Flexible Database Querying Based on Ordered Lattice Theory Extension

Minyar Sassi[1], Amel Touzi[1], and Habib Ounelli[2]
[1]Ecole Nationale d'Ingénieurs de Tunis, Tunisia
[2]Faculté des Sciences de Tunis, Campus Universitaire-1060 Tunis, Tunisia

Abstract: *This research reports on the synthesis of flexible database querying approach based on ordered lattice theory extension to deal with imprecise and structured data. This approach allows us to construct a multi-attributes type abstraction hierarchy structure for the case of decomposition according several attributes. This structure is defined from an ordered lattice theory extension. Our approach consists of two steps: the first step consists in data organization and the second at seeking, to interrogate them, relevant data sources for a given query. The contributions of this approach are a) the interdependence of the query research criteria, b) the research of the relevant data sources for a given query, and d) the scheduling of the results.*

Keywords: *Fuzzy cluster analysis, formal concept analysis, flexible database querying, concept query, relieving query.*

Received March 3, 2007; accepted June 13, 2007

1. Introduction

Scientific context of this article is the intersection of the following fields: "flexible querying" [1], "fuzzy subsets theory" [2] and data analysis methods such as: "formal concept analysis" [3] and "fuzzy clustering" [4].

So, when a user selects data from database, it expresses a strict constraint on the required data. The traditional flexible systems distinguish two data categories: those which satisfy the constraint and those which do not satisfy it.

The principle of the flexible querying aim at extending this binary behaviour by introducing preferences into the query criteria, what leads to qualitatively distinguish answers. Thus, an element turned over by a query will be more or less relevant according to the preferences that it will have satisfied.

Several works were proposed in the literature to introduce the database's flexible querying. The majority of this work used the fuzzy sets formalism to model the linguistic terms such as ("moderated", "means") and, thereafter, evaluate predicates container such terms [5].

In [6], a flexible and cooperative database querying approach within the fuzzy theory framework has been proposed. This approach contributes two promising shares compared to the similar approaches. The first taking into account the dependence's semantic between the query research criteria to determine its realisability or not. The second contribution relate to its cooperative aspect in the flexible querying. To ensure these functionalities, they have proposed to construct mono-attribute Type Abstraction Hierarchy (TAH) and a Multi-attribute Type Abstraction Hierarchy (MTAH). Theses structures are proposed initially in [7]. Theses structures are used for dependence's extraction from relieving attributes. Problems lie in:

- The generation of TAH's and MTAH from relieving attributes.
- The storage and the indexing of such structures.
- The update of HATM.

To cure theses problems, we propose a database flexible querying approach which contributes two principal shares compared to the others flexible querying approaches, in particular that in [6]. Theses contributions are:

- The automatic generation of TAH's and MTAH structures from relieving attributes.
- The research of relevant data sources for a given query.
- A detection of the query unrealisability.
- The scheduling of the results.

For satisfying the first contribution, we propose a new approach allowing combining two data analysis methods such as fuzzy clustering and Formal Concept Analysis (FCA) based on ordered lattice theory extension.

Of its part, fuzzy clustering has been a very successful technique of data analysis, as demonstrated by many successful applications in different domains [4]. Theses techniques allow objects of a data set to belong to several clusters simultaneously, with different membership degrees.

Despite being a very effective technique, difficulties arise when interpreting mutual relationships between specific clusters.

To cure this problem, we propose a new method based on ordered lattice theory extension for 1) structuring data, 2) generation of TAH's and TAHM, 3) extracting data dependencies, and 4) updating of TAH's and MTAH.

FCA is a method for data analysis, information management and knowledge representation that takes advantage of the features of formal concepts. The core of FCA is concept lattice. Theoretical foundation of concept lattice founds on the mathematical lattice theory [8]. Lattice is a popular mathematical structure for modelling conceptual hierarchies.

The concern always remains, after having stored theses structures, to be able to interrogate the database and to have relevant answers to the queries. Indeed, when it carries out a search for information in a database, the user is very often submerged by the mass of turned answers. To cure this problem, we propose formalism for relevant data sources research of for a given query. Relevant data sources can be sorted according to the distance separating the concepts in the lattice.

The rest of the paper is organised as follows. Section 2 makes a review of some data analysis methods and database flexible querying approaches closer to our approach. Section 3 presents problems and contributions. Section 4 describes our database flexible querying approach. Section 5 evaluates this approach. Finally, section 6 concludes the paper and gives some future works.

2. Backgrounds

2.1. Data Analysis Methods

Cluster analysis and FCA are well-known graphical methods in data analysis [8]. For its part, cluster analysis looses some information on the way from the original data to the graphical output. FCA may represent the original data without loss of information in the plane.

2.1.1. Cluster Analysis

The objective of cluster analysis [4] is the classification of objects according to similarities among them, and organizing of data into groups. Clustering techniques are among the unsupervised methods, they do not use prior class identifiers. The main potential of clustering is to detect the underlying structure in data for model reduction and optimization. Since groups (clusters) can formally be seen as subsets of the data set, one possible classification of clustering methods can be according to whether the subsets are fuzzy or crisp.

The goal of hard clustering is to assign each data point to one and only one cluster. In real applications, there is very often no sharp boundary between clusters so that fuzzy clustering is often better suited for the data. It assigns different degrees of membership to each point. The membership of a point is thus shared among various clusters.

Several researches were carried out for the automatic determination of the number of clusters [9] and the quality evaluation [10] of the obtained partitions. They are based on the definition of an objective function [10] making it possible to measure the quality of the obtained partitions. A prominent fuzzy clustering algorithm for determining the optimal number of cluster is presented in [11].

To make it easier for the readers understand the ideas behind clustering technique, we tried to unify the notation used in the rest of the paper.

The following notations are assumed: $X \in R^N$ is the vector of N objects (tuples) x_j for the attribute A_i, $C_k(A_i)$ is the k^{th} cluster of attribute A_i and $C(A_i)$ is the optimal number of clusters found for the attribute A_i. Fuzzy clustering methods allow objects to belong to several clusters simultaneously, with different degrees of membership. The data set X is thus partitioned into C fuzzy subsets [10]. The structure of the partition matrix $U = [\mu_{ik}]$.

Despite being a very effective technique, fuzzy clustering presents some difficulties when interpreting fuzzy clustering results. In the case of large samples, the large number of membership values with respect to the constructed clusters makes it almost impossible to effectively compare the fuzzy properties of the objects. In this case, the mutual relationships between specific clusters can be masked. In addition, the relationships between data structure and fuzzy clustering results are difficult to understand.

2.1.2. Ordered Lattice Theory

Concept is also important for data analysis in computer science. FCA is a graphical method for data analysis, information management and knowledge representation that takes advantage of the features of formal concepts.

The core of FCA is concept lattice. Theoretical foundation of concept lattice founds on the mathematical lattice theory [12, 13].

The application of concept lattice has been an area of active and promising research in various fields such as knowledge discovery, information retrieval, software engineer and machine learning.

Concept lattice describes the character of the set pair: intent and extent of concept.

This understanding has been formalized by a formal context, $\mathbf{K}$ defined by a triple (G, M, I) where G and M are sets and I is a binary relation between G and M (i.e., $I \subseteq G \times M$). The elements of G are called objects or transactions, while the elements of M are called attributes or items.

The formal concepts of $\mathbf{K}$ are the pair (A, B) with $A \subseteq G$ and $B \subseteq M$ such that (A, B) is maximal with respect to the property $A \times B \subseteq I$. The set A is called the extent and B is called the intend of the concept (A, B) [13].

The set C of all concepts of a context $\mathbf{K}$ with order $(A_1, B_1) \leq (A_2, B_2) :\Leftrightarrow A_1 \subseteq A_2$ is always a complete lattice, and is called the concept lattice of the context $\mathbf{K}$, noted in the rest as $\Im(C)$.

A data context is usually represented by the binary data, but in practice, the values of attribute are not binary, we can transform many-valued data context to binary values context by concept scaling [14].

Such data is formalized as a many valued context (G, M, W, I). G is a set of objects, M is a set of attributes, W is a set of attribute values, and $I \subseteq G \times M \times W$ is a relation such that $(g_1, m_1, w_1) \in I$ and $(g_2, m_2, w_2) \in I$ implies that $w_1 = w_2$.

Attributes m_i are understood as partial functions from G into W . A formal context in which there are no attribute values is often called a single-valued context. Many-valued contexts can be mapped into formal contexts using conceptual scales [14].

A conceptual scale for a set $Y \subseteq M$ is a single-valued context $S_Y := (G_Y, M_Y, I_Y)$ with $G_Y \subseteq \times_{m \in Y} W_m$. The idea is to replace the attribute values in W_m which are often too specific by more general attributes which are provided in M_Y .

The concept lattice of the derived context can be visualized in a nested line diagram. In nested line diagrams, the nodes of the concept lattice of the first scale are enlarged, so that the concept lattice of the second scale can be drawn inside. The second lattice is then used to further differentiate each of the extents of the concepts of the first lattice.

Though lattice-based information representation has the advantage of providing efficient visual interface over textual display, the complexity of a lattice may grow rapidly with the size of the database.

2.2. Principals Flexible Querying Approaches

In this section, we present the essential idea of the principal flexible querying approaches closest to our approach. These approaches differ primarily by the used procedure to find the set of the similar values to those presented by the user. The second difference is by the formalism which is used to model real world imperfection. Relieving in CoBase system [15] bases itself on the data field's decomposition by an automatic regrouping of elements which share a set of common characteristics. The operation of data regrouping is assured by the automatic construction of a TAH for an attribute and of a MTAH in the case of decomposition according to several attributes. MTAH structure allows representing the database fields on several abstraction levels. The root of the hierarchy constitutes the most abstract representation. CoBase used this data organization to carry out the query relieving. Relieving consists in extending the research space by the other values of the field containing this space or by the values of the field obtained after a succession of generalizations and specializations.

In [6], a relieving approach within the fuzzy set framework was proposed. This approach contributes two promising shares compared to the preceding approaches. The first contribution is the taking into account the semantic dependences between the query search criteria to determine its realisability or not. The second contribution relates to its co-operative aspect in the flexible querying. For the dependencies extraction, this approach consists in building TAH's and MTAH from relieving attributes. The problem here lies in storage, indexing of such structures and the incremental update of these structures.

3. Problems and Contributions

Despite being a very effective technique, fuzzy clustering presents some difficulties when interpreting fuzzy clustering results. In this case, the mutual relationships between specific clusters can be masked and difficult to understand.

In the same way, despite its mathematical foundations, one limit of using lattice is the lattice complexity (number of concepts).

To cure this problem, we propose to combine FCA data analysis method based on lattice theory with fuzzy cluster analysis method for:

- The reduction of the lattice complexity by minimizing the number of concepts using clustering technique.
- A cluster's hierarchy generation thus allowing the interpretation and the extraction of the dependences.

However, there are many situations in which uncertainty information also occurs. Traditional ordered lattice theory is hardly able to represent such vague information. To tackle this problem, we propose to combine fuzzy logic with the ordered lattice theory in which uncertainty information is directly represented by a real number of membership value in the range of [0,1]. As such, linguistic variables are no longer needed. In comparison with the fuzzy ordered lattice generated from the l-fuzzy [18-16] context, the fuzzy concept lattice generated using the ordered lattice extension will be simpler in terms of the number of formal concepts, and it also supports a formal mechanism for calculating concept similarities. Therefore, the proposed ordered lattice extension is a suitable representation for data structure.

This structure is applied in our approach to exceed the problems of proposed approaches such as: 1) the storage, indexing and the update of TAH's and MTAH, 2) the research of the relevant data sources for a given query and, 3) the scheduling of the results.

4. Proposed Approach

We propose in this section a relieving approach within the fuzzy set framework [2]. We consider a relational database containing relieving attributes i.e., attributes which the users can use in a predicate of comparison containing a linguistic term. In this paper, we limit ourselves to the relieving numerical attributes.

Figure 1 shows the proposed approach. It consists of two steps: the first step consists in data organisation and the second aims at seeking, to interrogate them, relevant data sources for a given query.

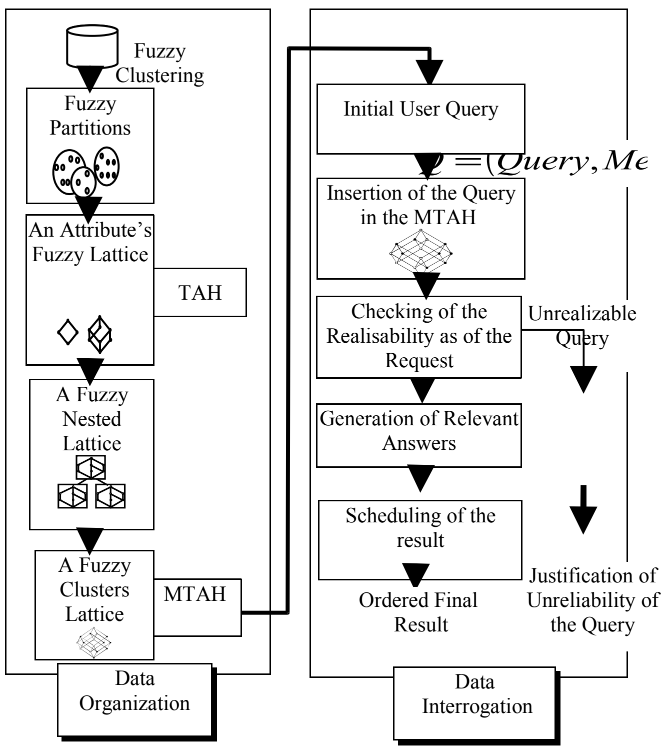


Figure 1. The proposed approach.

4.1. TAH's Generation

We have mentioned in section 2.1.1 that fuzzy clustering methods allow objects to belong to several clusters simultaneously, with different degrees of membership. The data set X is thus partitioned into C fuzzy partitions (clusters).

In many applications, training data relates individual objects to attributes that take on several values. For the generation of fuzzy formal context, we propose to relate objects with the clusters of each attribute that take on several values. These values represent the membership degrees of each object in each cluster. Fuzzy formal context incorporate fuzzy clustering, to represent vague information. We present the basic definitions used in our theoretical foundations of our approach.

Definition 1: a fuzzy formal context is a triplet $K = (G, M, I = \varphi(G \times M))$ where G is a set of objects, M is a set of clusters, and I is a fuzzy set on domain $G \times M$. Each relation $(g, m) \in I$ has a membership value $\mu(g, m)$ in $[0,1]$.

However, data is formalized as a many-valued context (G, M, W, I), where G is a set of objects, M is a set of attributes, W is a set of attribute values, and $I \subseteq G \times M \times W$.

Definition 2: A fuzzy conceptual scale for a set $Y \subseteq M$ is a (single-valued) fuzzy formal context $S_Y := (G_Y, M_Y, I_Y = \varphi(G_Y \times M_Y))$ with $G_Y \subseteq \times_{m \in Y} W_m$.

The idea is to allow objects G to belong to several clusters simultaneously. We replace the attribute values in W_m with different degrees of membership. Each relation $(g, m) \in I_Y$ has a membership value $\mu(g, m)$ in [٠,١]. The sum of the values of each fuzzy conceptual scale is equal to 1.

Example: let a relational database describing

Announce (<u>réfAn</u>, date_an, codPr, codAp)
Apartment (<u>codAp</u>, price, state, site, surface, num_ room, city)
Owner (<u>codPr</u>, name, surname, num_phone, address)

apartment announces, where the primary key of each relation is underlined:

Table 1 shows the fuzzy conceptual scales for Price and surface attributes.

Table 1. Fuzzy conceptual scales for price and surface attributes.

	Price			Surface	
	C1	C2	C3	C4	C5
A1	0.1	0.5	0.4	0.5	0.5
A2	0.3	0.6	0.1	0.4	0.6
A3	0.7	0.1	0.2	0.7	0.3
A4	0.1	0.4	0.5	0.2	0.8
A5	0.2	0.4	0.4	0.6	0.4
A6	0.5	0.3	0.2	0.5	0.5

Definition 3: given a fuzzy conceptual scale $S_Y := (G_Y, M_Y, I_Y = \varphi(G_Y \times M_Y))$, we define $\alpha - Cut(A_i) = (C(A_i))^{-1}$ where $C(A_i)$ the number of clusters of scale is A_i.

In our example, $\alpha - Cut(\mathrm{Pr}\,ice) = 0.3$ and $\alpha - Cut(Surface) = 0.5$.

Traditional ordered lattice is hardly able to represent fuzzy properties from uncertainly data. To tackle this problem, we use a new technique that incorporates fuzzy logic into the ordered lattice theory in which uncertainty information is directly represented by a real number of membership value in the range of [0,1].

Definition 4: given a fuzzy formal context $K = (G, M, I)$ and an $\alpha - Cut$, we define $\mathbf{X}^* = \{m \in M \mid \forall g \in \mathbf{X}: \mu(g, m) \geq \alpha - Cut\}$ for $\mathbf{X} \subseteq G$ and $\mathbf{Y}^* = \{g \in G \mid \forall m \in \mathbf{Y}: \mu(g, m) \geq \alpha - Cut\}$ for $Y \subseteq M$.

A fuzzy formal concept (or fuzzy concept) of a fuzzy formal context (G, M, I) with an $\alpha - Cut$ is a pair $(\mathrm{X}_f = \varphi(\mathrm{X}), \mathrm{Y})$ where $\mathbf{X} \subseteq G$, $Y \subseteq M$, $\mathrm{X}^* = \mathrm{Y}$ and $\mathrm{Y}^* = \mathrm{X}$. Each object $g \in \varphi(\mathbf{X})$ has a membership μ_g defined as $\mu_g = \min_{m \in Y} \mu(g, m)$.

Where $\mu(g, m)$ is the membership value between object g and attribute m, which is defined in I. Note that if $\mathrm{Y} = \{\ \}$ then $\mu_g = 1$ for every g.

Generally, we can consider the attributes of a formal concept as the description of the concept. Thus, the relationships between the object and the concept should be the intersection of the relationships between the objects and the attributes of the concept. Since each relationship between the object and an attribute is represented as a set of membership values in fuzzy formal context, then the intersection of these membership values should be the minimum of these membership values, according to fuzzy theory.

Definition 5: let (A_1, B_1) and (A_2, B_2) be two fuzzy concepts of a fuzzy formal context $\mathrm{K} = (G, M, I = \varphi(G \times M))$. $(\varphi(A_1), B_1)$ is a the subconcept of $(\varphi(A_2), B_2)$ denoted as $(\varphi(A_1), B_1) \leq (\varphi(A_2), B_2)$ if and only if $\varphi(A_1) \subseteq \varphi(A_2)$ $(\Leftrightarrow B_2 \subseteq B_1)$. Equivalently, (A_2, B_2) is the superconcept of (A_1, B_1).

Definition 6: a fuzzy concept lattice of a fuzzy formal context $\mathbf{K}$ with an $\alpha - Cut$ is a set C of all fuzzy concepts of $\mathbf{K}$ with the partial order $\leq$ with the $\alpha - Cut$ value, noted in the rest as $\Im(C)$.

Definition 7: the similarity of a fuzzy formal concept $C_1 = (\varphi(A_1), B_1)$ and its subconcept $C_2 = (\varphi(A_2), B_2)$ is defined as

$$E(C_1, C_2) = \frac{|\varphi(A_1) \cap \varphi(A_2)|}{|\varphi(A_1) \cup \varphi(A_2)|}.$$

The corresponding fuzzy concept lattices are shown in Figure 3.This very simple sorting procedure gives us for each fuzzy many-valued attribute the distribution of the objects in the fuzzy line diagram of the chosen fuzzy scale. The well-known histograms for one variable arise as special cases from fuzzy line diagrams.

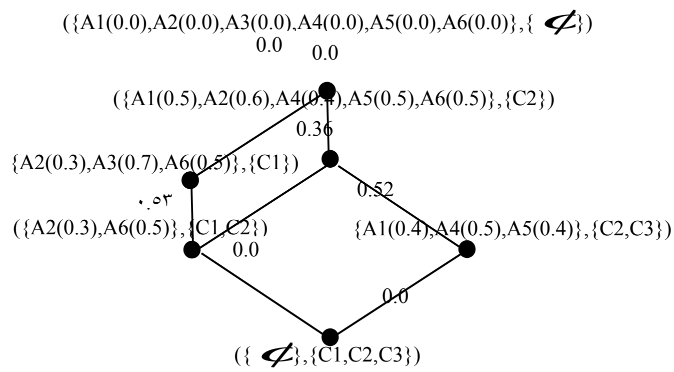


Figure 2. Fuzzy lattices for surface and price fuzzy scale: price TAH and surface TAH.

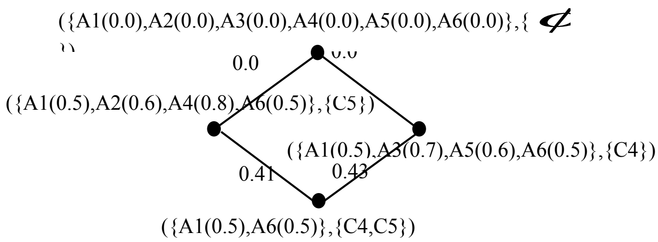


Figure 3. Fuzzy Lattices for Surface fuzzy scale: the Surface TAH.

4.2. MTAH Generation

Usually, we are interested in the interaction between two or more fuzzy many-valued attributes. This interaction can be visualized using the so called fuzzy nested line diagrams. This fuzzy nested line diagrams are an extension of fuzzy concept lattices. They are used for:

- Visualizing larger fuzzy concept lattices.
- Emphasizing sub-structures and regularities.
- Combining fuzzy conceptual scales on-line.

In this fuzzy nested line diagram, we are interested to see for each concepts of diagram represented in Figure 2 how its apartments are distributed in the surface TAH. We blow up each circle of price TAH of Figure 2 and insert the surface TAH presented in Figure 3.

Figure 3 shows the fuzzy nested line diagram constructed from Figure 2.

Hence, Figure 4 represents all pairs (c,d) of concepts c from the first and concepts d from the second fuzzy lattice. This structure is called the direct product of the two given fuzzy lattices.

From the fuzzy nested lattice, we can draw a nice usual fuzzy lattice of the same fuzzy context. This is illustrated by Figure 5.

Figure 5 shows that each concept is represented by a set of memberships generated from apartments belonging to the corresponding fuzzy clusters. Then, hierarchical relation generation is performed to extract dependencies from data.

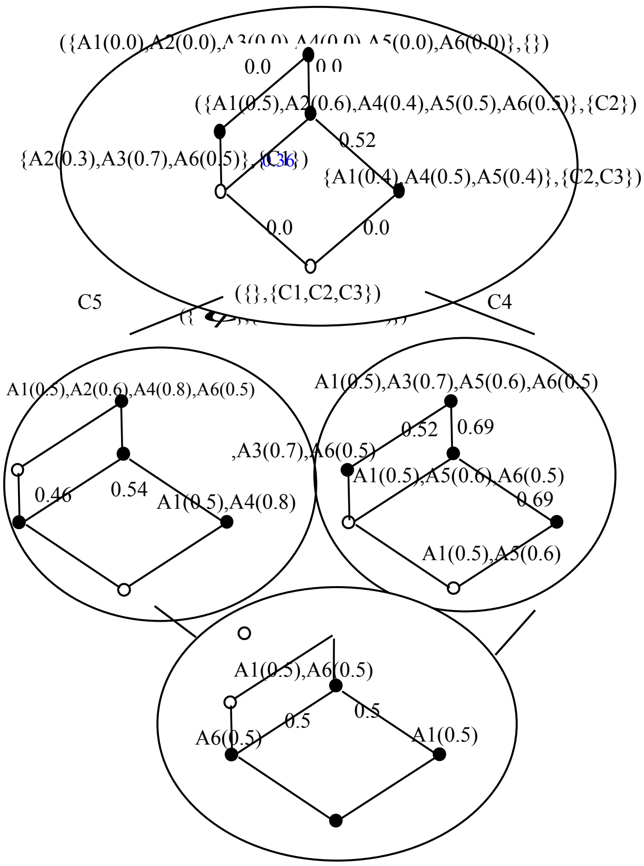
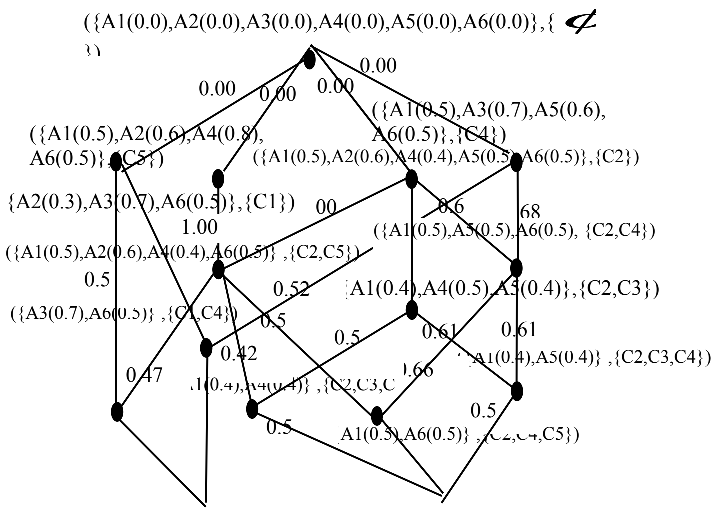


Figure 4. A fuzzy nested lattice.



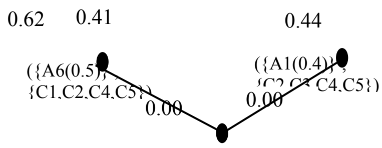


Figure 5. Fuzzy lattice: MTAH.

5. Database Flexible Querying

5.1. Generation of Approximate Answers

Let us consider the following query:

Select refAn, price, surface
From Announce, Apartment
Where price = 105 (A_1)
and surface = 75 (A_2)
and city = 'Paris' (A_3)
and place = '16ème arrondissement' (A_4)
and Apartment.codAp = Announce.codAp (A_5)

This step consists in building lattice to extract the dependences between the user query criteria. Initially, we determine tuples from database satisfying not relieving attributes (A_3, A_4, and A_5).

To improve the data representation, we define the terminals of the intervals of distribution in each cluster generated. With this intention, we must define the minimal and maximum values of each interval. If the case of price attribute, its first cluster contains 3 objects. The minimal value (respectively maximum) in this cluster is that of which the value of price attribute is minimal (respectively maximum).

With these generated intervals, the users can assign labels to each relieving attribute. For example, for the relieving attribute «price », the users can refer it by one of the three following linguistic terms: « weak », « means » and « raised ». They can use it also with an exact value.

Once that the fuzzy lattice $\Im(C)$ is built, the research of the relevant data sources can start. In the same way in [17 and 18], we define a concept query $Q = (Q_A, Q_B)$ where $Q_A = \{Query\}$, i.e. a name for the query extension and Q_B is the set of metadata (clusters) describing the sources sought by the query.

These metadata are given with part of the fuzzy clustering operation to determine the objects membership's degrees in the various clusters. In our case, value 105 for Price attribute and value 75 of surface attribute have the following degrees. Then, we apply the $\alpha-Cut$ for each attribute to minimize the number of concepts. According to our example, the query Q seek the data sources having the metadata $Q_B = \{(C2, C3, C5)\}$.Table 2 present the membership egrees of the query.

Once definite, the concept Q is inserted in the lattice by using the incremental construction algorithm of godin [19]. The lattice obtained is noted $\Im(C \oplus Q)$ where $C \oplus Q$ indicate the new set of concepts resulting from insertion of the query. The lattice $\Im(C \oplus Q)$ corresponds to the example mentioned previously is represented in Figure 5. The concepts surrounded by the dotted ones are either of the new concepts or of the concepts modified following insertion of the query Q in the lattice. These concepts are the only ones which share metadata with the query and which can contain relevant data sources.

Proposition: a data source S is relevant for a given query $Q=(Q_A,Q_B)$ if and only if S is characterized by at least one of the meta-data given from Q_B. The relevance degree of S is given by the number of meta-data that S divide with Q_B.

This proposition of relevance is at the base of the research process which detailed in the rest of this section and illustrated by an example. It is different from the vicinity concept used in [15], which can lead to obtaining the data divide no metadata with the query, what does not correspond to our needs.

Let given a query concept $Q=(Q_A,Q_B)$, all the relevant data sources are in the extension of Q and of its subsumers in the concepts lattice since the intention of each one of these concepts are included in Q_B (the intension of the query concept).

Table 2. Query memberships degrees.

Price			Surface	
C1	C2	C3	C4	C5
0.1	0.3	0.6	0.2	0.8

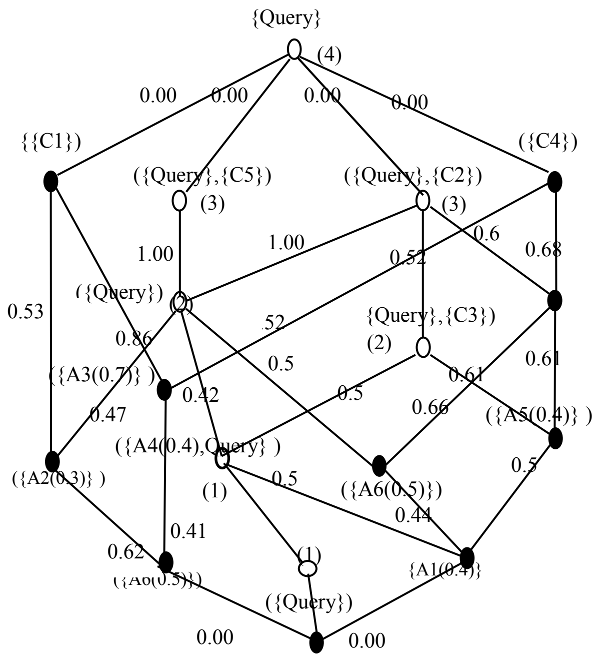


Figure 5. Query insertion in MTAH.

5.2. Checking of the Query Realisability

If the query criteria are in contradiction with their dependences extracted the database, it is known as unrealisable.

Proposition: let a query Q having the concept $Q=(Q_A,Q_B)$. A query Q is unrealisable if and only if $\exists$ data source in Q_A which dividing any metadata of the set Q_A.

5.3. Scheduling of the Result Answers

This step consists in ordering the n-uplets obtained according to their satisfaction degrees of the initial query. To ensure this operation, we note by $R(Q,C)$ the set of relevant data sources for the considered query as a set of formal concepts C of formal context K.

Initially, we seek to insert the query concept in the lattice $\Im(C)$ what produces the lattice $\Im(C \oplus Q)$. Then, the set of data sources appear in the extension of Q in the lattice $\Im(C \oplus Q)$ (if it exists) is inserted in the list $R(Q,C)$.

Let $SUBS(Q,C,١)$ the set of subsmers of Q in $R(Q,C)$. The set of the data sources which appear in the extensions of the concepts of $SUBS(Q,C,١)$ and which is not already in $R(Q,C)$ are added to the result with this stage. The following step consists in determining $SUBS(Q,C,٢)$ the set of the concept's subsmers of $SUBS(Q,C,١)$ (or subsumers of distance 2 of query). In the same way for $SUBS(Q,C,1)$, new data sources in the extensions of the concepts of $SUBS(Q,C,1)$ are added to the result $R(Q,C)$. The same operation is carried out until reaching a unit $SUBS(Q,C,n)$ which is empty (stop condition of the algorithm). With each step, the data sources appear in a concept with empty intension are ignored. The row of a source in $R(Q,C)$ can be memorized according to the distance from the source (or of the first concept in which the source appears) at the query in $\Im(C \oplus Q)$.

In Figure 5, numbers located meadows of concepts the iterations of the algorithm explained previously. In the first iteration the query concept $Q=(Q_A,\{C2,C3,C5\})$ is considered.

In this example extension $Q_A=\phi$. No data source is added to the result $R(Q,C)$ in this step.

The second iteration makes it possible to add to the result $R(Q,C)$ the data sources A1(0.4) and A4(0.4) because concept $(\{A١(٠.٤),A٤(٠.٤)\},\{C٢,C٣,C٤\})$ (forming the set $SUBS(Q, C,1)$) subsume the query concept $Q = (Q_A, \{C2, C3, C5\})$. With the third iteration, sources A2 (0.6), A6 (0.5) and A5 (0.4) are added to $R(Q, C)$ because $SUBS(Q, C,2) = \{(\{A2(0.6), A6(0.6)\}, \{C2, C$ $(\{A5(0.4)\}, \{C2, C3\})\}$. With the fourth iteration $SUBS(Q, C,3) = \{(\{A1(0.5), A2(0.6), A4(0.$ $(\{A1(0.5), A2(0.6), A4(0.4), A5(0.5), A6(0$. The sources data appearing in the extension of these concepts are already added. The only forming concept $SUBS(Q, C,4)$ is a concept having an empty intension. No source is thus added to $R(Q, C)$ with this step and the algorithm stops since $SUBS(Q, C, ٤) = \phi$. Thus $R(Q, C)$ is consisted of the sources ordered according to a satisfaction degree defined as follows.

Definition 8: the satisfaction degree corresponds to the similarity of a fuzzy formal concept $C_١ = (\varphi(A_١), B_١)$ and its subconcept $C_2 = (\varphi(A_2), B_2)$ is defined as:

$$E(C_١, C_٢) = \frac{|\varphi(A_١) \cap \varphi(A_٢)|}{|\varphi(A_١) \cup \varphi(A_٢)|}$$

(1)

Table 3 shows the satisfaction degrees of relevant answers.

Table 3. Scheduling of the result.

Data Sources	Meta Data	Satisfaction Degree
$\{A1, A4\}$	$\{C2, C3, C$	1.00
$\{A٥\}$	$\{C٢, C٣\}$	0.5
$\{A٢, A٦\}$	$\{C٢, C٥\}$	0.42

A search algorithm for the relevant data sources is as follows.

Input:

A query $Q = (Q_A, Q_B)$ *where* $Q_A = \phi$ *and* Q_B *is a set of metadata.*

The concept lattice $\Im(C)$ *corresponding to the formal context* K.

Output:

The concept lattice $\Im(C \oplus Q)$.

A set of ordered relevant sources for the query Q *and the satisfaction degrees.*

Begin:

Build the concept $Q = (Q_A, Q_B)$.

Insert Q *in lattice* $\Im(C)$ *to obtain the lattice* $\Im(C \oplus Q)$.

Seek in $\Im(C \oplus Q)$ *the new concept* $Q = (Q_A \cup Q_B^{'}, Q_B)$

level:=0

$SUBS(Q, K, level) := \{Q\}$ *(initialization of the query subsumers research* Q *in the lattice in function of their level or distance from* Q

$R(Q, C) := \phi$ *(initialization of the result)*

While $SUBS(Q, C, level) := \phi$ *do*

$R(Q, C, level) := \phi$

For concept $C = (A, B) \in SUBS(Q, C, level)$ *do*

a. If $B \neq \phi$ *then*

$R(Q, C, level) := R(Q, C, level) \cup A$

b.End If

End For

$R(Q, C) := R(Q, C) \cup R(Q, C, level)$

Build $SUBS(Q, C, level + 1)$ *the set of direct concepts subsumers of* $SUBS(Q, C, level)$

level := level +1

End While

Outputting $R(Q, C)$, *the set of relevant data Sources for* Q *and the concept's similarity as satisfaction degrees.*

End

6. Conclusion

We have proposed an approach for flexible database querying based on an extension of ordered lattice Theory. The proposed approach consists of the following steps: the first step consists of data organisation for dependence's extraction. For this, we have proposed to combine two data analysis techniques: fuzzy cluster analysis and FCA based on an ordered lattice theory extension. The first technique is applied for allowing objects of a data set to belong to several clusters simultaneously, with different degrees of memberships. Despite being a very effective technique, the mutual relationships between specific clusters of interest are masked. We have used a second data analysis method which is based on an extension ordered lattice theory for the generation of these relationships. The result is an incremental Multi-attributes Type Abstraction Hierarchy (MTAH).

The second step consists of database flexible querying. For this we have proposed to use the MTAH generated from the first step for seeking, to interrogate them, relevant data sources.

We have proposed a new formalism for scheduling the result data sources based on concept similarity. We can extent this work while applying our approach with large databases and to use FCA to discover data semantic. Future investigations should take this into account.

References

[1] Bosc P. and Pivert O., "Some Approches for Relational Databases Flexible Querying," *International Journal of Intelligent Information Systems*, vol.1, no.3-4, pp. 323-354, 1992.

[2] Zadeh L., "Fuzzy Sets,". *Information and Control 8*, vol. 8, no. 3, pp. 338-353, 1965.

[3] Priss U., "Formal Concept Analysis in Information Science," *Annual Review of Information Science and Technology (ARIST)*, vol. 40, 2006.

[4] Uri K. and Jianjun Z., "Fuzzy Clustering Principles, Methods and Examples," *IKS*, December 1998.

[5] Bosc P., Galibourg M., and Hamon G., "Fuzzy Quering with SQL: Extensions and Implementation Aspects," *Fuzzy Sets and Systems*, vol. 28, pp. 333-349, 1988.

[6] Habib O. and Ramzi B., "Interrogation Flexible et Coopérative D'une BD par Abstraction Conceptuelle Hiérarchique," *INFORSID*, pp. 41-56, *2004*.

[7] Chu W., Yang H., Chiang K., Minock M., Chow G., and Larson C., "CoBase: A Scalable and Extensible Cooperative Information System," *Journal of Intelligence Information Systems*, Kluwer Academic Publishers, Boston, Mass, vol. 6, no.2-3, pp. 233-259, 1996.

[8] Laszlo S. and Amedeo N., "Les Treillis de Galois Pour L'organisation et la Gestion des Connaissances," *in Proceedings of 11`emes Rencontres de la Société Francophone de Classification (SFC'04)*, Bordeaux, France, pp. 298-301, 2004.

[9] Halkidi M. and Vazirgiannis M., "Clustering Validity Assessment: Finding the Optimal Partitioning of a Data Set," *in Proceedings of the IEEE International Conference on Data Mining (ICDM.01)*, San Jose, California, USA, 2001.

[10] Bezdek J., *Pattern Recognition with Fuzzy Objective Function Algorithms*, Kluwer Academic Publishers, Nowell, MA, 1981.

[11] Sassi M., Touzi A., and Ounelli H., "Using Gaussians Functions to Determine Representative Clustering Prototypes," *in Proceedings of the 17th IEEE International Conference on Database and Expert Systems Applications*, Poland, pp. 435-439, 2006.

[12] Ganter B. and Wille R., *Formal Concept Analysis: Mathematical Foundations*, Springer, Berlin, Heidelberg, 1999.

[13] Wille R., "Lattices in Data Analysis: How to Draw them with a Computer," *in Rival I. (eds.)*, Algorithms and Order, Kluwer, Dordrecht-Boston, pp. 33-58, 1989.

[14] Wolff K., "Information Channels and Conceptual Scaling," *in Proceedings of the International Conference on Conceptual Structures (ICCS'2000)*, Darmstadt, 2000.

[15] Chu W., Chiang K., Hsu C., and Yau H., "An Error-Based Conceptual Clustering Method for Providing Approximate Query Answers," *Communications of the ACM*, vol. 39, no. 12, 1996.

[16] Huynh V. and Nakamori Y., "Fuzzy Concept Formation Based on Context Model," *in Baba N., et al.(eds) editors, Knowledge-Based Intelligent Information Engineering Systems and Allied Technologie, IOS Press*, Amsterdam, pp. 687-691, 2001.

[17] Godin R., Mineau G., and Missaoui R., "Méthodes de Classification Conceptuelle Basée sur les Treillis de Galois et Application," *Revue D'Intelligence Artificielle*, vol. 9 no. 2, pp. 105-137, 1995.

[18] Capipineto C. and Romano G., "Order-Theoretical Ranking," *Journal of the American Society for Information Science*, vol. 51, no. 7, pp. 587-6001, 2000.

[19] Godin R., Missaoui R., and Alaoui H., "Incremental Concept Formation Algorithms Based on Galois (Concept) Lattices," *Computational Intelligence*, vol. 11, no. 2, pp. 246-267, 1995.

Minyar Sassi received the diploma of engineering in computer science from National School of Engineering of Tunis, in 2003, and the Master degree in automatic and signal processing from the National School of Engineering of Tunis in 2004. Currently, she is a PhD candidate. Her researches concern query optimization, clustering and flexible querying.

Amel Touzi received the diploma of engineering in computer science and PhD in computer science from the Faculty of Sciences of Tunis, in 1989 and 1994, respectively. She is an assistant professor at the Department of Technologies of Information and Communications in the National School of Engineering of Tunis. She is also a member of the Systems and Signal Processing Laboratory

(LSTS). Her researches interest includes many aspects of logic programming, deductive databases, fuzzy relational databases, object-relational databases flexible querying and artificial intelligence.

Habib Ounelli received a PhD in computer science in 1987 from the University of Paris-Sud Orsay, Paris, France. Since 1988, he is a full professor at the Computer Sciences Department of Faculty of Sciences of Tunis. His researches concern fuzzy databases, flexible querying, and deductive databases.