Ontology-Driven Possibilistic Reference Fusion

Fatiha Saïs[1], Rallou Thomopoulos[2,3], and Sébastien Destercke[3]

[1] LRI and INRIA Saclay-Île-de-France, 2-4 rue J. Monod, F-91893 Orsay, France
[2] LIRMM (CNRS & Univ. Montpellier II), 161 rue Ada, F-34392 Montpellier cedex 5, France
[3] INRA/CIRAD, UMR1208, 2 place P. Viala, F-34060 Montpellier cedex 1, France
Fatiha.Sais@lri.fr, rallou@supagro.inra.fr,
sebastien.destercke@supagro.inra.fr

Abstract. It often happens that different references (i.e. data descriptions), possibly coming from different heterogeneous data sources, concern the same real world entity. In such cases, it is necessary: (i) to detect, through reconciliation methods, whether different data descriptions refer to the same real world entity and (ii) to fuse them into a unique representation. Here we assume the reference reconciliation is solved, and we propose a fusion method based on possibility theory, able to cope with uncertainty and with ontological knowledge. An implementation using W3C standards is provided. Rising from the fusion process, an ontology enrichment procedure is proposed to complete the global ontology.
Key words: Data integration, Data fusion, Ontologies, Hierarchical Fuzzy Set.

1 Introduction

In a context of increasing available information, modern integration systems must be able to deal with heterogeneous information sources and more specifically to provide consistency checking mechanisms. Key issues to obtain this consistency, by providing an integrated representation of data, concern the problem of schema/data reconciliation and fusion. Schema heterogeneity is a major cause of the mismatch of data descriptions between sources. Extensive research work has been done recently (see [1] for surveys) to reconcile schemas or ontologies through mappings.

However, the homogeneity or reconciliation of the schemas does not prevent variations between the data descriptions. Data reconciliation consists in deciding whether different data descriptions, here called references, concern the same real-world entity (e.g. the same person, the same experiment, the same paper). In this paper, reconciliation is assumed to be solved (see [2] for more details). Data fusion then consists in merging the reconciled references into a single one. This is the problem considered in this paper. Performing the fusion step offers several advantages: (i) it provides the user with more consistent and detailed answers, since they gather information from multiple references; (ii) it reduces the number of returned answers and consequently makes query evaluation faster and (iii) it makes query results more user-friendly, as the result returns only one reference for each group of redundant references.

The fusion procedure should be as automated as possible, as it is likely to deal with large amounts of data. The final merged references should also take account of the uncertainties arising from the data heterogeneity and from automatic reconciliation:

variability of attribute values (e.g. the same molecule can be named *"Vitamin B2"* in one reference and *"Riboflavin"* in another), lack of data, incorrect entries, *etc.* Most of reference reconciliation systems used in data cleaning (e.g. ETL systems in data warehouses) settle for detecting the reconciliation decisions and delegate the fusion task to the user. The few fusion procedures proposed up to now do not satisfy the previous requirements, as they need human intervention to be performed and only provide one value per attribute of the fused reference [3]. Besides, these existing methods do not take account of additional domain knowledge, coming for instance from an ontology.

In this paper, we propose a fusion method that satisfies the above issues. More precisely, we have chosen to preserve as much as possible the original values of the data descriptions. Owing to the potential uncertainty in the reconciliation decisions, the certainty in the choice of relevant values cannot be guaranteed. Therefore, all the values appearing in the data descriptions are kept and are given different confidence degrees modelling the final uncertainty.

We propose to use possibility theory to model the uncertainty, as this theory allows to model explicitly the imprecision in the information and to easily take account of source reliability. It is also computationally convenient and offers a simple interpretation. We also consider that some generic information about the references is available in the form of an ontology (formalised in OWL language). After introducing notations and basics about possibility theory and ontology in Section 2, we develop the proposed approach to build fused references in Section 3. In particular, we present how various features of the data are taken into account in the fused uncertainty models, through the use of criteria and of the ontology. As we are working under an open-world assumption (i.e. not all existing values are in the ontology), we also develop a simple enrichment approach allowing to integrate new values into the ontology. Once this fusion is done, classical or fuzzy query methods [4] can be applied to them. An implementation of the proposed approach in W3C standardised languages (RDF and SPARQL) is proposed in Section 5, allowing for the proposed approach to be implemented without having to build extensions of classical languages. The method is then experimented (see Section 6) on a real dataset of the scientific publication domain on which it has obtained promising results. Finally, we discuss the interest of our approach with respect to some existing related works in Section 7.

2 Materials

In this section, we present the basic notions of possibility theory [5] and ontology [6] needed in this paper, as well as the used notations.

2.1 Possibility theory

When the value assumed by a variable X over a (finite) domain $\mathcal{X}$ is uncertain, possibility theory can be used to model this uncertainty. In particular, it is able to model imprecision and incompleteness in the available information, a feature that classical probability is arguably unable to account for (see Walley [7] for a full discussion). The main tool of possibility theory are possibility distributions, defined as follows:

Definition 1 (Possibility distribution). *A possibility distribution π on a domain $\mathcal{X}$, is a mapping $\pi : \mathcal{X} \rightarrow [0, 1]$ from $\mathcal{X}$ to the unit interval such that there is at least one element $x \in \mathcal{X}$ for which $\pi(x) = 1$.*

From a possibility distribution π, two set-functions are then defined, namely the possibility and necessity measures, such that, for any subset $A \subseteq \mathcal{X}$:

$$\Pi(A) = \sup_{x \in A} \pi(x) \qquad \text{(Possibility measure)} \qquad (1)$$

$$N(A) = 1 - \Pi(A^c) = \inf_{x \in A^c} (1 - \pi(x)). \qquad \text{(Necessity measure)} \qquad (2)$$

While necessity is a lower confidence measure indicating how much A is certain, possibility measure indicates how much A is plausible. Both measures quantify our uncertainty about the true value of variabe X. A possibility distribution π is formally equivalent [8] to a fuzzy set $\mu : \mathcal{X} \rightarrow [0, 1]$ such that $\pi = \mu$.

2.2 Ontology

In this paper, we consider that we have a unique global ontology. We define an ontology by using a fragment of OWL DL, which is the description logic fragment of the Ontology Web Language recommended by the W3C.

We consider an ontology $\mathcal{O}$ as a tuple $(\mathcal{C}, \mathcal{P}, \mathcal{I}, \mathcal{D})$ composed of a set $\mathcal{C}$ of classes (unary relations), a set $\mathcal{P}$ of typed properties (binary relations), a set $\mathcal{I}$ of individuals (or concrete values) and a set $\mathcal{D}$ of data types, containing for example $rdfs : Literal$.

We consider that the ontology is also composed of a set of constraints between classes and properties (e.g. subsomption and equivalence relations) summed up in the following table:

Ontology Constraints	DL notation	OWL notation
Subsomption between classes	$C_1 \sqsubseteq C_2$	SubClassOf(C_1 C_2)
Class equivalence	$C_1 \equiv C_2$	EquivalentClasses(C_1 C_2)
Subsomption between properties	$P_1 \sqsubseteq P_2$	SubPropertyOf(P_1 P_2)
Domain typing of a property	$\exists P \sqsubseteq C$	Domain(P C)
Range typing of a property	$\exists P^- \sqsubseteq C$	Range(P C)

In OWL, two kinds of properties can be distinguished: the *abstract properties* which have classes as domain and range, and the *concrete properties* which have a class as domain and a basic data type as range (e.g. Integer, Date, rdfs:Literal).

Given two classes C_1 and C_2, we denote by $lcs(C_1, C_2)$ their least common subsumer, that is $lcs(C_1, C_2) = \{C \in \mathcal{C} | C_i \sqsubseteq C$, and $((\exists C'$ s.t. $C_i \sqsubseteq C') \Rightarrow (C \sqsubseteq C')),\ i \in \{1, 2\}\}$.

As usually done, we consider a *Universal* class subsuming all the other classes of the ontology, to ensure that such a lcs always exists. Note that the notion of lcs can be easily extended to any number of classes. Due to the semantics of subsumption $\sqsubseteq$, if a property has a class C as domain or range, then for any class C' such that $C' \sqsubseteq C$, the same property holds with respectively C' as domain or range.

Table 1. Specification of individuals $\mathcal{I}$ through assertional statements relating data to the domain ontology.

Ontology assertions	DL notation	OWL notation
Class assertion	$C(i)$	Individual($i : C$)
Data type assertion	$D(v)$	Individual($v : D$)
Abstract property assertion	$P(i_1, i_2)$	ObjectProperty(P domain($i_1 : C_1$) range($i_2 : C_2$))
Concrete property assertion	$P'(i, v)$	DataTypeProperty(P' domain($i : C$) range($v : D$))

A small part of the class hierarchy of $\mathcal{C}$ is given as an example in Figure 1. The properties are pictured by dashed Arrows from the domain class to the range class (or data type). The partial order $\sqsubseteq$ is pictured by $\rightarrow$ and describes the subsomption relation. The equivalence relation $\equiv$ is pictured by the relation $equivTo$.

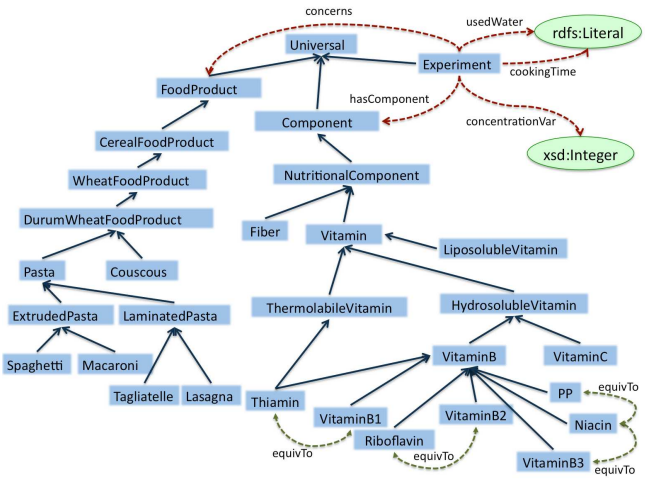

Fig. 1. A part of the domain ontology

In the sequel, we call values the following elements describing an individual i:

- Let *ObjectProperty(P domain($i : C$) range($i' : C'$))* be an abstract property assertion. The value i', referring to a class, is related to i through the property P. In this case, we thus consider hierarchical symbolic values. The hierarchical organization of values is induced by the subsomption relation and the associated ordering.
- Let *DataTypeProperty(P domain($i : C$) range($v : D$))* be a concrete property assertion. v is the value related to i through the property P. Two cases can be distinguished:
  - non-hierarchical symbolic values without referring to a class, e.g. String, Date;
  - numeric values, i.e., the range of P is a closed interval $[\underline{v}, \overline{v}]$.

Remark 1. In OWL, the intervals can be expressed by using XML data type restrictions ($xsd : minInclusive,\ xsd : maxInclusive$) and the OWL constructor $owl : allValuesFrom$ (see [9] for more details).

3 Reference Fusion Approach

After introducing some notations with an illustrative example and recalling how reference reconciliation is achieved, we introduce the fusion method used to merge reconciled references. To do so, we focus on one group of reconciled references. Both used criteria and ontology enrichment procedure (used in case of newly encountered hierarchical values) are detailed.

3.1 Problem statement and illustrative example

We consider N different references $ref_1, \ldots, ref_N$, coming from M different sources $S_1, \ldots, S_M$, with $M \leq N$. Note that all sources and references share the same ontology. These references are individuals of a given class C and have common descriptions represented by a set $\mathbb{P} = \{P_1, \ldots, P_K\}$ of K properties. In Example 1, used thorough the paper to illustrate the fusion approach, show references that are identified as individuals of class $Experiment$, and where the the property set is $\mathbb{P} =$ {cookingTime, usedWater, hasComponent, concentrationVar}.

Example 1. We consider two data sources describing experiments on vitamin rate variation during the cooking of food products. Data are summarized in Figure 2.

Source S1:

Ref.	cookingTime	usedWater	hasComponent	concentrationVar
idE11	12 mins	Distilled water	Thiamin	-53.3
idE12	12 mins	Tap water	Niacin	-45.6

Source S2:

Ref.	cookingTime	usedWater	hasComponent	concentrationVar
idE21	13 mins	Water	VitaminB6	-46
idE22	10 mins	Deionized water	Thiamine	-52.9
idE23	10 mins	Deionized water	VitaminB	-51.8

Fig. 2. *Data to be reconciled concerning the impact of cooking on vitamin level in pasta*

For readability reasons, we have chosen to represent data in a relational form, as it is shown in Figure 2. The OWL DL representation of source 1 data is illustrated in Figure 3 as a set of DL assertional statements on individuals (references and basic values).

We denote by $\mathcal{V}_k$ the set of possible values of the property P_k (numerical values, non-hierarchical values or hierarchical values). To simplify notations, when a property P_k has as range a class c_{P_k} of the ontology $\mathcal{O}$ (i.e. P_k is an abstract property), we denote by $\mathcal{O}_{P_k} = \{\mathcal{C}_{P_k}, \mathcal{P}_{P_k}\}$ the reduced ontology such that $\mathcal{C}_{P_k} = \{c \in \mathcal{C} | c \sqsubseteq c_{P_k}\}$ and the

Experiment(idE11); Thiamin(idVita100) ; cookingTime(idE11, “12 mins”);
usedWater(idE11,“Distilled water”); hasComponent(idE11, idVita100);
concentrationVar(idE11, -53.3);
Experiment(idE12); Niacin(idVita101) ; cookingTime(idE12, “12 mins”);
usedWater(idE11,“Tap water”); hasComponent(idE12, idVita101);
concentrationVar(idE12, -45.6);

Fig. 3. *Data of* Source 1 *in the form of DL assertionnal statements*

properties are limited to $\mathcal{C}_{P_k} \times \mathcal{C}_{P_k}$ (i.e., $\mathcal{P}_{P_k} = \mathcal{P} \cap (\mathcal{C}_{P_k} \times \mathcal{C}_{P_k})$). A given reference ref_n can therefore be described by a set $\{v_{n1}, \ldots, v_{nK}\}$ of K values, where v_{nK} is the value of the property P_k for reference ref_n. Note that missing data (null values) may exist.

Reconciliation problem. The first step of reference reconciliation consists in identifying the pairs of duplicated references (i.e. that represent the same real-world entity) by the use of a dedicated algorithm (e.g. the N2R method [2]). From the set of reconciled pairs, groups of duplicated references are then built by transitive closure. The obtained groups[4] provide a partition of $\{ref_1, \ldots, ref_N\}$. In Example 1 , the pairs $\{idE11, idE22\}$, $\{idE22, idE23\}$ are considered as duplicates. The group built from these pairs is $\{idE11, idE22, idE23\}$. In the sequel, we consider that L groups denoted by $Rec_1, \ldots, Rec_L$ are obtained by reconciliation; and the set of values taken by a property P_k among a group Rec_l will be denoted by V_{lk}, with $k = 1, \ldots, K$ and $l = 1, \ldots, L$.

Fusion problem Once reconciled groups are obtained, references within each group must be merged so that a unique reference is associated to each group (ending up with L references). We propose to base this fusion on possibility theory. The method handles ontological knowledge whenever a property takes as value a concept of the ontology $\mathcal{O}$ (i.e. it is an abstract property). For a given group Rec_l, it consists in two main steps:

- build, for each reference $ref_n \in Rec_l$, a possibility distribution $\pi_{n,k}$ defined on $\mathcal{V}_{lk}$ and describing the uncertainty concerning the real value of property P_k. This steps build a possibility distribution defined over $\mathcal{V}_{lk}$ for each reference, ending up with $|Rec_l|$ distributions for each property;
- the $|Rec_l|$ distributions are then fused in a single one, so that to each property P_k inside a group Rec_l is associated a unique possibility distribution.

To build this model, the method is based on a small set of criteria. These criteria corresponds to information that is always available and that appears sensible to evaluate the relevance of a given value. This allows the method to be general and applicable to the great majority of situations and problems where redundant references can exist. However, in specific situations or problems, there could be additional criteria that should be considered. In such situation, one would have to integrate them in a meaningful way to the uncertainty model. We now details these criteria.

[4] A reference that is not duplicated forms a group by itself.

3.2 Criteria for uncertainty modeling and reference fusion

Several features contribute to the evaluation of the relevance of the property values: variability of encountered values, lack of data, abstract or concrete property, commonness of a given value, uncorrect input, etc. Therefore, several criteria will be used to build the uncertainty model. The first criterion (conceptual similarity) concerns hierarchical symbolic values and thus applies to abstract properties. It is based on the classical Wu & Palmer measure [10] (again, other measures may be more adapted for specific problems). The other criteria concern all kinds of values and were already considered and their use justified in [11], but with an had-oc construction of the uncertainty models.

Consider a given group Rec_l and a fixed property P_k. Let v be the value taken by P_k in the considered reference of the group Rec_l. The criteria are:

- *Conceptual Similarity* ***(CS)***: measures the semantic similarity between two classes. Here, we use the Wu & Palmer measure [10]. Let c_1, c_2 be two classes, N_1, N_2 the path lengths between $lcs(c_1, c_2)$ and respectively c_1 and c_2, and N_3 the path length between $lcs(c_1, c_2)$ and the class Universal. Then, $CS(c_1, c_2)$ reads[5]:

$$CS(c_1, c_2) = \frac{2\ N_3}{N_1 + N_2 + 2\ N_3}.$$

  This criterion will be used to compare the values taken by two abstract properties, whose ranges are classes. Indeed, if one hierarchical value would have to be replaced by another one, the best replacement candidates are the one that are semantically closer to it.
- *Homogeneity* ***(Hom)***: measures the frequency of occurrence of a given value v inside a group of reconciled references $ref_n \in Rec_l$. This criteria is chosen for the reason that the more often a value appears in a group, the more likely it is to be the right one. Homogeneity reads:

$$Hom(v) = |\{v_{nk}=v | ref_n \in Rec_l\}| / |Rec_l|.$$

- *Syntactic similarity* ***(Sim)***: we will denote by $Sim(v, v')$ a syntactic similarity measure between two values v and v' taken by the property P_k in a group of reconciled references. There are many such measures [12], and choosing a particular measure is often dependant of the nature of the data. The argument for retaining this criteria is similar to the one of conceptual similarity (this latter one only applying to hierarchical values).
- *Data source reliability* ***(α_m)***: we consider that a reliability value α_m is associated with each source S_m, $m = 1, \ldots, M$, measuring the confidence we have in the information coming from this source. We consider that information coming from a highly reliable source should have more impact than the one coming from a poorly reliable one, without discarding completely any of these information. This reliability can be, for instance, a function of the last update date of the source [11].

[5] Note that Conceptual Similarity between two equivalent classes is 1.

– *Global frequency* ***(f)***: measures the frequency of a value v among all the references ref_n, $n = 1, \ldots, N$. Indeed, a value appearing numerous times is less likely to contain typographic errors, and is more reliable. It reads:

$$f(v) = {}^{|\{v_{nk}=v | n=1,\ldots,N\}|}\!/_{N}$$

These criteria form a basis from which uncertainty can be estimated. They are significant and general enough so as to be accessible in most situations. Other criteria, more problem specific, can then be added.

3.3 Uncertainty modeling

Three cases can occur: P_k takes hierarchical symbolic values (it is an abstract property), non-hierarchical symbolic values or numerical values (in the last two cases it is a concrete property). We mainly concentrate on the first case, the two other cases being simpler to deal with.

Symbolic hierarchical values (abstract property) We assume that all values in $\mathcal{V}_{lk}$ are present in the ontology as classes, i.e., any $v \in \mathcal{V}_{lk}$ is also in $\mathcal{C}_{P_k}$. A simple method using syntactic similarity (not taken into account here) to integrate newly encountered values is explained afterwards 4. If v is the value given by the reference, we denote by $\mathcal{V}_{lk}^{v} = \{v^{1,v}, \ldots, v^{|\mathcal{V}_{lk}|,v}\}$ the set of ordered values taken by the references of Rec_l, indexed with respect to their conceptual similarity with v, i.e. $i < j \Rightarrow CS(v, v^{i,v}) \leq CS(v, v^{j,v})$ (note that $v^{1,v} = v$). The order relation induced by CS values is a preorder, since multiple values can have the same Wu & Palmer measure with respect to v. For $j = 1, \ldots, |\mathcal{V}_{lk}|$, a first possibility distribution $\pi'_{n,k}$ is built as follows:

$$\pi'_{n,k}(v^{j,v}) = \tag{3}$$

$$\begin{cases} 1 & \text{if } j = 1 \\ \left(1 - \frac{f(v)}{\sum\limits_{v \in \mathcal{V}_{lk}} f(v)}\right)\left(1 - \frac{\sum\limits_{i<j} CS(v,v^{i,v})}{\sum\limits_{j=1}^{|\mathcal{V}_{lk}|} CS(v,v^{j,v})}\right) & \text{if } j>1 \text{ and } CS(v,v^{j,v})<CS(v,v^{j-1,v}) \\ \pi'_{n,k}(v^{j-1,v}) & \text{if } j>1 \text{ and } CS(v,v^{j,v})=CS(v,v^{j-1,v}) \end{cases}$$

In this distribution, the observed value is the most plausible. When the global frequency of this value v is high, other values are made less plausible (their possibility degree being inversely proportional to $f(v)$). In other words, our confidence that v is a reliable value increase with $f(v)$. The plausibility degree of other values than v are also made lower when their conceptual similarities with v are lower (note that equivalencies and equalities of conceptual similarities are treated by the last case). $\pi'_{n,k}$ thus takes account of both conceptual similarity and global frequency.

The reliability α_m of the source S_m from which the reference comes is then used in a classical discounting operation, which consists in transforming, for all $v \in \mathcal{V}_{lk}$, the distribution $\pi'_{n,k}$ into:

$$\pi_{n,k}(v) = \max(1 - \alpha_m, \pi'_{n,k}(v)).$$

This is equivalent to make the information more imprecise when it is less reliable, thus reducing its impact on the final model (original information is kept if $\alpha_m = 1$ and has no impact at all if $\alpha_m = 0$).

Example 2. We consider the subgroup of references $\{idE11, idE22, idE23\}$ from Figure 2, and the property $P_3 = hasComponent$. We also consider that $\alpha_1 = 0.9$ and $\alpha_2 = 0.8$. The set of possible values for this subgroup is $\mathcal{V}_{l3}$ =*{Thiamine, Thiamin, Vitamin B}*. The hierarchical symbolic value "Thiamine" is equivalent to "Thiamin", and thus considered as being in the same equivalence class. The conceptual similarity between "Thiamin" and "Vitamin B" is such that $N_1 = 1$, $N_2 = 0$ and $N_3 = 5$ ("VitaminB" being the Least Common Subsumer), hence *CS(Thiamin,VitaminB)*$= {}^{10}/_{11}$. Finally, we assume that $f(\{Thiamine, Thiamin\}) = 18/123$ and that *f({VitaminB})*$= 2/123$, since data about experiments generally give the precise name of the tested vitamin. For the reference $\{idE23\}$, we have $\mathcal{V}^v_{lk}$ *= {Vitamin B, Thiamine, Thiamin}*, and the distribution $\pi'_{3,3}$ is such that:

$$\pi'_{3,3}(VitaminB) = 1; \quad \pi'_{3,3}(\{Thiamine, Thiamin\}) = (1-{}^{2}/_{20})(1-\frac{1}{{}^{20}/_{11}}) = 0.405$$

and we have $\pi_{3,3} = \pi'_{3,3}$, all values of $\pi'_{3,3}$ being above $1 - \alpha_1 = 0.1$. Note that a missing value would have been modelled by the distribution $\pi(VitaminB) = \pi(\{Thiamine, Thiamin\}) = 1$

Non hierarchical symbolic values In this case, no hierarchical proximity has to be integrated to the uncertainty model, and we consider that $\mathcal{V}^v_{lk} = \{v^{1,v}, \ldots, v^{|\mathcal{V}_{lk}|,v}\}$ is indexed and ordered according to syntactic similarity of values with v, i.e., $i < j \Rightarrow Sim(v, v^{i,v}) \leq Sim(v, v^{j,v})$. The first distribution $\pi'_{n,k}$ is then computed by the same equation as Eq. (3), except that $CS(v, v^{j,v})$ is replaced by $Sim(v, v^{j,v})$, that is, for $j = 1, \ldots, |\mathcal{V}_{lk}|$,

$$\pi'_{n,k}(v^{j,v}) = \tag{4}$$

$$\begin{cases} 1 & \text{if } j = 1 \\ \left(1 - \frac{f(v)}{\sum_{v \in \mathcal{V}_{lk}} f(v)}\right)\left(1 - \frac{\sum_{i<j} Sim(v,v^{i,v})}{\sum_{j=1}^{|\mathcal{V}_{lk}|} Sim(v,v^{j,v})}\right) & \text{if } j>1 \text{ and } Sim(v,v^{j,v})<Sim(v,v^{j-1,v}) \\ \pi'_{n,k}(v^{j-1,v}) & \text{if } j>1 \text{ and } Sim(v,v^{j,v})=Sim(v,v^{j-1,v}) \end{cases}$$

The discounting operation is then applied as in the hierarchical case. Arguments justifying the uncertainty model are similar to those of the hierarchical case.

Numerical values Properties that take numerical values can possibly be subject to small variations between references. They can be, for example, physical measurements coming out from experiments. In general, such numerical values concern physical parameters. In these cases, assume $[v^-, v^+]$ is the interval given by the source (precise values are retrieved when $v^- = v^+$). The possibility distribution $\pi_{n,k}$ modeling the uncertainty for this property and reference is then

$$\pi_{n,p}(v) = \left\{ \begin{array}{cc} 1 & \text{if } v \in [v^-, v^+] \\ 1 - \alpha_m & \text{if } v \in [\underline{v}_k, \overline{v}_k] \setminus [v^-, v^+] \end{array} \right\}. \quad (5)$$

Other numerical values such as postal code, customer number, ID number, *etc.* are treated as symbolic values without hierarchical structure.

Missing data The treatment of missing data in databases is a well-known problem. In the present method, modeling the ignorance about a property value P_k for ref_n can be easily done, using the so-called vacuous (or non-iformative) possibility distribution, that is the distribution $\pi_{n,k}$ such that, for each $v \in \mathcal{V}_{lk}$, $\pi_{n,k}(v) = 1$. This distribution can then be merged with the others, with the effect of increasing the final imprecision. Note that that no additional assumptions has to be made about missing data in this method.

3.4 Fusion method using the uncertainty model

Given a group of reconciled references Rec_l, we denote by ref_{Σ_l} the single fused reference resulting from the fusion process. This fused reference will consist of K possibility distributions $\pi_{\Sigma_l,k}$ defined over spaces $\mathcal{V}_{lk}$, $k = 1, \ldots, K$ and obtained from the distributions described in Section 3.3.

There exists many rules to merge possibility distributions [13]. Here, using a simple arithmetic mean operator is a relevant choice, as it corresponds to a statistical counting and presents a natural way to integrate the homogeneity criterion in the final representation: a value will have all the more weight as it appears more frequently in the group of reconciled references. For a property P_k and a group Rec_l, the final representation $\pi_{\Sigma_l,k}$ is computed, for all $v \in \mathcal{V}_{lk}$, as follows:

$$\pi_{\Sigma_l,k}(v) = \sum_{ref_n \in Rec_l} \frac{1}{|Rec_l|} \pi_{n,k}(v) \quad (6)$$

which is then made consistent by applying the following transformation to all $v \in \mathcal{V}_{lk}$: $\pi_{\Sigma_l,k}(v) = {\pi'_{\Sigma_l,k}(v)}/{\max_{v \in \mathcal{V}_{lk}} \pi'_{\Sigma_l,k}(v)}$. Once this fusion step is achieved, we end up with L final representations, where each property value is described by a possibility distribution reflecting our uncertainty about the real value.

Example 3. Let us pursue example 2 by considering the same subgroup and the same property $A_3 = hasComponent$. As the values $Thiamin, Thiamine$ are considered as equivalent, we have $\pi'_{1,3} = \pi'_{2,3}$ (resp. the distributions induced by references $\{idE11\}$ and $\{idE22\}$). The different distributions are then

$$\pi'_{3,3}(VitaminB) = 1; \quad \pi'_{3,3}(\{Thiamine, Thiamin\}) = 0.405$$
$$\pi'_{1,3}(\{Thiamine, Thiamin\}) = 1; \quad \pi'_{1,3}(VitaminB) = 0.045.$$

However, since sources do not have the same reliability, we have, after the discounting operation, $\pi_{3,3} = \pi'_{3,3}$ and

$$\pi_{1,3}(\{Thiamine, Thiamin\}) = 1; \quad \pi_{1,3}(VitaminB) = 0.1$$
$$\pi_{2,3}(\{Thiamine, Thiamin\}) = 1; \quad \pi_{2,3}(VitaminB) = 0.2.$$

The obtained fused distribution $\pi_{\Sigma_l,3}$ (Using Eq. (6) on $\pi'_{i,3}$, $i = 1, 2, 3$) is

$$\pi_{\Sigma_l,3}(VitaminB) = \frac{0.115}{0.802} = 0.143; \quad \pi_{\Sigma_l,3}(\{Thiamine, Thiamin\}) = \frac{0.802}{0.802} = 1.$$

4 Ontology Enrichment

Up to now, we have assumed that every value of V_{lk} corresponding to an abstract property P_k was in the ontology $\mathcal{O}_{P_k}$. In practice, there are high chances that some references contain values absent from the ontology, since we work under open-world assumption. Therefore, in order to fuse the references of a group of duplicates Rec_l, we must integrate new values to the ontology. This section providesa simple method to enrich the original ontology with new values, before the fusion process. Algorithm 1

Algorithm 1: Ontology enrichment algorithm

Input:
- V_{lk}: the set of values taken by the considered property P_k in Rec_l;
- $\mathcal{O} = \{\mathcal{C}, \mathcal{P}, \mathcal{I}, \mathcal{D}\}$: an initial ontology;
- th: a similarity threshold.

Output: $\mathcal{O}'$: the enriched ontology.

(1) $\mathcal{O}' \leftarrow \mathcal{O}$
(2) $D \leftarrow subClasses(c_{P_k})$
(3) $E \leftarrow D \cap V_{lk}$
(4) **pour tout** $\{v \in V_{lk}\}$
(5) **si** $(v \notin E)$
(6) $bestScore \leftarrow 0$
(7) $bestClass \leftarrow null$
(8) **pour tout** $\{d \in D\}$
(9) $S \leftarrow Sim(v, d)$
(10) **si** $(S > th$ and $S > bestScore)$
(11) $bestScore \leftarrow S$
(12) $bestClass \leftarrow d$
(13) **si** $(bestClass \neq null)$
(14) $\mathcal{O}' \leftarrow addEquivalence(\mathcal{O}', v, bestClass)$
(15) **sinon**
(16) **si** $(E \neq \emptyset)$
(17) $bestClass \leftarrow LCS(E)$
(18) **sinon**
(19) $bestClass \leftarrow c_{P_k}$
(20) $\mathcal{O}' \leftarrow addSubClass(\mathcal{O}', v, bestClass)$
(21)

describes the method for a set of values V_{lk}. It consists in considering that a new value is either equivalent to another one in $\mathcal{C}_{P_k}$ if it is syntactically close enough to it, or is subsumed by the least common subsumer of the values of V_{lk} in $\mathcal{C}_{P_k}$. It contains the following functions:

– $subClasses(c) = \{c' \in \mathcal{C}_{P_k} | c' \sqsubseteq c\}$ computes the set of descendants of a class c including c itself;
– $addEquivalence(O, c_1, c_2)$ adds the class c_1 in the ontology O as equivalent to the class c_2 (i.e. $c_1 \equiv c_2$);
– $addSubClass(O, c_1, c_2)$ adds the class c_1 in the ontology O as a sub-class of the class c_2 (i.e. $c_1 \sqsubseteq c_2$ and $\nexists c_3$ s.t. $c_1 \sqsubseteq c_3 \sqsubseteq c_2$).

Example 4. Consider the subgroup made of references $\{idE11, idE22, idE23\}$ from Figure 2, and the property $P_3 = hasComponent$. We have $\mathcal{V}_{l3}$ =*{Thiamine, Thiamin, Vitamin B}* and $E = \mathcal{V}_{l3} \cap subClasses(A_3) = \{Thiamin, VitaminB\}$. Referring to Fig. 1, we have $LCS(E) = \{VitaminB\}$. However, assuming that *Sim(Thiamine, Thiamin) = 0.95* and that $th = 0.9$, applying Algorithm 1 leads to declare $Thiamine \equiv Thiamin$, and to consider them as equivalent values. The ontology is thus enriched by adding the class $Thiamine$ as equivalent to the existing class $Thiamin$.

5 Implementation of the Approach

In this section we show how we implement our approach for reference fusion and querying. To implement the proposed approach, we apply a maping beween the structural specification of OWL language and the sepecification of RDF language (see [14]). We have chosen to use the semantic web languages RDF and SPARQL for respectively describing fused references and querying them.

We first give a representation of the fused references by using an extension of RDF to a fuzzy-RDF language. Then we propose a transformation of the obtained fuzzy-RDF data into plain RDF data. In section 5.3, we present the flexible querying of fused references by using SPARQL.

5.1 Fuzzy-RDF representation of fused data

To represent uncertain data, Mazzieri [15] proposes an extension of RDF language into a fuzzy-RDF language, providing its syntax and semantics. The syntax extension consists in expressing RDF declarations in the form of triples <*subject, predicate, object*> by using declarations of the form α : <*subject predicate object*>. For each triple, a degree α in $[0, 1]$ is added, representing the truth value of the triple. Note that other representation choices are also possible, such as the syntax proposed in [16].

Using the fusion method, the obtained description of the reference Rec_l takes the form, for a value v: $\pi_{\Sigma_l,k}(v) \ :< \ Rec_l \ P_k \ v >$, where P_k is a property and v is a value in $\mathcal{V}_{lk}$. Note that only one value of each equivalence class has to be stored, as the synonyms can be obtained through the ontology.

Example 5. The fused datum given in example 3 leads to the following declarations describing the possibility distribution for the value of the attribute A_3 :*hasComponent* :
$1 \ :< \ Rec_1 \ hasComponent \ \text{“}Thiamin'' >$
$0.143 \ :< \ Rec_1 \ hasComponent \ \text{“}Vitamin \ B'' >$.

5.2 Transformation of Fuzzy-RDF data into plain RDF data

In order to guarantee the implementation of the fusion method in all platforms based on plain RDF, we propose a transformation of our fuzzy-RDF representation of the fused data into plain RDF. We use the reification mechanism (for the reification semantics, see [17]) that allows adding new elements to the descriptions of the RDF declarations, like data author, creation date,.... In our case, the reification consists in adding to the triples of the form $< \ Rec_l \ P_k \ v >$ the *possibility* property that has a resource as domain and a decimal as range.

Let ns be the namespace of the RDFS schema which we have enriched by the $possibility$ property. We obtain, for each fuzzy-RDF triple of the form $\pi_{\Sigma_l,k}(v) \ :< Rec_l \ P_k \ v >$, to which an identifier $tripleID - i$ is assigned, its reified representation. For example, for the triple $1 \ :< \ Rec_l \ hasComponent \ "Thiamin" >$ we obtain the following reified representation:

```
<tripleID-1 rdf:type rdf:Statement > .
<tripleID-1 rdf:subject ns:Rec1> .
<tripleID-1  rdf:predicate ns:hasComponent> .
<tripleID-1  rdf:object "Thiamin"^^xsd:string > .
<tripleID-1  ns:possibility 1^^xsd:decimal> .
```

By applying this transformation, the fused references can be queried by using the SPARQL language without any need of extension.

5.3 Fused Reference Querying

The SPARQL syntax is close to the SQL one classicaly used in relational databases. The queries in SPARQL are evaluated on the set of triples contained in RDF data. In the following we will use *select queries*, which return a set of triples that check the constraints expressed in the WHERE clause. The SPARQL queries are evaluated on the set of fused references represented in plain RDF, by reification.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX ns: <http://www.lri.fr/~sais/myRDFS-1/>
SELECT  ?ref ?comp ?confidence
WHERE {
?x rdf:type rdf:Statement  .
?x rdf:subject ?ref  .
?x rdf:predicate ns:hasComponent .
?x rdf:object ?comp .
?x rdf:object ?confidence .
}
ORDER BY ?confidence
LIMIT 1
```

6 Experiments

In this section we present some experiment results of the fusion method. In order to assess the quality of the method, we use a dataset that we can compare to ideal value, in this case references of articles, of conferences and of persons. First, we give a description of the dataset that we have used. Then, we present the evaluation criteria that we have considered to validate / invalidate the obtained results.

6.1 Presentation of *Cora* dataset

The fusion method has been implemented and evaluated on the *Cora* data set related to the scientific publication domain. It has been used as a benchmark by several reference reconciliation approaches [2, 18]. Cora dataset is a collection of 1295 citations of 124 different research papers in computer science. These citations have been collected from the research engine Cora specialized on scientific publications search. We associate a reference to each article, conference and author (person). An article is described by several properties: *title*, *year*, *pageFrom*, *pageTo* and *type* which takes values in {proceedings, journal, book, ...}. A person is described by his *name* and a conference is described by three properties: *confName*, *confYear* and a *city*. There are two relations (objectProperties) which link each article to its authors and to the conference where it is published.

	Article	**Conference**	**Person**
#Groups	124	134	68
#References	1295	1292	3521
#distinct-values-per-group	[1..5]	[1..28]	[1..37]
Avg(#disctinct values-per-group)	3	8	9

Table 2. *Cora dataset description.*

We have applied the fusion method on the gold-standard of Cora dataset. It is organized as a set of 328 groups of pairwise reconciled references for the three classes: article, conference and person. In table 2, we present some statistics of the characteristics of the gold-standard: the number of renconciled reference groups, the number of references, the interval bounded by the minimum and maximum number of distinct values per group and the average of distinct values per reconciled group.

6.2 Evaluation protocol

To evaluate the validity of the fusion method, we have compared, for a set of selected properties, the ranking of their values according to the confidence degree obtained by the fusion method with the ranking given by a humain expert.

In some applications domains, the identification of the right value can be purely subjective. For example, choosing between the two painting names "*La joconde*" and "*Mona-lisa*" is not obvious, as the two names are acceptable. Nevertheless, there are some obvious criteria that allow to differentiate a right from a wrong value, which mainly consists in features that contribute to the syntactic integrity of the values.

– Typographical errors, like "*Cr**i**ptographic*" instead of "*Cr**y**ptographic*".
– Syntactic errors that are due to the data extraction processing, like, "*for - -mulae*" instead of "*formulae*" or "**Bart (1993).** *Reasoning with characteristic models*" instead of "*Reasoning with characteristic models*".
– Missing words, like "*... free probabilistic concepts*" instead of "*... free probabilistic* **learning** *concepts*"
– Additionnal words, like "**some** *experiments with a new ...*" instead of "*experiments with a new*"

When the previous criteria do not help the expert to classify the values, the DBLP[6] browser is used to determine the right-value and the wrong ones.

The second evaluation step consists in reviewing the list of ranked values of each property and classifying them, according to the previous criteria, into two classes: the right-values and the wrong-values. In the case of the Cora dataset, there is only one right value which satisfy the defined criteria. However, in some application domains they can be several values which can correspond to the right value in case of synonymies. The expert gives a ranking of the values by putting the right-value in the top rank, i.e., before all the wrong-values. The third step consists in comparing the ranked lists of values obtained by the fusion method with those given by the expert. In this step we count:

1. *#well-ranked-RV*: the number of well-ranked right-values, that is the number of right values that appear in the top rank.
2. *#misranked-RV*: the number of misranked right-values, that is the number of cases where the right value appears after one or several wrong-values (it has a lower confidence degree).

A less strict evaluation protocol could be used: instead of considering the top rank of the value list, we can consider the top-k list of values and check if the right-value belongs to this top-k list of values or not.

6.3 Fusion method results

In Table 3, we give the results for the three properties which contain most of syntactic variations: *Title*, *ConfName* and person *Name*. We compute the precision for the right-values as the proportion of the number of well-ranked values in reconciled groups [7]:

$$Precision = \frac{\#well-ranked-right-values}{\#reconciled-groups}$$

We note that the recall value is equal the precision because of the strict evaluation protocol. Indeed, as we consider that the fusion method fails when the right-value does not appear in the top position, the recall value corresponds also to the proportion of the well-ranked right-values in the reconciled groups.

[6] The DBLP Computer Science Bibliography which provides bibliographic information on major computer science journals and proceedings.
[7] We have considered the reconciled groups where the size of value list is (≥ 2).

	Article-Title	Conference-Name	Person-Name
#reconciled-groups	66	66	44
#well-ranked-RV	62	49	33
#misranked-RV	4	17	11
Precision=Recall	93.9%	74.2%	75%

Table 3. Fusion results in terms of precision for the values of: Title, ConfName, Name

The results of Table 3 show that the fusion method has obtained a precision of 93.9% for the ranking of the right values of article title. It obtains also a precision of 74.2 % for the ranking of conference name and of 75% for the person names. We can notice that the precision for the conference names and for the person names are lower than the precision of article titles. This can be due to the important rate of syntactic variation in their corresponding possible values.

As it is shown in Table 2, the number of distinct values of the conference names variates between 1 to 28 values and between 1 to 37 for the person names. For the conference names, the variations are mostly caused in by abbreviations (e.g. proc./proceedings, symp./symposium), by a variety of codifications (e.g. 9th/ninth) and by extraction problems (e.g. net-works/networks). For the person names, even when only considering the English-speaking world, a name can have several different spelling forms for a variety of reasons. In the Anglo-Saxon region and most other Western countries, a personal name is usually made of a given name, an optional middle name, and a surname or family name. Hispanic names can contain two surnames. For example, in the dataset we have 11 variations for the person name *Umesh Virkumar Vazirani*: {*Umesh Vazirani*; *U. Vazirani*; *Umesh V. Vazirani*; *Vazirani U.V.*; etc.}. Hence the main difficulties arise from what we could consider as abbreviations or synonyms. We could therefore improve our results by declaring such values as synonyms in our ontology. However, the results about article title show a very good recognition rate in case of a strict evaluation protocol. We can guarantee that the results can only be better for a top-k evaluation.

By these experiments we have shown the good performances of the developed fusion approach where data are syntactically very heterogeneous.

7 Related work

There are some studies on reference reconciliation that deal to a certain degree with reference fusion. In [3], a rule-based language is used by the administrator of the integration system to define different functions of reference fusion. Particular constructors are used to specify information on reliable data sources that are exploited in case of conflicts between values. Thus, the fusion can be achieved without considering the values coming from the other sources. Consequently, the value conflicts are not even detected. In [19], the fusion is also performed by using fusion rules that are specified by the integration system administrator. The authors propose five strategies for the resolution of conflicts between values. In [20], the authors propose a new operator FUSE BY used in SQL queries. This operator takes as arguments a set of pre-defined functions (e.g. vote, max, min) which are associated with attributes that are involved in the SQL query.

In our method, heterogeneity and conflict between redundant references is handled through the use of these criteria and of possibility theory, whereas previous methods [3] tends to ignore or bypass these features. The computation of possibility degrees is based on a combination of various criteria which are related to the value features, like frequency, but also related to data source features, like reliability. Unlike other approaches[20, 19], our method does not need any extension of the query language to be able to query the fused data. Finally, the representation of the fusion result in the form of possibility distributions allows to rank values by their plausibilities and thus offers some flexibility when handling and querying the fused data. It is based on justified theoretical uncertainty treatment tools and in-depth uncertainty modeling (discounting operation, homogeneity criterion taken into account through the merging operator, etc.) allowing to model imprecision. To our knowledge, such possibilistic-based methods are new.

Also, the fusion method we propose takes into account the hierarchical organization of the vocabulary, as well as equivalence relations, provided by an ontology, which is not the case in previous studies. A cooperation between the reconciliation/fusion process and ontology completion is also firstly provided in this paper, through the ontology enrichment procedure.

A preliminary study to the present work was presented in [11]. This work did not yet take into account the ontology, and used a more had-oc construction of uncertainty models. Note that we consider crisp ontology in this paper and not fuzzy ontology [21]. Indeed, in our case, fuzzy sets are used to describe uncertainty and arise from the fusion process, not from the ontology definition. That is, even if they are defined over concepts of the ontology, they do not pertain to the ontology.

8 Conclusion

In this paper, we have proposed a method for reference fusion, driven by ontological knowledge including subsomption and equivalence relations between concepts. The fusion method allows the computation, for each candidate value of a given property, of a possibility degree. The set of values that is assigned to a property is expressed as a possibility distribution.

We have shown the applicability of our methods and illustrated them on agronomical data. By the experiments on scientific publication dataset, we have shown the efficiency of the developed fusion approach even where data are syntactically very heterogeneous. We plan in a short term to apply the fusion method on datasets of other application domains where the use of semantic knowledge is more relevant (subsumptions, synonymies, etc.).

From a methodological point of view, an assumption that we made in this study is that values encountered in the data are not necessarily declared in the ontology (open-world assumption); we proposed a method that uses the values missing in the ontology as candidates to complete the existing ontology. Another perspective is to combine this approach with methods allowing automatic detection of synonyms, which is currently under study.

References

1. Rahm, E., Bernstein, P.A.: A survey of approaches to automatic schema matching. The VLDB Journal **10**(4) (2001) 334–350
2. Saïs, F., Pernelle, N., Rousset, M.C.: Combining a logical and a numerical method for data reconciliation. J. Data Semantics **12** (2009) 66–94
3. Papakonstantinou, Y., Abiteboul, S., Garcia-Molina, H.: Object fusion in mediator systems. In: VLDB, San Francisco, CA, USA (1996) 413–424
4. Dubois, D., Prade, H.: Tolerant fuzzy pattern matching: an introduction. In Bosc, P., Kacprzyk, J., eds.: Fuzziness in Database Management Systems. Physica-Verlag, Heidelberg (1995) 42–58
5. Dubois, D., Prade, H.: Possibility Theory - An Approach to Computerized Processing of Uncertainty. Plenum Press, New York (1988)
6. Dean, M., Schreiber, G.: OWL Web Ontology Language Reference, W3C Recommendation, http://www.w3.org/tr/owl-ref/. Technical report (2004)
7. Walley, P.: Measures of uncertainty in expert systems. Artifical Intelligence **83** (1996) 1–58
8. Zadeh, L.: Fuzzy sets as a basis for a theory of possibility. Fuzzy Sets and Systems **1** (1978) 3–28
9. Motik, B., Horrocks, I.: Owl datatypes: Design and implementation. In: ISWC '08: Proceedings of the 7th International Conference on The Semantic Web, Berlin, Heidelberg, Springer-Verlag (2008) 307–322
10. Wu, Z., Palmer, M.: Verbs semantics and lexical selection. In: Proceedings of the 32nd annual meeting on Association for Computational Linguistics, Morristown, NJ, USA, Association for Computational Linguistics (1994) 133–138
11. Saïs, F., Thomopoulos, R.: Reference fusion and flexible querying. In Meersman, R., Tari, Z., eds.: ODBASE-OTM Conferences (2). Volume 5332 of LNCS., Springer (2008) 1541–1549
12. W. Cohen, P.R., Fienberg, S.: A comparison of string metrics for matching names and records. In: Proc. of the KDD-2003 Workshop on Data Cleaning, Record Linkage, and Object Consolidation. (2003)
13. Dubois, D., Prade, H.: Possibility theory in information fusion. In Riccia, G.D., Lenz, H., Kruse, R., eds.: Data fusion and Perception. Volume CISM Courses and Lectures N 431. Springer Verlag, Berlin (2001) 53–76
14. W3C: Owl ontologies to rdf graphs, http://www.w3.org/2007/owl/wiki/mapping-to-rdf-graphs (2007)
15. Mazzieri, M.: A fuzzy rdf semantics to represent trust metadata. In: In 1st Workshop on Semantic Web. Applications and Perspectives. (2004)
16. Buche, P., Dibie-Barthélemy, J., Hignette, G.: Flexible querying of fuzzy rdf annotations using fuzzy conceptual graphs. In Eklund, P.W., Haemmerlé, O., eds.: ICCS. Volume 5113 of Lecture Notes in Computer Science., Springer (2008) 133–146
17. Hayes, P.: RDF Semantics, http://www.w3.org/tr/rdf-mt/. Technical report (2004)
18. Dong, X., Halevy, A., Madhavan, J.: Reference reconciliation in complex information spaces. In: SIGMOD '05: Proceedings of the 2005 ACM SIGMOD international conference on Management of data, New York, NY, USA, ACM Press (2005) 85–96
19. Subrahmanian, V., Adali, S., Brink, A., Emery, R., Lu, J.L., Rajput, A., Rogers, T.J., Ross, R., Ward, C.: Hermes: A heterogeneous reasoning and mediator system (1995)
20. Bleiholder, J., Naumann, F.: Declarative data fusion – Syntax, semantics, and implementation. In: Proc. of the 9th East European Conference on Advances in Databases and Information Systems. (2005)
21. Calegari, S., Ciucci, D.: Integrating fuzzy logic in ontologies. In: Proc. of the 8th International Conference on Enterprise Information Systems. (2006) 66–73