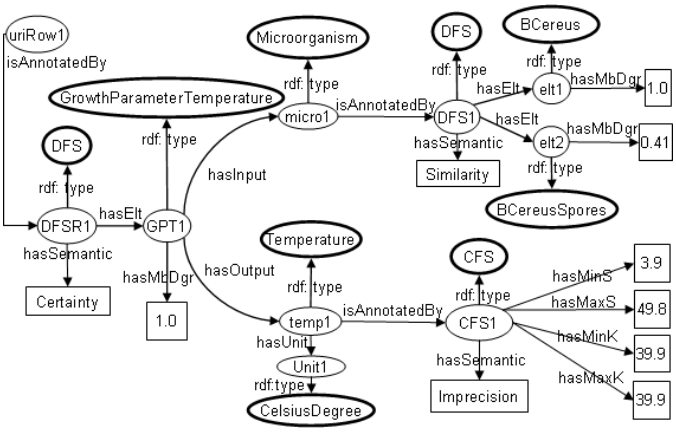

Fig. 10. Example of RDF descriptions generated by our annotation method from the data table of Figure 2 (resources corresponding to OWL classes are represented by ellipses framed in bold).

discrete fuzzy set. This fuzzy set, typed by the OWL class *DFS* (for Discrete Fuzzy Set), has a semantic of certainty and indicates the list of relations of the OTR which are the more certainty represented in the data table. Only the relation *GrowthParameterTemperature* belongs to this fuzzy set with the certainty score of 1.0. This score expresses the degree of certainty associated with the relation recognition by the semantic annotation method. The input concept of the relation, which is an instance of the symbolic target concept *Microorganism*, is annotated by the discrete fuzzy set $\{1.0/$*Bacillus_Cereus*, $0.81/$*Bacillus_Cereus_Spores*$\}$, typed by the OWL class *DFS*. The output concept of the relation, which is an instance of the target quantity *Temperature*, is annotated by a continuous trapezoid fuzzy set with $sup = [3.9, 49.8]$ and $ker = [39.9, 39.9]$, typed by the OWL class *CFS* (for Continuous Fuzzy Set).

V. The Fuzzy Querying Method

We present in this section the querying subsystem, called MIEL++, of ONDINE system. 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 (see [2] for more details about the SQL subsystem wrapper) and a SPARQL query in the XML/RDF datawarehouse (see [5] 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, we present the extension of MIEL++ subsystem which allows the end-user to query fuzzy RDF annotations of data tables, represented in XML documents, by means of SPARQL queries. We remind the notions of view and MIEL++ query (see [2] for more details). We then present the construction of a MIEL++ answer retrieved from the XML/RDF data warehouse. We conclude this section with experimental results.

A. 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 discrete fuzzy 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) (see Subsection IV-A). When a MIEL++ query is asked by the end-user into the XML/RDF datawarehouse 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. The second point is studied in Section V-B.

Example 5: Let us define a MIEL++ query Q expressed in the view *GrowthParameterTemperature*: Q = {*Microorganism, Temperature* | *GrowthParameterTemperature(Microorganism, Temperature)* $\wedge$ *(Microorganism* $\approx$ *MicroPreferences)* $\wedge$ *(Temperature* $\approx$ *TemperaturePreferences)* $\wedge$ *(thresh* $\geq$ *0.5)*}. The discrete fuzzy set *MicroPreferences*, which is equal to $\{1.0/$*Gram+*, $0.5/$*Gram-*$\}$, means that the end-user is interested in microorganisms which are first *Gram+* and second *Gram-*. The continuous fuzzy set *TemperaturePreferences*, which is equal to $[39.0, 40.0, 41.0, 42.0]$, means that the end-user is first interested in temperature values in the interval $[40.0, 41.0]$ which corresponds to the kernel of the fuzzy set; but he/she accepts to enlarge the querying till the interval $[39.0, 42.0]$ which corresponds to the support of the fuzzy set. Instances of the relation *GrowthParameterTemperature* having a certainty score smaller than 0.5 are discarded.

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, we propose 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 (i) from the signature of the relation represented by the view and associated with the MIEL++ query and (ii) from the sets of projection and selection attributes of the MIEL++ query. [5] provides a complete description of the SPARQL query generation.

B. 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 (see [5] for more details). 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 in [18] to compare a fuzzy set representing preferences to a fuzzy set having a semantic of imprecision: a possibility degree of matching denoted Π and a necessity degree of matching denoted N. On the other hand, we propose to use the adequation degree as defined in [19] to compare a fuzzy set representing preferences to a fuzzy set having a semantic of similarity.

Definition 1: Let $(a \approx v)$ be a selection criterion of the MIEL++ query Q, v' a fuzzy value of the attribute a stored in the XML/RDF data warehouse, $sem_{v'}$ the semantic of v', μ_v and $\mu_{v'}$ being their respective membership degrees defined on the domain Dom. The comparison result depends upon the semantic of the fuzzy set: If $sem_{v'} =$ *imprecision*, the comparison result is given by the *possibility degree of matching* between v and v' denoted $\Pi(v, v') = sup_{x \in Dom}(min(\mu_v(x), \mu_{v'}(x))$ and the *necessity degree of matching* between v and v' denoted $N(v, v') = inf_{x \in Dom}(max(\mu_v(x), 1 - \mu_{v'}(x))$. If $sem_{v'}$ =*similarity*, the comparison result is given by the *adequation degree* between v and v' denoted $ad(v, v') = sup_{x \in Dom}\ (min(\mu_v(x), \mu_{v'}(x))$.

The comparison results between fuzzy sets having the same semantic (similarity or imprecision) are aggregated using the min operator (classically used to interpret the conjunction). An

TABLE VI
RDF DATASET SIZE FOR THE QUERYING EXPERIMENTAL RESULTS.

size / domain	# RDF triples	# RDF graphs
microbial risk	22,000	312
chemical risk	94,580	1,407
aeronautics	18,000	330

answer is a set of tuples composed of the certainty score cs associated with the relation represented in the query, three comparison scores associated with the selection criteria: a global adequation score ad_g associated with the comparison results having a semantic of similarity and two global matching scores Π_g and N_g associated with the comparison results having a semantic of imprecision, and, the values associated with each projection attribute. Based on those scores, we propose to define a total order on the answers which gives greater importance to the most pertinent answers compared with the OTR. Thus, the answers are successively sorted according to cs, ad_g and a total order defined on N_g and Π_g, N_g being considered as of greater importance than Π_g.

Example 6: The answer to MIEL++ query of Example 5 compared with the data table presented in Figure 2 of which the first row is annotated in Figure 10 is given below:
$\{$ $\{cs = 1,\ ad_g = 1.0,\ N_g = 0,\ \Pi_g = 0.9,$ Microorg $= (1.0/$Bacillus Cereus $+0.5/$Bacillus Cereus Spores$)$, Temperature $= [3.9, 39.9, 39.9, 49.8]\}$, $\{cs = 1,\ ad_g = 0.5,\ N_g = 0,\ \Pi_g = 0.9,$ Microorg $= (1.0/$Escherichia Coli$)$, Temperature $= [4.9, 41.1, 41.1, 45.8]\}$ $\}$.

C. Experimental results

Experimental results about the querying of the RDF dataset of the XML/RDF data warehouse in three domains (microbial risk, chemical risk and aeronautics) are presented in Tables VII, VIII and IX. The size of the data warehouse in terms of RDF triples and RDF graphs (a RDF graph is an instance of a n-ary relation) are given in Table VI. In preliminary tests performed on a RDF dataset associated with the microbial risk application, five queries were evaluated: in Table VII p is the precision, r is the recall and G is the set of answer graphs. Better results were obtained in the queries where the selection criteria concerns microorganisms than in the ones concerning food products. This is due to the fact that microorganism names are more standardized in data tables than food product names. Therefore, the quality of the fuzzy annotations associated with the symbolic concept *Microorganism* is better than the ones associated with the symbolic concept *Food_product*. Nevertheless, a precision of 100% was obtained for the two last queries concerning food product if a threshold of 0.7 was added for the adequation degrees (ad_g) in the query.

The second application on which experimental results were obtained concerns chemical risk in food. Five queries (see Table VIII) were evaluated on the associated RDF dataset. The results are similar to the ones obtained for the microbial risk application: they are better when the selection criteria concerns contaminant names which are more standardized than food products names. Nevertheless, as for the microbial risk application, a precision of 100% was obtained for the last query concerning *Red wine* $10°$ if a threshold of 0.5 was added for the adequation degree (ad_g) in the query, discarding other types of wines (white, rose, ...).

Our querying subsystem was also experimented on an aeronautic application. Four queries (see Table IX) were evaluated on the associated RDF dataset. The recall of 64% for the third query is due to the fact that six answer graphs were annotated,

TABLE VII
EVALUATION OF QUERY RESULTS FOR THE MICROBIAL RISK APPLICATION.

Queried relation	Selection criteria	p	r	# G
Lag Time	Microorganism = L. Monocytogenes	100%	100%	47
Lag Time	Microorganism = P. Fluorescens	100%	100%	29
Growth Kinetics	Microorganism = E. Coli	100%	100%	39
Lag Time	FoodProduct = Egg salad	50%	100%	24
Growth Kinetics	FoodProduct = Salad	54%	100%	26

TABLE VIII
EVALUATION OF QUERY RESULTS FOR THE CHEMICAL RISK APPLICATION.

Queried relation	Selection criteria	p	r	# G
Contamination Level	Food = Breakfast cereal *and* Contaminant = Ochratoxin A	100%	100%	83
Limit of Quantification	Food = Wheat *and* Contaminant = Ochratoxin A	78%	100%	33
Sample Positive	Food = {1.0/Wheat + 0.9/Breakfast cereal}	93%	100%	79
Mean Contamination	Contaminant = Ochratoxin A	100%	100%	394
Mean Contamination	Food = Red wine $10°$	31%	100%	35

TABLE IX
EVALUATION OF QUERY RESULTS FOR THE AERONAUTICS APPLICATION.

Queried relation	Selection criteria	p %	r %	#G
Aircraft Width	Width $\approx [35, 40, 50, 65]$m	100	100	12
Aircraft Width	AircraftName $\approx$ {1.0/A350 + 0.9/A340 }	100	100	10
Aircraft Cruise Speed	Width $\approx [0.7, 0.79, 0.83, 0.9]$ mach	100	64	11
Deliveries Number	NbDeliveries $\approx [10, 20, 30, 60]$	100	100	48

by our annotation method, with the unit *km/h* instead of *mach* which was specified in the query.

VI. Comparison with the State of the Art

Recent propositions in the Semantic Web community propose to extract, filter, annotate and query Web data tables (see [20]–[22], [23]), but they have not been designed with the same objectives as ours. TableSeer (see [20]) for instance allows a set of predefined metadata (caption, cell content, geographical position of the data table in the HTML page, ...) to be extracted from Web data tables, but it does not compare the schema of the Web data tables with preexisting schemas defined in an ontology. We can also cite WebTables (see [21], [22]) which proposes a system to identify relational tables in a huge amount of tables included in HTML documents and to index them, this in order to query and rank them. Nevertheless, the WebTables querying language is only composed of a set of key-words which are compared with the attribute names of the Web data tables. The row content of the Web data tables is not used in the querying process which is only based on global co-occurrences frequencies statistics of attribute names. In [24], relations from an ontology are instantiated using various HTML structures including tables. However, they only identify binary concept-role relations between instances which are assumed to be already annotated (manually or using another information extraction system). Our work differs as we focus on the recognition of n-ary relations and we propose a step-by-step algorithm including the recognition of concepts. From this point of view, the work presented in [25] is closer to ours, as they transform data tables of different structures into a common relational database schema with n-ary relations. However, our approach extends [25] in several ways: a better distinction in the OTR between the concepts and the terminology, annotation of cells with either the most similar terms of the OTR or imprecise values; flexible querying of n-ary relations handling those fuzzy annotations. The work of [23] focuses on the recognition of quantities in Web data tables guided by an ontology of measure units. Heuristic rules permit to define disambiguation strategies when the same term refers to different quantities. This case is unusual in our context where the OTR is dedicated to a given application domain, in which only a subset of the ontology of measure units defined in [23] is involved. But, this work is complementary to our's and could be integrated as an extension of our method for the recognition of quantities.

Our proposal in this paper can also be compared with papers studying flexible querying extending XPATH or SPARQL. Different approaches have been proposed. [26] defines FUZZYXPATH, a fuzzy extension of XPATH to query XML documents. [27] proposes an extension of the SPARQL ?Optional? clause (called Relax). This clause allows the computation of a set of generalizations of the RDF triplets involved in the SPARQL query using especially declarations done in the RDF Schema. [28] also proposes the same kind of extension of the SPARQL query using a distance function applied to the classes and properties of the RDF Schema. The originality of our approach in flexible SPARQL querying is that we propose a complete and integrated solution which allows one (1) to annotate Web data tables with the vocabulary defined in an OTR, (2) to perform a flexible querying of the annotated tables using the same vocabulary and taking into account the fuzzy degrees generated by the annotation method according to their associated semantic. Our work did not use the fuzzy extension of SPARQL based on a fuzzy extension of DL-Lite proposed by [29] for two main reasons: (i) our OTR requires a higher level of expressiveness (OWL2-DL) which is useful for consistency checking (for example, in order to express that the class Quantity is distinct from the class Symbolic_Concept); (ii) the SPARQL extension does not yet allow the distinction between fuzzy sets having a semantic of similarity and imprecision.

VII. Conclusion

We have presented in this paper a complete system, 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. ONDINE system has been implemented through the development of @Web software on the one hand and the development of MIEL++ software on the other hand. Moreover, ONDINE system has been implemented in the Sym'Previus predictive microbiology modeling system which allows the behavior of a microorganism in a given food matrix to be predicted (see [30] for more details). To the best of our knowledge, ONDINE is the only software which allows one to simultaneously (1) annotate accurately a data table with an OTR and (2) perform approximate reasoning during the flexible querying process, comparing preferences expressed by the end-user with fuzzy annotations. ONDINE has been successfully tested on three different applications (microbial risk in food, chemical risk in food and aeronautics) which illustrate the generic potential of the proposal. In the very next future, we want to explore four new ideas to extend our approach. The first one consists in associating the data tables, which have been extracted from Web documents, with a reliability degree which takes into account several criteria to qualify the trust in the data source as for example the type or the reputation of the data source. 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 and (3) managing OTR evolution by taking into account annotation results and other ontologies. For example, we would like to integrate the unit conversion rules defined in OM [31] to manage standardization of measure units associated with a

given quantity. Those perspectives will allow us to test the genericity of our OTR, which we pretend to be dedicated to the data integration task.

REFERENCES

[1] P. Buche and O. Haemmerlé, "Towards a unified querying system of both structured and semi-structured imprecise data using fuzzy views," in *ICCS*, ser. LNAI, vol. 1867, 2000, pp. 207–220.

[2] P. Buche, C. Dervin, O. Haemmerlé, and R. Thomopoulos, "Fuzzy querying of incomplete, imprecise, and heterogeneously structured data in the relational model using ontologies and rules." *IEEE T. Fuzzy Systems*, vol. 13, no. 3, pp. 373–383, 2005.

[3] G. Hignette, P. Buche, J. Dibie-Barthélemy, and O. Haemmerlé, "An ontology-driven annotation of data tables," in *WISE Workshops. Web Data Integration and Management for Life Sciences.*, ser. LNCS, vol. 4832, 2007, pp. 29–40.

[4] G. Hignette, P. Buche, J. Dibie-Barthélemy, and O. Haemmerlé, "Fuzzy annotation of web data tables driven by a domain ontology," in *ESWC*, ser. Lecture Notes in Computer Science, vol. 5554, 2009, pp. 638–653.

[5] P. Buche, J. Dibie-Barthélemy, and H. Chebil, "Flexible sparql querying of web data tables driven by an ontology," in *FQAS*, ser. Lecture Notes in Computer Science, vol. 5822, 2009, pp. 345–357.

[6] P. Cimiano, P. Buitelaar, J. McCrae, and M. Sintek, "Lexinfo: A declarative model for the lexicon-ontology interface," *J. Web Sem.*, vol. 9, no. 1, pp. 29–51, 2011.

[7] J. McCrae, D. Spohr, and P. Cimiano, "Linking lexical resources and ontologies on the semantic web with lemon," in *ESWC (1)*, ser. Lecture Notes in Computer Science, vol. 6643. Springer, 2011, pp. 245–259.

[8] T. Declerck and P. Lendvai, "Towards a standardized linguistic annotation of the textual content of labels in knowledge representation systems," in *LREC*, 2010.

[9] A. Reymonet, J. Thomas, and N. Aussenac-Gilles, "Modelling ontological and terminological resources in OWL DL," in *OntoLex 2007 - Workshop at ISWC07*, 2007.

[10] C. Roche, M. Calberg-Challot, L. Damas, and P. Rouard, "Ontoterminology - a new paradigm for terminology," in *KEOD*, 2009, pp. 321–326.

[11] A. Reymonet, J. Thomas, and N. Aussenac-Gilles, "Ontology based information retrieval: an application to automotive diagnosis," in *International Workshop on Principles of Diagnosis*, 2009, pp. 9–14.

[12] R. Yangarber, W. Lin, and R. Grishman, "Unsupervised learning of generalized names," in *International Conference on Computational Linguistics*, 2002, pp. 1–7.

[13] C. J. van Rijsbergen, *Information Retrieval*. Butterworth, 1979.

[14] J. C. Platt, *Fast training of support vector machines using sequential minimal optimization*. MIT Press, 1999, pp. 185–208.

[15] L. Zadeh, "Fuzzy sets," *Information and control*, vol. 8, pp. 338–353, 1965.

[16] ——, "Fuzzy sets as a basis for a theory of possibility," *Fuzzy Sets and Systems*, vol. 1, pp. 3–28, 1978.

[17] D. Dubois and H. Prade, "The three semantics of fuzzy sets," *Fuzzy Sets and Systems*, vol. 90, pp. 141–150, 1997.

[18] ——, *Possibility theory - An approach to computerized processing of uncertainty*. Plenum Press, New York, 1988.

[19] M. Baziz, M. Boughanem, H. Prade, and G. Pasi, "A fuzzy logic approach to information retrieval using a ontology-based representation of documents," in *Fuzzy Logic and the Semantic Web*, ser. Capturing Intelligence. Elsevier, 2006, vol. 1, pp. 363–377.

[20] Y. Liu, K. Bai, P. Mitra, and C. L. Giles, "Tableseer: automatic table metadata extraction and searching in digital libraries," in *JCDL*. ACM, 2007, pp. 91–100.

[21] M. J. Cafarella, A. Y. Halevy, Y. Zhang, D. Z. Wang, and E. Wu, "Uncovering the relational web," in *WebDB*, 2008.

[22] M. J. Cafarella, A. Y. Halevy, D. Z. Wang, E. Wu, and Y. Zhang, "Webtables: exploring the power of tables on the web," *PVLDB*, vol. 1, no. 1, pp. 538–549, 2008.

[23] M. van Assem, H. Rijgersberg, M. Wigham, and J. Top, "Converting and annotating quantitative data tables," in *Proceedings of the 9th international semantic web conference - Volume Part I*, 2010, pp. 16–31.

[24] S. Tenier, Y. Toussaint, A. Napoli, and X. Polanco, "Instantiation of relations for semantic annotation," in *Int. Conf. on Web Intelligence*, 2006, pp. 463–472.

[25] D. W. Embley, C. Tao, and S. W. Liddle, "Automatically extracting ontologically specified data from HTML tables of unknown structure," in *Conceptual Modeling - ER 2002*, 2002, pp. 322–337.

[26] A. Campi, E. Damiani, S. Guinea, S. Marrara, G. Pasi, and P. Spoletini, "A fuzzy extension for the xpath query language," in *FQAS*, ser. LNCS 4027, 2006, pp. 210–221.

[27] C. A. Hutardo, A. Poulovassilis, and P. T. Wood, "A relaxed approach to rdf querying," in *ISWC*, ser. LNCS, vol. 4273, 2006, pp. 314–328.

[28] O. Corby, R. Dieng-Kuntz, C. Faron-Zucker, and F. Gandon, "Searching the semantic web: Approximate query processing based on ontologies," *IEEE Intelligent Systems Journal*, vol. 21, no. 1, pp. 20–27, 2006.

[29] J. Z. Pan, G. B. Stamou, G. Stoilos, S. Taylor, and E. Thomas, "Scalable querying services over fuzzy ontologies," in *WWW 2008*, 2008, pp. 575–584.

[30] P. Buche, O. Couvert, J. Dibie-Barthélemy, G. Hignette, E. Mettler, and L. Soler, "Flexible querying of web data to simulate bacterial growth in food," *Food Microbiology*, vol. 28, no. 4, pp. 685–693, 2011.

[31] H. Rijgersberg, M. Wigham, and J. L. Top, "How semantics can improve engineering processes: a case of units measure and quantities," *Advanced Engineering Informatics*, vol. 25, no. 2, pp. 276 – 287, 2011.

Patrice Buche Patrice Buche received the PhD degree in computer science from the University of Rennes, France, in 1990. He is research engineer with INRA, Agricultural Research Institute. His research works mainly concern data integration from heterogeneous sources and fuzzy querying in structured and weakly structured databases.

Juliette Dibie Juliette Dibie-Barthélemy received her PhD degree in Computer Science from the University Paris Dauphine, France, in 2000. Since 2000 she is Assistant Professor in Computer Science at AgroParisTech, Paris, France. Her research activities mainly concern knowledge representation, knowledge validation and data integration of heterogeneous data guided by an ontology.

Liliana Ibanescu Liliana Ibanescu received her PhD degree in Computer Science from *Institut National Polytechnique de Lorraine*, Nancy, France, in 2004. Since 2008 she is Assistant Professor in Computer Science at AgroParisTech, Paris, France. Her current research activities mainly concern knowledge representation and data integration of heterogeneous data.

Lydie Soler Lydie Soler received the Professional Master's degree in Computer Sciences from the University of Versailles (France) in 2003. She has been an engineer at the Applied Mathematical and Computer Science department of INRA since 2004. She is a Java analyst and software developer.