CONTEXT – SENSITIVE SEMANTIC QUERY EXPANSION

Giorgos Akrivas, Manolis Wallace, Giorgos Andreou, Giorgos Stamou and Stefanos Kollias

Image, Video and Multimedia Laboratory,
Department of Electrical and Computer Engineering,
National Technical University of Athens,
15773 Zografou,Greece
{wallace, gakrivas, geand}@image.ntua.gr, {gstam, stefanos}@softlab.ntua.gr

ABSTRACT

Modern Information Retrieval Systems match the terms included in a user's query with available documents, through the use of an index. A fuzzy thesaurus is used to enrich the query with associated terms. In this work, we use semantic entities, rather than terms; this allows us to use knowledge stored in a semantic encyclopedia, specifically the ordering relations, in order to perform a semantic expansion of the query. The process of query expansion takes into account the query context,which is defined as a fuzzy set of semantic entities. Furthermore, we integrate our approach with the user's profile.

1. INTRODUCTION

An *Information Retrieval System (IRS)* consists of a database, containing a number of documents, an *index*, that associates each document to its related terms, and a *matching mechanism*, that maps the user's query (which consists of *terms*), to a set of associated documents. Quite often, the user's query and the index are fuzzy, meaning that the user can somehow supply the degree of importance for each term, and that the set of associated terms for each document also contains degrees of association. In this case, the returned documents are sorted, with the one that best matches the user's query returned first [1].

It is possible that a query does not match a given index entry, although the document that corresponds to it is relevant to the query. For example, a synonym of a term found in a document may be used in the query. This problem is typically solved with the use of a fuzzy thesaurus containing, for each term, the set of its related ones. The process of enlarging the user's query with the associated terms is called *query expansion*; it is based on the associative relation A of the thesaurus, which relates terms based on their probability to co-exist in a document [2], [3].

In order to make query expansion more "intelligent", it is necessary to take into account the *meaning* of the terms [4]. The notion of a *semantic encyclopedia* can be used as a means to provide semantics to the user's query [5]. The user queries, on one hand, and the index entries, on the other, are represented with the use of *semantic entities*, defined in the encyclopedia. The former task is called *query interpretation*, and maps each term of a textual query to its corresponding semantic entity, thus producing the *semantic query*. This is naturally performed on query time. The latter task is called *detection of semantic entities* and produces the *semantic index*, i.e. a mapping of semantic entities to related documents [6]. This is performed off line.

Matching of semantic queries and semantic indices is not a trivial task; semantic entities that are not found in a query might be highly associated with it, and their inclusion might result in enhancing the process of information retrieval. Again, as in the case of textual queries and textual indices, a statistically generated associative relation A can be used for the expansion of the semantic query. Moreover, the various *semantic relations* amongst semantic entities, that are defined in the semantic encyclopedia, provide the possibility to perform a query expansion based on semantics rather than statistics. We will refer to this approach as *semantic query expansion*.

In this work, we use the semantic relations of the encyclopedia, specifically the ordering relations, in order to construct the *Inclusion* relation, which resides in the thesaurus. We use this relation to extract the common meaning of the terms in the query; we refer to this as the *context* of the query. Based on the latter, we propose an algorithm for *context – sensitive* semantic query expansion.

The paper is organized as follows: In section 2 we present the ordering relations of the encyclopedia and construct the Inclusion relation based on them. In section 3 we explain how, using the semantic entities in the query, we can mine the context of the query. In section 4 we integrate the context with the user profile, while in section 5 we use the context to map terms to semantic entities. Sections 6 and

This work has been partially funded by the EU IST-1999-20502 FAETHON project.

7 present our method for context – sensitive query expansion, as well as simulation examples. Finally, in section 8 we propose some possible extensions of our work.

2. THE INCLUSION RELATION OF THE FUZZY THESAURS

Detection of context, as mentioned above, depends on the Inclusion relation of the thesaurus. Before continuing, we provide a few details on mathematical notation in general and this relation in particular.

2.1. Mathematical Notation

Let $S = \{s_1, s_2, \ldots, s_n\}$, denote the set of semantic entities.

A *fuzzy binary relation* on S is a function $R : S^2 \to [0,1]$. The *inverse* relation R^{-1} is defined as $R^{-1}(x,y) = R(y,x)$.

The *intersection*, *union* and $\sup -t$ *composition* of two fuzzy relations P and Q defined on the same set S are defined as:

$[P \cap Q](x,y) = t(P(x,y), Q(x,y))$
$[P \cup Q](x,y) = u(P(x,y), Q(x,y))$
$[P \circ Q](x,y) = \sup\limits_{z \in S} t(P(x,z), Q(z,y))$

where t and u are a t-norm and a t co-norm, respectively. The *standard* t-norm and t-conorm are, respectively, the $\min$ and $\max$ functions. An *Archimedian* t-norm satisfies the property of subidempotency, i.e. $t(a,a) < a, \forall a \in (0,1)$.

The *identity* relation, I, is the identity element of the $\sup -t$ composition: $R \circ I = I \circ R = R, \forall R$.

The properties of *reflectivity*, *symmetricity* and $\sup -t$ *transitivity* are defined as following:

R is called reflective iff $I \subseteq R$

R is called symmetric iff $R = R^{-1}$

R is called antisymmetric iff $R \cap R^{-1} \subseteq I$

R is called sup-t transitive (or, simply, transitive) iff $R \circ R \subseteq R$

A *transitive closure* of a relation is the smallest transitive relation that contains the original relation . The transitive closure of a relation is given by the formula

$Tr\,(R) = \bigcup\limits_{n=1}^{\infty} R^{(n)}$

where $R^{(n)} = R \circ R^{(n-1)}, R^{(1)} = R$

If R is reflective, then its transitive closure is given by $Tr\,(R) = R^{(n-1)}$, where $n = |S|$ [7].

A fuzzy *ordering* relation is a fuzzy binary relation that is antisymmetric and transitive. A *partial ordering* is, additionally, reflective.

A fuzzy partial ordering relation R defines, for each element $s \in S$, the fuzzy set of its *ancestors* (dominating class) $R_{\geq[s]}(x) = R(s,x)$, and its *descendants* (dominated class) $R_{\leq[s]}(x) = R(x,s)$. For simplicity, we will use the alternative notation $R(s)$ instead of $R_{\leq[s]}$.

As described in section 1, a query q is a fuzzy set defined on S. This means that any element $s_i \in S, i \in \mathbb{N}_n$ belongs to q in some degree $w_i = q(s)$. Of course, for most semantic entities this degree is expected to be zero. Nevertheless, we assume that $w_i = 1$ for at least one semantic entity (i.e. q is normal, the height $h(q) \doteq \max\limits_{i \in \mathbb{N}_n} w_i = 1$). From now on, for the query q, we will use the vector notation $\mathbf{q} = [q_i]$, or the sum notation $q = \sum\limits_{i \in \mathbb{N}_n} s_i / w_i$.

2.2. The Fuzzy Inclusion Relation

Construction of the Inclusion relation is typically based on the *specialization* relation Sp , which is a partial ordering on the set of the semantic entities[8]. $Sp(a,b)$ means that the meaning of a “includes” the meaning of b. The most common forms of specialization are subclassing (i.e. a is a generalization of b) and thematic categorization (i.e. a is the thematic category of b). The role of the specialization relation in query expansion is that if the user query contains a, then a document containing b will be of interest, since it contains a special case of a.

Another important ordering found in the encyclopedia is the *part* relation P [8]. $P(a,b)$ means that b is a part of a. Moreover, it is expected that the role of P for query expansion is the opposite of that of Sp, i.e. when the user query contains b, a document containing a will probably be of interest, because a contains a part b.

Given the above considerations, we construct the Inclusion relation I of the thesaurus as follows:

$$I = (Sp \cup P^{-1})^{n-1} \qquad (1)$$

where $n = |S|$. This means that I is the transitive closure of $Sp \cup P^{-1}$. Since the composition of transitive relations is not necessarily transitive, this closure is necessary. Based on the roles that Sp and P have in information retrieval, it is easy to see that (1) combines them in a way that implies that, if the user query contains a, then $I(a,b)$ indicates that documents that contain b will also be of interest.

In this work, fuzziness of the I relation has an important role. High values of $I(a,b)$ imply that the meaning of b approaches the meaning of a, in the sense that when the user query contains a, then the user will certainly be satisfied with documents containing b. On the other hand, as $I(a,b)$ decreases, the meaning of b becomes “narrower” than the meaning of a, in the sense that a document containing just b will probably not be of interest to the user. Therefore,

$$a \neq b \implies I(a,b) < 1$$

Moreover, the t-norm is an Archimedian norm. This means that $I(a,c) \geq \max\limits_{s \in S} t(I(a,s), I(s,c))$, $t(a,a) < a$ and, therefore, $t(a,b) < \min(a,b), \forall a \in (0,1)$.

3. DETECTION OF THE QUERY CONTEXT

By using the above interpretation of the I relation, we define the *context* of a semantic entity as the set of semantic entities that are included in it. Therefore, the context of s is simply the set of descendants $I(s)$. Assuming that q is crisp, the context $K(q)$ of q, which is a set of semantic entities, can be defined simply as the set of their common descendants, i.e.:

$$K(q) = \bigcap_{i \in \mathbb{N}_n} I(s_i)$$

Obviously, $q_1 \subseteq q_2 \implies K(q_1) \supseteq K(q_2)$, i.e. the presence of more query terms will make the query context narrower.

We will show that a direct extension of the above definition in the fuzzy case, for example $K^*(q) = \bigcap\limits_i w_i I(s_i)$, is not meaningful. A low degree of importance w_i for the semantic entity s_i implies that the meaning of s_i is relatively insignificant for the query. On the other hand, it is implied by the above definition of K^* that a low value of w_i will narrow the context more than a high one; this is the opposite effect than what is desired. In order to achieve the desired effect, the following conditions must be satisfied, for the weighted context $K(s_i) = \sum\limits_j s_j / K(s_i)_j$ of the semantic entity s_i:

- if $w_i = 0$, then $K(s_i) = S$ (no narrowing of context)
- if $w_i = 1$, then $K(s_i) = I(s_i)$
- $K(s_i)_j$ decreases monotonically with w_i

Our approach is linear:

$$K(s_i)_j = 1 - w_i(1 - I(s_i)_j)$$

The context of the fuzzy query is the fuzzy intersection of the individual weighted contexts:

$$K(q) = \bigcap_i K(s_i)$$

When the query terms are highly correlated in I, then the query context will contain high values. We will use the term *context intensity* for the greatest of them, i.e. for the height $h_q = h(K(q))$ of the query context.

4. USER PROFILE

In section 6, we propose a query expansion method, which considers the context. In this section, we use the user's preferences to alter this context, thus providing the capability for query personalization.

Since the context is defined on the set S of semantic entities, it makes sense to define the user profile on the same set [9], as follows:

$$U^+ = \sum_i s_i / u_i^+ \subseteq S$$

$$U^- = \sum_i s_i / u_i^- \subseteq S$$

U^+ is the fuzzy set of a user's *positive preferences* and U^- is the fuzzy set of the user's *negative preferences*. Positive preferences refer to the degree to which we believe that a semantic entity is of interest to the user, and negative preferences refer to the degree to which we believe that a semantic entity is not of interest to the user. As a minimum restriction for consistency we demand that

$$u_i^+ > 0 \implies u_i^- = 0$$

Neutral preference for a semantic entity is denoted by $u_i^+ = u_i^- = 0$.

We propose the following method for shifting the original context $K(q)$ in the direction of the user's preferences:

$$K'(q)_j = (K(q)_j)^{1 + n u_i^- - p u_i^+}$$

where $K'(q)$ is the adjusted (personalized) context, and p, n are parameters that specify the degree to which the user's positive and negative preferences affect the context. The proposed formula and its properties have been studied extensively in the field of fuzzy logic, under the general category of *modifiers* or *linguistic hedges*. It has the following properties, that are desired in the process of personalizing the context:

- it does not affect weights when the preference is neutral
- it does not alter weights that are equal to one or zero
- it is monotonous with respect to the initial weight of the context
- it is monotonous with respect to the user's preference.

5. CONTEXT – SENSITIVE QUERY INTERPRETATION

In section 3, we supposed that the mapping of the terms provided by the user to the corresponding semantic entities is one-to-one, and therefore trivial. This is true for most cases; still, exceptions exist, as distinct semantic entities may have common textual descriptions. As a simple example, let us consider the case of the term "element". At least two distinct semantic entities correspond to it: "element$_1$", which is related to chemistry, and "element$_2$", which is related to XML.

Let us now suppose that a query containing the term "element" is given by a user. If the remaining terms of the query are related to chemistry, then it is quite safe to suppose that the user is referring to semantic entity "element$_1$" rather than to semantic entity "element$_2$". This implies that the context of the query can be used to facilitate the process of semantic entity determination. However, the detection of the query context, as it was described in section 3, cannot be performed before the query interpretation is completed. Therefore, query interpretation needs to take place simultaneously with context detection. We propose the following method:

Let the textual query contain the terms $\mathbf{t} = [t_i], i = 1, \ldots, T$. Let also t_i be the textual description of semantic entities $s_{ij}, j = 1, \ldots, T_i$. Then, there exist $\prod\limits_i T_i$ distinct combinations of semantic entities that may be used for the representation of the user's query; for each one of those we calculate the corresponding context. The combination that produces the most intense context is the one we select.

The algorithm we have proposed for query interpretation is exhaustive. Still, this is not a drawback, as:

- queries do not contain large numbers of terms
- the terms for which more that one semantic entities may be chosen are rare
- the number of distinct semantic entities that may have a common textual description is not large.

6. CONTEXT – SENSITIVE QUERY EXPANSION

As mentioned in section 1, query expansion enriches the query in order to increase the probability of a match between the query and the document index. The presence of several entities in the query defines a context, which we use, in this section, to direct the expansion process.

6.1. Handling of Semantic Entities in Query Expansion

In section 1, we explain that the search engine uses the query q, and the document index D, which is a fuzzy relation between the set of semantic entities S and the set of documents T, to produce the result r; r is a fuzzy set on T. When the query is comprised of a single semantic entity s, then the result is simply the respective line of D, i.e. $r(q) = D(s)$. When the query contains more than one semantic entities, then the result is the set of documents that contain all the semantic entities, i.e. $r(q) = D(s_1) \cap D(s_2) \cap D(s_3)$.

In query expansion, we replace each semantic entity s with a set of semantic entities $X(s)$; we will refer to this set as the expanded semantic entity. When querying, we treat $X(s)$ considering a union operation, i.e. documents that match any entity contained in $X(s)$ are selected. Therefore, in order to preserve the intersection operation among the original query entities, we need to expand each entity separately.

6.2. Semantic Entity Expansion

Using the above principle, we formally define the expanded entity $X(s_i) = \sum\limits_j s_j / x_{ij}$ as a fuzzy set on S; we compute it using the query q, the context $K(q)$ of the query, and the I relation of the thesaurus. The weight x_{ij} denotes the degree of significance of the entity s_j in $X(s_i)$.

In a context – insensitive query expansion, the value of x_{ij}, is proportional to the weight w_i and the degree of inclusion $I(s_i, s_j)$. Therefore, $x_{ij} = w_{ij} = w_i I(s_i, s_j)$. In a context – sensitive query expansion, x_{ij} increases monotonically with respect to the degree to which the context of s_j is relative to the context of the query. We will use the quantity

$$h_j = \frac{h(I(s_j) \cap K(q))}{h_q}$$

as a measure of this relevance. Therefore, we demand that the following conditions be satisfied by our method :

- x_{ij} increases monotonically with respect to w_{ij}.
- $h_j = 1 \implies x_{ij} = w_{ij}$
- $h_q = 0 \implies x_{ij} = w_{ij}$
- $h_q = 1 \implies x_{ij} = w_{ij} h_j$
- x_{ij} increases monotonically with respect to h_j

Again, we follow a linear approach and propose the following formula:

$$x_{ij} = w_{ij}(1 - h_q(1 - h_j))$$

It is easy to observe that the expanded entity produced by this method satisfies the above conditions.

7. SIMULATION EXAMPLE

In this section, we give examples of the proposed algorithms for context personalization and context – sensitive query expansion.

$K(q)_j$	$K'(q)_j$	$K'(q)_j - K(q)_j$
0,00	0,00	0,00
0,10	0,20	0,10
0,20	0,32	0,12
0,30	0,43	0,13
0,40	0,53	0,13
0,50	0,62	0,12
0,60	0,70	0,10
0,70	0,78	0,08
0,80	0,86	0,06
0,90	0,93	0,03
1,00	1,00	0,00

Table 1. Positive preference 0.6

7.1. Context Personalization

In order to simulate the proposed context personalization method, we assume that a user profile contains a positive preference for semantic entity s_j to a degree 0.6. Formally, we assume that $u_i^+ = 0.6$. Table 1 presents the importance $K'(q)_j$ of entity s_j in the personalized context, for $p = 0.5$ and $K(q)_j$ ranging form 0 to 1. It also presents the difference between $K'(q)_j$ and $K(q)_j$; this difference is a measure of the influence of the personalization process on the original context. We can observe that (a) the influence is positive, i.e. the context is shifted in the direction of the preference, as well as that (b) the influence becomes smaller as $K(q)_j \to 0$ or $K(q)_j \to 1$; on the contrary, it becomes larger as $K(q)_j$ becomes fuzzier. The later implies that the role of the profile becomes more important, as the uncertainty about the participation of a semantic entity in the context is greater.

Table 2 presents the corresponding values, if we assume a negative preference $u_i^- = 0.6$. Although the process's behavior is similar (the values are shifted in the direction of the preference), the influence on the original context is smaller. This results from the fact that $n < p$, and is desired, as we choose to treat positive preferences with greater confidence.

7.2. Query Expansion

In order to demonstrate the proposed method, we provide an example of the constructed Inclusion relation in Figure 1. Thick lines correspond to the Part relation, while thin lines correspond to the Specialization relation. The archimedian norm we use is the product. Values that are implied by transitivity are omitted for the sake of clarity.

We present the result of expansion of the following queries:

q_1 =“Motor”$/1+$“Airplane”$/1$
q_2 =“Motor”$/1+$“Airplane”$/1+$“Propeller”$/0.7$
q_3 =“Motor”$/1+$“Airplane″$/1+$“Propeller”$/1$

$K(q)_j$	$K'(q)_j$	$K'(q)_j - K(q)_j$
0,00	0,00	-0,00
0,10	0,08	-0,02
0,20	0,16	-0,04
0,30	0,26	-0,04
0,40	0,36	-0,04
0,50	0,46	-0,04
0,60	0,56	-0,04
0,70	0,67	-0,03
0,80	0,78	-0,02
0,90	0,89	-0,01
1,00	1,00	-0,00

Table 2. Negative preference 0.6

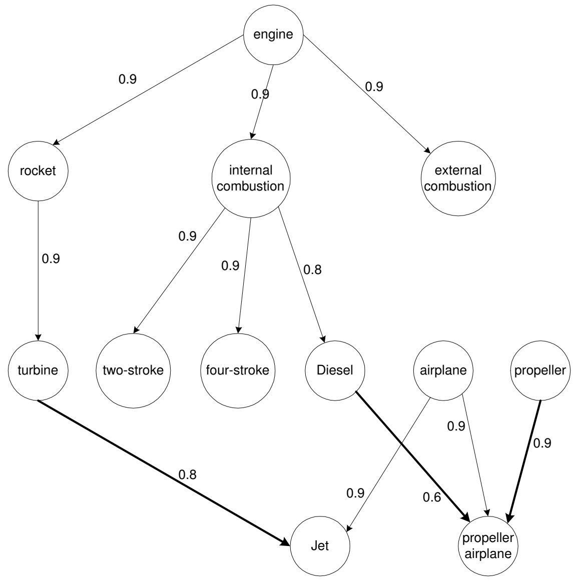

Fig. 1. The Inclusion relation

Entity	No Context	q_1	q_2	q_3
motor	1	1	1	1
ext-combustion	0.9	0.38	0.51	0.51
int-combustion	0.9	0.77	0.9	0.9
4-stroke	0.81	0.34	0.46	0.46
2-stroke	0.81	0.34	0.46	0.46
rocket	0.8	0.8	0.7	0.46
diesel	0.72	0.61	0.72	0.72
turbine	0.72	0.72	0.63	0.41
jet	0.58	0.58	0.51	0.33
prop-plane	0.43	0.37	0.43	0.43

Table 3. Expanded entity “motor”

Entity	No Context	q_1	q_2	q_3
airplane	1	1	1	1
prop-plane	0.9	0.77	0.9	0.9
jet	0.9	0.9	0.78	0.51

Table 4. Expanded entity “airplane”

Entity	No Context	q_1	q_2	q_3
propeller	1	-	0.7	1
prop-plane	0.9	-	0.63	0.9

Table 5. Expanded entity “propeller”

Column “No Context” in table 3 shows the degree to which various semantic entities participate in $I($“Motor”$)$; this is in fact their importance in $X(\text{“Motor}'')$, if the context is not considered. Columns “q_1”, “q_2” and “q_3” show the degree to which they are used for the expansion of the semantic entity “Motor”, according to our method, if the query is q_1, q_2 and q_3, respectively. Tables 4 and 5 present the corresponding data for semantic entities “Airplane” and “Propeller”. We observe the following.

The use of the context in the expansion of term “Motor” in query q_1 results in the drastic diminishing of the importance of the terms “four-stroke”, “two-stroke” and “external combustion” in the expanded entity. This is desirable, as it is easy to see that, according to our encyclopedia, these terms are not related to the context of “Airplane”. Furthermore, the semantic entities that are related to the context are not filtered. Thus, the entity expansion is successfully performed in the direction that the query context specifies. We observe the same when considering the remaining terms or queries.

The three simulated queries are not independent. They are all of the form:

$q =$“Motor”$/1+$“Airplane”$/1+$“Propeller”$/w$

where w assumes the values 0, 0.7, 1. It is easy to see that all membership degrees in the expanded q_2 lie between their corresponding values for queries q_1 and q_3. This implies that the transition from $w = 0$ to $w = 1$ is gradual; therefore, fuzziness of queries in the extraction of the context is meaningful.

8. CONCLUSIONS AND FUTURE WORK

In this work, we propose a novel definition of the fuzzy Inclusion relation, which uses knowledge stored in a semantic encyclopedia. The problem of semantic query expansion is tackled through the notion of context, which is based on the Inclusion relation.

In the proposed method, linear approaches are applied, for the sake of simplicity. We believe that more general, non – linear approaches might be interesting to investigate. Another open issue is the choice of the archimedian norm in the fuzzy transitivity of the Inclusion relation. Finally, the result of the semantic query expansion must be combined with the result of the associative query expansion in an efficient manner.

9. REFERENCES

[1] Kraft D.H., Bordogna G. and Passi G., Fuzzy Set Techniques in Information Retrieval, in James C. Berdek Didier Dudas and Henri Prade (Eds.) Fuzzy Sets in Approximate Reasoning and Information Systems, (Boston: Kluwer Academic Publishers, 2000).

[2] Miyamoto S., Fuzzy sets in information retrieval and cluster analysis, (Dordrecht/Boston/London: Kluwer Academic publishers, 1990)

[3] Wen-Syan Li and Divyakant Agrawal, Supporting web query expansion efficiently using multi-granularity indexing and query processing, Data & Knowledge Engineering, Volume 35, Issue 3, December 2000, Pages 239-257

[4] Kraft D.H., Petry F.E., Fuzzy information systems: managing uncertainty in databases and information retrieval systems, Fuzzy Sets and Systems, 90 (1997) 183-191, Elsevier.

[5] Akrivas G., Stamou G., Fuzzy Semantic Association of Audiovisual Document Descriptions, Proc. of Int. Workshop on Very Low Bitrate Video Coding (VLBV), Athens, Greece, Oct. 2001

[6] Avrithis Y. and Stamou G., FAETHON: Unified Intelligent Access to Heterogenous Audiovisual Content, Proc. of Int. Workshop on Very Low Bitrate Video Coding (VLBV), Athens, Greece, Oct. 2001

[7] Klir G. and Bo Yuan, Fuzzy Sets and Fuzzy Logic, Theory and Applications, New Jersey, Prentice Hall, 1995

[8] ISO/IEC JTC 1/SC 29 M4242, Text of 15938-5 FDIS Information Technology – Multimedia Content Description Interface – Part 5 Multimedia Description Schemes, October 2001.

[9] Manolis Wallace, Giorgos Akrivas, Giorgos Stamou and Stefanos Kollias, Representation of User Preferences and Adaptation to Context in Information Retrieval, submitted.