Dealing with P2P semantic heterogeneity through query expansion and interpretation

Anthony Ventresque, Sylvie Cazalens, Philippe Lamarre, Patrick Valduriez

Dealing with P2P Semantic Heterogeneity through Query Expansion and Interpretation

Anthony Ventresque
LINA, University of Nantes
Anthony.Ventresque@univ-nantes.fr

Sylvie Cazalens
LINA, University of Nantes
Sylvie.Cazalens@univ-nantes.fr

Philippe Lamarre
LINA, University of Nantes
Philippe.Lamarre@univ-nantes.fr

Patrick Valduriez
INRIA and LINA, University of Nantes
Patrick.Valduriez@inria.fr

ABSTRACT

In P2P systems where query initiators and information providers do not necessarily share the same ontology, semantic interoperability generally relies on ontology matching or schema mappings. Information exchange is then not only enabled by the established correspondences (the "shared" parts of the ontologies) but, in some sense, limited to them. Then, to what extent the "unshared" parts can also contribute to and improve information exchange ? In this paper, we address this question by considering a system where documents and queries are represented by semantic vectors. We propose a specific query expansion step at the query initiator's side and a query interpretation step at the document provider's. Through these steps, unshared concepts contribute to evaluate the relevance of documents wrt. a given query. Our experiments show that our method enables to correctly evaluate the relevance of a document even if concepts of a query are not shared. In some cases, we are able to find up to 90% of the documents that would be selected when all the central concepts are shared.

1. INTRODUCTION

In P2P systems where query initiators and information providers do not necessarily share the same ontology, semantic interoperability generally relies on ontology matching or schema mappings. Several works in this domain focus on what (*i.e.* the concepts and relations) the peers share [8, 15]. This is quite important because, obviously if nothing is shared between the ontologies of two peers, there is a little chance that they be able to understand the meaning of the information exchanged. However, no matter how the shared part is obtained (through consensus or mapping), there might be concepts (and relations) that are not consensual, and thus not shared. The question is then to know whether the unshared parts can still be useful for information exchange.

In this paper, we focus on semantic interoperability and information exchange between a query initiator p_1 and a document provider p_2, which use different ontologies but share some common concepts. The problem we address is to *find documents which are relevant to a given query although the documents and the query may be both represented with concepts that are not shared*. This problem is very important because in semantic web applications with high numbers of participants, the ontology (or ontologies) is rarely entirely shared. Most often, participants agree on some part of a reference ontology to exchange information and internally, keep working with their own ontology [15, 18].

We represent documents and queries by *semantic vectors* [20], a model based on the vector space model [1] using concepts instead of terms. Although there exist other, richer representations (conceptual graphs for example), semantic vectors are a common way to represent unstructured documents in information retrieval. Each concept of the ontology is weighted according to its representativeness of the document. The same is done for the query. The resulting vector represents the document (respectively, the query) in the n-dimensional space formed by the n concepts of the ontology. Then the relevance of a document with respect to a query corresponds to the proximity of the vectors in the space.

In order to improve information exchange beyond the "shared part" of the ontologies, we promote both *query expansion* (at the query initiator's side) and *query interpretation* (at the document provider's side). Query expansion may contribute to weight linked shared concepts, thus improving the document provider's understanding of the query. Similarly, by interpreting an expanded query with respect to its own ontology (*i.e.* by weighting additional concepts of its own ontology), the document provider may find additional related documents for the query initiator that would not be found by only using the matching concepts in the query and the documents. Although the basic idea of query expansion and interpretation is simple, query interpretation is very difficult because it requires to precisely weight additional concepts given some weighted shared ones, while

the whole space (i.e. the ontology) and similarity measures change.

In this context, our contributions are the following. First, we propose a specific query expansion method. Its property is to keep separate the results of the propagation from each central concept of the query, thus limiting the noise due to inaccurate expansion. Second, given this expansion, we define the relevance of a document. Its main, original characteristic is to require the document vector to be requalified with respect to the expanded query, the result being called *image* of the document. Third, a main contribution is the definition of query interpretation which enables the expanded query to be expressed with respect to the provider's ontology. Fourth, we provide two series of experiments with still very good results although few concepts are shared.

This paper is organized as follows. In Section 2 we show a (very) simple scenario. Section 3 gives preliminary definitions. Section 4 presents our query expansion method and the image based relevance of a document. For simplicity, we assume a context of shared ontology. This assumption is relaxed after in Section 5, where we consider the case where the query initiator and the document provider use different ontologies and present the query interpretation. Section 6 discusses the experiments and their results. The two last sections are respectively devoted to related work and conclusion.

2. MOTIVATING SCENARIO

Assume a P2P information system where peer Ann issues a query represented by {(*swing*,1.0)}. Ann has got two neighbours Bob and Charlie to which she sends her query. Each three of them uses a simple ontology as shown in Figure 1. There exist total or partial mappings between the ontologies. Assume Bob has data about {(*jazz*,0.4)} and Charlie about {(*scat*,0.8)}. As nobody has data related to the concept *swing*, Ann will not get any answer in a semantic vector space system with the cosine as relevance measure. This is because of independence of dimensions, as it concerns Bob, and because of semantic heterogeneity as it concerns Charlie.

A first solution is query expansion as in [19]. For example, Ann could send an expanded query {(*swing*,1.0),(*jazz*,0.6),(*bebop*,0.4),(*free*,0.4)}, because the three last concepts are linked to the central one of her query. Obviously, weights are assigned wrt. the similarity to central concept (*swing*): *jazz* is more similar to *swing* than *bebop* or *free*, and then has a greater weight. It enables to assess a certain but limited relevance to Bob's data.

In our opinion, Charlie's data about *scat* is relevant too, as its data is expressed on sibling concept of the query central one. In fact, we think that wrt. the expansion of the query, Charlie should be able to assert that Ann would put a weight on *scat* if she knew it. This is what we call the interpretation of the (expanded) query. The aim of this process is to "interpret" a query expansion according to another ontology, within another semantic space. That is to say, using the information extracted from the query expansion (central concept, propagated weights), Charlie should infer that *scat* should have a 0.4 weight if it were in Ann's ontology. Then, the query interpreted by Charly is {(*swing*,1.0),(*jazz*,0.6),(*scat*,0.4),(*funk*,0.4)}. Thus Charlie's data is relevant for Ann's query, even if the ontologies are not the same.

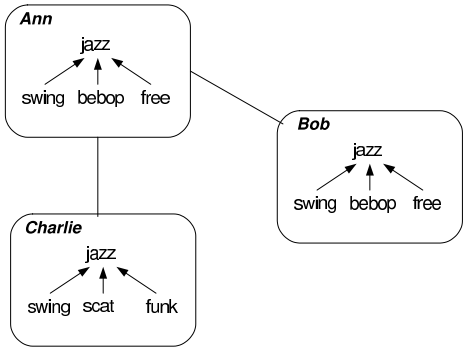

Figure 1: Ann, Bob and Charlie and their ontologies. In this example, mappings between ontologies rely on using the same word for the same concept.

The latter proposal does not increase the number of exchanged messages between Ann and her neighbours. There might only be a slight increase of the peers load due to expansion and interpretation. The flooding of the query across the system is out of the scope of this paper. Just notice that there are several solutions as for the forward of the query (either the original one or the interpreted one, or both). This increases the length of the messages, although not significantly, but does not change the number of exchanged messages.

3. PRELIMINARY DEFINITIONS

We define an ontology as a set of concepts together with a set of relations between these concepts. In our experiments, we consider an ontology with only one relation: the is-a relation (specialization link). This does not restrict the generality of our relevance computation. Indeed, the presence of several relations only affects the definition of the similarity of a concept wrt. another. A *semantic vector* $\overrightarrow{v_\Omega}$ is an application defined on the set of concepts $\mathcal{C}_\Omega$ of the ontology Ω : $\forall c \in \mathcal{C}_\Omega, \overrightarrow{v_\Omega} : c \rightarrow [0..1]$. A popular way to compute the relevance of a document is to use the cosine-based proximity of the document and query vectors in the space [16]. The problem with cosine is the independence of dimensions. Query expansion is generally used to express these links, by propagating initial weights on other linked concepts. To define a query expansion, we need a *similarity function* [14] which expresses how much a concept is similar to another within the ontology : sim_c: $\mathcal{C}_\Omega \rightarrow [0, 1]$, is a similarity function iff $sim_c(c) = 1$ and $0 \leq sim_c(c_j) < 1$ for all $c_j \neq c$ in $\mathcal{C}_\Omega$. Then, propagation from a central concept c of weight v assigns a weight to every value of similarity with c.

DEFINITION 1 (PROPAGATION FUNCTION). *Let c be a concept of Ω valued by v; and let sim_c be a similarity function. A function* $\mathcal{P}f_c$: $[0..1] \mapsto [0..1]$, $sim_c(c') \rightarrow \mathcal{P}f_c(sim_c(c'))$ *is a* propagation function from c *iff*

- $\mathcal{P}f_c(sim_c(c)) = v$, *and*
- $\forall c_k, c_l \in \mathcal{C}_\Omega \ sim_c(c_k) \leq sim_c(c_l) \Rightarrow \mathcal{P}f_c(sim_c(c_k)) \ \leq \ \mathcal{P}f_c(sim_c(c_l))$

Among different types of propagation functions those inspired by the membership functions used in fuzzy logic work fine (see Figure 2) in our experiments. Each one is defined by three parameters v (weight of the central concept), l_1 (similarity value until which concepts have the same weight : v) and l_2 (similarity value until which concepts have non zero weight) such that, $\forall x = sim_c(c'), c' \in c_\Omega$:

$$\mathcal{P}f_c(x) = f_{v,l_1,l_2}(x) = \begin{cases} v & if\ x \geq l_1 \\ \frac{v}{l_1 - l_2}x + \frac{l_2 \times v}{l_1 - l_2} & if\ l_1 > x > l_2 \\ 0 & if\ l_2 \geq x \end{cases}$$

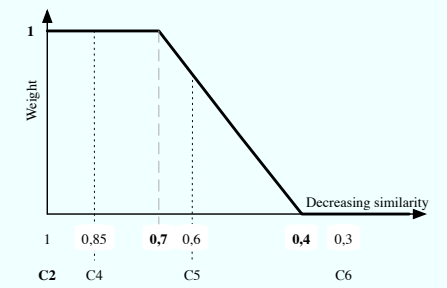

Figure 2: Example of a propagation function $f_{1,0.7,0.4}$ with central concept c_2 .

4. QUERY EXPANSION AND IMAGE BASED RELEVANCE

In this section, we present our method to compute the relevance of a document wrt a query. For the sake of simplicity, we assume that the query initiator and the document provider use the same ontology. However, they can still differ on the similarity measures and the propagation functions. First, we compute a *query expansion*, and then an *image of a document vector* to compute the relevance of the document wrt. a query in a single space.

To our knowledge, most propagation methods propagate the weight of each weighted concept in *the same vector*, thus directly adding the expanded terms in the original vector. When a concept is involved in several propagations conducted from different central concepts, an aggregation function (e.g. the maximum) is used. We call this kind of method "rough" propagation. Although its results are not bad, such a propagation has some drawbacks among which a possible unbalance of the relative importance of the initial concepts [12]. First, let us denote by $\mathcal{C}_{\overrightarrow{q}}$ the set of the *central concepts* of query $\overrightarrow{q}$, *i.e.* those weighted concepts which represent the query. To keep separate the effects of different propagations, each central concept of $\mathcal{C}_{\overrightarrow{q}}$ is *semantically enriched* by propagation, in a separate vector.

Definition 2 (Semantically Enriched Dimension). *Let $\overrightarrow{q}$ be a query vector and let c be a concept in $\mathcal{C}_{\overrightarrow{q}}$. A semantic vector $\overrightarrow{sed}_c$ is a* semantically enriched dimension, *iff* $\forall c' \in \mathcal{C}_\Omega, \overrightarrow{sed}_c[c'] \leq \overrightarrow{sed}_c[c]$.

Definition 3 (Expansion of a query). *Let $\overrightarrow{q}$ be a query vector. An* expansion *of $\overrightarrow{q}$, noted $\mathcal{E}_{\overrightarrow{q}}$ is a set defined by:* $\mathcal{E}_{\overrightarrow{q}} = \{\overrightarrow{sed}_c : c \in \mathcal{C}_{\overrightarrow{q}}, \ \forall c' \in \mathcal{C}_\Omega, \overrightarrow{sed}_c[c'] = \mathcal{P}f_c(c')\}$

Figure 3 illustrates the expansion of a query $\overrightarrow{q}$ with two weighted concepts c_4 and c_7. It contains two semantically enriched dimensions . In vector $\overrightarrow{sed}_{c_7}$, concept c_7 has the same value as in the query. Concepts c_3, c_{11} and c_6 have been weighted according to their similarity with c_7. The other dimension is obtained from c_4 in the same way.

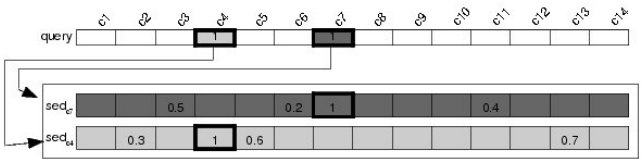

Figure 3: A query expansion composed of 2 semantically enriched dimensions.

The expanded query is composed of several semantic vectors (the SEDs). Our aim is then to transform the semantic vector of a document in an *image* through the expanded query, i.e. to characterize the document wrt. each central concept c (dimension) of the query, as far as it has concepts related to c, in particular even if c is not initially weighted in $\overrightarrow{d}$. Given a SED $\overrightarrow{sed}_c$, we aim at valuating c in the image of the document $\overrightarrow{d}$ according to the relevance of $\overrightarrow{d}$ to $\overrightarrow{sed}_c$. To evaluate the impact of $\overrightarrow{sed}_c$ on $\overrightarrow{d}$ we consider the product of the respective values of each concept in $\overrightarrow{sed}_c$ and $\overrightarrow{d}$. Intuitively, all the concepts of the document which are linked to c through $\overrightarrow{sed}_c$ have a nonnull value. The image of $\overrightarrow{d}$ keeps track of the best value assigned to one of the linked concepts if it is better than $\overrightarrow{d}[\ c\ ]$, which is the initial value of c . This process is repeated for each SED of the query. Algorithm 1 gives the computation of the image of document $\overrightarrow{d}$, noted $\overrightarrow{i}_d$. This algorithm ensures that all the central concepts of the initial query vector are also weighted in the image of the document as far as the document is related to them, as for c_4 and c_7 in the Figue 4. Wrt. the query, the image of the document is more accurate because it enforces the documents characterization over each dimension of the query (e.g. c_4 as a greater value in $\overrightarrow{i}_d$ than in $\overrightarrow{d}$ because c_4 is related to c_2). However, in the image, we keep unchanged the weights of the concepts which are not linked to any concept of the query (*i.e.* which are not weighted in any SED). For example c_1 and c_9.

We define the relevance of $\overrightarrow{d}$ wrt. $\overrightarrow{q}$ by $\cos(\overrightarrow{i}_d, \overrightarrow{q})$. Considering the image enables to take into account the documents that have concepts linked to those of the query. Using a cosine, and thus the norm of the vectors, assigns a lower importance to the documents with an important norm, which are often very general.

Algorithm 1: Image of a document wrt a query.

input : a semantic vector $\overrightarrow{d}$ on an ontology Ω; an expanded query $\mathcal{E}_{\overrightarrow{q}}$
output: a semantic vector $\overrightarrow{i}_d$, image of $\overrightarrow{d}$.
forall $c \in \mathcal{C}_{\overrightarrow{q}}$ **do**
  forall $c' : \overrightarrow{sed}_c[c'] \neq 0$ **do**
    $\overrightarrow{i}_d[c] \leftarrow \max(\overrightarrow{d}[c'] \times \overrightarrow{sed}_c[c'], \overrightarrow{i}_d[c])$;
forall $c \notin \mathcal{C}_{\overrightarrow{q}}$ **do**
  if $\exists c' \in \mathcal{C}_{\overrightarrow{q}} : \overrightarrow{sed}_{c'}[c] \neq 0$ **then** $\overrightarrow{i}_d[c] \leftarrow 0$
  else $\overrightarrow{i}_d[c] \leftarrow \overrightarrow{d}[c]$;
return $\overrightarrow{i}_d$;

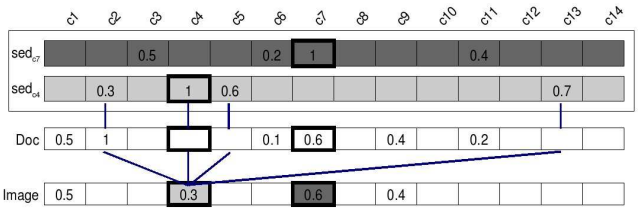

Figure 4: Obtaining the image of a document.

5. RELEVANCE IN THE CONTEXT OF UNSHARED CONCEPTS

In this section, we assume that the query initiator and the document provider do not use the same ontology. We follow the approach adopted in Section 4, using a query expansion at the query initiator's side and the computation of the image of the document at the provider's side. But things get complicated by the fact that the query initiator and the document provider do not use the same vector space. An additional step is needed in order to evaluate relevance in a same and single space. Thus, we introduce a *query interpretation* step at the provider's side.

5.1 Computing Relevance: Overview

As shown in Figure 5, the query initiator, denoted by p_1, works within the context of ontology Ω_1, while the document provider, noted p_2, works with ontology Ω_2. Through its semantic indexing module, the query initiator (respectively the document provider) produces the query vector (respectively the document vector), which is expressed on Ω_1 (respectively Ω_2). Both p_1 and p_2 also have their own way of computing both the similarity and the propagation.

We assume that the query initiator and the document provider *share* some common concepts, meaning that each of them regularly, although may be not often, runs an ontology matching algorithm. Ontology matching results in an *alignment* between two ontologies, which is composed of a (non empty) set of correspondences with some cardinality and, possibly some meta-data [4]. A *correspondence* establishes a relation (equivalence, subsumption, disjointedness...) between some entities (in our case, concepts), with some confidence measure. Each correspondence has an identifier. In this paper, we only consider the equivalence relation between concepts and those couples of equivalent concepts of which confidence measure is above some threshold. We call them the *shared* concepts. For simplicity, when there is an equivalence, we make no difference between the name of the given concept at p_1's, its name at p_2's, and the identifier of the correspondence, which all refer to the same concept. Hence, the set of shared concepts is denoted by $\mathcal{C}_{\Omega_1} \cap \mathcal{C}_{\Omega_2}$.

Given these assumptions, computing relevance requires the following steps :

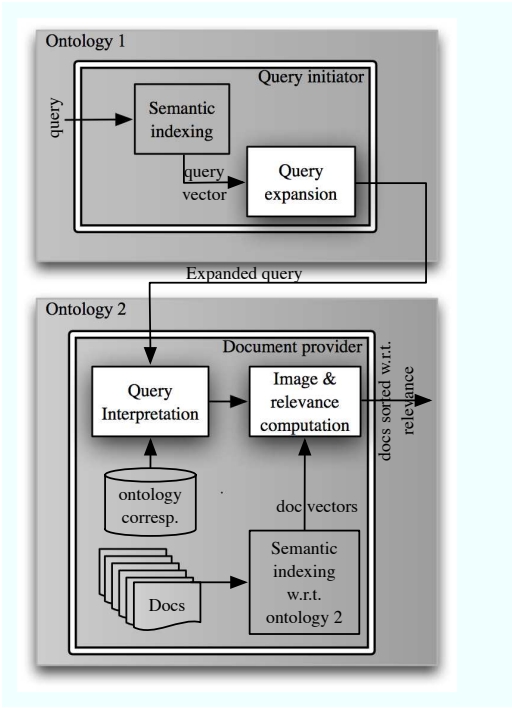

Figure 5: Overview of relevance computation

Query Expansion. It remains unchanged. The query initiator p_1 computes an *expansion* of its query, which results in a set of SEDs. Each SED is expressed on the set $\mathcal{C}_{\Omega_1}$, no matter the ontology used by p_2. Then, the expanded query is sent to p_2, together with the initial query.

Query Interpretation. Query interpretation by p_2 provides a set of interpreted SEDs on the set $\mathcal{C}_{\Omega_2}$ and an interpreted query. Each SED of the expanded query is interpreted separately. Interpretation of a SED $\overrightarrow{sed}_c$ is decomposed in two problems, which we address in the next subsections:

- The first problem is to find a concept in $\mathcal{C}_{\Omega_2}$ that corresponds to c, noted $\tilde{c}$. This is difficult when the central concept is not shared. In this case, we use the weights of the shared concepts to guide the search. Of course, this is only a "contextual" correspondence as opposed to one that would be obtained through matching.
- The second problem is to attribute weights to shared and unshared concepts of $\mathcal{C}_{\Omega_2}$ which are linked to $\overrightarrow{sed}_c$. This amounts to interpret the SED.

Image of the Document and Cosine Computation. They remain unchanged. Provider p_2 computes the image of its documents wrt. the interpreted SEDs and then, their cosine based relevance wrt. the interpreted query, no matter the ontology used by p_1.

In the following, we describe the steps involved in the interpretation of a given SED.

5.2 Finding a Corresponding Concept

The interpretation of a given SED $\overrightarrow{sed}_c$ leads to a major problem: finding a concept in $\mathcal{C}_{\Omega_2}$ which corresponds to the central concept c. This corresponding concept is noted $\tilde{c}$ and will play the role of the central concept in the interpretation of $\overrightarrow{sed}_c$, noted $\overrightarrow{sed}_{\tilde{c}}$. If c is shared, we just keep it as the central concept of the interpreted SED. When c is not shared we have to find a concept which seems to best respect the "flavor" of the initial SED.

Theoretically, all the concepts of $\mathcal{C}_{\Omega_2}$ should be considered. Several criterias can apply to choose one which seems to best correspond. We propose to define the notion of *interpretation function*. Definition 4 consists of four points. The first one requires the interpretation function to assign $\overrightarrow{sed}_c[c]$ to the similarity value 1. In the second point, we use the weights assigned by $\overrightarrow{sed}_c$ to the shared concepts ($c1$, $c2$, $c3$ and c_6 in figures 6 (a) and (b)) and the ranking of concepts in function of $sim_{\tilde{c}}$. However, there might be several shared concepts that have the same similarity value wrt. $\tilde{c}$, but have a different weight according to $\overrightarrow{sed}_c$. Thus, we require function $f_i^{\overrightarrow{sed}_c,\tilde{c}}$ to assign the minimum of these values to the corresponding similarity value. This is a pessimistic choice and we could either take the maximum or a combination of these weights. As for the third point, let us call c_{min}, the shared concept with the lowest similarity value (c_6 in Figure 6 (a) and c_3 in Figure 6 (b)). We consider that we have not enough information to weight the similarity values lower than $sim_{\tilde{c}}(c_{min})$. Thus we assign them the zero value. The fourth point is just a mathematical expression which ensures that the segments of the affine function are only those defined by the previous points.

DEFINITION 4 (INTERPRETATION FUNCTION). *Given a SED $\overrightarrow{sed}_c$ and a concept $\tilde{c}$, $f_i^{\overrightarrow{sed}_c,\tilde{c}} : [0..1] \rightarrow [0..1]$, noted f_i if no ambiguity, is an* interpretation function *iff it is a piecewise affine function and:*

- $f_i(1) = \overrightarrow{sed}_c[c]$;
- $\forall c' \in \mathcal{C}_{\Omega_1} \cap \mathcal{C}_{\Omega_2}, \quad f_i(sim_{\tilde{c}}(c')) = \min_{\substack{c'' \in \mathcal{C}_{\Omega_1} \cap \mathcal{C}_{\Omega_2} \\ sim_{\tilde{c}}(c') = sim_{\tilde{c}}(c'')}} (\overrightarrow{sed}_c[c''])$;
- $\forall x \in [0..1], \; x < sim_{\tilde{c}}(c_{min}) \;\Rightarrow\; f_i(x) = 0$;
- $Seg = \|\{x : \exists c' \in \mathcal{C}_{\Omega_1} \cap \mathcal{C}_{\Omega_2}, c' \neq \tilde{c} \text{ and } sim_{\tilde{c}}(c') = x\}\| + 1$ *where Seg is the number of segments of f_i.*

Intuitively, the criterias for choosing a corresponding concept among all the possible concepts can be expressed in terms of the properties of the piecewise affine function f_i. Of course, there are as many different function f_i as candidate concepts. The idea is to choose a concept which function f_i resembles the more a propagation function. Let us consider the example of Figure 6 (a) and (b) where c_1, c_2, c_3 and c_6 are shared. The function in Figure 6 (a) is obtained considering c'_1 as the corresponding concept (and thus ranking the other concepts in function of their similarity with $c'1$). The function in Figure 6 (b) is obtained similarly, considering c'_2. Having to choose between c'_1 and c'_2 we would prefer c'_1 because function $f_i^{\overrightarrow{sed}_c,c'_1}$ is monotonically decreasing whereas $f_i^{\overrightarrow{sed}_c,c'_2}$ shows a higher "disorder" wrt. the general curve of a propagation function.

Several characteristics of the interpretation function can be considered to evaluate "disorder". For example, one could choose the function which minimizes the number of local minima (thus minimizing the number of times the sign of the derivated function changes). Another example is to choose the function which minimizes the variations of weight between local minima and their next local maximum (thus penalizing the functions which do not decrease monotonically). A third could to combine these criterias.

5.3 Interpreting a SED

We define the interpretation of a given SED $\overrightarrow{sed}_c$ as another SED, with central concept $\tilde{c}$ which has been computed at the previous step. We keep their original weight to all the shared concepts. The unshared concepts are weighted using an interpretation function as defined above.

DEFINITION 5 (INTERPRETATION OF A SED). *Let $\overrightarrow{sed}_c$ be a SED on $\mathcal{C}_{\Omega_1}$ and let $\tilde{c}$ be the concept corresponding to c in $\mathcal{C}_{\Omega_2}$. Let $sim_{\tilde{c}}$ be a similarity function and let $f_i^{\overrightarrow{sed}_c,\tilde{c}}$, noted f_i, be an interpretation function. Then SED $\overrightarrow{sed}_{\tilde{c}}$ is an interpretation of $\overrightarrow{sed}_c$ iff:*

- $\overrightarrow{sed}_{\tilde{c}}[\tilde{c}] = f_i(1)$;
- $\forall c' \in \mathcal{C}_{\Omega_1} \cap \mathcal{C}_{\Omega_2}, \; \overrightarrow{sed}_{\tilde{c}}[c'] = \overrightarrow{sed}_c[c']$;
- $\forall c' \in \mathcal{C}_{\Omega_2} \setminus \mathcal{C}_{\Omega_1}, \; \overrightarrow{sed}_{\tilde{c}}[c'] = f_i(sim_{\tilde{c}}(c'))$;

Figure 6 (c) illustrates this definition. Document provider $p2$ ranks its own concepts in function of $sim_{\tilde{c}}$. Among these concepts, some are shared ones for which the initial SED $\overrightarrow{sed}_c$ provides a given weight. This is the case for c_1, c_2, c_3 and c_6 which are in bold face in the figure. The unshared concepts are assigned the weight they obtain by function f_i (through their similarity to $\tilde{c}$). This is illustrated for concepts c_4 and c_5 by a dotted arrow.

6. EXPERIMENTAL VALIDATION

In this section, we use our approach based on *image based relevance* to find documents which are the most relevant to given queries. We compare our results with those obtained by the *cosine based method* and the *rough propagation method*, because they are well-known and often used issues

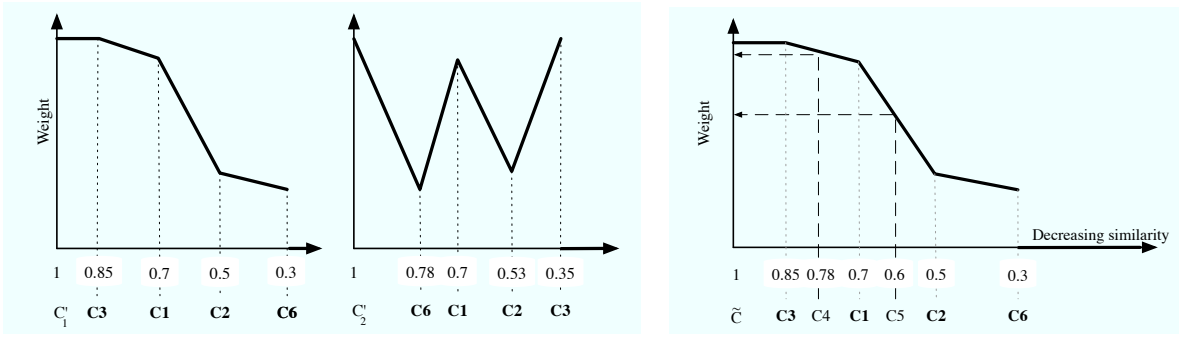

Figure 6: Two steps of the interpretation : (a) candidate concept c'_1, (b) candidate concept c'_2 and (c) weighting the unshared concepts.

in the litterature [16, 19]. In the former method, relevance is defined by the cosine between the query and document vectors. In the latter, the effects of propagating weights from different concepts are mixed in a single vector; then relevance is obtained using the cosine.

6.1 General Setup for the Experiments

We use the Cranfield corpus, a testing corpus consisting of 1400 documents and 225 queries in natural language, all related to aeronautical engineering. For each query, each document is scored by humans as relevant or not relevant (boolean relevance). Our ontology is lightweight, in the meaning of [7], *i.e.* an ontology composed of a taxonomy of concepts : WordNet [5]. Semantic indexing [17] is the process which can compute the semantic vectors from documents or queries in natural language. The aim is to find the most representative concepts for documents or queries. We use a program made in our lab : RIIO [3], which is based on the selection of synsets from WordNet. Although it is not the best indexing module, one of its advantages is that there is no human intervention in the process. The semantic similarity function we use is that of [2] for properties and accuracy reasons. We slightly modified that function due to normalization considerations.

In order to evaluate whether our solution is robust, we would need ontologies which agree on different percentages of concepts : 90%, 80%, 70%, ..., 10%. This is very difficult to obtain. We could build artificial ontologies, but this would force us to give up the experiments on a real corpus. Thus, we decided to stick to WordNet and simulate semantic heterogeneity.

Both the query initiator and the provider use WordNet, but we make so that they are not able to understand each other on some concepts (a given percentage of them). To do so, we remove some mappings between the ontology of the query initiator and the ontology of the document manager. Thus it simulates the case where the query intiator and the document provider use the same ontology but are not aware of it. It is then no more possible to compare queries and documents on those concepts. The aim is to evaluate how the answers to queries expressed with ontologies partially unshared, change. Note that the case with no concept removed reduces to a single ontology.

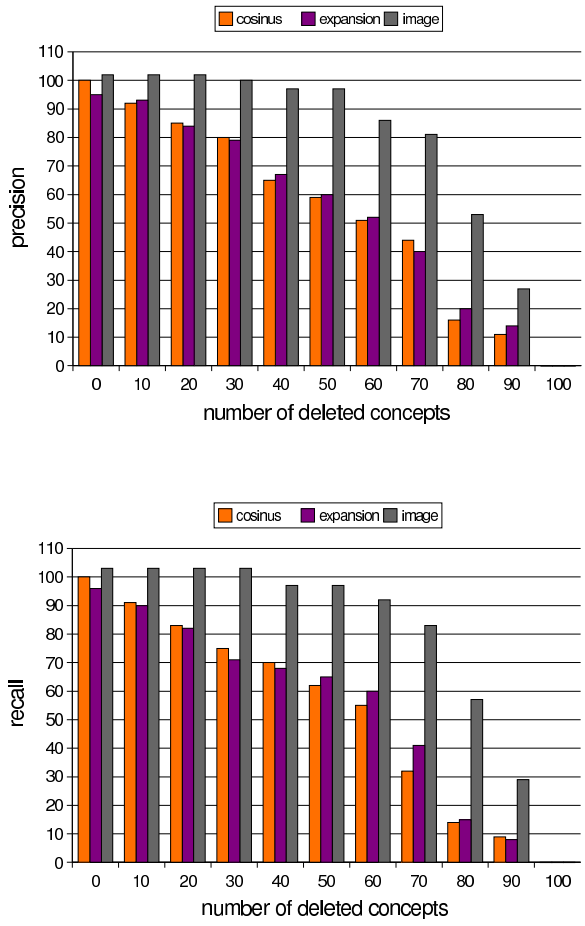

Figure 7: Evolution of precision and recall in function of the percentage of concepts randomly unknown from the set of shared concepts.

In a first experiment, we progressively reduce the number of mappings, thus increasing the percentage of deleted mappings (10%, 20%, ... until 90%). The progressive deletion in their common knowledge is done randomly. In a second experiment, we remove the mappings concerning the central concepts of the queries in the ontology of the document

manager. This is now an intentional removing, which is the worst case for most of the techniques in IR : loosing only the elements that match. For both experiments, we take into account the results obtained with the 225 queries of the corpus.

6.2 Results

Figure 7 shows the results obtained in average for the all 225 queries of the testing corpus. The reference method is the cosine one when no concept is removed, which gives a given reference precision and recall. Then, for each method and each percentage of removed concepts, we compute the ratio of the precision obtained (respectively recall) by the reference precision. When the percentage of randomly removed concepts increases, precision and recall (Figure 7) decrease *i.e.* the results are less and less relevant. However, our "image and interpretation based" solution shows much better results. When the percentage of removed concepts is under 70%, we still get 80% or more of the answers obtained in the reference case.

In the second experiment, we consider that the document manager does not understand (*i.e.* share with the query initiator) the central concepts of the query (see Figure 8). With the cosine method, there is no more matching between concepts in queries and concepts in documents. Thus no relevant document could be retrieved. With the query expansion, some of the added concepts in the query allow to match with concepts in documents that are close to the central concepts of the query. This leads to precision and recall at almost 10%. Our image-based retrieving method has more than 90% of precision and recall in the retrieval. This is also an important result. Obviously, as we have the same ontology and the same similarity function, the interpretation can retrieve most of the central concepts of the query. But the case presented here is hard for most of the classical techniques (concepts of the query unshared) and we obtain a very important improvement. This second experiment also highlights the limits of the approximation algorithm used to compute the corresponding concept when the central concept is not shared. Indeed, the algorithm used in the experiments considers the lowest common ancestor. However, if it would compute the correct concept each time, we would obtain 100% in our results. However, we do not get more than 90%.

7. RELATED WORK

The most common way to represent documents and queries after indexing is to use a vector space model as in [1]. Documents and queries in natural language are represented as vectors of keywords (terms). If there are n keywords, each document is represented by a vector in the $n-$dimensional space. Relevance of a document can then be computed by comparing the deviation of angles between the document vector and the original query vector. An approach based on *semantic* vectors [20, 10] uses the same kind of multi-dimensional linear space except that it no longer considers keywords but *concepts* of an ontology: the content of each document (respectively query) is abstracted to a semantic vector by characterizing it according to each concept. The more a given document is related to a given concept, the higher is the value of the concept in the semantic vector of

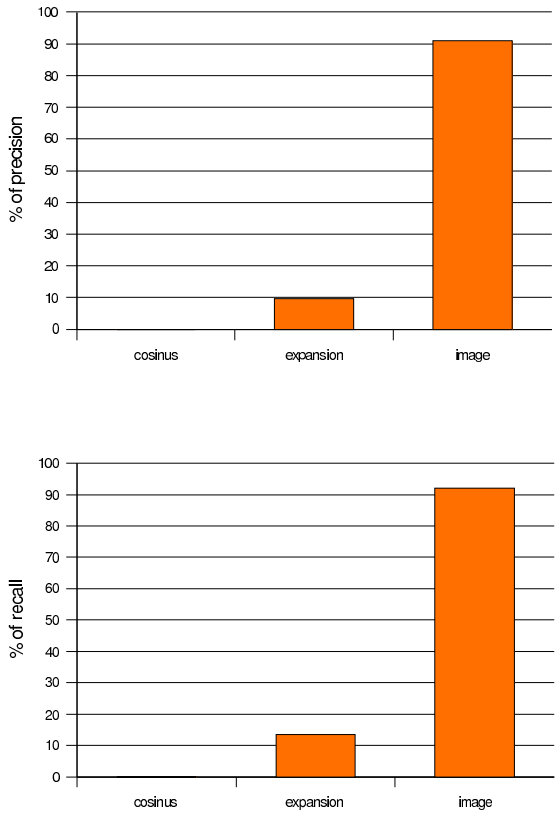

Figure 8: Precision and recall when the central concepts of the query are unshared.

the document. Our approach uses this kind of representation.

The idea of query expansion is shared by several fields. It was already used in the late 1980's in Cooperative Answering Systems [6]. Some of the suggested techniques expanded SQL queries considering a taxonomy. In this paper, we do not consider SQL queries, and we use more recent results about ontologies and their interoperability. Expansion of query vectors is used for instance in [13, 19]. However, this expansion produces a single semantic vector only. This amounts to mix the effects of the propagations from different concepts of the query. Although this method avoids some silence, it often generates too much noise, without any highly accurate sense disambiguation [19]. Consequently, the results can be worse than in the classical vector space model [1]. Our major difference with this approach is that (1) the propagations from the concepts of the query are kept separate and that (2) they are not directly compared with the document. Rather, they are used to modify its semantic vector. In our experiments, our method gives better results. Also, we join [12] on their criticism of the propagation in a single vector, but our solutions are different.

Our approach also relies on the correspondences resulting from the matching of the two ontologies. Several existing matching algorithms could be used in our case [4]. In the interpretation step, we provide a very general algorithm to

find the concept corresponding to the central concept of a SED. In case the concept is not shared, one could wonder whether matching algorithms could be used. In the solution we propose, the problem is quite different because the *weights of the concepts are also used* to find the corresponding concept (through the interpretation function). This is not the case in traditionnal ontology matching, which aim is to find general correspondences. In our case, one can see the problem as finding a "contextual" matching, the results of which cannot be used in other contexts. Because it is difficult to compute all the interpretation functions, one can use an *approximation algorithm* (for example, taking the least common ancestor as we did in our experiments). In that case, existing proposals can fit like [9, 11]. But it is clear that they do not find the best solution everytime.

Finally, the word *interpretation* is used very often and reflects very different problems. However, to the best of our knowledge, it never refers to the case of interpreting a query expressed on some ontology, within the space of another ontology, by considering the weights of the concepts.

8. CONCLUSION

The main contribution of this paper is a proposal improving information exchange between a query initiator and a document provider that use different ontologies, in a context where semantic vectors are used to represent documents and queries. The approach only requires the initiator and the provider to share some concepts and also uses the unshared ones to find additional relevant documents. To our knowledge, the problem has never been addressed before and our approach is a first, encouraging solution. In short, when performing query expansion, the query initiator makes more precise the concepts of the query by associating an expansion to each of them (SED). The expansion depends on the initiator's characteristics: ontology, similarity, propagation function. However, as far as shared concepts appear in a SED, expansion helps the document provider interpreting what the initiator wants, especially when the central concept is not shared. Interpretation by the document provider is not easy because the peers do not share the same vector space. Given its own ontology and similarity function, it first finds out a correspondent concept for the central concept of each SED, and then interprets the whole SED. The interpreted SEDs are used to compute an image of the documents and their relevance. This is only possible because the central concepts are expanded separately. Indeed if the effects of propagations from different central concepts were mixed in a single vector, the document provider wouldn't be able to interpret the query as precisely.

Although our approach builds on several notions (ontology, ontology matching, concept similarity, semantic indexing, relevance of a document wrt a query...) it is not stuck to a specific definition or implementation of them and seems compatible with many instantiations of them. It is important to notice that there is no human intervention at all in our experiments, in particular for semantic indexing. Clearly, in absolute, precision and recall could benefit from human interventions at different steps like indexation or the definition of the SEDs. Results show that our approach significantly improves the information exchange, finding up to 90% of the documents that would be found if all the concepts were shared.

9. REFERENCES

[1] M. W. Berry, Z. Drmac, and E. R. Jessup. Matrices, vector spaces, and information retrieval. *SIAM Rev.*, 41(2), 1999.

[2] A. Bidault, C. Froidevaux, and B. Safar. Repairing queries in a mediator approach. In *ECAI*, 2000.

[3] E. Desmontils and C. Jacquin. *The Emerging Semantic Web*, chapter Indexing a web site with a terminology oriented ontology. 2002.

[4] J. Euzenat and P. Shvaiko. *Ontology matching*. Springer-Verlag, Heidelberg (DE), 2007.

[5] C. Fellbaum. *WordNet : an electronic lexical database*. 1998.

[6] T. Gaasterland, P. Godfrey, and J. Minker. An overview of cooperative answering. *J. of Intelligent Information Systems*, 1(2):123–157, 1992.

[7] A. Gómez-Pérez, M. Fernández, and O. Corcho. *Ontological Engineering*. Springer-Verlag, London, 2004.

[8] Z. G. Ives, A. Y. Halevy, P. Mork, and I. Tatarinov. Piazza: mediation and integration infrastructure for semantic web data. *Journal of Web Semantics*, 2003.

[9] G. Jiang, G. Cybenko, V. Kashyap, and J. A. Hendler. Semantic interoperability and information fluidity. *Int. J. of cooperative Information Systems*, 15(1):1–21, 2006.

[10] R. Krovetz and W. B. Croft. Lexical ambiguity and information retrieval. *Information Systems*, 1992.

[11] E. Mena, A. Illaramendi, V. Kashyap, and A. Sheth. Observer: An approach for query processing in global information sytems based on interoperation across preexisting ontologies. *Int. J. distributed and Parallel Databases*, 8(2):223–271, 2000.

[12] J.-Y. Nie and F. Jin. Integrating logical operators in query expansion in vector space model. In *SIGIR workshop on Mathematical and Formal methods in Information Retrieval*, 2002.

[13] Y. Qiu and H. P. Frei. Concept based query expansion. In *SIGIR*, 1993.

[14] P. Resnik. Using information content to evaluate semantic similarity in a taxonomy. In *IJCAI*, 1995.

[15] M.-C. Rousset. Somewhere: a scalable p2p infrastructure for querying distributed ontologies. In *CoopIS/DOA/ODBASE*, 2006.

[16] G. Salton and M. MacGill. *Introduction to Modern Information Retrieval*. MacGraw-Hill, 1983.

[17] M. Sanderson. Retrieving with good sense. *Information Retrieval*, 2000.

[18] C. Tempich, H. S. Pinto, and S. Staab. Ontology engineering revisited: An iterative case study. In *ESWC*, pages 110–124, 2006.

[19] E. M. Voorhees. Query expansion using lexical-semantic relations. In *SIGIR*, Dublin, 1994.

[20] W. Woods. Conceptual indexing: A better way to organize knowledge. Technical report, Sun Microsystems Laboratories, 1997.