An Extension of a Fuzzy Ontology for Flexible Querying

Nouredine Tamani, Ludovic Lietard, Daniel Rocacher

An Extension of a Fuzzy Ontology for Flexible Querying

Nouredine Tamani
IRISA/ENSSAT/Univ. de Rennes 1
BP 80518, 22305
Lannion Cedex, France
Email: tamani@enssat.fr

Ludovic Liétard
IRISA/IUT/Univ. de rennes 1
BP 30219, 22302
Lannion Cedex, France
Email: ludovic.lietard@univ-rennes1.fr

Daniel Rocacher
IRISA/ENSSAT/Univ de Rennes 1
BP 80518, 22305
Lannion Cedex, France
Email: rocacher@enssat.fr

***Abstract*—In this paper, we propose a personalized approach for flexible querying of information systems. This approach consists in the combination of the reasoning capabilities of the fuzzy DLR-Lite ontology and the expressivity of the SQLf language. The interpretation of the gradual inclusion (subsumption) axioms of the ontology is based on the Gödel fuzzy implication. Its generalization to a tree of inclusions is also proposed. This tree and its property of propagation of degrees are the basic theoretical elements of our application, which consists in querying of a multimodal transport information system which is embedded in a mobile terminal characterized by limited storage and processing capabilities.**

***Keywords*—Ontology, flexible querying, personalization, fuzzy DLR-Lite, SQLf, fuzzy sets theory, multimodal transportation.**

I. Introduction

The personalization of flexible querying of information systems dedicated to multimodal transportation networks is addressed in this paper. We propose an approach which consists in the development of a fuzzy semantic layer in order to extend the functionalities of information systems to reasoning. This layer is made of reasoning functions which combine domain application data with ontology axioms. Ontologies are able (i) to model the domain knowledge, (ii) to unify its vocabulary, (iii) to reason with available knowledge and (iv) to deduce implicit data from explicit data.

The proposed approach is developed in the framework of flexible querying of information systems which allows the user to express preferences in his/her queries. In such a system, the delivered answers are ranked from the most to the least preferred. In this context, user preferences can be expressed by vague or fuzzy predicates such as *fast*, *expensive*, *near*, etc. which are defined by fuzzy sets [16]. From the querying point of view, the SQLf language [4], [3] is a suitable language to express query with preferences. It is an extension of the SQL language to fuzzy predicates (conditions) and provides more expressive SQL-like statements.

From the reasoning point of view, several fuzzy ontologies (based on description logics [1]) have been developed (see [2], [8], [9], [10], [11], [12]) in which theoretical aspects have been widely studied (the definition of fuzzy concepts and relations, the definition and the interpretation of fuzzy inclusion or subsumption axioms, the fuzzy extension of subsumption algorithms, etc.). However, these ontologies are not suitable to address querying issue. Indeed, ontologies save axioms, individuals and related data in text files which are not appropriate for the needs of real applications. Ontology-based querying systems are faced with a large amount of data to such extent that the combination of ontologies and databases become an absolute necessity. In addition, the developed ontology-based querying systems, such as [15], [5], [10], [13], are based on limited description logics in terms of expressivity. To overcome this limitation, databases are combined with ontologies to provide DBMS functionalities such as data persistence and query evaluation. Ontology axioms are then used in the query rewriting process to take into account the logical relationships between concepts.

In this context, we consider the Description Logic extended to *m*-ary Relations [5] denoted DLR-Lite. More precisely, we are interested in the fuzzy extension of the DLR-Lite, called fuzzy DLR-Lite [7], [14], [15], which is made of a reduced set of fuzzy operators and abstraction rules which are connected to databases through SQL statements.

In this paper, we aim to design a flexible querying system that combines a fuzzy DLR-Lite knowledge base with the SQLf language, so as to take profit from the expressivity of the SQLf language and the reasoning mechanisms of the fuzzy DLR-Lite. Firstly, we show the relevance of the Gödel fuzzy implication to the interpretation of gradual inclusion (subsumption) axioms of the fuzzy ontology. Then, we consider the enrichment of the fuzzy DLR-Lite knowledge base with the SQLf language. The resulting knowledge base is finally applied on a sample mobile application in the field of multimodal transportation. The developed application is embedded in a lightweight terminal in which the accessible data are limited to a subset of the knowledge base. This restriction is due to the limited storage and processing capabilities of such terminals.

The remainder of the paper is structured as follows. Section II introduces the fuzzy DLR-Lite knowledge base. In section III, the Gödel fuzzy implication is used to define an interpretation to the gradual inclusion axiom; its contrapositive is also used to define a restricted form of the negation operator. We also show in section III the way in which our interpretation is extended to a tree of gradual inclusion axioms and its degree propagation property. Section IV introduces briefly the SQLf language and points out the relevance of the enrichment of the fuzzy DLR-Lite knowledge base with this language. Our sample application is described in section V. Section VI recalls our contributions and draws some lines for future works.

II. The Fuzzy DLR-Lite Ontology

In this section, we introduce some elements of both the fuzzy sets theory and the fuzzy DLR-Lite knowledge base.

A. The Fuzzy Sets Theory

The fuzzy sets theory is introduced by Zadeh [16] to express the gradual membership of an element to a set. Formally, a fuzzy set F is defined on a referential U by a membership function $\mu_F : U \mapsto [0,1]$ such that $\mu_F(x)$ denotes the membership grade of x in F.

In particular, $\mu_F(x) = 1$ denotes the full membership of x in F, $\mu_F(x) = 0$ expresses the absolute non-membership and when $0 < \mu_F(x) < 1$, it reflects a partial membership (the more $\mu_F(x)$ is close to 1, the more x is a member of F).

A fuzzy set generalizes a crisp set in which membership grades are in $\{0,1\}$. If a fuzzy set is a discrete set then it is denoted by $F = \{(x_1, \mu_F(x_1)), ..., (x_n, \mu_F(x_n))\}$, otherwise, it is characterized by its membership function.

The union $\cup$ and the intersection $\cap$ operators are defined with a couple of (t-norm, t-conorm) such as (min, max). Let F, G be two fuzzy sets; $\mu_{F\cup G}(x) = max(\mu_F(x), \mu_G(x))$, $\mu_{F\cap G}(x) = min(\mu_F(x), \mu_G(x))$ and the complement of F, noted F^c, is defined by $\mu_{F^c}(x) = 1 - \mu_F(x)$. The logical counterparts of $\cap$, $\cup$ and the complement are resp. $\wedge, \vee$ and $\neg$. Other operators are also defined such as fuzzy implications.

B. The Fuzzy DLR-Lite Knowledge Base

The fuzzy DLR-Lite knowledge base $\mathscr{K}$ consists of three components $(\mathscr{O}, \mathscr{F}, \mathscr{A})$ defined in what follows.

1) The Facts Component $\mathscr{F}$: It contains the extensions of relations made of tuples of the form $R(c_1, ..., c_n)[s]$, where R is an n-ary relation, $c_{i,i=1,...,n}$ are constants and $s \in]0,1]$ is a grade indicating in which extent the tuple $\langle c_1, ..., c_n \rangle$ is an instance of the relation R.

2) The Ontology Component $\mathscr{O}$: The ontology defines the concepts of the application domain by means of projection, intersection and inclusion axioms:

- *The gradual inclusion (subsumption) axiom* has the form $(R_1 \sqcap R_2 \sqcap ... \sqcap R_l \sqsubseteq R_r)[s]$, where $l \geq 1$ and all the involved relations are of the same arity and $s \in]0,1]$ is the truth degree of the axiom.
- *The projection axioms* allow to define new concepts and their attached facts by extracting one or more columns from other relations. Two kinds of projection axioms are defined in the ontology component: *the simple projection* and *the restricted projection*.
  The simple projection is of the form $\exists[i_1, ..., i_k]R$ that expresses the projection of the relation R on columns $i_1, ..., i_k$, and *the restricted projection* is of the form $\exists[i_1, ..., i_k]R.(Cond_1 \sqcap ... \sqcap Cond_h)$ which restricts the projection of $\exists[i_1, ..., i_k]R$ according to the boolean conditions $Cond_{j,j=1,...,h}$ of the form $([i]\theta v)$, where $[i]$ corresponds to a column number of the relation R, $\theta \in \{<, \leq, =, \neq, >, \geq\}$ and v is a value.

3) The Abstraction Component $\mathscr{A}$: It forms the set of statements that connect concepts and relations, defined in the ontology component, to physical relational tables defined in a database. Simple and complex abstraction statements are defined as follows:

- *Simple abstraction statement* has the form:
  $R_1 \longmapsto R_2(c_1[t_1], ..., c_n[t_n])[c_s]$ which states that the n-ary relation R_1 of the ontology component is mapped into the projection on columns $c_1, ..., c_n$ of the m-ary table R_2 ($n \leq m$) each of which is of type t_i. The score of the delivered tuples is provided by the column c_s.
- *Complex abstraction statement* has the form:
  $R_1 \longmapsto (t_1, ..., t_n)[c_s].sql$ where R_1 is an n-ary relation of the ontology component and sql is an SQL query which returns the n-ary tuples $\langle t_1, ..., t_n \rangle$ ranked in decreasing order of their scores c_s.

C. Interpretation of the Ontology Axioms $\mathscr{O}$

An interpretation of the fuzzy DLR-Lite ontology, denoted $\mathscr{I} =< \Delta, \cdot^{\mathscr{I}} >$, consists of a finite interpretation domain Δ and of an interpretation function $.^{\mathscr{I}}$ which associates for each relation R (of arity m) an interpretation $R^{\mathscr{I}} : \Delta^m \rightarrow [0,1]$ where $R^{\mathscr{I}}(c)$ denotes the grade of membership of the tuple c to the relation R.

It is worth noticing that the interpretation function $R^{\mathscr{I}}$ has the same meaning as the membership function μ_R as defined in the fuzzy sets theory. Therefore, both $R^{\mathscr{I}}(x)$ and $\mu_R(x)$ notations are identical.

For sake of clarity, we only introduce the interpretation of the gradual inclusion axiom; see [14] for more details about the interpretation of simple and restricted projection axioms.

The interpretation of the gradual inclusion axiom is defined as follows:

$$\mathscr{I} \models (R_1 \sqcap ... \sqcap R_l \sqsubseteq R_r)[n] \Leftrightarrow \forall c \in \Delta^m, n \otimes R_1^{\mathscr{I}}(c) \otimes \ldots \otimes R_l^{\mathscr{I}}(c) \leq R_r^{\mathscr{I}}(c), \quad (1)$$

where m is the arity of the relations $R_{i,i=1,...,m}$ and R_r and $\otimes$ is a t-norm.

The formula (1) states that if c is an instance of $R_{i,i=1,...,l}$ at the degree $s_i = R_i^{\mathscr{I}}(c)$ then c is an instance of R_r at the minimal degree $n \otimes s_1 \otimes ... \otimes s_l$.

D. The Query Language

A fuzzy DLR-Lite query consists in a conjunction query combined with a scoring function to rank answers. The query is of the following form:

$$\underbrace{q(x)[s]}_{head} \leftarrow \underbrace{\exists y R_1(z_1)[s_1], ..., R_l(z_l)[s_l],}_{body} \underbrace{OrderBy(s = f(s_1, ..., s_l)),}_{scoring\, function}$$

where q is a k-ary relation, each R_i is a relation, x (resp. y) is a vector of distinguished (resp. non distinguished) variables, z_i is a tuple of constants or variables from x or y, f is a scoring function which combines the scores $s_1,...,s_l$ and s is the degree of satisfaction of a tuple x to the query q.

Such a query is evaluated using the gradual inclusion axioms defined in the ontology component. Each query atom $(R_i(z_i)[s_i])$ appearing in the right-hand side of an inclusion axiom is substituted by its left-hand side. Each axiom substitution generates a new query which is added to a disjunctive query. Each subquery of the resulting disjunctive query is finally translated into an SQL query to be evaluated over the facts component.

Example 1. Let *Journey* be a relational table of journeys defined from a multimodal transportation system. We define the fuzzy relation *FastJourney* from the *Journey* relation and the following inclusion (subsumption) axiom:

$(FastJourney \sqsubseteq FavoriteJourney)[0.7]$,

which states that each *fast* journey is also a *favorite* journey at the minimal degree of 0.7.

Any query involving *favorite* journeys, such as:

$q(t)[s] \leftarrow FavoriteJourney(t), OrderBy(s)$

is rewritten, based on the above gradual inclusion axiom, in terms of *fast* journeys (the t-norm used is *min*) as follows:

$q'(t)[s] \leftarrow FastJourney(t)[s'], OrderBy(s = min(0.7, s'))$,

where $s' = fast^{\mathscr{I}}(t)$ is the grade which expresses in what extent the journey t is *fast*.

We obtain for each journey t a minimal grade of satisfaction to the query q equals to $min(0.7, fast^{\mathscr{I}}(t))$ since the interpretation of the gradual inclusion axiom means:

$\forall t, Favorite^{\mathscr{I}}(t) \geq min(0.7, fast^{\mathscr{I}}(t))$. ▪

III. The Gradual Inclusion Axiom and the Gödel Fuzzy Implication

In this section, we show that the interpretation of the gradual inclusion axioms of the fuzzy DLR-Lite ontology is based on the Gödel fuzzy implication. We also show the interest of its contrapositive for the definition of a restricted form of the negation operator. We finally generalize the Gödel fuzzy implication based interpretation to a tree of gradual inclusion axioms.

A. Gödel Fuzzy Implication Based Interpretation of the Gradual Inclusion (Subsumption) Axiom

Let $(A \sqsubseteq B)[\alpha]$ be a gradual inclusion axiom, denoted by $A \sqsubseteq^{\alpha} B$. It is interpreted in the framework of the fuzzy DLR-Lite [14] as follows:

$$\forall x \in \Delta, B^{\mathscr{I}}(x) \geq min(\alpha, A^{\mathscr{I}}(x)) \quad (2)$$

We can prove that this interpretation is based on a gradual inclusion defined by the Gödel fuzzy implication ($\mu_A(x) \Rightarrow_{G\ddot{o}} \mu_B(x) = 1 \ \ if \ \ \mu_A(x) \leq \mu_B(x); \mu_B(x), otherwise$):

$$A \sqsubseteq^{\alpha}_{G\ddot{o}} B \Leftrightarrow min_x(\mu_A(x) \Rightarrow_{G\ddot{o}} \mu_B(x)) = \alpha \quad (3)$$

In the remainder of the paper, we use equally the notations $A \sqsubseteq^{\alpha} B$ and $A \sqsubseteq^{\alpha}_{G\ddot{o}} B$.

Proof: The Gödel fuzzy implication is an R-implication defined by:

$\mu_A(x) \Rightarrow_{G\ddot{o}} \mu_B(x) = \sup_{[0,1]}\{u | min(\mu_A(x), u) \leq \mu_B(x)\}$.

We denote $(\mu_A(x) \Rightarrow_{G\ddot{o}} \mu_B(x))$ by $I(\mu_A(x), \mu_B(x))$.

As $I(\mu_A(x), \mu_B(x))$ is a u value then:

$$\forall x, min(\mu_A(x), I(\mu_A(x), \mu_B(x))) \leq \mu_B(x) \quad (4)$$

having $\alpha = min_x(I(\mu_A(x), \mu_B(x)))$ then

$\forall x, \alpha \leq I(\mu_A(x), \mu_B(x)) \Rightarrow$

$\forall x, min(\mu_A(x), \alpha) \leq min(\mu_A(x), I(\mu_A(x), \mu_B(x)))$.

From the formula (4), we obtain: $\forall x, min(\mu_A(x), \alpha) \leq \mu_B(x)$. ■

This interpretation, based on the Gödel fuzzy implication, allows the definition of a minimal threshold of satisfaction to queries in which substitutions of the gradual inclusion axioms are performed.

In addition, the Gödel fuzzy implication provides effective means of computing the minimal grade of the gradual inclusion axioms (formula (1)).

B. The Gradual Inclusion Axiom and the Contrapositive of the Gödel Fuzzy Implication

The contrapositive of the Gödel fuzzy implication could be used to define a new type of fuzzy inclusion (subsumption) denoted $\sqsubseteq_{ContG}$:

$$A \sqsubseteq^{\beta}_{ContG} B \Leftrightarrow min_x(1 - \mu_B(x) \Rightarrow_{G\ddot{o}} 1 - \mu_A(x)) = \beta \quad (5)$$

The interpretation of this form of fuzzy inclusion axioms is deduced from the interpretation based on the Gödel fuzzy implication as follows:

$$A \sqsubseteq^{\beta}_{ContG} B \Rightarrow \forall x, \neg A^{\mathscr{I}}(x) \geq min(\beta, \neg B^{\mathscr{I}}(x)) \quad (6)$$

The proof of this interpretation is similar to the previous proof. This form of gradual inclusion axioms allows to express a restricted negation operator.

Example 2. Let *TouristJourney* $\sqsubseteq^{0.6}_{ContG}$ *FavoriteJourney* be a fuzzy inclusion (subsumption) axiom which means that the fuzzy set of *touristic* journeys is included in the fuzzy set of *favorite* journeys at the minimal degree of 0.6. In the case that the access is limited to *favorite* journeys, the system can answer queries about journeys which aren't *touristic*, based on the above fuzzy inclusion axiom, as follows:

$\neg TouristJourney(t) \geq min(1 - FavoriteJourney(t), 0.6)$ (formula (5)). ▪

Please notice that this negation is only applicable to relations involved in gradual inclusion axioms.

C. Interpretation of a Tree of Gradual Inclusion Axioms

The proposed interpretation of the gradual inclusion axiom is generalized in the case of a tree of inclusion axioms. This tree structure and its property of degree propagation are the basis of the developed application in section 5.

Without loss of generality, we restrict the tree structure to two levels of inclusion as depicted in Fig. 1a.

Such a tree structure means: $A \sqsubseteq^{\alpha}_{G\ddot{o}} B$ and $B \sqsubseteq^{\beta}_{G\ddot{o}} C$, where $\alpha, \beta \in]0,1]$.

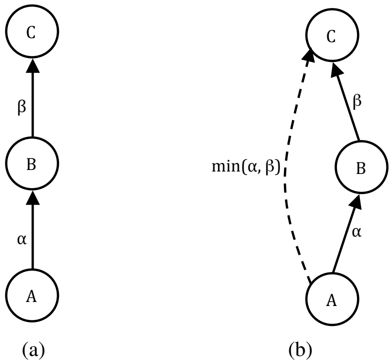

Fig. 1: A tree structure with two levels.

By applying the formula (2), we obtain:

$$\mu_B(x) \geq min(\mu_A(x), \alpha) \tag{7}$$

$$\mu_C(x) \geq min(\mu_B(x), \beta) \tag{8}$$

The propagation property allows the deduction of the grade of inclusion of the concept A in the concept C (see Fig. 1b).

Proof: From the formula (7), we deduce:

$$\mu_B(x) \geq min(\mu_A(x), \alpha) \Rightarrow$$
$$min(\beta, \mu_B(x)) \geq min(\beta, min(\mu_A(x), \alpha)) =$$
$$min(\alpha, \beta, \mu_A(x)) = min(min(\alpha, \beta), \mu_A(x)).$$

$$\text{From} \begin{cases} min(\beta, \mu_B(x)) \geq min(min(\alpha, \beta), \mu_A(x)) \\ \mu_C(x) \geq min(\mu_B(x), \beta)\ (cf.\ formula\, 8) \end{cases}$$
$$\Rightarrow \mu_C(x) \geq min(min(\alpha, \beta), \mu_A(x)).$$

■

This property of degree propagation can be generalized to a series of gradual inclusion axioms of the form:

$$A_1 \sqsubseteq_{Gö}^{\alpha_1} A_2 \sqsubseteq_{Gö}^{\alpha_2} \dots \sqsubseteq_{Gö}^{\alpha_{n-1}} A_n.$$

Effectively, transitive relationships of the form: $A_i \sqsubseteq_{Gö}^{\alpha} A_j$, where $i < j$ can be interpreted as follows:

$$\forall x : \mu_{A_j}(x) \geq min(\mu_{A_i}(x), \alpha),$$
$$\text{where } \alpha = min(\alpha_i, \alpha_{i+1}, ..., \alpha_{j-1}).$$

IV. The Enrichment of the Fuzzy DLR-Lite With the SQLf Language

The SQLf language [4], [3] is an extension of the SQL language to flexible querying by extending its statements to fuzzy conditions. An SQLf query delivers a fuzzy relation r which is a fuzzy set in which the grade of membership of tuple t to the relation r, denoted $\mu_r(t)$, expresses its level of satisfaction. The closer to 1 $\mu_r(t)$, the more t is preferred.

The SQLf language is based on the extension of the relational algebra operators to fuzzy conditions. The basic form of an SQLf query (fuzzy restriction) is as follows:

Select [distinct] *[n|t|n,t] attributes* ***From*** *relations*
Where *fuzzy_cond;*

The above query returns a set of ranked tuples with their grades of satisfaction, where n is a limitation of the size of the returned set and $t \in]0,1]$ is a minimal threshold of satisfaction.

Furthermore, the SQLf language allows more complex statements such as partitioning, fuzzy nesting, division involving fuzzy relations and queries based on fuzzy quantified propositions. These complex statements haven't any boolean counterpart [3].

The limitation of the fuzzy DLR-Lite lies in its inability to express complex preferences. This is due to the fact that abstraction statements, based on SQL queries, do not allow the user to define complex concepts such as *"journeys that satisfy almost all important criteria"*, which is based on a fuzzy quantified proposition.

In order to improve the expressiveness of the fuzzy DLR-Lite knowledge base, we propose the extension of the complex abstraction statements to the SQLf language. So that, it becomes possible to express preferences using complex SQLf statements. We redefine, then, the extended complex abstraction statements as follows:

$R_1 \mapsto (t_1, ..., t_k)[c_s].sqlf$, where R_1 is a concept from the ontology and *sqlf* is an SQLf query which delivers a fuzzy relation.

Example 3. The extended complex abstraction statements allow to define concepts and relations corresponding to complex preferences, such as *MyFastJourney* that corresponds to *"journeys having most of steps are fast"*, which is a fuzzy relation that can be defined by the following extended abstraction statement using an SQLf statement:

MyFastJourneys ↦ (int)[c_s].(***Select*** *#Journey* ***From*** *Journey* ***Group By*** *#Journey* ***Having*** *most (#step) are fast(durLink)*); ▪

From the semantic point of view, our extension fits in well with the fuzzy DLR-Lite knowledge base and ensures backward compatibility, since the SQLf language covers the SQL language and the fuzzy sets theory is the common formal framework of interpretation of both SQLf queries and axioms of the fuzzy DLR-Lite, which is based on the Gödel fuzzy implication and the t-norm *min*.

V. Illustrative example

In this section, a sample application is developed in the field of multimodal transportation, in order to show the interests of our approach, and to emphasize the benefit gained from the interpretation of the gradual inclusion axiom, using the Gödel fuzzy implication and the enrichment of the knowledge base with the SQLf language. The main features and the context of the application are described in following subsections.

A. The Context of the application

The considered application is developed in the context of mobile applications. The user terminal is a lightweight terminal which is limited in terms of main storage and processing capabilities. Therefore, the user terminal only embeds a subset of the knowledge base that consists of (i) fuzzy relations defined by the abstraction statements, (ii) fuzzy predicates and (iii) gradual inclusion axioms.

The user can query the embedded application with conjunctive queries which will be evaluated over the available data, based on the gradual inclusion axioms.

We assume that a user interface is provided to define membership functions corresponding to fuzzy concepts. The embedded system can deduce, from the gradual inclusion axioms, the minimal grade of inclusion of the already defined fuzzy concepts in the user-defined fuzzy concepts.

B. An Embedded Multimodal Transport Information System

We design a mobile application in the field of multimodal transportation. The conceptual data model of the system is consistent with the Transmodel standard [6]. Transmodel is an european standard defining a unified conceptual data model of multimodal transport networks which combine bus, tramway, trolley bus and underground networks. Transmodel provides operating data model for network management activities such as timetables, staff, fleet and rolling stock management, real time monitoring and user information circulation.

In practice, travelers are more interested in information about available networks, timetables and services. So, a subset of the database is enough, from the user point of view, to answer his/her queries which are generally about journeys.

For the sake of clarity, the knowledge base of our application is limited to one level of gradual inclusion enriched after that with a user defined fuzzy predicate.

We introduce below the 3 steps for the design of the embedded application: (i) data preprocessing, (ii) querying of the embedded system and (iii) the enrichment of the system.

1) Step 1. Data Preprocessing: Let the following relational tables be a subset of a database from a multimodal transport information system:

Network, Line, Station, Link and *LinkLine*.

We define the *Journey* relation as follows:

Journey (#journey, #link, departure, arrival, source, destination, duration, DepLink, ArrLink, modeLink, durLink, costLink, comfLink, waitLink).

It is assumed that journeys from source s to destination d are preprocessed.

Based on the SQLf queries, the following fuzzy relations are defined from the *Journey* relation:

1) *TouristJourney* $\mapsto$ [#*Journey*]***.Select*** *#Journey* ***From*** *Journey* ***Group By*** *#Journey* ***Having*** *(most #Link are Touristic);*
2) *PleasantJourney* $\mapsto$ [#*Journey*]***.Select*** *#Journey* ***From*** *Journey* ***Group By*** *#Journey* ***Having*** *(most #Link are Short(waitLink));*

The above extended complex abstraction statements state respectively that a *touristic* journey is a journey in which most of its steps (named link in the statements) cut across touristic sites and a *pleasant* journey is a journey in which the waiting time of most of its steps is *short*. Both *touristic* and *short* are fuzzy predicates.

The fuzzy predicate *MorningDeparture* is also defined by the membership function described in Fig. 2.

The following gradual inclusion axioms could be obtained by a processing using the formula (2) over the database:

- TouristJourney $\sqsubseteq_{G\ddot{o}}^{0.7}$ MorningDepartureJourney
- PleasantJourney $\sqsubseteq_{G\ddot{o}}^{0.8}$ MorningDepartureJourney

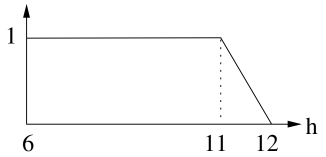

Fig. 2: The membership function of the fuzzy predicate *Morning*.

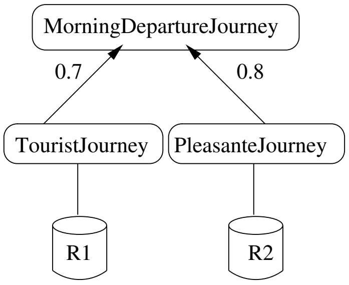

Fig. 3: The embedded concepts and data.

2) Step 2. Querying the Embedded System: The data embedded in the user terminal are illustrated in Fig. 3. The fuzzy relations $R_{i,i=1,2}(\#Journey)$ (in the sense of the SQLf language) are resulting from SQLf queries of the extended complex abstraction statements and mapped respectively to fuzzy concepts (relations in the sense of the fuzzy DLR-Lite) *TouristJourney* et *PleasantJourney*.

Due to the storage limitation of the user terminal, both the relations $R_{i,i=1,2}$ only contain the *#Journey* and the score columns. The membership function of the fuzzy predicate *MorningDeparture* is also embedded.

The user can query the system based on the embedded fuzzy predicates. The grade of satisfaction to both fuzzy conditions *Touristic* and *pleasant* are embedded within the relations $R_{i,i=1,2}$. About the fuzzy relation *MorningDepartureJourney*, a minimal grade of satisfaction can be obtained using the formula (2). For example:

$\forall t, \mu_{Morning}(t) \geq min(\mu_{Touristic}, 0.7) = min(\mu_{R_1}(t), 0.7)$,

and

$\forall t, \mu_{Morning}(t) \geq min(\mu_{Pleasant}(t), 0.8) = min(\mu_{R_2}(t), 0.8)$.

Finally, we can write:

$$\forall t, \mu_{Morning}(t) \geq max(min(\mu_{R_1}(t), 0.7), min(\mu_{R_2}(t), 0.8)).$$

3) Step 3. Enrichment of the Embedded System: The user can define his/her own fuzzy concepts to express his/her preferences. These concepts could be connected to the previously defined tree structure of concepts in order to personalize the embedded system. The query evaluation process is then made more accurate and close to user expectations.

Example 4. Through the fuzzy concept definition interface, the user can define *FavoriteDepartureJourney* as a fuzzy concept based on the favorite departure time (see Fig. 4). *FavoriteDepartureJourney* is then bound to *MorningDepartureJourney* fuzzy concept as follows:
$MorningDepartureJourney \sqsubseteq_{G\ddot{o}}^{\alpha} FavoriteDepartureJourney$.

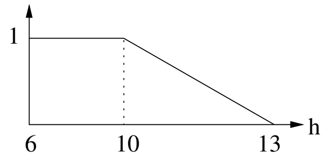

Fig. 4: The membership function of the fuzzy predicate *Favorite*.

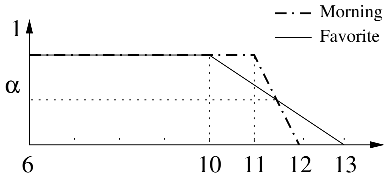

Fig. 5: Computation of the value of α.

The value of α in the above gradual inclusion axiom is $\alpha = min_t(\mu_{Morning}(t) \Rightarrow_{Gö} \mu_{Favorite}(t))$ and can be computed from the membership functions of both *Morning* and *Favorite* fuzzy predicates as depicted in Fig. 5.

Once the value of α is obtained ($\alpha = 0.5$ in this example), the embedded system, by applying the propagation property, deduces the grade of gradual inclusion links that bind the added concept to the existing ones as described in Fig. 6.

Any query which is about the fuzzy predicate *FavoriteDepartureJourney* is evaluated over the available data which covers *pleasant* and *touristic* journeys such that:

$$\forall t, \mu_{Favorite}(t) \geq max(min(0.5, \mu_{R_1}(t)), min(0.5, \mu_{R_2}(t))). \blacksquare$$

Remark. The deduction of the minimal grade of inclusion of a fuzzy concept in a user-defined fuzzy concept is based on both their respective membership function and the formula:

$$A \sqsubseteq^{\alpha}_{Gö} B \Leftrightarrow min_x(\mu_A(x) \Rightarrow_{Gö} \mu_B(x)) = \alpha.$$

This computation is only significant if the concept B generalizes the concept A (in the same way as concept hierarchies in ontologies) which means that $Support(A) \subseteq Support(B)$. Indeed, if $Support(A) \nsubseteq Support(B)$ then the grade of inclusion is equal to 0; which is in fact worthless.

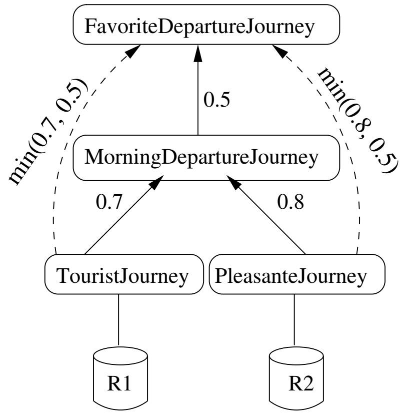

Fig. 6: Enrichment of the system.

VI. Conclusion and future works

The flexible querying approach developed in this paper is based on the combination of a fuzzy description logic (denoted fuzzy DLR-Lite) and the SQLf language that allows the expression of more complex fuzzy relations than those could be expressed by the SQL language. An interpretation of the gradual inclusion axiom based on the Gödel fuzzy implication is introduced. This interpretation provides a formal framework in which a restricted form of the negation operator is defined and a generalization to a tree of fuzzy relations is also proposed.

An implementation, based on algorithms for query evaluation introduced in [14], is in progress. It remains the development of an effective algorithm to compute the minimum grade of inclusion of fuzzy concepts added to the knowledge base. We also plan to extend the enrichment process to complex fuzzy predicates as those defined by division involving fuzzy relations.

Acknowledgments

This work is funded by the Brittany region and the department of Côtes-d'Armor and is partially supported by the National Agency of Research (AOC Ref. ANR-08-CORD- 009).

References

[1] F. Baader, D. Calvanese, D. L. M. Guinness, D. Nardi, and P. F. Patel-Schneider, editors. *The Description Logic Handbook : Theory, Implementation and Applications*. Cambridge University Press, 2003.

[2] F. Bobillo and U. Straccia. fuzzyDL : An expressive fuzzy description logic reasoner. In *IEEE International Conference on Fuzzy Systems*, pages 923–930, 2008.

[3] P. Bosc, L. Liétard, O. Pivert, and D. Rocacher. *Base de données - Gradualité et imprécision dans les bases de données Ensembles flous, requêtes flexibles et interrogation de données mal connues*. Technosup. 1 edition, 2004.

[4] P. Bosc and O. Pivert. SQLf: A relational database langage for fuzzy querying. *IEEE Transactions on Fuzzy Systems*, 3(1):1–17, Feb 1995.

[5] D. Calvanese, G. D. Giacomo, D. Lembo, M. Lenzerini, and R. Rosati. Data complexity of query answering in description logics. In *KR*, pages 260–270, 2006.

[6] O. Carles. Modèle conceptuel de données générique pour la représentation des réseaux intermodaux. Tech. report, INRETS, 2002.

[7] S. Colucci, T. D. Noia, A. Ragone, M. Ruta, U. Straccia, and E. Tinelli. *Semantic Web Information Management*, chapter 19 : Informative Top-k retrieval for advanced skill management, pages 449–476. Springer-Verlag Belin Heidelberg, 2010.

[8] C. Hudelot, J. Atif, and I. Bloch. Fuzzy spatial relation ontology for image interpretation. *Fuzzy Sets and Systems*, 159:1929–1951, 2008.

[9] J. Lu, Y. Li, B. Zhou, and D. Kang. Reasoning within extended fuzzy description logic. *Knowledge-Based Systems*, 22:28–37, 2009.

[10] J. Z. Pan, G. Stamou, G. Stoilos, and E. Thomas. Expressive querying over fuzzy DL-Lite ontologies. In *20th International workshop on description logics, Brixen-Bressanone, Italy*, 2007.

[11] G. Stoilos, G. Stamou, V. Tzouvaras, J. Z. Pan, and I. Horrocks. Fuzzy OWL: Uncertainty and the semantic web. In *International workshop on OWL: Experience and Directions*, 2005.

[12] U. Straccia. Towards a fuzzy description logic for the semantic web (preliminary report). *LNCS*, 3532:167–181, 2005.

[13] U. Straccia. Answering vague queries in fuzzy DL-lite. In *In Proceedings of the 11th. Inter. Conf. on Information Processing and Management of uncertainty in Knowledge-Based Systems*, pages 2238–2245, 2006.

[14] U. Straccia. Softfacts : a top-k retrieval engine for a tractable description logic accessing relational databases. Technical report, ISTI-CNR, 2009.

[15] U. Straccia and G. Visco. DLmedia: an ontology mediated multimedia information retrieval system. In *In Proceedings of DL 2007*, 2007.

[16] L. Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.