Knowledge Representation and Inference

– An Approach from Object-Oriented Computing and Fuzzy Theory –

Kazuhiro Ogata *Norihisa Doi*

Department of Computer Science
Graduate School of Science and Technology
Keio University

Abstract

A new knowledge representation method based on object-oriented computing and fuzzy theory is proposed. Many types of knowledge representation have been studied, for example, *frame* [Minsky 75], *script* [Schank 77] and *semantic network* [Charniak 85]. In this paper, we assume that humans have limited masses of knowledge, and that the masses are not independent but mutually related with each other. Then we make a model of one's intellectual actions. We call a mass of knowledge a chunk, and we regard it as an object in object-oriented computing. For representing vague knowledge, we use fuzzy relations of type 2 which relates between chunks. Fuzzy relations of type 2 is an extension of fuzzy relations. We regard inference in this model as moving from one chunk to another by way of the relation between the chunks.

As an application of this model, an information retrieval system is dealt with.

1 Introduction

We call a space where knowledge is stored a knowledge space, and we suppose that knowledge is divided into modules and is put in the knowledge space. We call these modules, or masses of knowledge, "*chunks*". Chunks are not put at random in the knowledge space but are put according to the relations between chunks.

It is more natural if we do not regard the relations between chunks as *crisp* relations (i.e. having relations or not between chunks) but as vague(*fuzzy*) relations. To show the relations between chunks, we use fuzzy relations of type 2 which is an extension of fuzzy relations. The fuzzy relations of type 2 are obtained by extending the fuzzy relations in the same manner the fuzzy sets of type 2[Mizumoto 76] are obtained by extending the fuzzy sets[Zadeh 65]. A given chunk has relations with itself, in short self-relations.

In our proposed knowledge representation, an inference is regarded as consciousness moving from one chunk to another by way of the relation between the chunks. Generally, since a chunk has more than one relation with other chunks, we can infer in parallel by moving consciousness in parallel. We can implement the above inference mechanism by regarding a chunk as an object in object-oriented concurrent computing.

The purpose of our proposed knowledge representation is to model the flexibility of one's thought process and information processing with object-oriented computing and fuzzy theory. By taking the case of information retrieval as an example, it will be shown that a machine based on a model of human thought process and information processing can be very useful. An example of information retrieval in using fuzzy theory is in [Miyamoto 89].

However, our approach is different from it in that we use fuzzy relations of type 2 for relations between keywords and we retrieve documents in stages and also in parallel by grouping keywords.

2 Relations between Chunks and Inference

In this section, first we mention the reason we connect chunks with fuzzy relations of type 2. Then we describe a method of inferring by using the fuzzy relations of type 2.

2.1 The Relations between Chunks

The fuzzy sets of type 2 can be defined as an extension of ordinary fuzzy sets. A fuzzy set of type 2 A in a universal set X is characterized by a fuzzy membership function μ_A which is a fuzzy set between $[0,1]$. In general, if we express the mapping of set U to set V as V^U, we can express the fuzzy membership function as follows. [1]

$$\mu_A : X \to [0,1]^J \tag{1}$$

where J is the subset of $[0,1]$. We also call $\mu_A(x)$ a fuzzy grade, where x is an element of X.

We can similarly define fuzzy relations of type 2 as an extension of ordinary fuzzy relations. The fuzzy relations of type 2 are the fuzzy sets of type 2 where the cartesian product $X \times Y$(X and Y are sets) is a universal set and are characterized by the following fuzzy membership function μ_{R_2}.

$$\mu_{R_2} : X \times Y \to [0,1]^J \tag{2}$$

The purpose of our proposed knowledge representation is to model the flexibility of one's thought process and information processing with object-oriented computing and fuzzy theory. Next we investigate one's thought process and information processing [Sugeno 87].

We may regard "language" as an important element in one's thought process. In our everyday life, whenever we think consciously, we always think with language. We can not imagine thinking without any language. So we can consider language and thinking to be the same thing. Like thinking, communication is one of the fundamental human actions, and it is also done by using a language. So we can say that language is indispensable in everyday life. We can also say that language is essential for the human existence. In fact, linguists claim that what distinguishes human beings from other animals is not the use of tools but the use of language.

Language is indispensable in one's thought process and fuzziness always exists in languages. Generally, the ambiguity in a language means multiple meanings. It is inevitable that the number of words in languages are finite because the number of alphabetic characters is finite and the length of words is limited. By contrast, the things language has to describe is infinite. This must be the principal reason for the occurrence of multiple meanings. But the general ambiguity of language is not multiple meanings but the vagueness which exists in each meaning. The term "articulation" in the field of linguistics means to cut off part

[1] Since the grade of fuzzy set of type 2 is a fuzzy set in $J \subseteq [0,1]$, the ordinary fuzzy set is renamed as a fuzzy set of type 1. Similarly, we can define a fuzzy set of type n ($n = 1, 2, \cdots$) as follows.

$$\mu_A : X \to [0,1]^{J_1^{J_2^{\cdot^{\cdot^{\cdot^{J_{n-1}}}}}}}$$

where J_1, J_2, $\cdots$, J_{n-1} are the subsets of $[0,1]$.

of the continuous world and assign a word to it. Since language itself is not continuous, we can regard articulation as the dispersion of the continuous world. In linguistics, articulation is emphasized to be arbitrary as it is independent of the structure of the objective world, although it is dependent on the subject's arbitrariness. For example, rainbow is articulated into seven colors in Japanese but six colors in English (i.e. red, orange, yellow, green, blue, purple). So articulation is indicated to be arbitrary and independent of the optical structure of a rainbow. The region we cut off from the objective world is labeled by using language. What this label points to seems to be a definite object. But if we look at its perimeter, we will find some indefinable vagueness. This vagueness inevitably occurs because of the dispersion of the continuous world with incontinuous language.

As shown above, ambiguity(fuzziness) in language always exists. We can also say that the existence of language depends on vagueness. So we can also say that vagueness in thinking always exists. The unquestionable difference between humans and machines is that we human beings can recognize vagueness in the objective world, and effectively judge and make a decision even in uncertain situations without excluding vagueness in the thought process .

Since we can consider language and one's thought to be the same thing, it seems appropriate to express relations between chunks, or masses of knowledge used in one's thought process, with natural language. For example, we can express the similarity between concepts that knowledge has by using the terms, "relation very","relation more or less", "relation slightly" and so on. It is clear that we can not express these relations by using two-valued logic. It is possible to express these relations by using numbers between $[0,1]$. But since we express relations between knowledge with natural language, we may lose the vagueness in the concept that knowledge has and the ambiguity in language. So, it seems more appropriate to express relations between knowledge by using fuzzy sets. In short, we express relations between knowledge(chunks) by using fuzzy relations of type 2 which is an extension of fuzzy relations.

2.2 A Method of Inference

We use fuzzy relations of type 2 for inference. First we describe the composition of fuzzy relations of type 2, and then parallel inference with an object-oriented concurrent computing.

2.2.1 Composition of Fuzzy Relations of Type 2

We use two operators $\sqcup$ and $\sqcap$ in the composition of fuzzy relations of type 2. We can define $\sqcup$ and $\sqcap$ as an extension of $\vee$ (max) and $\wedge$ (min) defined in the interval $[0,1]$ by using the extension principle [Zadeh 75a]. Let X and Y be fuzzy sets of J(a subset of $[0,1]$), and $\sqcup$ and $\sqcap$ are defined as follows, where $x_i \in X$, $y_i \in Y$.

$$\begin{aligned} X \sqcup Y = & \ (\textstyle\sum_i \mu_X(x_i)/x_i) \sqcup (\textstyle\sum_j \mu_Y(y_j)/y_j) \\ = & \ \textstyle\sum_{i,j} (\mu_X(x_i) \wedge \mu_Y(y_j))/(x_i \vee y_j) \end{aligned} \tag{3}$$

$$\begin{aligned} X \sqcap Y = & \ (\textstyle\sum_i \mu_X(x_i)/x_i) \sqcap (\textstyle\sum_j \mu_Y(y_j)/y_j) \\ = & \ \textstyle\sum_{i,j} (\mu_X(x_i) \wedge \mu_Y(y_j))/(x_i \wedge y_j) \end{aligned} \tag{4}$$

Let X, Y and Z be sets. We can also define the composition of fuzzy relations of type 2 $R \circ S$, where $R \subset X \times Y$ and $S \subset Z \times Y$, as follows.

$$R \circ S \Leftrightarrow \mu_{R \circ S}(x,z) = \sqcup_y [\mu_R(x,y) \sqcap \mu_S(z,y)], \quad x \in X, y \in Y, z \in Z \tag{5}$$

2.2.2 Parallel Inference

Suppose each chunk has a state of knowledge representing the state of the chunk. The state of knowledge is the knowledge in the chunk and their respective fuzzy grades.

We regard one chunk as one object in object-oriented concurrent computing. An inference is regarded as consciousness moving from one chunk to another by way of the relation between the chunks. Using the concept of object-oriented computing, we represent this inference as follows. Consciousness moving from one chunk to another is equal to sending a message. The state of knowledge in the chunk is sent as the message. The chunk that receives the message makes its own state of knowledge with the received state of knowledge and the relation between the chunks, or to be exact, the composition of fuzzy relations of type 2 of the received state and the relation between chunks.

Generally, since a chunk has more than one relation with other chunks, when we start inferring at a chunk, we do not send a message one by one. We can infer in parallel by sending messages to all the chunks related to the chunk at the same time.

3 Application to Information Retrieval

We deal with information retrieval as an example for our proposed knowledge representation. We regard knowledge and chunk as keywords and their subsets respectively. Each keyword has documents which has the concept that the keyword has.

Let's compare information retrieval done by using a machine and by a human at a reference corner in a library. When we retrieve documents with a machine, it is necessary to know the keywords representing exactly the contents of the searched documents. But, with a human in a reference corner, even if we do not know the exact keywords for the searched documents, we can find corresponding documents. In case of retrieving with a human, it may be inferior to a machine in speed and in accuracy. But even if we do not know the exact keywords, we can find needed documents because of the flexibility of one's thought process and information processing, and because we use approaches which do not make a fetish of precision and exactness but are tolerant with imprecision and partial truths. This application attempts to realize this human characteristic.

In this section, first we describe the inference method in the information retrieval system and explain the inference process with an example. Next we describe the objects which we need for the implementation of the information retrieval system. We also show an example of the information retrieval system which we implemented with the object-oriented concurrent language ABCL/1 [Yonezawa 87].

3.1 Inference Method in Information Retrieval System

We need two fuzzy relations of type 2. One is a relation between the user and keywords, and the other is a relation between keywords. The relation between the user and a keyword shows how much the user is interested in the keyword. The relation between keywords shows how much one keyword is related to another.

An inference in the information retrieval system means that when relation K between keywords is given and then relation U between the user and a keyword is given, we infer from K and U a new relation U' between the user and a keyword. We use a composition in fuzzy relations of type 2 for inferring U' from K and U:

$$\begin{aligned} &U' = U \circ K \\ &\Leftrightarrow \mu_{U'}(user, key') = \sqcup_{key}[\mu_U(user, key) \sqcap \mu_K(key', key)], \qquad (6) \\ &\qquad (user, key) \in U, (key', key) \in K, (user, key') \in U' \end{aligned}$$

In general, the set of fuzzy grades is not closed under the logical connectives. So we must transform the result of the composition into a given fuzzy grade by using a concept called linguistic approximate[Zadeh 75].

When we transform a result of the composition into a given fuzzy grade, we should select the most similar fuzzy grade to the result. We express the similarity between fuzzy grades with Euclidean distance as

$$d(S, S^*) = \sqrt{\sum_i (\mu_S(x_i) - \mu_{S^*}(x_i))^2} \tag{7}$$

where S and S^* are fuzzy grades. The smaller d is, the more similar S is with S^*.

By defining the similarity between fuzzy grades as (7), a number of possible fuzzy grades may be gotten for a result. In this case, we choose the greatest fuzzy grade.

3.2 A Method of Inference

For example, we give a set of keywords as

$$\begin{aligned} K = \quad & \{key1, key2, key3, key4, key5, key6, \\ & key7, key8, key9, key10, key11, key12\} \end{aligned}$$

We divide the universal set K into four subsets $K1$, $K2$, $K3$, $K4$. There are direct relations between only $K1$ and $K2$, $K1$ and $K4$, $K2$ and $K3$. For instance, each subset is as follows.

$$\begin{aligned} K1 &= \{key1, key2, key3\}, \quad & K2 &= \{key4, key5, key6\} \\ K3 &= \{key7, key8, key9\}, \quad & K4 &= \{key10, key11, key12\} \end{aligned}$$

We also define each keywords for some documents as follows.

$$\begin{array}{l} key1 \cdots book1, \quad key2 \cdots book2, \quad key3 \cdots book3 \\ key4 \cdots book4, \quad key5 \cdots book5 \\ key7 \cdots book6, book7 \\ key11 \cdots book8, book9, book10, book11 \end{array}$$

We relate between keywords by using fuzzy relations of type 2.

We regard each subset as an object chunk. These relations are as follows

$$\begin{array}{ll} K1 \cdots chunk1, & K2 \cdots chunk2 \\ K3 \cdots chunk3, & K4 \cdots chunk4 \end{array}$$

The above relations are shown in Figure 1($Rij(i, j = 1, 2, 3, 4)$ is the relation between *chunki* and *chunkj*).

Inputs to this information retrieval system are conjunctions and disjunctions of keywords. We do not allow negation as inputs. We also assume the followings.

- Object *createChunk* creates each *chunk* and object *createChunki* ($i = 1, 2, \cdots$) has information regarding each chunk(keywords, relations between keywords, etc). For instance, when creating *chunk3*, first *createChunk3* is sent a message [:new] and then *createChunk3* sends a message [:create arg1 arg2 $\cdots$](arg1, arg2,$\cdots$ are information regarding chunk3) to *createChunk*.

- There is an object *retrievalSystem* and an object *controller*. Object *controller* is created by object *createController*.

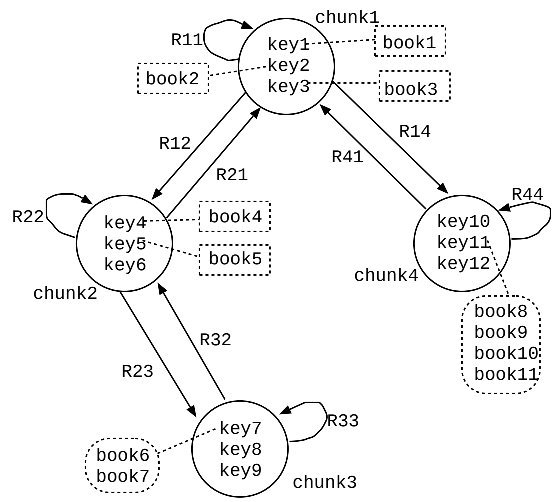

Figure 1: The Relation between Chunks

Given the structure shown in Figure 1, the process of retrieval to the inputs (*key*1 *and key*7) *or key*4 is as follows.

1. Stage 0

   (a) Object *retrievalSystem* receives (*key*1 *and key*7) *or key*4 as a message.

   (b) The *retrievalSystem* transforms the input (*key*1 *and key*7) *or key*4 into the conjunctive normal form (*key*1 *or key*4) *and* (*key*7 *or key*4).

   (c) The *retrievalSystem* divides keywords into clauses. *Key*1 *or key*4 and *key*7 *or key*4 will be called *clause*1 and *clause*2 respectively.

   (d) The *retrievalSystem* creates the same number of *controllers* as clauses, in this case two, by sending a message [:new] to *createController*. The two *controllers* will be called *controller*1 and *controller*2.

   (e) The *retrievalSystem* sends *clause*1 and *clause*2 to *controller*1 and *controller*2 respectively as messages.

   (f) Each of the *controllers* checks which chunk has the keywords in *clause*1 and *clause*2. In this case, *clause*1's *key*1 and *key*4 belongs to *chunk*1 and *chunk*2 respectively, *clause*2's *key*7 and *key*4 belongs to *chunk*3 and *chunk*2 respectively.

   (g) **controller1** — *Controller*1 sends a message [:new] to *createChunk*1 and *createChunk*2, and creates *chunk*1 and *chunk*2 respectively. The system infers at *chunk*1 and *chunk*2. The results of the inference in *chunk*1 are *key*1, *key*2, *key*3 and their respective fuzzy grades with the user. In *chunk*2, *key*4, *key*5, *key*6 and their respective fuzzy grades with the user are the results. So the results of this *controller* are *key*1, *key*2, *key*3, *key*4, *key*5, *key*6 and their respective fuzzy grades with the user.

   controller2 — *Controller*2 sends a message [:new] to *createChunk*3 and *createChunk*2, and creates *chunk*3 and *chunk*2 respectively. The system infers at *chunk*3 and *chunk*2. The results of the inference in *chunk*3 are *key*7, *key*8, *key*9 and the respective fuzzy grades with the user. In chunk2, *key*4, *key*5, *key*6 and the respective fuzzy grades with the user are the results. So the results of this *controller* are *key*4, *key*5, *key*6, *key*7, *key*8, *key*9 and their respective fuzzy grades with the user.

(h) The results of stage 0 of the inference is the conjunction of the fuzzy logic [Zadeh 75] of the results in the *controller*1 and the *controller*2. In short, they are *key*4, *key*5, *key*6 and their respective fuzzy grades with the user. So what the system shows to the user are documents related to *key*4, *key*5, *key*6 and their respective fuzzy grades with the user. The fuzzy grade for the degree of interest for the user in the document is the grade between the keyword and the user. For example, suppose that the keyword *key*4 relates directly to documents *d*1, ..., *dm* and the fuzzy grades between key4 and the user is "interest little". Then the fuzzy grade for the documents *d*1, ..., *dm* will also be "interest little".

2. Stage 1

   (a) **controller1** — In the inference starting at *chunk*1, since *chunk*1 relates to *chunk*2 and *chunk*4, *chunk*1 sends a message [:new] to *createChunk*2 and *createChunk*4, and creates *chunk2* and *chunk*4 respectively. *Chunk*1 sends the result of stage 0 of inference (at *chunk*1) as a message to *chunk*2 and *chunk*4 where inference is to continue. In the same way, in the inference starting at *chunk*2, inference continues at *chunk*1 and *chunk*3. The result of the *controller*1 is the disjunction of the fuzzy logic of the results of the inference started at chunk1 and the results of the inference started at *chunk*2. The result is *key*1, *key*2, ..., *key*12 and their respective fuzzy grades with the user.

   controller2 — In the inference starting at *chunk*3, the system continues inferring at *chunk*2. In the inference starting at the *chunk*2, the system continues inferring at *chunk*1 and *chunk*3. The result of the *controller*2 is the disjunction of the fuzzy logic the results of the inference started at *chunk*1 and the results of the inference started at *chunk*3. The result is *key*1, *key*2, ..., *key*9 and their respective fuzzy grades with the user.

   (b) The result of stage 1 of the inference is the conjunction of the results of *controller*1 and the results of *controller*2. That is, they are *key*1, *key*2, ..., *key*9 and their respective fuzzy grades with the user. So what the system shows to the user are documents related to *key*1, *key*2, ..., *key*9 and their respective fuzzy grades with the user.

Should we want to infer more deeply, we should infer stage 2, stage 3, and so on of the inference. We show the process of this inference in Figure 2.

3.3 Representation of Relations

We use "relation-very", "relation-more-or-less", "interest" and "interest-slightly" as the fuzzy grades between keywords, and between the user and a keyword. Next, we describe how these fuzzy grades are represented.

For example, suppose we define a universal set of fuzzy grades as

$$J = 0 + 0.1 + 0.2 + 0.3 + 0.4 + 0.5 + 0.6 + 0.7 + 0.8 + 0.9 + 1 \tag{8}$$

where + stands for the union in fuzzy sets. In this case, the fuzzy grades "relation" and "interest" are given as follows.

$$\text{relation} = \text{interest} = 0.5/0.7 + 0.7/0.8 + 0.9/0.9 + 1/1 \tag{9}$$

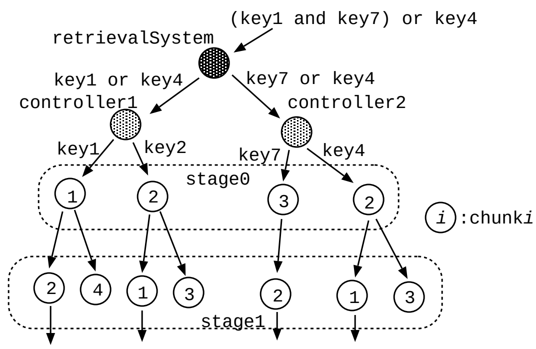

Figure 2: The process of the inference

If we define the fuzzy grades "relation" and "interest" as in (9), we can define the other fuzzy grades, for example, "relation-very","relation-almost","interest-more-or-less","interest-rather" , etc , with the concept of a linguistic hedge[Zadeh 72] as follows.

$$\begin{array}{l} \text{relation-very} = 0.25/0.7 + 0.49/0.8 + 0.81/0.9 + 1/1 \\ \text{relation-almost} = 0.5/0.7 + 0.6/0.8 + 1/0.9 + 1/1 \\ \text{interest-more-or-less} = 0.3/0.5 + 0.5/0.6 + 0.7/0.7 + 0.9/0.8 + 1/0.9 + 1/1 \\ \text{interest-rather} = 0.27/0.7 + 1/0.8 + 0.39/0.9 \end{array} \quad (10)$$

We can express relations between keywords and relations between the user and a keyword by using matrices, and use the above fuzzy grades as their elements. For example, $R12$ of Figure 1 is as follows.

$$R12 = \begin{bmatrix} \text{relation} & \text{relation-rather} & \text{relation-almost} \\ \text{relation-more-or-less} & \text{relation} & \text{relation} \\ \text{relation-almost} & \text{relation} & \text{relation-rather} \end{bmatrix} \quad (11)$$

The elements of line 1 in the above matrix "relation", "relation-rather" and "relation-almost" express the fuzzy grades of the relations between *chunk*1 and *chunk*4, *chunk*5 and *chunk*6.

Let $Ru1$ be the relation between the user and keywords in *chunk*1(Figure 1), and then we can express $Ru1$ as follows.

$$Ru1 = \begin{bmatrix} \text{relation-very} & \text{relation-slightly} & \text{relation} \end{bmatrix} \quad (12)$$

When representing the composition of fuzzy relations of type 2 with matrices, the operations of addition and multiplication are replaced by $\sqcup$ and $\sqcap$ respectively.

3.4 An Implementation of the Information Retrieval System with an Object-Oriented Concurrent Language

Roughly speaking, we need three objects in the information retrieval system. These objects are *retrievalSystem*, *controller* and *chunk*. Furthermore, for creating *controller* and *chunk* we have objects *createController* and *createChunk* respectively. We also have objects *createChunki* $(i = 1, 2, \cdots)$ for having the information regarding each *chunk*.

The role of these objects are as follows.

- **retrievalSystem** — This is the interface of this system. This handles the inputs and outputs, and decides how many *controllers* are to be created.

- **createController** — This creates object *controller*.
- **controller** — This decides how many *chunks* are to be created. A *controller* deals with a sentence connecting keywords by disjunction only, and each controller is connected by conjunction.
- **createChunk** — This creates object *chunk*.
- **chunk** — This holds a subset of keywords, fuzzy relations between keywords, etc. Inference takes place here.
- **createChunk**i ($i = 1, 2, \cdots, n$) — This contains the information regarding each chunk. It sends the information of a *chunk* as a message to *createChunk* where the *chunk* is created.

The following is an example of this information retrieval system.

```
<ABCL/1> [main <= [:query]]
*** Welcome to Fuzzy Retrieval System ***

*** Input Your Interested Keywords

? (or (and key1 key7) key4)

*** RESULT ***
(book4 interest-very)
(book5 interest)
**************

Would you like to continue?(yes/no)yes

*** RESULT ***
(book4 interest-very)
(book5 interest)
(book6 interest-rather)
(book7 interest-rather)
(book1 interest-rather)
(book2 interest-rather)
(book3 interest-rather)
**************

Would you like to continue?(yes/no)yes

*** RESULT ***
(book4 interest-very)
(book5 interest)
(book1 interest-rather)
(book2 interest-rather)
(book3 interest-rather)
(book8 interest-rather)
(book9 interest-rather)
(book10 interest-rather)
(book11 interest-rather)
```

```
(book6 interest-rather)
(book7 interest-rather)
**************

Would you like to continue?(yes/no)no

thank you!!!
Next Retrieval?(yes/no)no

good-bye!!!
<ABCL/1>
```

4 Conclusion

We proposed a new knowledge representation method and described an information retrieval system as its example. The purpose of this paper is to model the flexibility of one's thought process and information processing. We compared humans and machines in the case of information retrieval. With humans, even if we do not know the exact keywords for the searched documents, we can find corresponding documents. The purpose of this information retrieval system is to be able to find corresponding documents even without accurate knowledge.

We can apply our proposed knowledge representation and the information retrieval system to a distributed environment.

References

[Charniak 85] Charniak, E. and McDermott, D., *Introduction to Artificial Intelligence*, Addison-Wesley (1985).

[Minsky 75] Minsky, M., "A Framework for Representing Knowledge", In *The Psychology of Computer Vision*, Winston, P.H.(Ed.), McGraw-Hill (1975).

[Miyamoto 89] Miyamoto, S., "Two Approaches for Information Retrieval Through Fuzzy Associations", *IEEE Trans. Syst. Man Cybern.*, Vol. SMC-19, No. 1, 123-130 (1989).

[Mizumoto 76] Mizumoto, M. and Tanaka, K., "Some Properties of Fuzzy Sets of Type 2", *Inform. Control*, Vol. 31, 312-340 (1976).

[Schank 77] Schank, R.C., *Scripts, Plans, Goals, and Understanding*, Lawrence Erlbaum (1977).

[Sugeno 87] Sugeno, M., "On Recent Applications of Fuzzy Theory", *Mathematical Sciences*, No. 284, 5-9 (1987) (in Japanese).

[Yonezawa 87] Yonezawa, A., Shibayama, E., Takada, T. and Honda, Y., "Modelling and Programming in an Object-Oriented Concurrent Language ABCL/1", In *Object-Oriented Concurrent Programming*, Yonezawa, A. and Tokoro, M.(Eds.), The MIT Press (1987).

[Zadeh 65] Zadeh, L.A., "Fuzzy Sets", *Inform. Contr.*, Vol. 8, 338-353 (1965).

[Zadeh 72] Zadeh, L.A., "A Fuzzy-Set-Theoretic interpretation of linguistic hedges", *J. of Cybernetics*, Vol. 2, 4-34 (1972).

[Zadeh 75] Zadeh, L.A., "Fuzzy Logic and Approximate Reasoning", *Synthese*, Vol. 30, 407-428 (1975).

[Zadeh 75a] Zadeh, L.A., "The Concept of a Linguistic Variable and its Application to Approximate Reasoning–I", *Inform. Sciences*, Vol. 8, 199-249 (1975).