



# Fuzzy Indexing and Retrieval in Case-Based Systems

BING CHIANG JENG AND TING-PENG LIANG

Department of Information Management, National Sun Yat-sen University, Kaohsiung, Taiwan

**Abstract**—*Case-based reasoning is a technique recently developed to alleviate limitations of the rule-based expert systems. Instead of relying solely on rules, a case-based system maintains old cases in a case base. When a new problem is encountered, the system retrieves similar cases from the case base and constructs a solution to the new problem based on existing solutions. A key issue in case-based reasoning is how to index and retrieve similar cases. In this paper, we present a new approach that integrates fuzzy set concepts into the case indexing and retrieval process. This approach has a few advantages over existing methods. First, it allows numerical features to be converted into fuzzy terms to simplify the matching process. Second, it allows cases in different domains to be comparable. Finally, it allows greater flexibility in the retrieval of candidate cases.*

## 1. INTRODUCTION

RECENTLY, CASE-BASED REASONING (CBR) has received much attention in developing knowledge-based systems. A special issue that included papers on CBR theory and applications was published recently by *Expert Systems with Applications* (Liang & Turban, 1993). Unlike the traditional rule-based approach in which expert knowledge must be represented in "if-then" rules, a case-based system allows knowledge to be grouped and stored as cases. The development of this approach and advances in neural networks, genetic algorithms, and related techniques have surged as key tools of the second generation hybrid system (Hedberg, 1993; Liebowitz, 1993). A case is a combination of a problem state and its corresponding solution. A case-based system maintains a case base in which old cases are stored. When a new problem is encountered, instead of relying solely on chaining rules, the system retrieves similar cases from the case base and uses their solutions as a basis for constructing a solution to the new problem.

There are several advantages of the case-based approach. First, it provides a means for storing experience in expert systems. This is important because literature in cognitive science has indicated that a major difference between experts and novices is that the former relies more on experience (i.e., previously solved cases that they can remember), whereas the latter relies more on rules. Although rules are good at capturing general

principles for problem solving, the case base allows unique experience to be memorized. Furthermore, the case-based approach enables knowledge engineers to handle poorly structured domains such as legal litigation, strategic planning, and design. It is well-known that decisions are more an art than a science, and knowledge is very difficult to represent completely in rules in these domains. In fact, even human beings are trained using case methods in schools of law, business, and the arts. Finally, the case-based approach makes self-learning of expert systems easier. Because the case base maintains a bank of previously solved cases, learning is only a matter of representing and storing cases in the case base. When similar cases are cumulated to a certain number that warrants general rules, inductive learning methods can be applied to induce rules from the cases (Barletta, 1992; Hammond, 1989; Kolodner, 1991; Riesbeck & Schank, 1989; Simoudis, 1992; Slade, 1991; Stotler, 1992). Applications of CBR have been reported in many domains (e.g., Ketler, 1993; Riesbeck & Schank, 1989; Chi, Chen, & Kiang, 1993; Lee & Kim, 1993; Mott, 1993).

A typical CBR process includes the steps as shown in Figure 1. First, when a problem is encountered, key features of the problem are identified. These features are then used to retrieve one or more similar cases from the case base. Case retrieval may include feature match performed on case indices and actual retrieval of case data. If more than one case are found to be similar to the new problem, the one considered most similar to the new problem is chosen. Finally, the solution to the chosen case is adapted and modified to produce the solution to the new problem. The solved problem becomes a new case that can be saved in the case base. There are four key issues in the CBR process:

---

Note: Both authors contributed equally and are listed alphabetically.

Requests for reprints should be sent to Ting-Peng Liang, Department of Information Management, National Sun Yat-sen University, Kaohsiung, Taiwan 80424 ROC.

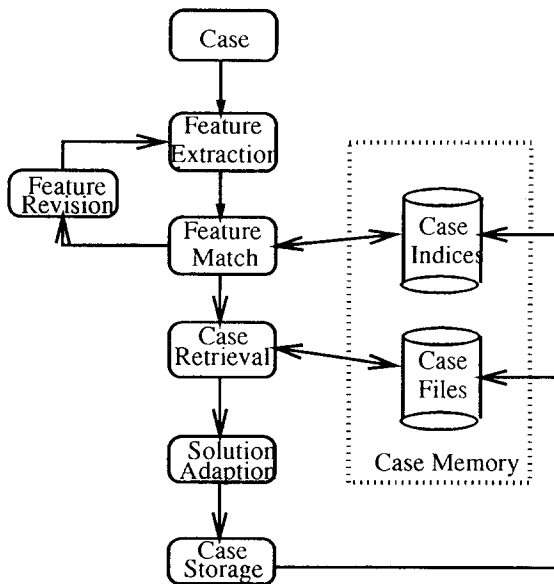


FIGURE 1. Process of case-based problem solving.

(a) identifying key features, (b) indexing and retrieving similar cases in the case base, (c) measuring case similarity to select the best match, and (d) modifying the existing solution to fit the new problem. In this paper, we focus our discussion on case indexing and retrieval.

Case indexing and retrieval are two separate but closely related processes. Since a case base may contain thousands of cases, case indices organize their key features to expedite the search process. Case retrieval searches the case base to find candidate cases that share significant features with the new case. Existing literature in case-based reasoning has proposed several mechanisms for case indexing and retrieval. For example, Kolodner (1988) presented a parallel approach to retrieving events from case memory. Seifert (1988) proposed a goal-oriented retrieval model. Sycara and Navinchandra (1989) proposed mechanisms for index transformation and generation to improve case retrieval. Veloso and Carbonell (1991) analyzed the tradeoffs involved in balancing the costs of retrieval and search. Wall, Donahue, and Hill (1988) and Cain, Pazzani, and Silverstein (1991) used domain semantics to facilitate case retrieval. A good review of early literature can be found in Rissland, Kolodner, and Waltz (1989).

Although these existing approaches have solved some problems in case indexing and retrieval, there are a few problems that remain unsolved. One is that most existing methods assume qualitative features such as weak light or powerful weapon, but provide little discussion about how to deal with numerical ones. In reality, cases often have both kinds of features. We need mechanisms that can convert numerical features into qualitative terms for indexing and retrieval. In this paper, we present an approach that applies fuzzy

sets concepts to case indexing and retrieval to alleviate the problem.

There are at least four advantages of using fuzzy indexing and retrieval. First, it allows numerical features to be converted into fuzzy terms to simplify comparison. For example, we can convert the age of a patient into a categorical scale (e.g., old, mid-age, or young). Second, fuzzy sets allow multiple indexing of a case on a single feature with different degrees of membership. This increases the flexibility of case matching. For example, a 50-year old patient may be classified as old (0.6) and mid-age (0.5), where 0.6 and 0.5 are the degrees that the 50-year-old patient is classified as old or mid-age, respectively. This allows the case to be considered as a candidate when we are looking for either an old patient or a mid-age patient. Third, fuzzy sets make it easier to transfer knowledge across domains. For instance, we have cases showing that persons older than 50 years of age (i.e., old persons) will need special effort to get a good job. We can use these cases to derive a guideline that computer software older than 2 years on the market (i.e., old software) will need special effort to make profit. The absolute age scales are different in these two domains, but the fuzzy transformation provides a bridge for comparison. Finally, fuzzy sets allow term modifiers to be used to increase the flexibility in case retrieval. For example, we can search *very old* patients from a case base containing *old* patients with possibilities ranging from .5 to 1.0. Here, *very* is a modifier of *old*, which can be used to modify the membership grade of *old* and result in a subset of *old* patients (considered *very old*) being retrieved. This enhances the flexibility of retrieval.

In the remainder of the paper, we shall present the fuzzy approach to case indexing and retrieval. First, the concepts of fuzzy sets and their values in case-based reasoning are presented. This is followed by our approach to fuzzy indexing and retrieval of cases. Finally, applications of the proposed approach and illustrative examples are discussed.

## 2. FUZZY SETS CONCEPTS

Fuzzy set theory was introduced by Zadeh in 1965 as a generalization of the conventional set theory (Zadeh, 1965). Its primary objectives are to represent and model the meaning of ambiguous terms such as *old*, *tall*, and *very beautiful* and process multiple memberships in classification. Traditionally, an object either belongs or does not belong to a set. The membership is binary. For example, a person who is classified as *honest* cannot be considered *not honest* at the same time. A person who is classified as an old man cannot be a young man at the same time. This is called *classical set* or *crisp set*. The characteristic function that represents whether an object  $x$  belongs to a set  $S$  is:

$$m_s(x) = \begin{cases} 1 & \text{if } x \text{ is an element of } S; \\ 0 & \text{otherwise.} \end{cases}$$

$$A \cup B \Leftrightarrow \forall x \in U,$$

$$m_{A \cup B}(x) = \max[m_A(x), m_B(x)]; \quad (4)$$

Although classical sets are useful, they have limitations. For instance, if we define *old* as someone whose age is 60 or older, then a 59-year-old person cannot be classified as old. In reality, human beings often adopt a more flexible approach. A person can be honest and dishonest with different degrees of possibility. For instance, a person may be 80% honest, which implies a possibility of 20% that he or she is dishonest. A 55-year-old person may be considered old and mid-age of different degrees at the same time.

A set that admits partial membership is called a *fuzzy set*. The membership function of a fuzzy set maps the domain of the set to an interval of [0,1]. Let  $X = \{x\}$  denote a space of objects, then a fuzzy set  $F$  in  $X$  is a set of ordered pairs  $F = \{(x, m_f(x)) | x \in X\}$ , where  $m_f(x)$  is called "the grade of membership of  $x$  in  $F$ ." Here, membership grades reflect the degree that an object belongs to a set. For instance, Figure 2 shows the membership function of *old* persons. We can see that the membership grade of a person who is older than 70 is 1.0, whereas the membership grade of a 60-year-old person is 0.66. Compared with classical sets, the fuzzy set representation does not use a single threshold value to define set membership. Instead, it allows a range of gray area for classification.

The membership function of a class may be represented as a mathematical equation. For example, the membership function of *old* persons as shown in Figure 2 can be represented as  $m_{old}(x) = (x - 40)/30$  if  $40 \leq x \leq 70$ , 0 if  $x < 40$ , and 1 if  $x > 70$ , whereas the membership function of *young* is  $m_{young}(x) = 1 - m_{old}(x)$ . Since crisp sets are a special case of fuzzy sets, most operations defined on crisp sets can also be applied to fuzzy sets. The following are some examples:

$$A = B \Leftrightarrow \forall x \in U, \quad m_A(x) = m_B(x); \quad (1)$$

$$\sim A \Leftrightarrow \forall x \in U, \quad m_{\sim A}(x) = 1 - m_A(x); \quad (2)$$

$$A \cap B \Leftrightarrow \forall x \in U,$$

$$m_{A \cap B}(x) = \min[m_A(x), m_B(x)]; \quad (3)$$

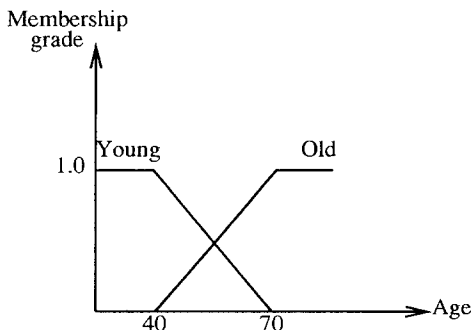


FIGURE 2. The membership functions of *old* and *young*.

In addition to fuzzy terms such as *old* or *good*, we often use *modifiers* such as *very*, *somewhat*, and *more or less* to describe reality. Another advantage of fuzzy sets is their ability in handling these modifiers in human languages. A typical approach adopted in fuzzy sets for handling modifiers is to add operations that can change the membership function of the fuzzy term the modifiers modify. Given the membership function of old patients  $m_{old}(x)$ , the membership function of *very old* patients is  $[m_{old}(x)]^2$ , whereas the membership function of *somewhat old* patients is  $[m_{old}(x)]^{1/2}$ .

Since fuzzy sets use possibilities rather than binary membership values, a hurdle value is often necessary to differentiate those considered highly likely to be a member of a set from those considered relatively unlikely. For instance, when we are looking for old patients, we may want to consider only those whose membership grades of "old" are above 0.5. This value is generally called an  $\alpha$ -cut. An object belongs to a set if its membership grade is greater than or equal to the  $\alpha$ -cut. In our previous example, if the  $\alpha$ -cut is set at 0.5 for age, then patients older than 55 are considered old, whereas patients older than 61 are very old.

### 3. FUZZY INDEXING AND RETRIEVAL OF CASES

Since case-based reasoning involves finding similar cases from the case base and using them to construct new solutions, indexing and retrieving of cases play critical roles in case-based problem solving. Unless the cases are properly indexed and ready for retrieval, they may not be useful. Generally speaking, case indexing and retrieving are implemented on the attribute level. A case is composed of many attributes available for indexing. For example, the risk of a firm may be assessed by a set of financial ratios such as quick ratio and inventory turnover. Case attributes can be divided into two categories: qualitative and quantitative. Qualitative attributes accept nominal values. A firm's quality of management, for instance, is a qualitative attribute whose value may be excellent, good, average, or poor. Quantitative attributes allow values to be measured on a numerical scale. Inventory turnover of a firm is an example of numerical attributes.

Fuzzy indexing and retrieval are useful in domains where cases have quantitative attributes. For cases with qualitative attributes only, indexing can be performed on attributes directly. For example, the risk of a firm can be assessed as high, medium, or low (three classes) and its quality of management can be classified as excellent, good, average, or poor (four classes). We can easily index firms by their risks or quality of manage-

ment. If we also want to include inventory turnover, however, indexing becomes more complicated. The value of inventory turnover can be any positive real number. Like age or other numerical attributes, it has an infinite number of possible values and is easier to index with a proper transformation into a few discrete classes.

Traditionally, we can choose hurdle values for converting quantitative values into qualitative classes. For example, we may define that young stands for the age range below 29, mid-age is between 30 and 49, and old is 50 and above. This transforms ages into three classes. Similarly, we may define rich people as persons whose wealth is worth more than a million dollars. This approach is useful in some situations. However, it has a few drawbacks. First, it does not differentiate different degrees of memberships. Persons having 1 million and 100 million are both classified as rich but, in reality, they are very different in their degrees of richness. Similarly, a person at 50 *may be* considered old but a person at 75 is *definitely* old. Using the fuzzy approach can overcome this problem.

Another problem with the traditional approach is that it does not provide adequate flexibility for handling marginal cases. A 50-year-old person is classified as old but a 49-year-and-360-day person is not old. This discrete nature can cause major problems in case-based reasoning. For instance, suppose we have three persons whose ages and wealth (30, \$999,999), (25 with \$200,000), and (60 with \$10 million) are in our case base and we need to find someone who is young and rich. Based on our previous definitions, none of them is qualified because the first is neither young nor rich, the second is young but not rich, whereas the third is rich but not young. A case-based system may retrieve the second or the third person based on partial match. The first one has no chance in the traditional approach, though he is very close in both age and wealth and may be the best match in reality.

The major value of fuzzy indexing and retrieval is that they can effectively offset the above problem in case retrieval. Fuzzy indexing and retrieval allow multiple class memberships to be defined on a single attribute. In other words, a person may be classified as old and young at the same time with different membership grades. In our previous example, fuzzy indexing may classify the first case as (old/.6, young/.5, rich/.8), which would make him qualify for the retrieval criteria of rich and young.

Fuzzy indexing is a two-stage process as shown in Figure 3. Quantitative attributes are first processed by the *fuzzifier* (called *fuzzification*) and then indexed on the resulting classes (*indexing*) before being stored in the case base. The fuzzification process includes the following steps:

1. When a case is encountered, quantitative attributes are identified;

2. For each quantitative attribute, proper classes are determined based on practical needs;
3. The membership function of each class and its associated  $\alpha$ -cut are determined;
4. Numerical values of each case are converted into proper classes for indexing.

To illustrate the fuzzification process, we use inventory turnover as an example. To simplify representation, a case consisting of firm risk, quality of management, and inventory turnover is represented as a three-tuple following the name of the case, that is, casename(risk value, value of quality of management, value of inventory turnover). Hence, Mitech(low, good, 4.0) means a firm called Mitech whose risk is low, quality of management is good, and inventory turnover is 4.0. When a case is to be stored and indexed in the case base, the fuzzifier finds that inventory turnover is quantitative that needs transformation. If business analysts usually classify inventory turnover into three levels: high, moderate, and low, then three membership functions must be constructed:

1.  $m_{\text{high}}(x) = \min(1, x/7)$ ;
2.  $m_{\text{moderate}}(x) = 1/(4-x)$  if  $x \leq 3$ ;  $1/(x-2)$  if  $x > 3$ ;
3.  $m_{\text{low}}(x) = 1 - \min(1, x/5)$ .

For Mitech(low, good, 4.0), the fuzzifier converts inventory turnover value 4.0 into membership grades of the respective classes: .57 for high, .5 for moderate, and .2 for low, which can be represented as (high/.57, moderate/.5, low/.2). These mean that the inventory turnover of the firm has a .57 possibility to be considered high, .5 possibility of moderate, and .2 possibility of low in the industry. If  $\alpha$ -cuts are set at .5, then Mitech can be classified as firms having high and moderate inventory turnovers (represented as (high/.57, moderate/.5)) and indexed accordingly. Following the same procedures, the cases in Table 1 can be organized by their classes in inventory turnover as shown in Figure 4.

Once cases are indexed and stored in the case base, they can be used for problem solving. When a new case

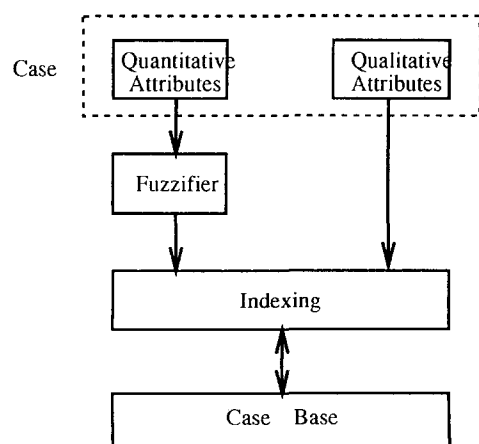


FIGURE 3. The fuzzy indexing process.

**TABLE 1**  
Cases in a Sample Case Base

Firm	Risk	Quality	Inventory Turnover	Fuzzy Inventory Turnover
A	High	Average	1.5	(low/.7)
B	High	Average	1.8	(low/.62)
C	High	Average	0.5	(low/.9)
D	High	Average	2.2	(low/.56, moderate/.55)
E	High	Poor	2.0	(low/.6, moderate/.5)
F	High	Poor	2.8	(moderate/.83)
G	High	Poor	1.8	(low/.64)
H	Medium	Average	2.6	(moderate/.71)
I	Medium	Average	3.2	(moderate/.83)
J	Medium	Average	4.5	(high/.64)
K	Medium	Good	3.5	(moderate/.67, high/.5)
L	Medium	Good	2.8	(moderate/.83)
M	Medium	Good	3.0	(moderate/1.0)
N	Medium	Excellent	2.5	(low/.5, moderate/.67)
O	Low	Good	3.8	(moderate/.56, high/.54)
P	Low	Good	5.0	(high/.71)
Q	Low	Good	6.5	(high/.93)
R	Low	Excellent	3.6	(moderate/.83, high/.51)
S	Low	Excellent	5.6	(high/.80)
T	Low	Excellent	6.3	(high/.90)
U	Low	Excellent	8.4	(high/1.0)

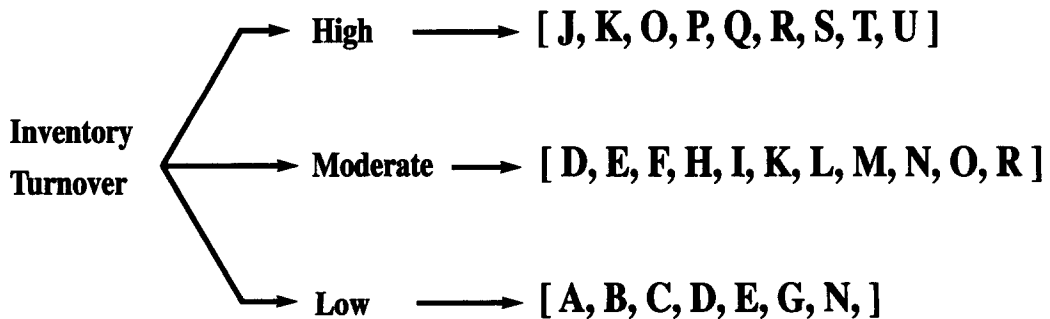
is encountered, the CBR engine searches the case base to retrieve similar cases. The retrieval process also needs fuzzy treatment if quantitative attributes are involved. The fuzzy retrieval process includes the following steps:

1. Quantitative attributes are converted into fuzzy terms based on membership functions defined in the fuzzifier;
2. The resulting fuzzy terms combined with known qualitative attributes are used as keys for searching similar cases;
3. The matched cases are retrieved as candidates, and the one that has the highest similarity is used to construct a solution to the new case.

Given the cases in Table 1, suppose we want to assess the risk of NewTech whose quality of management is good and inventory turnover is 4.1. If we try to match the attribute values with the data in the case base directly, none will be found because no firm has the same inventory turnover. Based on the membership function

defined previously, the inventory turnover of 4.1 can be converted to high/.59, and the case becomes NewTech(*X*, good, high/.59), where *X* stands for an unknown value. We can easily find that firms K, O, P, Q match the known attributes of the new case and can be used as bases for assessing the risk of NewTech.

Fuzzy retrieval often results in a set of candidate cases for reasoning. The issue following fuzzy retrieval is to find the most similar case among candidates. In our example, case K indicates that the risk is medium, whereas cases O, P, and Q indicate that the risk is low. There are several ways of finding the most similar case. For a classification problem, as in the example we use, the most straightforward way is to count the number of cases showing a particular result. Since three of the four candidate cases have low risk, we can conclude that the risk of NewTech is low. Of course, there are situations where this approach is not appropriate. In strategic planning problems, for example, we may need



**FIGURE 4.** Case organized by inventory turnover.

to find the most similar case based on certain similarity measures and revise its plan to suit the new situation.

Another approach is to use functions to measure the distance between the new and candidate cases and choose the one having the shortest distance as the most similar case. The distance can be compared on the original quantitative attribute value or on the converted fuzzy membership grades. If we use the original value, the distance function for attribute  $i$  can be defined as  $d_i = \text{abs}(x_{ij} - x_{in})$ , where  $x_{ij}$  and  $x_{in}$  are the value of attribute  $i$  for candidate  $j$  and new case  $n$ , respectively. If more than one quantitative attribute is involved, the overall distance is the sum of individual distances, that is,  $d = \sum d_i$ . In our example, case O has the shortest distance ( $4.1 - 3.8 = 0.3$ ) if the original values are used.

Similar results can be obtained if converted fuzzy grades are used. In this approach, distance is defined on each converted class and then aggregated for each attribute. That is,  $d_i = \sum_j \text{abs}(x_{ijk} - x_{ijn})$ , where  $x_{ijk}$  and  $x_{ijn}$  are the grades of attribute  $i$ , class  $j$ , for cases  $k$  and  $n$ , respectively. The overall distance  $d = \sum d_i$ . In the previous example, the distances for candidate cases are shown in Table 2. It happens that these two methods result in the same conclusion—case O is the most similar to NewTech. This, however, may not always be true. The selection of methods can have significant impact on the basis chosen for constructing new solutions. Detailed comparison and discussion are beyond the scope of this paper.

#### 4. APPLICATIONS OF THE FUZZY APPROACH

In addition to providing better indexing of cases in the case base, the fuzzy conversion of quantitative attributes presented in the previous section is very useful in at least three areas: extended retrieval flexibility, cross-sectional comparison, and more flexible induction of rules. We shall examine them in this section.

##### 4.1. Extended Retrieval Flexibility

In some cases, we may want to use the case base in a more flexible way. For instance, we may want to find firms with moderate inventory turnover and low risk or firms with very high inventory turnover. Without the fuzzy conversion, it would not be possible to meet these demands.

Using our sample data, if we want to find firms with moderate inventory turnover and low risk, the system will respond with cases O and R. From these cases, we can infer that the firm must have at least a good management team in order to have low risk. In addition, the inventory turnover value should be higher than 3.6 if an excellent management team exists and higher than 3.8 if a good management team exists.

Suppose we want to find firms with very high inventory turnover. Since *very* is a modifier, we can use the quadratic function defined in Section 2 to modify the membership function of inventory turnover. In other words,  $m_{\text{very high}}(x) = [m_{\text{high}}(x)]^2$ . Using this membership function, we can calculate that the membership grades of cases O, P, Q, R, S, T, and U belonging to the class of very high inventory turnover. The results are .29, .50, .86, .26, .64, .81, and 1, respectively. That is, the system will find that cases O and R do not have very high inventory turnover if .5 is used as the  $\alpha$ -cut. Similarly, we can also find cases whose inventory turnover is somewhat high, as long as we define the modifier *somewhat* properly. This makes the use of existing cases much more flexible.

##### 4.2. Cross-Sectional Comparison

Sometimes, a particular attribute has different properties in different sections. For example, proper inventory turnovers are often affected by the nature of the industry. A turnover ratio of 4.0 may be considered high in one industry, but low in another. Fuzzy conversion allows this characteristic to be handled properly by defining different membership functions. Suppose we need to assess the risk of a firm, Fasturn, in another industry whose membership functions of inventory turnover are defined as follows:

1.  $m_{\text{high}}(x) = \min(1, x/14)$ ;
2.  $m_{\text{moderate}}(x) = 2/(8 - x)$  if  $x \leq 6$ ;  $2/(x - 4)$  if  $x > 6$ ;
3.  $m_{\text{low}}(x) = 1 - \min(1, x/10)$ .

We know that the firm's management is good and its inventory turnover is 6.5. If we search through the case base in Table 1 directly, we find that case Q matches the given information of Fasturn exactly and conclude that the risk of Fasturn is low. After applying the proper membership function, however, we can find that, instead of having a high inventory turnover, Fasturn's inventory turnover is moderate in the industry, (i.e., membership grades = low/.35, moderate/.80, and

TABLE 2  
Distances Between NewTech and Candidate Cases

Firm	Inventory Turnover	Fuzzy Inventory Turnover	Distance
K	3.5	(low/.3, moderate/.67, high/.5)	.40
O	3.8	(low/.24, moderate/.56, high/.54)	.19
P	5.0	(low/0, moderate/.33, high/.71)	.45
Q	6.5	(low/0, moderate/.22, high/.93)	.78
NewTech	4.1	(low/.18, moderate/.48, high/.59)	0

**TABLE 3**  
**Distances Between Fasturn and Candidate Cases**

Firm	Inventory Turnover	Fuzzy Inventory Turnover	Distance
K	3.5	(low/.3, moderate/.67, high/.5)	.22
L	2.8	(low/.44, moderate/.83, high/.4)	.18
M	3.0	(low/.4, moderate/1.0, high/.43)	.28
O	3.8	(low/.24, moderate/.56, high/.54)	.41
Fasturn	6.5	(low/.35, moderate/.80, high/.46)	0

high/.46). This classification results in a candidate set of {K, L, M, O}; all have good management and moderate inventory turnover. After measuring their grade distances (as shown in Table 3), case L is found to be the closest. This suggests the conclusion that the risk of Fasturn is medium rather than low. This example shows that cross-sectional comparisons can be critical in some cases. Our fuzzy approach makes cross-sectional comparison possible and easier for cases involving quantitative attributes.

#### 4.3. Fuzzy Induction of Rules

Another advantage of the fuzzy approach is that it sometimes makes it easier to induce rules or decision trees from the case base after the conversion. Rule induction is a generalization process that finds causal relationships, represented as if-then rules, among attributes from existing cases (Liang, 1992; Quinlan, 1979). A popular approach is Quinlan's ID3 that uses entropy to measure the relative contribution of each attribute. One problem with many existing induction approaches is that they often have to discretize numerical attributes such as inventory turnover ratios before calculating entropy. A typical approach used by existing methods is to find crisp hurdle values that cut the range of a numerical attribute into several discrete sections.

The fuzzy approach presented in this paper provides a more flexible way to process numerical attributes. First, we can use the fuzzy membership function to convert numerical values into proper categories, as we did in Table 1. This procedure allows some cases to be classified into more than one category. For instance, the inventory turnover ratio of case D is classified as both low and moderate. In other words, case D can be considered as two training cases, one is D1(high, average, low) and the other is D2(high, average, moderate).

After converting all numerical attributes into fuzzy terms, regular induction procedures can be applied to generate rules or decision trees. Since we may have conflicting cases, the possible correctness of each rule can be represented by the percentage of cases supporting the rule. For example, the ratio of 6/7 means that, among the 7 cases that meet the condition, 6 match the conclusion of the rule. From the cases in Table 1, the following rules can be generated:

1. If inventory turnover = low  
then risk = high, correctness = 6/7;
2. If inventory turnover = moderate and quality = poor  
then risk = high, correctness = 1/1;
3. If inventory turnover = moderate and quality = average  
then risk = medium, correctness = 2/3;
4. If inventory turnover = moderate and quality = good  
then risk = medium, correctness = 3/4;
5. If inventory turnover = moderate and quality = excellent  
then risk = low, correctness = 1/2;
6. If inventory turnover = high and quality = average  
then risk = medium, correctness = 1/1;
7. If inventory turnover = high and quality = good  
then risk = low, correctness = 3/4;
8. If inventory turnover = high and quality = excellent  
then risk = low, correctness = 4/4.

Through the application of these rules, we can solve new cases that we encounter. This also indicates the possible integration of the case-based and rule-based approaches.

#### 5. CONCLUDING REMARKS AND FUTURE WORK

Case-based reasoning has received significant attention in recent years. Since cases may contain quantitative attributes that are hard to index and manage in the case base, it is important to develop effective methods for handling them. In this paper, we have proposed a fuzzy-set-based approach that uses fuzzy membership functions to convert numerical attributes into qualitative terms for indexing and retrieval. We have shown that this new approach allows numerical data to be handled easily. We have also shown that the proposed approach increases retrieval flexibility, allows cross-sectional comparison, and supports fuzzy induction of rules.

In addition to converting quantitative attributes, the fuzzy approach can be extended to handle qualitative attributes. The classes of qualitative attributes are often fuzzy terms that can have membership grades. For instance, good management is a fuzzy description. Firm A's management may be 0.7 good and 0.2 excellent, whereas firm B's management may be 0.8 good and

0.5 excellent. In this case, the distance measures presented in the paper can apply to both types of attributes. The overall distance is the sum of distances of all attributes.

For practical applications, the proposed approach may have a few limitations that need further research. First, the determination of classes and membership functions has a critical impact on the performance of the resulting system. However, this determination is, by no means an easy task. In the future, we plan to develop theoretically sound procedures for defining membership functions.

The second issue for future research relates to the information bias or information loss due to data conversion from one metric to another. This happens in most situations when data is converted from a more accurate metric (such as real numbers) to a less accurate one (such as a categorical scale). The fuzzy conversion is no exception. Therefore, it is desirable to study the tradeoffs involved in the conversion and when fuzzy conversions are appropriate. We also need to find the situations in which the fuzzy approach is inappropriate and why.

The third issue is the appropriateness of using  $\alpha$ -cuts in fuzzy classification. We used  $\alpha$ -cuts in this paper to increase the efficiency of feature matching by removing cases with low membership grades. This, however, may also limit the retrieval flexibility of the resulting system. Delicate tradeoffs are involved in choosing proper  $\alpha$ -cuts.

In addition to the above three issues, we also need to examine the integration of the fuzzy approach with existing rule-based and other types of systems. Although it is common to use fuzzy logics to process uncertainties in rule-based systems, the integration of fuzzy rules and fuzzy cases can be very challenging, not to mention their further integration with fuzzy neural networks.

Finally, fuzzy retrieval often results in a set of candidate cases that need further evaluation before the best case can be chosen. Verification and validation are important issues for using case-based systems (O'Leary, 1993). Therefore, developing more effective methods for case selection, system evaluation, and validation are other important areas for further research.

## REFERENCES

- Barletta, R. (1992). An introduction to case-based reasoning. *AI Expert*, 6(8), 43–49.
- Cain, T., Pazzani, M.J., & Silverstein, G. (1991). Using domain knowledge to influence similarity judgments. *Proceedings of the Case-based Reasoning Workshop* (pp. 191–199). San Mateo, CA: Morgan Kaufmann.
- Chi, R.T., Chen, M., & Kiang, M.Y. (1993). Generalized case-based reasoning system for portfolio management. *Expert Systems with Applications*, 6, 67–76.
- Hammond, K.J. (1989). *Case-based planning: Viewing planning as a memory task*. Boston, MA: Academic Press.
- Hedberg, S. (1993, July). New knowledge tool. *Byte*, 18, pp. 106–111.
- Kamel, M., & Hadfield, B. (1990). Fuzzy query processing using clustering techniques. *Information Processing & Management*, 26, 279–293.
- Kandel, A. (1986). *Fuzzy mathematical techniques with applications*. Reading, MA: Addison-Wesley.
- Ketler, K. (1993). Case-based reasoning: An introduction. *Expert Systems with Applications*, 6, 3–8.
- Kolodner, J.L. (1988). Retrieving events from a case memory: A parallel implementation. *Proceedings of the case-based reasoning workshop* (pp. 233–249). San Mateo, CA: Morgan Kaufmann.
- Kolodner, J.L. (1991). Improving human decision making through case-based decision aiding. *AI Magazine*, 12(2), 52–68.
- Lee, J.K., & Kim, M.Y. (1993). Case-based learning for knowledge-based optimization modeling system: UNIK-CASE. *Expert Systems with Applications*, 6, 87–96.
- Liang, T.P. (1992). A composite approach to inducing knowledge for expert systems design. *Management Science*, 38(1), 1–17.
- Liang, T.P., & Turban, E. (1993). Case-based reasoning and its applications. (Guest editors' note). *Expert Systems with Applications*, 6(Special Issue), 1–2.
- Liebowitz, J. (1993, July). Roll your own hybrids. *Byte*, 18, 113–115.
- Mantaras, R.L., Cortes, U., Manero, J., & Plaza, E. (1990). Knowledge engineering for a document retrieval system. *Fuzzy Sets and Systems*, 38, 223–240.
- Mott, S. (1993). Case-based reasoning: Market, applications, and fit with other technologies. *Expert Systems with Applications*, 6, 97–104.
- O'Leary, D.E. (1993). Verification and validation of case-based systems. *Expert Systems with Applications*, 6, 57–66.
- Quinlan, J.R. (1979). Discovering rules from large collections of examples: A case study. In D. Michie (Ed.), *Expert systems in the micro electronic age*. Edinburgh, Scotland: Edinburgh University Press.
- Riesbeck, C., & Schank, R. (1989). *Inside case-based reasoning*. Hillsdale, NJ: Erlbaum.
- Rissland, E.L., Kolodner, J., & Waltz, D. (1989). Case-based reasoning from DARPA: Machine learning program plan. *Proceedings of the Case-based Reasoning Workshop*, (pp. 1–13). San Mateo, CA: Morgan Kaufmann.
- Seifert, C.M. (1988). A retrieval model for case-based memory. *Proceedings of the 1988 AAAI Case-based Reasoning Workshop*, (pp. 120–125). Minneapolis, MN.
- Simoudis, E. (1992, October). Using case-based retrieval for customer technical support. *IEEE Expert*, 7, 7–12.
- Slade, S. (1991). Case-based reasoning: A research paradigm. *AI Magazine*, 12(1), 42–55.
- Stotler, R.H. (1992). CBR for bid preparation. *AI Expert*, 7(3), 44–49.
- Sycara, K.P., & Navinchandra, D. (1989). Index transformation and generation for case retrieval. *Proceedings of the Case-based Reasoning Workshop*, (pp. 324–328). San Mateo, CA: Morgan Kaufmann.
- Veloso, M.M., & Carbonell, J.G. (1991). Variable-precision case retrieval in analogical problem solving. *Proceedings of the Case-based Reasoning Workshop*, (pp. 93–106). San Mateo, CA: Morgan Kaufmann.
- Wall, R.S., Donahue, D., & Hill, S. (1988). The use of domain semantics for retrieval and explanation in case-based reasoning. *Proceedings of the Case-based Reasoning Workshop*, (pp. 447–462). San Mateo, CA: Morgan Kaufmann.
- Zadeh, L.A. (1965). Fuzzy sets. *Information and Control*, 8, 338–353.