THE ANALYSIS OF USER'S QUERY THROUGHT LEARNING: HOW SIFADE DIAGNOSES USER'S GOALS

Bernadette BOUCHON *, Charles Albert TIJUS** & Mohamed-Nazih OMRI*

*LAFORIA-IBP
Université Paris 6, boite 169
4, Place Jussieu, 75252 Paris Cedex 05
Tel : (33-1) 44 27 70 10 - Fax : (33-1) 44 27 70 00
Email : omri@laforia.ibp.fr

**Laboratoire de Psychologie Cognitive, URA CNRS 1297
Université Paris 8 - 2, rue de la liberté, 93526 Saint-Denis - France

RESUME: Ce rapport présente une méthode de représentation des connaissances procédurales basée sur la notion de classes d'objets flous. Les valeurs d'attributs, définissant l'ensemble des classes du système d'aide à l'utilisation de dispositifs technique, présentent la distinction entre ensemble de valeurs nécessaires et ensemble de valeurs supplémentaires ou possibles. On définit également pour chaque valeur d'attribut une zone de valeurs utilisateurs permettant de faciliter l'identification et l'intégration des objets et procédures fournis par un interlocuteur novice en termes du langage naturel, pour les ranger dans des catégories de classes d'objets. L'intégration de valeurs floues dans la description de classes d'objets entraîne une valuation des liens hiérarchiques entre classes d'une part et entre instances et classes d'autre part.

MOTS-CLES: Classes floues, attributs flous, zones floues, imprécision, incertitude, représentation des connaissances procédurales, aide à l'utilisation de dispositifs techniques.

Abstract

This paper presents a method based on fuzzy object's class notion that is used to store procedural knowledge representation, to diagnose goals of planning actions and to provide on-line instructions for users of technical systems. Both a fuzzy range of uncertain values and a fuzzy range of supplementary values are specified for attributes describing object's classes and procedures applied to these classes. Various kinds of inclusion relations are defined between classes, as well as membership relations between instances and classes. Finally, in order to facilitate the analysis of user's query through learning, we define for each attribute a fuzzy user's area.

Keywords : Fuzzy objects, uncertainty, vagueness, procedural knowledge representation, user's assistance, fuzzy instancies, fuzzy attributes, fuzzy areas.

1. Expert Systems for User's Assistance

In order to provide on-line assistance, there are formalisable models of user's knowledge [2, 6, 11, 12] which provide the simulation the carrying out of tasks issued of user's procedures through human-computer interaction. These models seem suitable to be used as methods for device evaluation and for the conception of "on-line instructions" for effective planning and technical decision support. However, these models do not capture the intention of the user and do not supply the understanding of "how the system works" when an applied procedure does not provide the predicted results.

Learning how to use a device is mainly a learning by doing process. Man-machine studies show that users do not read documentation: they start using a device with the knowledge of what they know from analogous devices in quite like problem solving situations [3, 17]. Doing so, applied procedures usually failed and when new procedures are discovered, they are valid in certain states of the device but not in others. It follows that we must conceive an intelligent user's assistance with two obvious needs: (i) assistance for the choice of a valid procedure given the state of the device and (ii) assistance when a procedure failed in order to provide the understanding of why the user did not get the results he expected. This require mainly the diagnosis of the user's goal.

SIFADE is an expert-system that diagnoses the goal of the user from the point of view of the functioning of the system, given the task of the user. The central core of SIFADE is an "ideal user knowledge-representation" which is the valid knowledge of objects and procedures that describes the "how to use it", given the "how it works". The ideal user knowledge is used to provide on-line instructions after goals diagnosis. This enables the novice to evaluate applied procedures and to solve task-problem by learning new procedures. The on-line instructions mechanisms of our Intelligent Assistance System can be seen as a supervisor of task execution that has the ideal user's knowledge of prerequisites of procedures, of subgoals structure, of the semantic network of the elements of the device where applied procedures are used as properties as well as the knowledge of perceptible and imperceptible effects of user's actions.

The main expertise is to diagnose the user's goal and intention. There is a diagnosis stage because the assistance-system might be able to evaluate the level of the user's expertise and the task situation, in order to provide on-line instructions that are able to help the learning of valid procedures and of planning the execution of the current task. From actions on the task-device, the Assistance System looks for procedures that contain those actions, given that some may be invalid. But that is often the case that the Assistance System must know what is the user's intention with an interactive dialogue. In such situations, the Expert Assistant System tries to match the knowledge of the semantic network (both objects and procedures) of the ideal user's knowledge to items provided by users in natural language.

2. Matching User's knowledge to Ideal User's Network

Given the polysemic aspects of natural language (verbs and nouns), but the necessity of a man-machine interface involving queries of users, the problem that is under investigation is how to match the content of a query (the label of an object and the label of a procedure applied to this object) to their corresponding items (class of objects and procedures as properties) in the ideal user's network.

The example of the technical system we consider is a Word processor software, with objects such as "chain-of-characters", and procedures such as "cut" or "copy". For a novice user of the software, the list of standard denominations is not obvious and he often would like to ask an expert operator about how to execute an action such as "how to rub letters". The system SIFADE [0] we propose takes the place of an expert user. If it does not understand the meaning of an instruction, it discusses with the user until it is able to interpret the query in its own language. With the learning of new words in natural language as the interpretation which was produced in agreement with the user, the system improves its representation scheme at each experiment with a new user and, in addition, takes advantage of previous discussions with users : (i) the standard objects and procedures recognized by the software are described in a semantic network where procedures stand for properties of objects, (ii) as the queries of a user are expressed in natural language and as they correspond more or less to these standard denominations, the system establishes fuzzy connections between its primary knowledge and the new labels of objects or procedures expressed by the user.

For instance, since a new attribute can be identified with more than one primary attribute, we found convenient to use degrees of appropriateness of each such identification. For this purpose, we used the framework of fuzzy set theory and we describe fuzzy characterizations of classes of objects with procedures as properties. Possibility and/or certainty measures enable us to quantify the opinion of users regarding the degree of similarity or convenience of associations between descriptions of both objects and procedures.

In the next section, we present the fuzzy object based representation of knowledge [7, 8]. The proposed system deals with degrees of the membership of an object or of a procedure formulated in the user's language with regard to one of the classes of the fuzzy semantic network. It also takes into account the degree of inclusion between fuzzy classes and the grade of membership of an instance to one or several classes. If Rossazza [14, 11] introduced the concept of fuzzy objects and Dubois and Prade [4] addressed the problem of typicality of class hierarchy, very few works have been proposed in this direction, specially as one way of planning actions.

3. The main description

To describe common sense knowledge, different kinds of relationships must be used. They are necessary, sufficient and supplementary for Torasso and Console [13], necessary and typical for Padgham [9]. At a theoretical level, the first organisation as defined by Torasso and Console appears to be powerful. However, since it is difficult to define the sufficient property in practical applications, it is not used [13]. In our present work, we distinguish necessary properties from supplementary ones and we do not consider sufficient conditions. It follows that each object or procedure is described in SIFADE by three fuzzy areas : a necessary, supplementary and an user area.

3.1 Fuzzy areas

The necessary area ZN(P, A) of a property P of an attribute A is the set of couples of values (necessary property, necessity coefficient) admissible for A relatively to P. This area is fuzzy because some values are less admissible than others.

Example:

attribute :	**procedure**
domain :	[Erase, Select, Cut, Paste, Go-to, Move]
Erase :	[**(Cut .8) (Move .75)**]

The value of ZN(Erase, Procedure) heavily represented is obtained from a necessity distribution. The latter is given by one expert using **Cut** and **Move** to design the system property **Erase**.

The supplementary area ZS(P, A) of a property P of an attribute A is the set of couples of values (supplementary property, possibility coefficient) admissible for A relatively to P. This area is also obtained from a distribution of possibility given by one system expert.

Example :

attribute :	**procedure**
domain :	[Erase, Select, Cut, Paste, Go-to, Move]
Erase :	[**(Select .8) (Move 1) (cut 1)**]

The user area ZU(P, A) of a property P of an attribute A is the set of couples of values (user property, possibility coefficient) admissible for A relatively to P. This set is obtained from the user reasoning.

Example:

attribute :	**procedure**
domain :	[Erase, Select, Cut, Paste, Go-to, Move]
Erase :	[**(Gum .8) (rub-out 1)**]

(gum .8) and **(rub-out 1)** are the values of ZU(Arase, procedure). They are given by one user to design the system property **Erase**.

Different fuzzy areas described in this passage allow to define attributes of objects classes.

3.2 Fuzzy attributes

Let $\{P_1, P_2,..., P_n\}$ the set of properties of an attribute A, the structure of an attribute of an object is:

attribute: #:[(P_1 (ZN_1) (ZS_1) (ZU_1))...(P_i (ZN_i) (ZS_i) (ZU_i)) ..(P_k (ZN_k) (ZS_k) (ZU_k))]

Example: procedure: [(**Erase** ((Cut .5)) ((Move 1)) ((gum .75)))
(**Select** ((Cut .25)) ((Go-to .75)) ((blacken 1)))]

Three kinds of attributes can therefore be defined: necessary attribute AN, supplementary attribute AS and user attribute AU. They are defined respectively by :

$$A^*(C) = Z^*(P_1, A)xZ^*(P_2, A)x...xZ^*(P_k, A)$$

where C is a class of objects and * designate N (Necessary) or S (Supplementary) or U (User).

These attributes are fuzzy as results of the Cartesian products of fuzzy areas. The set of attributes describes classes and instances. These classes and instances are only defined by the sets of necessary and supplementary attributes. Properties of AU are user properties which can not be used in the theoretical definition of classes and instances.

3.3 Classes of fuzzy objects

Lets $\{AN_1, AN_2, ..., AN_n\}$,$\{AS_1, AS_2, ..., AS_n\}$,$\{AU_1, AU_2, ..., AU_n\}$ sets of necessary, supplementary and user attributes defining a class C. We then define respectively necessary class CN, supplementary class CS and user class CU by :

$$C^* = \{ A^*_1(C)x...xA^*_n(C) \ / \ A^*_i \in C, i \in <1, n>\}.$$

* designate N (Necessary) or S (Supplementary) or U (User).

3.4 Fuzzy instances

Lets $\{AN_1, AN_2, ..., AN_n\}$,$\{AS_1, AS_2, ..., AS_n\}$,$\{AU_1, AU_2, ..., AU_n\}$ sets of necessary attributes, supplementary attributes and user attributes defining an instance I. We define respectively necessary instance IN, supplementary instance IS and user instance IU by :

$$I^* = \{ A^*_1(I)x...xA^*_n(I) \ / \ A^*_i \in I, i \in <1, n>\}.$$

* designate N (Necessary) or S (Supplementary) or U (User).

4. Hierarchical relationships

The integration of fuzzy properties; as "to efface", in the object description implies the valuation of relationships in [0, 1]. There's two kinds of relationships: the relation **kind-of** between two classes and **is-a** between a class and an instance. One class instead of being (or not being) a **kind-of** an other class may be this to some extend. Each kind of relationship is described by a couple of values (necessity coefficient, possibility coefficient). This couple of values is obtained from inclusion between areas and between attributes.

example

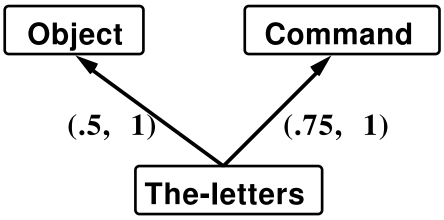

figure 4: **kind-of**

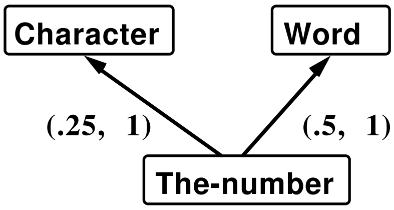

figure 5: **is-a**

In figure 4, The-letters appears as a **kind-of** Command with the couple of values (.75, 1), the first number is the necessity coefficient and the second is the possibility coefficient. It is also a **kind-of** object with (.5, 1) as a couple of values. In figure 5, The-number **is-a** Character with necessity coefficient .25, possibility coefficient 1 and **is-a** Word with .5 and 1 as necessity and possibility coefficients respectively.

4.1 Inclusion between fuzzy areas

Lets $\{p_{q1}, p_{q2}, ..., p_{qk}\}$ the set of linguistic values of the property P_q of an attribute A of a class C, $\{n_{q1}, n_{q2}, ..., n_{qk}\}$ the corresponding set of necessity degrees. We define the inclusion degree between two fuzzy areas by :

$$D(Z^*(P_q, A) \supset Z^*(S_p, A)) = \min_{1 \le j \le l} \{ \min_{\substack{p_{qi} = s_{pj} \\ 1 \le i \le k}} (n_{qi}, m_{pj})\}.$$

It can be applied for necessary and supplementary areas.

* designate N (necessary) or S (supplementary).

4.2 Inclusion relation between attributes

Lets i the number of properties P of an attribute A of a class C_1 and j the number of properties S of the same attribute A of a class C_2. We define the degree of inclusion between two necessary attributes and between two supplementary attributes by :

$$D((A^*, C_1) \supset (A^*, C_2)) = \min_{1 \le p \le j} \{ \min_{1 \le q \le i} \{ D(Z^*(P_q, A^*) \supset ZN(S_p, A^*))\}\}$$

* designate N (necessary) or S (supplementary).

4.3 Inclusion between fuzzy classes

We define the inclusion degree between two necessary classes and between two supplementary classes by :

$$D(C^*_1 \supset C^*_2) = \min_{1 \le i \le n} \{ D((A^*_i, C^*_1) \supset (A^*_i, C^*_2))\}.$$

The A^*_i are attributes defining the class C.

* designate N (necessary) or S (supplementary).

4.4 Instances and classes

In the case of a class and instances, we deal with degrees of membership of instances to a class. This degrees measure the physical representation of the class by the instances. It is obtained from inclusion degrees between fuzzy areas and between attributes. We define inclusion between necessary instances and classes and between supplementary instances and classes by :

$$D(C^* \supset I^*) = \min_{1 \le i \le n} \{ D((A^*_i, C^*) \supset (A^*_i, I^*))\}.$$

* designate N (necessary) or S (supplementary).

In a classical way, if I is an instance of a class C and C is included in E, then I is instance of E. This property is preserved by the following rule :

$$\text{if } D(C \supset I) > 0 \text{ and } D(E \supset C) > 0 \text{ then } D(E \supset I) > 0 .$$

5. Goal diagnosis

Goal diagnosis is realised by tasks descriptions and from requests of the user. Tasks descriptions are provided with a decomposition of procedures that starts with the main goal decomposed into subgoals till primitive actions on the device :

For instance
To move [Word]
    cut [Word]
        select [Word]
            Insert before word [curser-key]
            Drag [CharactersOf Word]
        transfertToClipboard[SelectdItem] & delete [selectedItem]
            press [option&X-keys]
    move to the target position [Curser]
    past [Word]

From sequences of primitive actions (in bold), we try to interpret goals of the user, but in order to diagnose goals, we mainly need to interpret queries in natural language.

To interpret the user goal , we need the identification of both object and procedure which can be applied to this object.

Four situations can be considered. In the first one, both object and procedure are known. In this case, there is no ambiguity and SIFADE returns an eventual answer to the user. In the second situation, where the object is known and the procedure is unknown, SIFADE proposes a set of procedures that can be applied to this object, it asks the user to associate a value between 0 and 1 to every procedure. The procedure in which the associated value is the greatest one is selected by SIFADE to build an equivalent query and returns an eventual answer. When the user is not satisfied, SIFADE considers the next procedure with the greatest associated value, when the last selected one has been removed. The process continues until all possibities have been sold out or satisfied the user. In the third situation, the object is unknown and the procedure is known. The same principle as in the second situation is used. The problem becomes more difficult in the fourth situation where both object and procedure are unknown. SIFADE uses a degree called stability degree associated to every object and every attribute. This degree is calculated by using vagueness or fuzzy areas. SIFADE proposes the list of the objects that present the smaller degrees and asks the user to associate a value between 0 and 1 for every element in the list. The set of procedures that can be applied to the object that presents the smallest stability degree is also proposed to the user who associates a value to every one. The process continues as in the second and third situation.

When the user is satisfied, the semantic network becomes richer with the object or the procedure expressed by the user.

6. Conclusion : the Knowledge Search Problem

The approach described here allows to use fuzzy objects classes for the representation of imprecise and uncertainty knowledge. This representation has a great practical interest due to the possibility to realize reasoning on classification with a fuzzy semantic network based system. For instance, the distinction between necessary, supplementary and user classes allows to take into account exceptions that may appear on fuzzy knowledge-base [7] and facilitate integration of user's objects in the base.

Nevertheless, the distinction between necessary and supplementary properties to improve description of common sense procedural knowledge and the means offered by this approach to manipulate this description seems insufficient to rapidly permit a successful knowledge identification. There is no other supplementary mean to contribute to the search of knowledge. The quest through a knowledge fuzzy-based is almost hazardous. There is no priority to analyse objects and often, the search of knowledge needs to go over all the semantic network. This has the inconvenience to reduce the quality of performance of the system. This problem is our first preoccupation. We think about a search mechanism to catch the more appropriate knowledge towards one situation. When an agent is incapable to reach his goal, a knowledge fuzzy-based system [8] provided with such a mechanism and facing more than one solution, will make a decision to help him.

REFERENCES

[1] Bouchon-Meunier B., Yager R.R., Zadeh L.A. (eds.) (1991). Uncertainty in knowledge bases. Lecture Notes in Computer Science, 521, Springer Verlag.

[2] Card, S.K., Moran, T.P., Newell A. (1983). The Psychology of Human-Computer Interaction. Hillsdale, N.J.: Erlbaum.

[3] Carrol J.M., Mack R.L. (1985). Metaphor, computing systems and active learning. International Journal of Man-Machine Studies, 212, 39-57.

[4] Dubois D. et Prade H. (1985), Théorie des possibilité- applications à la représentation des connaissances en informatique- masson.

[5] Dubois D. et Prade H. (1989), Vagueness and typicality in class hierarchies, Proceedings 3rd IFSA Congress, Seattle.

[6] Green T.R.G., Schiele F, Payne S.J. (1992). Formalisable models of user knowledge in human-computer interaction in working with computers: Theory versus Outcome Van Der Veer G.C., Green T.R.G., Hoc J.M, Murray D.M. (eds.) Londres Academic Press.

[7] Omri M.N. (1993). Interface Homme-Machine acceptant l'imprécision: SIFADE. Congrès INFORSID, Lille 93.

[8] Omri M.N., (1993). Définition de classes d'objets flous pour la représentation des connaissances procédurales imprécises et incertaines, 3ème Congrès National sur les applications des ensembles flous, Nîmes.

[9] Padgham L.(1988). A Model and Representation for Type Information and Its Use in Reasonning with Défaults. Proceedings of AAAI, vol 2, p 409-414.

[10] Rossazza J.P. (1990). utilisation de hiérarchie de classes floues pour la représentation des connaissances imprécises et sujettes à exceptions: système SORCIER. Thèse de 3ème cycle, Université Paul Sabatier.
[11] Rossazza J.P. (1992). Représentation centrée objet, 2ème Congrès National sur les applications des ensembles flous, Nîmes.
[12] Tauber M. (1988). On mental models and the user interface. Human-computer interaction in Working with Computers: Theory versus Outcome Van Der Veer G.C., Green, T.R.G., HOC J.M., Murray, D.M. (eds.) London. Academic Press.
[13] Tijus C.A., Poitrenaud S. (1992). Semantic Networks of Action. NATO conference on Psychological and Educational Foundations of Technology-Based Learning Environments. Kolymbari (Crete).
[14] Torasso P., Console L. (1989). Approximate Reasonning and Protypical knowledge. International journal of Approximate Reasonning, vol 3, no 2, pp 157-177.
[15] Vincent A.(1989). Filtrage sémantique dans une base de données imprécises et incertaines : un système souple autorisant la formulation de requêtes composites pondérées. Thèse de 3ème cycle, Université Paul Sabatier.
[16] Weber S. (1983). A general concept of fuzzy connectives, negations and applications based on norms and t-co-norms. Fuzzy Sets and Systems, 11, p115-134.
[17] Wright P., Barnard P. (1975). Just fill in this form: a review for designers, Applied Ergonomics, 6, 4, 213-220.
[18] Zadeh L.A. (1965). Fuzzy sets-information and control, vol.8, pp 338-353.
[19] Zadeh L.A. (1978). Fuzzy sets as basis for theory of possibility-Fuzzy sets and systems, vol. 1, N°1.

ERROR: undefined
OFFENDING COMMAND: ÿ

STACK: