Why and How to Define a Similarity Measure for Object Based Representation Systems

Gilles BISSON
IMAG-LIFIA, INRIA Rhône-Alpes, Project SHERPA
46 rue Félix Viallet, 38031 Grenoble CEDEX, France
phone : (16) 76-57-45-88 ; email : gilles.bisson@imag.fr

Abstract
Currently, in the Objects-Based Representation Systems, both classification and categorization are based on subsumption criterion. In practice, this criterion seems too strong as soon as we need to deal with incomplete or incoherent knowledge. The notion of similarity has been successfully used in many domains such as Data Analysis, Pattern Recognition, or Machine Learning to compare and to structure noisy knowledge. In this paper we present a preliminary work aiming at demonstrating the advantage of using a similarity measure instead of a subsumption criterion in the object-based representations.

keywords
Object-Based Representation, Similarity Measure, Classification, Categorization.

1. Motivations

The aims of Object Based Representation Systems (OBRS) are twofold: on the one hand they allow to store and to organise a domain knowledge; on the other hand, they allow to answer some user's queries through several inference mechanisms. Among the set of inference mechanisms, two of them are particularly important: the *classification* and the *categorization.* In the OBRS that distinguish between classes and instances, the classification process of these two kinds of objects has a different goal. The classification of an instance into a Knowledge Base, expressed as a hierarchy of classes, aims at finding the set of the most specific classes whose definition *subsumes* the slot values of the instances. Practically, the classification of instances aims at *identifying* the studied instances and allows to *deduce* some new information about these instances. The classification of a class aims at *setting* a class into a Knowledge Base by finding the two sets of the most specific classes (respectively most general ones) whose definitions subsume (respectively is subsumed by) the definition of the studied class. This operation can be used to build a KB incrementally, or to compare two KB.

The categorization aims at building automatically a hierarchy of classes from a set of instances describing the different objects found in the application domain. The categorization process has been extensively studied from a *Numerical Standpoint* in Data Analysis (Anderberg 73; Sneath *et al.* 73; Everitt 80) or more recently, from a *Symbolical Standpoint* in Conceptual Clustering (Michalski *et al.* 83; Gennari *et al.* 89; Bisson 92b).

However, it is worth noting that classification and categorization operations are not totally independent since some categorisation processes are based on the classification of instances (Fisher 87) or the classification of classes (Euzenat 93).

Currently, in the OBRS framework the inference processes of classification and categorization are based on the subsumption criterion. This criterion allows the system to compare the degree of generality between two any concepts A and B. In this way, the relation "A subsumes B" means that the set of individuals described by A, namely the extension of A, includes the set of individuals described by B. Here we propose to replace the subsumption criterion by a similarity criterion during the inference processes[1]. Now, the classification and the categorization will be based on a numerical measure allowing to quantify the similarities and the differences between any pair of objects.

The notion of similarity measure has been used and studied in many domains such as Data Analysis, Pattern Recognition, Machine Learning or Cognitive Sciences. However, the similarity measures defined in these domains are often unable to deal with representation languages as complex as those used in the OBRS framework. In the frame of OBRS few work dealing with the similarity notion are to be found, among which we can cite the work of Euzenat (93) about the building (by categorization) of concept hierarchies for the knowledge representation system TROPES (Mariño *et al.* 90) or the work of Salotti (92), Rossazza (90) et Granger (88) concerning the fuzzy classification of objects. In the neighbouring domain of Conceptual Graphs some work has also been done by Myang *et al.* (93) and Maher (93). Nevertheless, all of these similarity measures are based on the comparison of the attribute-value expressing the *local structure* of the objects, however they hardly deal with the *relational structure* existing between the objects. By relational structure, we mean the graph of links between the objects. This paper proposes an approach allowing to deal with *both* local and relational structure.

However, before defining such a similarity measure between objects, we have to answer two questions: first, what relations do exist between similarity and subsumption; secondly, what kind of advantages brings the similarity with respect to the subsumption ?

Let's consider the function SUB (O1, O2) testing during the classification if the description of the object O1 *strictly subsumes* the description of the object O2. This function is valued in the set {true, false}. Now, let's consider the function SIM (O1, O2) that allows to evaluate the degree of similarity between the objects O1 and O2. We must precise that such a similarity function is obviously *asymmetrical*[2]. As a matter of fact, the meaning of SIM (O1, O2) is: "What is the degree of inclusion between the definition of object O1 and the definition of object O2". Thus, the similarity function is valued in the continuous interval [0..1], the value 1 means a perfect inclusion of the first object within the second. In this way, the similarity function can be seen as an *extension* of the

---

[1] However, the classes in the KB are classically hierarchised according to a strict subsumption criterion.

[2] It is worth noting that from the mathematical point of view a similarity function is always symmetrical (Sim(x, y) = Sim(y, x)). However, in this paper we use the word similarity in a more general meaning, which is closely related to the one used in Cognitive Sciences (Tversky 77). In the classification problem, this function is clearly not symmetrical because each argument of the function plays a very different role: we compare a *target* object O1 with a *reference* object O2.

subsumption function because it provides a more precise information about the degree of generality between the objects. These functions verify the following implications[3].

$$\mathrm{SIM}\ (O1, O2) = 1 \quad \Rightarrow \quad \mathrm{SUB}\ (O1, O2) = \text{true}$$
$$\mathrm{SIM}\ (O1, O2) \in [0..1[\quad \Leftarrow \quad \mathrm{SUB}\ (O1, O2) = \text{false}$$

Henceforth, by using a similarity criterion the classification does not return the list of the most specific classes whose definitions strictly subsumes the definition of the object to classify, but it returns a list of classes ordered by decreasing degree of subsumption. This capacity to deal with “weak” subsumption is interesting for several reasons.

On the one hand, a large Knowledge Base is never built in one step and in the course of its use, one always needs to refine some definitions of classes. Therefore, at a given moment a KB can be partially incomplete or incoherent. For instance, the definition of classes can be incorrect (some attributes are missing or irrelevant) or the slot definitions contain some errors (the ranges of the intervals are not correct). Clearly, the subsumption criterion can help the developers of the KB to detect such problems. But in the same time this criterion prevents from using this KB because the classification process always stops as soon as it meets with a contradiction. By using a similarity criterion this problem disappears and the user can try to classify an object in the KB even if the definition of this object is partially incoherent with respect to the information contained in this KB.

On the other hand, when a user classifies an object, the relevance of the attributes is determined by his work and his goal. For instance, when a biologist and an ethologist want to compare two animals they obviously use a very different set of features. The system TROPES (Mariño *et al.* 90) proposes an interesting modelization of this problem through the notion of *Viewpoints*. In TROPES, each concept can be described following different viewpoints that correspond to a specific hierarchy of classes; each viewpoint uses a subset of the attributes of the concept. In practice, the vocabulary (attributes) used in a viewpoint is defined by a Boolean vector $[A_1, ..., A_n]$ where each Boolean A_i corresponds to an attribute of the concept. Thus, A_i equals true when the corresponding attribute is used in the classes of the viewpoint and A_i equals false otherwise. Again, the notion of similarity allows to go further by associating a relevance value to each attribute. In this way, we can express the fact an attribute is used or not in the viewpoints (0 meaning not used), but we also can express a *relevance scale* between the attributes.

Finally, the possibility of having a similarity measure between the instances is very interesting for the categorization problem. As a matter of fact, there are many efficient methods able to build a taxonomy from a similarity[4] matrix. For instance, we can use some techniques coming from Data Analysis as in the system T-TREE (Euzenat 93). Machine Learning approach also provides some methods allowing to build a hierarchy of concepts from a set of instances and a similarity measure; we could cite the system ADECLU (Decaestacker 93) and KBG (Bisson 92b) or the work of Lebbe *et al.* (91).

---

[3] We don’t have a full equivalence because the similarity computation describes in this paper is based on some heuristics. However, these heuristics allows to compute the similarity in a polynomial-time.

[4] Here, the similarity is a symmetrical function because during the categorization process each instance plays the same role. Thus, there is no reference object nor target object.

This paper is composed of four parts. First, we describe (§2) the Object Based Representation and the terminology used in this paper. Next, we present (§3) a rapid state of the art of the similarity measures in domains such as Data Analysis, Cognitive Sciences and Machine Learning and we propose a method allowing to evaluate the degree of similarity between two items in the classification and categorization contexts. In section §4 we apply the method previously defined in the frame of OBRS. Finally, we examine how our similarity can be used in the classification and categorisation processes.

2. Representation and Terminology

As in an OBRS like SHIRKA (Rechenmann *et al.* 91), we make a distinction between the notion of class (defining a concept) and the notion of instance (example of concept). A class is composed of a set of pairs *attribute/facet* with the following properties. First, we associate a numerical weight to each attribute (positive number) expressing the degree of relevance of this attribute in the class[5]. Secondly, we distinguish between two general kinds of facets: the facets that express a value (such that $default or $value) and the facets that define a type of data (such that $type, $range, $interval or $cardinality). When the type of an attribute is another object, we use the term *relation* to speak about this attribute. Finally, we do not consider the facets expressing a method or an action. We distinguish between two kinds of objects: the simple objects, which are objects without relation, and the complex objects, which correspond to a set of objects linked together by a set of relationship. In this paper, when we speak about the attributes (or properties) of an object we speak about all of its properties, including the inherited ones.

A Knowledge Base can be described as a graph whose each node expresses an object (class). In this graph, we can distinguish between a horizontal structure and a vertical structure (figure 1). The *horizontal structure* is correlated with the inheritance links (AKO) expressing the hierarchy of classes, the *vertical structure* is made of the binary relations existing between two classes[6]. Clearly, if the inheritance links always constitute an acyclic graph, that is not the case for the graph of relations.

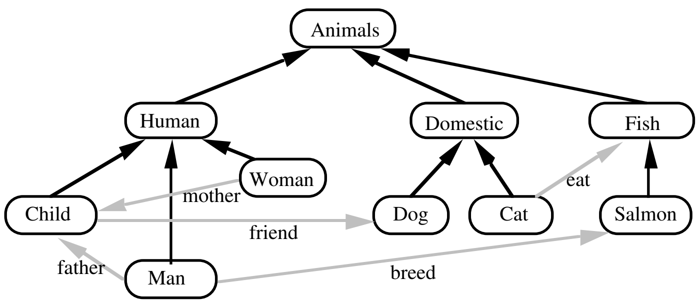

Figure 1: vertical and horizontal structures in a Knowledge Base.

Therefore, both classification and categorization processes are equivalent to a graph matching problem. As a matter of fact, the problem of classifying a complex object (class or instance) into a Knowledge Base comes down to find an *optimal matching* (figure 2)

[5] We will not discuss in this paper the way this relevance scale has been established.
[6] In a complex instance, we just have a horizontal structure between the objects.

between the components of this object and the components of the KB. By optimal matching, we mean the matching that maximizes the value of the similarity function. In the same way, to compute the similarity between two instances we need to finding the optimal matching between the simple objects that compose these instances.

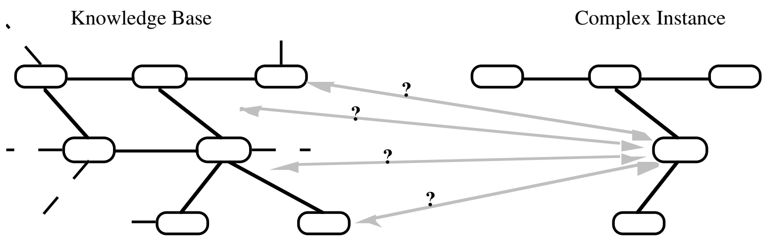

Figure 2: Matching problem during the classification.

As detailed in section §3.2, in order to be relevant the similarity measure must be able to take into account both *local* and *relational* structures of the matched objects. By local structure, we mean that the measure must evaluate the degree of likeness between the matched objects, that is to say the similarity between the shared attributes. By relational structure, we mean that the measure also must evaluate the degree of relationship between the matched objects, that is to say the similarity between the shared horizontal structures.

3. Work about Similarity

3.1. Similarity between attributes

The notion of distance has been extensively studied in the frame of Data Analysis (DA). Classically, a distance is defined as follow: Let Ω the set of individuals of the domain and let the metric $\mathfrak{d}$ that is an application from Ω x Ω to $\Re^+$.

$\forall$ a, b, c $\in \Omega$

1) $\mathfrak{d}(a, a) = 0$ (minimality property)
2) $\mathfrak{d}(a, b) = \mathfrak{d}(b, a)$ (symmetry property
3) $\mathfrak{d}(a, b) = 0 \Rightarrow a = b$ (identity property)
4) $\mathfrak{d}(a, c) \leq \mathfrak{d}(a, b) + \mathfrak{d}(b, c)$ (triangular inequality)

A distance must verify these four properties. It is worth noting that the individuals of Ω, as described in Data Analysis, are closely related to simple objects described by a set of attributes. Currently, many distances have been developped according to the type of attributes, such as χ^2 or Manahalobis (Diday *et al.* 82). However, as soon as we are working with real values, the most used (and simplest) metric is the *Minkowski measure*:

$$\mathfrak{d}_p(x, y) = \left[\sum_{i=1}^{K} W_i \,|\, x_i - y_i \,|^{\,p}\right]^{1/p} \qquad \text{with } p > 0 \qquad \textbf{(1)}$$

In this measure, x_i and y_i express the value of the i^{th} attribute describing the individual x and y. Wi expresses the numerical weight correlated with this attribute. According to the value of the parameter P, we retrieve some well-known distances such as *Manhattan distance* (P=1) and *Euclidian distance* (P=2). To transform the Minkowski distance **(1)** into a similarity measure we just need to introduce a value Di corresponding to the difference between the upper and the lower bounds of the range of the i^{th} attribute:

$$\mathbb{S}_p\,(x, y) = \left[\sum_{i=1}^{K} W_i \frac{D_i - |x_i - y_i|^p}{D_i}\right]^{1/p} \qquad \text{with } p > 0 \qquad \textbf{(2)}$$

The notion of distance between individuals has been also studied in Cognitive Sciences. From this point of view, Tversky (77) emphasises that the mathematical properties defined in Data Analysis (minimality, symmetry and triangular inequality) are generally not verified when one analyses the way people feel and deal with the notion of similarity. In other respect, the individuals to compare are often described by different sets of attributes. Therefore, Tversky proposes to evaluate the degree of similarity $\mathbb{S}$ (x, y) between two individuals x and y, respectively described by a set of attributes A and B, by combining the four terms $A \cap B$, $A \cup B$, A - B et B - A into the formula:

$$\mathbb{S}\,(x, y) = \frac{f\,(A \cap B)}{f\,(A \cup B) + \alpha\, f\,(A - B) + \beta\, f\,(B - A)} \qquad \text{with } \alpha, \beta \geq 0$$

According to the way the parameters f, α et β are instanciated, we can express different kinds of cognitive models of similarity. For instance, if we want to compare a pair of individuals described by a set of numerical attributes, we can combine the definitions proposed by Tversky and Minkowski into two different similarity measures **Sym** and **Asy** that can be respectively used in the frame of classification and categorization. In these measures, we use the Tversky's model to compare the two set of attributes describing the individuals; the function f of this model is the Minkowski's formula as rewritten in the equation **(2)**. The parameter P of this formula equals 1 since in the Tversky's model the function f corresponds to a linear combination of the features.

The function **Sym** is a symmetrical similarity measure that is usable in a categorization process. The parameters of the Tversky's model are instanciated to the values $\alpha=\beta=0$, which lead to a measure like $f(A \cap B)$ / $f(A \cup B)$. The function **Asy** is an asymmetrical similarity measure that is usable in a classification process. Here, the parameters of the Tversky's model are instanciated to the values $\alpha=0$ et $\beta=-1$, which lead to a measure like $f(A \cap B)$ / f(A). Hence, **Asy** allows to evaluate the degree of inclusion between the first argument (reference) into the second argument (target). In other words, **Asy** allows to quantify the degree of subsumption between the individuals B and A.

- Let the function: $\mathit{Att\text{-}similarity}\,(i) = \frac{D_i - |x_i - y_i|}{D_i}$

$\mathbf{Sym}(X,Y) = \dfrac{\sum_{i}^{A \cap B} W_i \; \mathit{Att\text{-}similarity}\,(i)}{\sum_{i}^{A \cup B} W_i}$	$\mathbf{Asy}(X,Y) = \dfrac{\sum_{i}^{A \cap B} W_i \; \mathit{Att\text{-}similarity}\,(i)}{\sum_{i}^{A} W_i}$

Both measures return a value belonging to the interval [0..1]. Here is an example of use of **Sym** and **Asy** with two individuals I1 and I2 described by a set of four attributes:

Attribute	Wi	I1	I2	Di	*Att-similarity* (i)
high	1	10	20	[0..99]	90%
length	2	2	2	[0..9]	100%
weight	1	8	-	[0..59]	-
strength	2	-	10	[-5..5]	-

⇒ **Sym** (I1, I2) = 48%
Asy (I1, I2) = 73%
Asy (I2, I1) = 58%

3.2. Similarity between relations

As soon as we work with a knowledge representation dealing with the notion of relations between the individuals (such that first order logic or object oriented modelization), the approach proposed in the previous section (§ 3.1) becomes too rough since we need to compare two graphs of individuals. Moreover, in our approach we consider that all the individuals in the graphs are relevant for the similarity computation. Therefore, when we compare any two individuals A and B, the similarity measure between these individuals must express the similarity between their attributes, but also the similarity between the individuals *connected* to A and B through the medium of the relations.

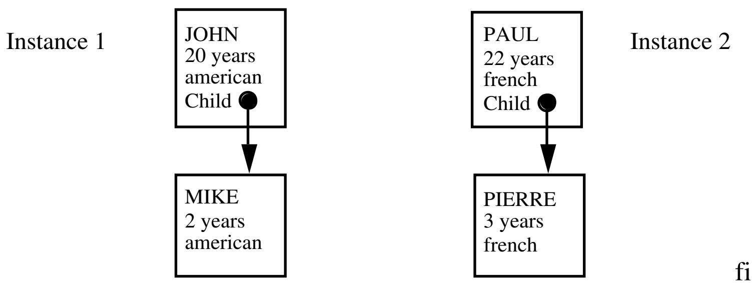

figure 3.

Thus, in the previous example (figure 3) to compare together the individual JOHN in the first instance and the individual PAUL in the second one, we need to evaluate the similarity between the common attributes (age and nationality), but we also need to take into account the similarity between their children MIKE and PIERRE. Intuitively, the closer MIKE and PIERRE are, the closer JOHN and PAUL are too. If we consider the relation CHILD as symmetrical, the problem is the equivalent when we compare the children: we must take into account the degree of similarity between their parents.

Many researches have been done to define a relevant and efficient similarity measure to compare any graphs of individuals. We can cite Pattern Recognition (Sanfeliu et Fu 83; Wong et You 85), Cognitive Sciences (Falkenhaimer *et al.* 89; Holyoak *et al.* 90) and Machine Learning (Esposito *et al.* 91; Bisson 92a). A detailed comparison between these approaches is beyond the scope of this paper, however we can emphasise some salient features. In Pattern Recognition people use informed tree search methods, such as *branch and bound search*, to explore the space of possible matchings. This method allows the system to find the optimal matching, nevertheless the process can be very expensive if the cost function used in the branch and bound is not efficient. Moreover, these methods provide an evaluation of the similarity between the whole graphs and not between the individuals. The methods implemented in Cognitive Sciences aim generally at studying the notion of *analogy*. Thus, the algorithms are not very efficient since they use very weak hypothesis to constrain the search space; for instance, they allow the system to match individuals without common attribute. However, this work brings some interesting evidences about the way people feel the notion of similarity: on the one hand, these theories emphasise that it is crucial to take into account both attributes and relations during the comparison of two individuals; on the other hand, for human being the soundness of a comparison increases when the matching choices preserves the relational structure between the individuals (the *systematicity principle* in Gentner 83).

In the system KBG[7] (Bisson 92b, 93) we have defined a similarity measure having two advantages. First, the complexity of the similarity computation is quadratic in terms of the number of individuals involved in the graphs and linear in terms of common attributes between individuals. Secondly, the general behavior of this similarity measure is consistent with the cognitive theories previously evoked. Without detailing, we think this second point is crucial in the frame of the OBRS: when the system has an apparent comportment closely related to the human behavior, the user can understand more deeply the way the system is working and thus he can use more efficiently this system.

In practice, the method used in KBG is quite simple. To measure the similarity between a pair of individuals, two kinds of information must be considered: the attributes and the relations. In section § 3.1 we have already proposed a simple approach allowing to evaluate the similarity between two sets of (numerical) attributes. Concerning the relations, the computation is based on the idea presented on the previous example (figure 3), that is to say: the result of the similarity evaluation between the relations CHILD of JOHN and PAUL depends on the similarity of MIKE with PIERRE, and reciprocally. More formally the similarity measure can be defined in the following way:

- Let X and Y a pair of individuals. These individuals are respectively described by two sets of the numerical attributes Ax and Ay and two sets of relations Rx and Ry.

1) Computation of the similarity between attributes

- Depending on whether we want to compute a symmetrical or asymmetrical similarity we use the functions **Sym** (X, Y) or **Asy** (X, Y) with Ax and Ay.

2) Computation of the similarity between relations

- Let the function Cnct (R, X) returning the individual connected to X through of the relation R. For instance: Cnct (Child, PAUL) → PIERRE

- Let the function *Rel-similarity* (i) computing the similarity between of the i^{th} relations of the list Rx∩Ry. This function plays the same role that the similarity function *Att-similarity* (i) between attributes. Depending on whether we compute a symmetrical or asymmetrical similarity the function is slightly different:

---

[7] This tool was implemented in the frame of the MLT ESPRIT contract. It can be used to categorize a set of instances and to generate a diagnosis KB. The representation language is based on the first order logic.

Symmetrical: $\textit{Rel-similarity}\ (i) = \frac{1}{2}\,(1+ \mathbf{FSsym}\ \ (\mathrm{Cnct}\ (R_i,X),\ \mathrm{Cnct}\ (R_i,Y))$

Asymmetrical: $\textit{Rel-similarity}\ (i) = \frac{1}{2}\,(1+ \mathbf{FSasy}\ \ (\mathrm{Cnct}\ (R_i,X),\ \mathrm{Cnct}\ (R_i,Y))$

Here, the similarity between a common relation R_i between the individuals X and Y equals the average similarities between the individuals involved in this relation. The first term "1" expresses the similarity between X and Y with respect to the common relation. The second terms (**FSsym** or **FSasy**) express the Final Similarity (symmetrical or not) between the two[8] individuals connected to X and Y through the medium of the relation R_i.

- Let the function **Relsym** and **Relasy** computing the similarity (symmetrical or not) for all the *common relations* between X and Y. In practice, these functions are equivalent to the functions **Sym** et **Asy** defined for the attributes.

$\mathbf{Relsym}(X,Y)= \dfrac{\sum_{i}^{Rx\cap Ry} W_i\ \textit{Rel-similarity}\ (i)}{\sum_{i}^{Rx\cup Ry} W_i}$	$\mathbf{Relasy}(X,Y)= \dfrac{\sum_{i}^{Rx\cap Ry} W_i\ \textit{Rel-similarity}\ (i)}{\sum_{i}^{Rx} W_i}$

The final similarity (**FSsym** or **FSasy**) between X and Y equals to the average similarities computed for the attributes and the relations.

$$\mathbf{FSsym}\ (X,Y) = \frac{1}{2}\ [\mathbf{Sym}\ (X, Y) + \mathbf{Relsym}\ (X, Y)] \quad \in [0..1]$$

$$\mathbf{FSasy}\ (X,Y) = \frac{1}{2}\ [\mathbf{Asy}\ (X, Y) + \mathbf{RelAsy}\ (X, Y)] \quad \in [0..1]$$

By reading this definition we can easily notice the following phenomena: to compute the similarity between the common relations of X and Y (**Relsym** or **Relasy**) one needs to know the overall similarity values (**FSsym** or **FSasy**) between the individuals connected to X and Y. Obviously the situation is fully symmetrical and the individuals connected to X and Y also needs in their similarity computations the overall similarity value between X and Y. Therefore, the similarity computation between two graphs of individuals comes down to the problem of *setting and solving a system of linear equations in several unknowns*. In these equations the unknowns express the pairs of individuals whose we try to evaluate the similarity. This method allows to take into account the *local* and the *relational* structures simultaneously during the computation. In this way, the higher the similarity value between two individuals is, the closer their properties and their connected individuals are too. Finally, in this method no *tree search* is required to evaluate the quality of matching. Of course, as this approach is based on a heuristic, the similarity values found do not always correspond to an optimal matching.

[8] If the cardinality of the relation is higher than 1, we can have a list of individuals connected to X and Y rather than only one individual. We will detailed this point in the section § 4.2. However, it is worth noting in this case the similarity measure always keeps the properties previously evoked.

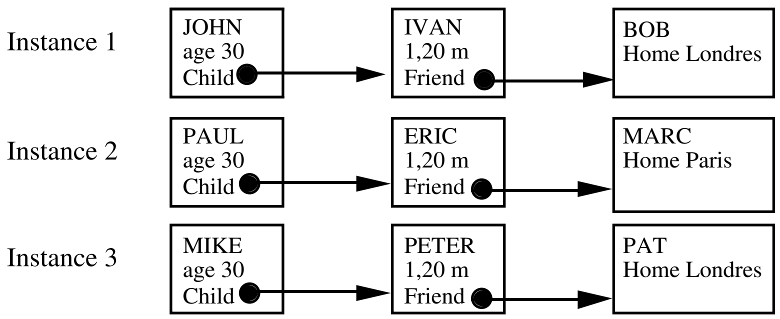

figure 4.

In figure 4, an example illustrates the advantages of our approach. If we compare the similarity between JOHN and PAUL and the similarity between JOHN and MIKE, we observe that FSsym (JOHN, PAUL) is smaller than FSsym (JOHN, MIKE). Indeed, although the fathers and their children are totally identical, there exist a difference concerning the place where the children's friends are living. With our method this difference is automatically taken into account in the similarity between the fathers. In practice, the scope of a piece of information (here, the attribute Home) into the graph is *infinite*. However, the influence an information has on the similarity between a pair of individuals *decreases as a geometrical law* according to the distance; namely, the number of relations between this information and the individuals (here, there are two relations: Friend and Child). Finally, we can demonstrate (Bisson 92b, 93) that the system of linear equations has always a solution even when they are some cycles between the relations.

3.3. Similarity between graphs of individuals

We have evaluated the degree of similarity between all the pairs of individuals (sharing at least one common property) occurring in a pair of graphs. Now, we need to study the way to compute the global similarity between these two graphs (figure 5).

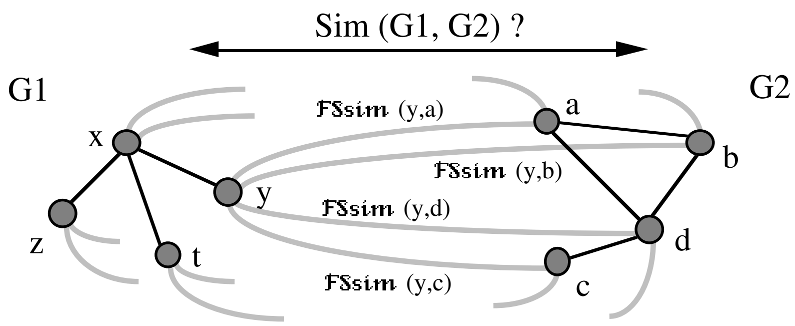

Figure 5: computation of the similarity between two graphs of individuals

Roughly speaking, we must search the 1-1 matching between the individuals of G1 and those of G2 that optimises the global similarity. From a theoretical standpoint, this problem comes down to searching the *optimal weighted assignment in a bipartite graph*. Several polynomial algorithms have been developped (Ahuja *et al.* 93) to deal with this problem. Moreover, Brissac *et al.* 93 proposes a general discussion about this problem. Once the matching has been chosen, the global similarity between the two graphs can be evaluated by computing the average similarity between the matched individuals.

4. Object and similarity

Obviously, the work presented in the previous section can be easily reused in the framework of OBRS. As a matter of fact, the individuals described in the previous section can be seen as a kind of elementary objects. Thus, we are going to study how to redefine the functions *Rel-similarity* and *Att-similarity* described in section § 3.2 in order to take into account the notion of facet. We split the discussion into two parts: the first part concerns the attributes (§ 4.1) and the second part concerns the relations (§ 4.2).

4.1. Similarity between attributes

Depending on the problem to solve, classification or categorization, the facets are compared in a different way. During categorization, we want to evaluate the likeness between two values (*symmetrical similarity*). During classification of instances or classes, we want to evaluate the degree a subsumption (*asymmetrical similarity*) that exists between a value and a type definition or between two type definitions. Although a facet definition can be rather complex, it seems possible to take into account the most part of the information it contains during the similarity evaluation. We are going to illustrate this point by defining such a measure in the case of the numerical attributes[9] used in the system SHIRKA (Rechenmann et al. 91). In this system an attribute (integer or real) can be defined either as monovalued or multivalued with cardinality constrains.

4.1.1 Similarity between two set of values (categorization of instances)

The comparison between a pair of values can be achieved with the help of a similarity measure (figure 6) denoted **SV** (x, y), which is identical to the one used in the section § 3.1. However, when the attributes are multivalued we have to deal with two sets of values. Thus, we need to introduce a new function denoted **BMatch** (S1, S2, **SV**) that searches for the Best MATCHing between the values of S1 and the values of S2, that is to say, the matching that optimises the global similarity between the two sets[10]. This function returns the average similarity between the matched values. Finally, we must take into account the fact that the cardinal of the two sets S1 and S2 can be different.

- Let two sets of values S1 = (V_1, V_2, ..., V_n) and S2 = (U_1, U_2, ..., U_m).
- Let D the difference between the bounds of the attribute domain.
- Let the similarity function **SV** (x,y) = $\frac{D - |x - y|}{D}$

$$\mathit{Att\text{-}similarity}\,(S1, S2) = \frac{1}{\text{Maximum}\,(n,m)} \;.\; \mathbf{BMatch}\,(S1, S2, \mathbf{SV}))$$

4.1.2 Similarity between a set of values and a definition (classification of instances)

We have to evaluate the degree of inclusion between the values of the instance and the definition of the attribute (described in the form of an interval). The similarity measure must return 1 when all the values belong to the definition and a number between [0..1[else. Thus, we introduce a function (figure 6) denoted **SI** allowing to evaluate the

[9] We could also define a similarity measure on the other types such that: taxonomy or nominal value.

[10] As in section § 3.3, the problem is to find the optimal weighted assignment in the bipartite graph.

similarity between one value and an interval. Moreover, as the cardinality constraints on the multivalued attributes are expressed in the form of an interval, the function **SI** is also used to evaluate if the number of values is coherent with the cardinality constraint.

- Let a set of values $S = (V_1, V_2, ..., V_n)$.
- Let an attribute Att whose range is defined through the interval Dom = $[D_1 .. D_2]$ and its cardinality through the interval Card = $[C_1 .. C_2]$.
- Let the function **SI** (x, [A..B]) = If $x \in [A..B]$ return 1

  Else return $\dfrac{|A-B|}{|A-B| + \text{Minimum}(|x-A|, |x-B|)}$

$$\textit{Att-similarity}\ (\text{Att}, S) = \mathbf{SI}\ (n, \text{Card}) \cdot \frac{1}{n} \sum_{i}^{n} \mathbf{SI}\ (V_i, \text{Dom})$$

4.1.3 Similarity between two definitions of attributes (classification of classes)

We have to evaluate the degree of subsumption between the reference definition and the target definition. This comparison involves two intervals and is achieved with the help of the function denoted **SD**. This function is quite similar to the function **SI** .

- Let the reference attribute Att_1 (respectively the target attribute Att_2) whose domain is expressed by the interval $\text{Dom}_1 = [D_1 .. D_2]$ and the cardinality by the interval $\text{Card}_1 = [C_1 .. C_2]$ (respectively $\text{Dom}_2 = [D_3 .. D_4]$ and $\text{Card}_2 = [C_3 .. C_4]$).

- Let **SD** ([A..B],[C..D]) = Case
  - $[A..B] \subset [C..D]$ return 1
  - $A \subset [C..D]$ return **SI** (B,[C..D])
  - $B \subset [C..D]$ return **SI** (A,[C..D])
  - $[C..D] \subset [A..B]$ return $|C-D| / |A-B|$
  - Else return $\frac{1}{2}$(**SI**(A,[C..D]) + **SI**(B,[C..D]))

$$\textit{Att-similarity}\ (\text{Att}_1, \text{Att}_2) = \mathbf{SD}\ (\text{Card}_2, \text{Card}_1) \cdot \mathbf{SD}\ (\text{Dom}_2, \text{Dom}_1)$$

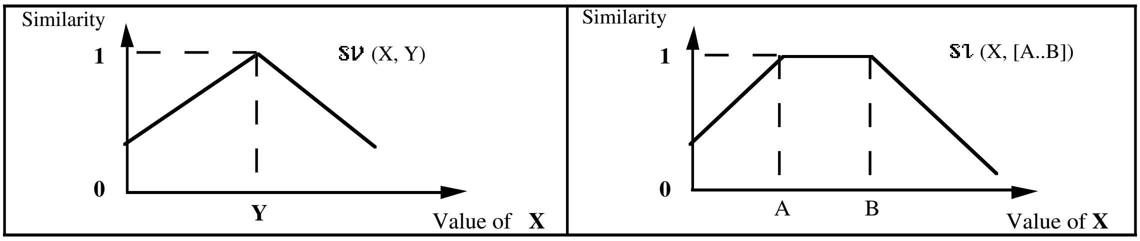

Figure 6: behavior of the functions **SV** and **SI**.

Obviously, the similarity measures between numerical attributes that we described in this section are somewhat arbitrary. However, the user can easily modify these definitions without changing the properties of the general method described in section § 3. Practically, the very advantage of using a similarity measure is to provide a flexible framework that the user can freely customise according to his application domain and to his needs. More generally, it seems important (Capponi 94) in the OBSR to allow the users to define their own types of data with the corresponding similarity measures.

4.2. Similarity between relations

With respect to the method detailed in section § 3.2 few modifications are needed to deal with relations in an object based representation. However, depending on the problem to solve, classification or categorization, the similarity measure *Rel-similarity* is different.

4.2.1 Similarity between two lists of objects (categorization)
In the section § 3.2 we did not take into account the fact that an object can be connected to a set of objects (that is typically the case for the Children relation). This problem can be easily solved by using a function quite similar to **BMatch** (see § 4.1.1). The function **SBMatch** (S1, S2, **FSsym**) searches for an optimal matching between the objects of the two sets S1 and S2. The only difference between **BMatch** and **SBMatch** is that the second function returns the summation of the similarity between the matched objects instead of the average of this summation. As the categorization process needs a symmetrical similarity, we use the function **FSsym** to evaluate each matching.

- Let two sets of objects $S_1 = (O_1, O_2, ..., O_n)$ and $S_2 = (P_1, P_2, ..., P_m)$.

$$\mathit{Rel\text{-}similarity}\ (S_1,S_2) = \frac{1}{\text{Maximum}(n+1,m+1)}.(1+\mathbf{SBMatch}\ (S_1,S_2,\mathbf{FSsym}))$$

4.2.2 Similarity between a list of objects and a definition (classification of instances)
During the classification of instances and for a given relation, we need to compare the set of objects (instances) in the slot value of the relation with the object (class) describe in the attribute definition. The final similarity equals the average similarity (computed with the asymmetrical function **FSasy**) between each object of the slot value with the object of the attribute definition. The cardinality constraint is computed as in section § 4.1.2.

- Let the set S of the objects in the slot value of the relation: $S = (O_1, ..., O_n)$.
- Let the object P (class) occurring in the definition of the studied relation. The cardinality of this relation is expressed in the form of an interval: $Card = [C_1 .. C_2]$.

$$\mathit{Rel\text{-}similarity}\ (P, S) = \mathbf{SI}\ (n, \text{Card}) . \frac{1}{n+1} . (1+ \sum_{i}^{n} \mathbf{FSasy}\ (P, O_i))$$

4.2.3 Similarity between two definitions of relations (classification of classes)
In that case, we need to compare the objects (classes) occurring in the definition of the studied relation. The cardinality constraint is computed as in section § 4.1.3.

- Let two objects (classes) P_1 et P_2 respectively occurring in the definitions of the relation R of the reference class O_1 and the target class O_2 we are studying. The cardinality of the relation R in O_1 (respectively in O_2) is expressed in the form of an interval $Card_1 = [C_1 .. C_2]$ (respectively $Card_2 =[C_3..C_4]$).

$$\mathit{Rel\text{-}similarity}\ (P_1, P_2) = \mathbf{SD}\ (\text{Card}_2, \text{Card}_1) . \frac{1}{2}\ (1+ \mathbf{FSasy}\ (P_1, P_2))$$

Once defined the functions *Att-similarity* and *Rel-similarity*, the similarity between a pair of complex objects is evaluated with the method describes in sections § 3.2 and § 3.3.

5. Use of the Similarity Measure

In the two previous sections we have defined a similarity measure allowing to evaluate the degree of similarity between two complex objects. Now we are going to see how such a similarity measure can be used to categorize and to classify a set of objects.

The categorization of a set of instances can be achieved by using a *Conceptual Clustering* approach (Michalski *et al*. 83). Given a set of instances and some information about the domain, a Conceptual Clustering system aims at performing the following task:

- To find a set of relevant categories (class) grouping these instances.
- To provide a comprehensive symbolic definition of each cluster.
- To build a hierarchical organisation of these clusters

Currently, there are two kinds of Conceptual Clustering systems whose algorithms are based on a similarity measure. The system ADECLU (Decaestacker 93) is representative of the Concept Formation approach (Fisher 87) and it allows to build a hierarchy of classes in an incremental way. The system KBG (Bisson 92b) uses another method based on a non-incremental agglomerative clustering algorithm stemming from Data Analysis community. However, it is worth noting that most of these Conceptual Clustering systems use a representation language based on a logical framework. Nevertheless, it is possible to adapt the existing algorithms to an object representation framework: the system T-TREE (Euzenat 93) is a good example of such approach.

Concerning the classification tasks, the use of a similarity measure instead of the subsumption function involves some changes in the classification algorithms. As a matter of fact, the notion of impossible class (namely, a class that does not verify the subsumption criterion) disappears. Therefore, we lose the stopping criterion of the classification algorithms. The classification process can be controlled by using a kind of *branch-and-bound* algorithm: at each step, we explore the children of the classes having the higher score of similarity with the object to classify. However, the similarity measure proposed in this paper does not always monotonically increase during the specialization; eventually, we cannot guarantee to select the optimal classes in the Knowledge Base.

6. Conclusion

Currently, in the objects-based representation systems, both classification and categorization are based on a subsumption criterion. However, when we need to deal with incomplete or incoherent knowledge this criterion is difficult to use. In such a domain, the notion of similarity seems more relevant. In this paper we have proposed a general approach of the similarity combining some results stemming from Data Analysis, Cognitive Sciences and Machine Learning. Our measure allows to evaluate the degree of similarity between complex objects and this measure can be used both in the frame of classification and categorization. Moreover, the computational cost of this measure remains interesting since it is (at most) quadratic in terms of number of objects involved in the comparison and linear in terms of common attributes. However, this study must be pursued in order to define some new classification and categorisation algorithms.

Bibliography

AHUJA R.K, MAGNANTI T.L., ORLIN J.B. 1993. *Network Flows: Theory, Algorithms and Applications*. Prentice Hall.

ANDERBERG M. 1973. *Cluster Analysis for Applications*. Academic Press, New-York.

BISSON G. 1992a. Learning in FOL with a similarity measure. In Proceedings of 10th National Conference on Artificial Intelligence (AAAI) . San-Jose. 82-87.

BISSON G. 1992b. Conceptual Clustering in a First Order Logic Representation. Proceedings of 10th European Conference on Artificial Intelligence (ECAI). Vienna. 458-462.

BISSON G. 1993. KBG : Induction de bases de connaissances en logique des prédicats. Thèse de l'université de Paris-sud (n° 2670) soutenue le 30 avril 1993.

BRISSAC O., LIQUIÈRE M. 1990. Un algorithme optimal pour l'association d'entités à partir de leurs similarités. Actes des 9eme JFA. Strasbourg. B1-B14.

CAPPONI C. 1994. Exploitation des types dans un modèle de représentation des connaissances par objets. Actes 9e RFIA. 11-14 janvier. Paris. 171-183.

DECAESTECKER C. 1993. Apprentissage et outils statistiques en classification conceptuelle incrémentale. *Revue d'Intelligence Artificielle*. Volume 7. 33-71.

DIDAY E., LEMAIRE, J., POUGET J., TESTU F., 1982. *Eléments d'analyse des données*. Edition Dunod.

ESPOSITO F., MALERBA D., SEMERARO G. 1991. Flexible Matching for Noisy Structural Descriptions. In proceeding of 12th IJCAI, 658-664. Sydney.

EUZENAT J. 1993. Brief Overview of T-TREE : The TROPES Taxonomy Building Tool. Proceedings of the 4th ASIS SIG/CR Classification Research Workshop. 69-87

EVERITT B. 1980. *Cluster Analysis*. Halsted press. New York.

FISHER D.H 1987. Knowledge Acquisition via Incremental Conceptual Clustering. *Machine Learning Journal* 2, 139-172.

FALKENHAINER B., FORBUS K., GENTNER D. 1989. The Structure-Mapping Engine: Algorithm and Examples. *Artificial Intelligence Journal*. Number 41. 1-63.

GENNARI J., LANGLEY P., FISHER D. 1989. Model of Incremental Concept Formation. *Artificial Intelligence Journal*, Volume 40, 11-61.

GENTNER D. 1983. Structure-Mapping: A Theoritical Framework for Analogy. In *Readings in Cognitive Science*. Morgan Kaufmann 1988. 303-310. (from *Cognitive Science* 7, 1983).

GENTNER D., RATTERMANN M., FORBUS K. 1993. The Roles of similarity in Transfer: Separating Retrieviability from Inferential Soundness.*Cognitive Psychology* 25. 431-467.

GRANGER C. 1988. An application of possibility theory to object recognition. In *Fuzzy Sets and Systems*. Volume 28. 351-362.

HOLYOAK K., THAGARD P. 1989. Analogical Mapping by Constraint Satisfaction. In *Cognitive Science Journal*. number 13. 295-355.

LEBBE J., NICOLAS J., VIGNES R. 1991. From Knowledge to Similarity. In Proceedings of the International Conference Symbolic-Numeric, Data Analysis and Learning. 585-597. Paris (France).

MAHER P. 1993. A Similarity Measure for Conceptual Graphs. *International Journal of Intelligent Systems*. Volume 8 Number 8. 819-837.

MARIÑO O. 1991. Classification d'objets composites dans un système de représentation de connaissances multi-points de vue. Actes de Reconnaissance des Formes et Intelligence Artificielle (RFIA). 233-242.

MARIÑO O., RECHENMANN F., UVIETTA P. 1990. Multiple Perspectives and Classification Mechanism in Object-Oriented Representation. Proceedings of 9th European Conference on Artificial Intelligence (ECAI). Stockholm. 425-430.

MICHALSKI R.S., STEPP E. 1983. Learning from Observation : Conceptual Clustering. In *Machine Learning 1 an Artificial Intelligence Approach*, Tioga, 331-363.

MYAENG S., LOPEZ-LOPEZ O. 1992. Conceptual graph matching: a flexible algorithm and experiments. Journal of Experimental and Theoritical Artificial Intelligence 4. 107-126.

NAPOLI A. 1992. Représentations à objets et raisonnement par classification en intelligence artificielle. Thèse d'état de l'université Nancy I, soutenue le 31 janvier 1992.

NAPOLI A. 1994. Catégorisation, raisonnement par classification et raisonnement à partir de cas. Actes des 9eme JFA. E1-E14. Strasbourg 23-25 mars.

RECHENMANN F., UVIETTA P. 1991. SHIRKA - An object-centered knowledge base management system. In *Artificial Intelligence in Numerical and Symbolic Simulation*. ALEAS publisher. Lyon.

ROSSAZZA J.P 1990. Utilisation de hiérarchies de classes floues pour la représentation de connaissances imprécises et sujettes à exceptions : le système SORCIER. Thèse de l'université Paul Sabatier de Toulouse soutenue le 15 mai 1990.

SALOTTI S. 1992. Filtrage flou et représentation centrée objet pour raisonner par analogie : le système FLORAN. Thèse de l'université Paris-sud (2339) soutenue le 4 décembre 1992

SANFELIU A., FU K. 1983. A Distance Measure between Attributed Relational Graphs for Pattern Recognition. In *IEEE Transactions on System, Man and Cybernetics*. 13. 353-362.

SNEATH P.H.A., SOKAL R. 1973. *Numerical Taxonomy*. Freeman.

TVERSKY A. 1977. Features of similarity. In *Readings in Cognitive Science*. Morgan Kaufmann 1988. (from *Psychological Review* 84, 327-352, 1977).

VOß (Ed) 1994. *Similarity Concepts and Retreival Methods*. FABEL project Technical Report number 13. Gesselschaft für Mathematik und Datenverarbeitung (GMB). Sankt Augustin.

WONG A., YOU M. 1985. Entropy and Distance of Random Graphs with Application to Structural Pattern Recognition. In *Transactions on Pattern Analysis and Machine Intelligence*. 7. 599-609.