Odd Similarity Investigation: A CBR Alternative

Savvas J. Nikolaidis
Department of Informatics
Aristotle University of Thessaloniki
54124 Thessaloniki, Greece
e-mail: snikol@csd.auth.gr

Abstract: Case-based reasoning systems tend to have a rather restricted view at similarity. The majority of case-based reasoning work today invokes a small number of standard similarity concepts, related to the measurement of distances of objects. Such a view at similarity may be adequate in a number of situations, but there are also cases where it does not meet the necessities quite well. The present paper is looking into non-standard concepts of similarity that can be used in case-based reasoning systems. A number of such approaches are described. Experiments are conducted as to study under which circumstances some of the alternative methods have the performing edge against the classic approach. Depending upon the nature of the data and the problem specifics, an alternative approach can be more suitable than the classic one.

Keywords: Knowledge Based Systems, Case-Based Reasoning, Case Retrieval, Similarity Measurement.

I. INTRODUCTION

Case-based reasoning is in essence analogical reasoning where the system tries to predict the outcome of a given situation according to the outcome of the most similar case, or cases, from a knowledge base of solved cases. Case Based Reasoning is a technique used in situations where we want to reduce the burden of knowledge acquisition, avoid repeating mistakes made in the past, work in domains where a well understood model doesn't exist, learn from past experiences, reason with incomplete or imprecise data, provide means of explanation and reflect human reasoning [14]. There are four key issues in the case-based reasoning process: (a) identifying key features, (b) 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. The most important part of a case-based reasoning system is the retrieval stage, where the system must find, in a sometimes-huge case base, the best matching case or cases from which to produce the prediction for the outcome of a given situation. The case based reasoning system must be able to identify the suitable cases for retrieval. The efficiency of this stage is a critical factor for the overall system performance. If the system is not able to properly identify a suitable case this case may not be retrieved, although it might be useful. Improving retrieval is an open problem in case-based reasoning research and case-based reasoning system development [11]. Case adaptation may also be needed. Case attributes can be qualitative, quantitative, descriptive or other.

II. STANDARD AND ALTERNATIVE APPROACHES TO SIMILARITY

The retrieval stage in Case-Based Reasoning systems requires the use of some kind of similarity measurement for the best case to match. Search for similarity, is a problem which occurs in diverse applications, such as stock market prediction [17], [20], plagiarism detection [19], forest fire prediction [18], and protein and DNA sequencing [16]. A number of similarity measuring techniques have been used in different systems. The selection of the similarity measurement is very important, because, if the one selected is not the appropriate, the system will produce erroneous results. The selection depends on being able to identify relevant attributes and make use of them. There is no similarity measurement that can fit in all situations. The Question of defining similarity is one of the most subtle and critical issues raised by case-based reasoning [12]. Very serious consideration must be given to the nature of the data, which dictate the selection of the suitable similarity measurement.

In this paper we present different approaches for representing similarity and identifying cases for the retrieval stage in case based reasoning systems. First we study the Euclidean distance. Euclidean distance is a similarity measure frequently used in the literature. The Euclidean distance is employed in the Nearest Neighbor algorithm and the k-Nearest Neighbor algorithm. With the NN algorithm we try to find from our Knowledge Base the closest match for a given situation and with the k-NN algorithm we are searching for the k closest matches. Then from the results we derive our evaluation. Two different measures of similarity, namely the unweighted Euclidean distance and the weighted Euclidean distance are in use. Each one of them is detailed below. The Euclidean distance can only be used with quantitative attributes. All formulas presented assume that x and y represent size attributes. If our attributes are of a different kind we have to devise a mapping to the set of real numbers so that we can use this method.

The unweighted Euclidean distance d between the points (x_0,y_0) and (x_1,y_1) is given below. The number of attributes employed will determine the number of dimensions used.

$$d = \sqrt{(x_0 - x_1)^2 + (y_0 - y_1)^2}$$

It is common in CBR for the attributes, i.e., features vectors to be weighted to reflect the relative importance of each feature. The weighted Euclidean distance d between the points (x_0,y_0) and (x_1,y_1) is given by the formula:

$$d = \sqrt{w_x(x_0 - x_1)^2 + w_y(y_0 - y_1)^2}$$

where w_x and w_y are the weights of x and y respectively.

Depending upon the problem we are called to confront and the nature of the data involved on top of the classic distance-based approach we have a useful array of alternative techniques that can be used. In the case of textual similarity, as for example when we are trying to compare documents or web pages the "edit distance" is an interesting approach. This similarity function is discussed in detail in [15]. Edit Distance *ed(s1, s2)* between two strings s1 and s2 is the minimum number of character edit operations (delete, insert, and substitute) required to transform s1 into s2, normalized by the maximum of the lengths of s1 and s2.

$$ed(s1,s2) = \frac{eo(s1,s2)}{\max(len(s1),len(s2))}$$

For example the edit distance between the strings 'company' and 'corporation' is 7/11 0.64, and the sequence of edit operations is shown:

```
c o m p     a       n y
↓ ↓ ⇩ ↓     ↓       ↓
c o r p o r a t i o n
```

Simple arrows indicate exact matches (cost is 0) and strong arrows substitutions (cost is 1). Characters in italics are deleted or inserted and always have a unit cost.

Cases that are not simple in form can be described as graphs Pin graphs are finite collections of particular graphs which have certain distinguished properties. There are special nodes which may be replaced by complete individual graphs. For more details the reader is referred to [1] and [9]. Pin graphs can be used for case representation. These structures can naturally describe certain technical objects and there are algorithms known to be more efficient on hierarchically structured graphs than on flat ones. In these structures structural similarity is employed as the similarity function.

A much promising alternative approach to similarity measurement is the incorporation of fuzzy logic to case based reasoning systems. Fuzzy logic techniques have been increasingly used in CBR during the last years. [5], [8], [10]. One such approach is proposed and implemented below.

More complex applications may require the combination of different concepts and techniques. Formal concepts as well as sophisticated heuristics may be employed to deal with more complicated situations.

III. PROPERTY DEFINITION FOR QUALITATIVE CASE ATTRIBUTES

Experts describe similarity fluently using a fuzzy vocabulary. For example, an expert may quote, "Two features are slightly similar when the difference between their values is near 10". This kind of ambiguous knowledge is difficult to encode with classic techniques. Fuzzy sets on the other hand can do exactly that. Lotfi A. Zadeh, the founder of fuzzy logic, argues that "fuzzy logic lets people compute with words". He says that "the fuzzy approach is necessary when the available information is too imprecise to justify the use of numbers, and second, when there is a tolerance for imprecision which can be exploited to achieve tractability, robustness, low solution cost, and better rapport with reality" [22]. Fuzzy logic uses fuzzy sets to represent properties. According to classic logic every proposition must either be *True* or *False*, *A* or *not-A*, either *this* or *not this*. For example, a rose is either red or not red. It cannot be red *and* not red. Every statement or sentence is true or false or has the truth value 1 or 0. Having such a property makes an item belong to a classic (also called "crisp") set. Conventional computer logic has great difficulty manipulating data representing subjective or vague human ideas such as "an attractive person" or "pretty hot". Fuzzy logic was designed to allow computers to determine the distinctions among data with shades of gray, similar to the process of human reasoning. This theory proposed making the membership function (or the values *False* and *True*) operate over the range of real numbers [0.0, 1.0]. Elements In fuzzy sets have different degrees of membership in the range from 0 to 1. *Zero* (0) means absolute exclusion from the set and *one* (1) means that the element definitely belongs to the set. All the numbers in between, declare a different degree of membership. Crisp sets are a special case of fuzzy sets. Most operations defined on crisp sets can also be applied to fuzzy sets. A detailed description of fuzzy logic methods is given by Zimmerman [23].

Fuzzy methods have been extensively used in case based reasoning. Fuzzy techniques in CBR have been used since the early nineties. [3], [4], [6]. Fuzzy logic is especially useful for CBR because CBR is fundamentally analogical reasoning [11], analogical reasoning can operate with linguistic expressions, and fuzzy logic is designed to operate with linguistic expressions. We combine fuzzy logic with CBR because fuzzy logic is helpful for acquiring knowledge and it provides methods for

applying knowledge to real-world data. Fuzzy logic simplifies elicitation of knowledge from domain experts, such as knowledge of how similarity between two cases depends on the difference between their individual, collective, and temporal attributes. Fuzzy logic emulates human reasoning about similarity of real-world cases, which are fuzzy, that is, continuous and not discrete. [8]. Main et al [13] explain how fuzzy logic applies to CBR. One of the main tasks involved in the design of CBR systems is determining the features that make up a case and finding a way to index these cases in a case-base for efficient and correct retrieval. Common types of variables used to describe features in case-based systems are: Boolean, continuous, and multi-valued (ordinal, nominal, and interval-specific). Fuzzy variables allow one to represent features in another way: A large number of features that characterize cases frequently consist of linguistic variables which are best represented using fuzzy vectors. After testing fuzzy features in case selection, they found that the cases retrieved matched the current case in at least 95% of the tests. There are at least four advantages of using fuzzy techniques in the retrieval stage [10]. First, it allows numerical features to be converted into fuzzy terms to simplify comparison. For example we can convert the age of a person into a categorical scale (e.g., old, middle-aged, 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. A 50-year old person may be classified as old (0.6) and middle-aged (0.5) where 0.6 and 0.5 are the degrees that the 50-year old is classified as old and middle-aged respectively. This allows the case to be viewed as a candidate when we are looking for either an old person or a middle-aged person. Third, fuzzy sets make it easier to transfer knowledge across domains. For instance, we have cases showing 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 a 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 persons from a case base containing old persons with possibilities ranging from 0.5 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. With fuzzy set membership functions we can have gradual membership to a set. Fuzzy logic also enables us, to use common words as case-based reasoning attributes. Fuzzy sets can use term modifiers to modify the membership grade, define subsets and increase the flexibility of case retrieval. Terms like "very", "somewhat", "definitely" etc. can be represented and used for the similarity measurement.

Another significant concern is the nature of the data. Qualitative attributes have discrete nominal values. People understand better and feel more comfortable with labels and descriptions than actual numerical values. Justifying a decision can thus be made straightforwardly and so sometimes it is useful to devise a way to translate quantitative values to nominal ones. If we try to classify cases using classic "crisp" sets we will probably encounter problems:

Marginal cases

Classic sets do not describe qualitative attributes adequately because they give the same degree of membership to all their members and the same degree of exclusion to all the non-members. Let's discuss for example a system that is called to decide upon the appropriate medication for a hospital patient. One factor on which the choice of the appropriate medication depends is the patient's age. If we have a "crisp" set classification procedure then we have to set arbitrary limits to say when a man is old. This kind of classification has its drawbacks: If we say that the limit is at 60 years of age, then, for our system, a 61 year old man is as "old" as a 90 year old, but in reality they are very different in their degree of oldness: a man aged 61 is "old" but a man of 90 is "definitely old". Another problem with the traditional approach is that it does not provide adequate flexibility for marginal cases. A man aged 59 years and eleven months is classifies as a "not old" when a man aged 60 is classified as "old" even thought they practically have the same age. This is not the way humans see things and can lead to problems. If, for example, we consider medical care in a hospital for a patient near the age limit, classifying him in one category can result in treatment not suitable for him.

Partial Match

Case based reasoners using the standard set theory are risking taking the wrong decisions by making wrong assumptions. Misinterpretations of a given situation can lead to errors. Suppose we have a system which searches for people who are "young and rich" and let's say that someone is defined as "young" if his age is less than 30 years of age and "rich" if his wealth is more than $1.000.000. Suppose now that we have three persons whose age and wealth are (26, $50.000), (67, $10.000.000) and (30, $999.990). None of them qualifies the

criteria as being "young and rich". The first is young but not rich, the second rich but not young, and the third is neither. A classic set based case based reasoner might retrieve the first or the second person based on *partial match* but it may miss the third, although in reality he is the one closest to the criteria. Using fuzzy sets can overcome the above problems. An attribute may have different degrees of membership in sets that would be mutually exclusive with the classic set membership definition. A person can be at the same time classified as young and old, with different degrees of membership.

IV. CASE IDENTIFICATION AND RETRIEVAL: OUR APPROACH

A case-based reasoner has to find a case or a set of cases similar to the target problem. Most CBR systems are based on similarity relations between the target and the cases. These relations are vague by nature. In the CBR cycle, during the analysis of the similarity among cases (Initially Match Process during the Retrieve Step [7]), not crisp classification methods can be used in order to improve the performance and the efficiency of the CBR system. The cases can be "preprocessed" and assigned to one or more of a number of fuzzy categories. The retrieval of previously solved problems similar to the current one is a two-step process. The initial matching process can be described as a procedure to isolate interesting, or as we say, potentially applicable cases. For example we could have four preprocessed categories of a person's age: "child", "young", "middle aged" and "old" and search only the appropriate one(s) when we encounter a new case. During this step the system reduces the set of cases to be compared to the current case in the second step of the similarity computation. Using the Similarity Function in the second stage we evaluate selected cases to determine the definitely applicable ones. The algorithm can be described:

1. Locate and retrieve potentially applicable cases from the case-base.
2. Evaluate selected cases to determine the applicable ones.
3. Transfer knowledge from the old case(s) to the current one.

The algorithm is described in *Figure 1.*

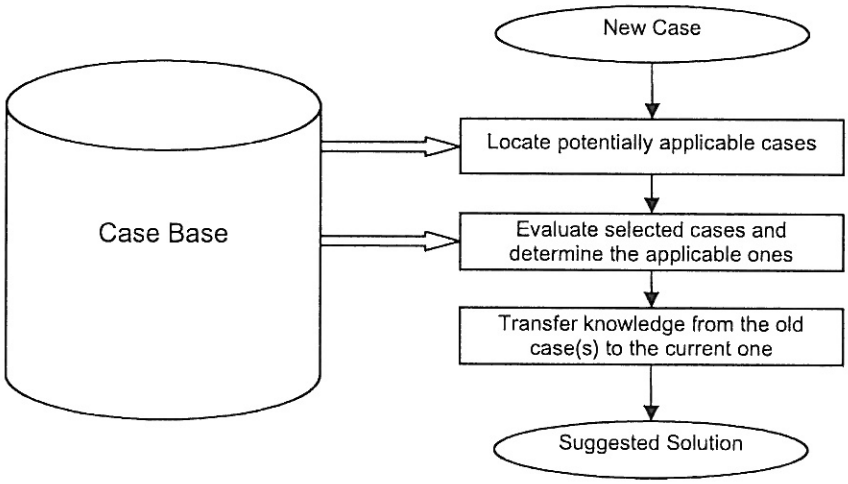

Figure 1. The F-CIR algorithm

The fuzzification process is a feature selection algorithm and creates fuzzy categories that are used for indexing. The fuzzyfier dictates the way an attribute is turned into a classifiable one in a fuzzy set. At the end of the process the case base has been converted into an indexed case base according to the fuzzy categorization.

The fuzzification process works through the case base according the following steps:

1. Eliminate one attribute and test (try to find irrelevant attributes).
2. Eliminate all but one attributes and test (find the relative significance of different attributes).
3. Assign weights to the attributes.
4. Identify significant or irrelevant ranges in attribute values. Enhance the former and discard the later.
5. Normalize to the fuzzy set belonging range [0, 1].
6. Repeat the above steps until there is no significant result improvement (the performance "tops up").

The fuzzification process is shown in *Figure 2* and *Figure 3.*

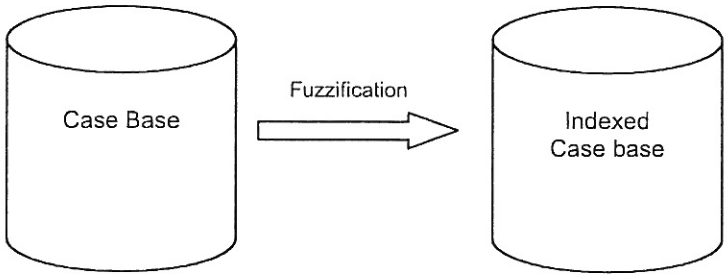

Figure 2. The fuzzification process

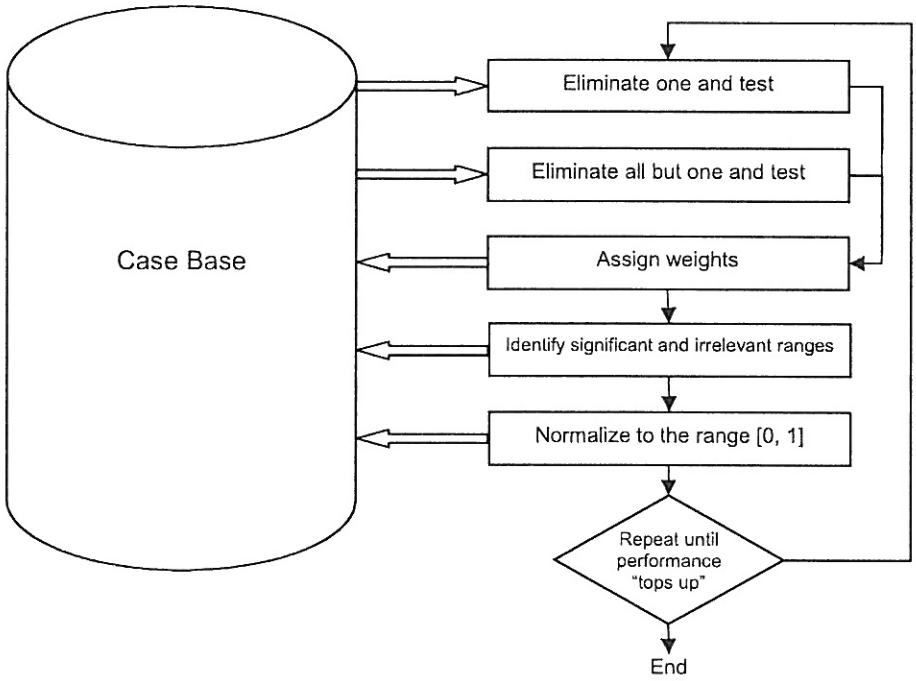

Figure 3. The fuzzification algorithm

For similarity function we use the one described below [2]. Suppose we try to find the similarity between a case $c = (f_1^c, f_2^c, \ldots, f_n^c)$ and the target problem $t = (f_1^t, f_2^t, \ldots, f_n^t)$. The similarity is given by the weighted sum:

$$\sum_{i=1}^{n} \frac{w_i \cdot SIM(f_i^t, f_i^c)}{\sum_{i=1}^{n} w_i}$$

where w_i is the weight for the i-th feature and for the function $SIM(f_1^c, f_2^t)$ is the value for f_2^c of the Gaussian curve build by the system with mean value f_2^t and fixed standard deviation σ:

$$SIM(f_i^t, x) = e^{\frac{(x-f_i^t)^2}{\sigma^2}}$$

By using the above procedure we preserve a lot of the information that would be lost if we were using crisp sets, useless information that could produce erroneous predictions is discarded and finally our prediction results become more easily justifiable.

V. THE EXPERIMENTS

We have tested our methods by executing some experiments. Initially we used two unrelated sets of data describing the real estate market situation. The common aim of all the experiments is to estimate the real value of the property from the data in each data set.

The first data set consists mainly of quantitative data. Attributes in these set include: property surface area in m^3, the flour on which the property is located, distance from the city center and from major highways in km, age of the building etc. The data has been tested with the unweighted Euclidean distance method, the weighted Euclidean distance method *(NN)* and *k-NN* with different values of *k*, and our Fuzzy method. From the dataset of more than 500 cases 50 were randomly selected for evaluation and the rest for training. The experiment

was repeated 400 times for statistical purposes. The results are shown in *Table I* and *Figure 4.*

	"Hits"	"Misses"	Percentage
Unweighted Eucledian	13,8	36,2	27,6%
Weighted Eucledian	31,1	18,9	62,2%
k-NN	34,6	15,4	69,2%
F-CIR	35,2	14,8	70,4%

Table I Mean values of correct and wrong predictions using randomly selected 50 member testing sets, according to different methods for the first data set.

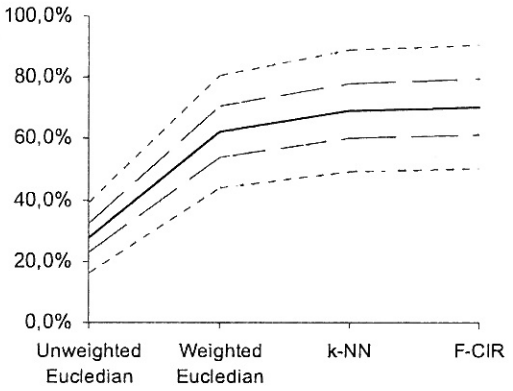

Figure 4 Performance comparison between different methods for the first data set

All Algorithms except the unweighted Euclidean Nearest Neighbor performed well. The slightly better performance of F-CIR compared with the weighted Euclidean is derived from the identification of significant and irrelevant attribute ranges during the fuzzification process.

The data set used in the second set of experiments, consists mainly of nominal data. Attributes in these set include: city area where the property is located (city center, east side, name of suburb), description of the apartment (studio, 2-bedroom etc), accessibility to the employment centers, traffic conditions in the area, apartment orientation, level of pollution in the area etc. The experiments were conducted with the same philosophy as the ones for the first dataset (400 repetitions). The results are shown in *Table II* and *Figure 5.* F-CIR worked satisfactory.

	"Hits"	"Misses"	Percentage
Unweighted Eucledian	14,3	35,7	28,6%
Weighted Eucledian	16,6	33,4	33,2%
k-NN	17,2	32,8	34,4%
F-CIR	29.2	20,8	58,4%

Table II Mean values of correct and wrong predictions using randomly selected 50 member testing sets, according to different methods for the second data set.

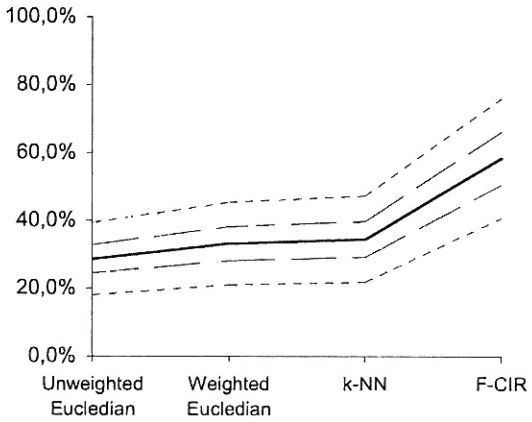

Figure 5 Performance comparisons between different methods for the second data set

Another set of experiments was executed using hydrologic and climatic data from the hydroelectric complex in Agras, Greece. The aim was the prediction of the level of water in the natural lake of Arnissa which works as a natural reservoir for the complex. The lowering of the waters of the lake in recent years has given rise to environmental concerns. The raw data were used with the k-NN method. The fuzzification process was applied before the data were used with the fuzzy method. The results are shown in *Table III.*

	"Hits"	"Misses"	Percentage
k-NN	23,6	16,4	59%
F-CIR	28,9	11,1	72,2%

Table III Mean values of correct and wrong predictions for the Arnissa lake water level using the Agras hydroelectric complex hydrologic data.

VI. CONCLUSIONS

The classic distance based approach may, in a number of situations, be inadequate to describe efficiently the similarity between objects in a case base. Under these circumstances the use of a non-standard similarity function is required. Depending

on the nature of the problem and the data involved alternatives include a number of methods. The selection of the appropriate one is critical for the performance of the system. As the problems faced by case based reasoning systems get increasingly complex the need for alternative approaches, or even combinations of them, is certain to increase.

REFERENCES

[1] Helmut Alt and Johannes Blomer. Resemblance and symmetries of geometric patterns. In Burkhard Monien and Thomas Ottmann, editors, Data Structures and Effcient Algorithms, volume 594 of Lecture Notes in Computer Science, pages 1-24. Springer-Verlag 1992.

[2] Bandini S., and Manzoni, 2001, Proceedings of the 2001 ACM symposium on Applied computing, pp.462-466

[3] Bento, C. and Costa E., 1993: A similarity metric for retrieval of cases imperfectly described and explained, in: Proceedings First European Workshop on Case-Based Reasoning, Richter, Wess, Althoff and Mauer (eds.), Vol. 1, 1993, pp 8-13.

[4] Bonissone, P.P., and Ayub, S., 1992: Similarity measures for case based reasoning systems, in Proceedings of the IPMU-92 Conference, 1992, pp. 483-487.

[5] S. Chaudhuri, K. Ganjam, V. Ganti, R, Motwani, 2003: Robust and Efficient Fuzzy Match for Online Data Cleaning, Procceedings of SIGMOD 2003.

[6] Dubois and Prade, 1992: Gradual inference rules in approximate reasoning, Information Sciences, 61, 103-122

[7] Kolodner. Case-Based Reasoning. Morgan Kaufmann, San Mateo (CA), 1993.

[8] Hansen, B. K. (2000) Weather prediction using similarity between temporal cases and fuzzy sets, Master of Computer Science thesis, Dalhousie University – Daltech.

[9] Franz Hofting, Thomas Lengauer, and Egon Wanke. Processing of hierarchally defined graphs and graph families. In Burkhard Monien and Thomas Ottmann, editors, Data Structures and Efficient Algorithms, volume 594 of Lecture Notes in Computer Science, pages 44-69. Springer-Verlag, 1992.

[10] Jeng, B. C., and Liang, T.-P. (1995) Fuzzy indexing and retrieval in case-based systems, Expert Systems With Applications, Vol. 8., No. 1, 1995. Elsevier Science Ltd., 135–142.

[11] Leake, D. B. (1996) CBR in context. The present and future; in Leake, D. B. (editor) (1996) Case-Based Reasoning: Experiences, Lessons & Future Directions, American Association for Artificial Intelligence, Menlo Park California, USA, 3–30.

[12] Luger, G. F., and Stubblefield, W. A. (1998) Artificial Intelligence: Structures and Strategies for Complex Problem Solving, Addison Wesley Longman, Reading, Massachusetts, USA, pg. 238.

[13] Main. J., Dillon, T. S., and Khosla. R. (1996) Use of fuzzy feature vectors and neural vectors for case retrieval in case based systems, NAFIPS 1996 Biennial Conference of the North American Fuzzy Information Processing Society, IEEE, New York, NY, 438–443.

[14] Main, J.; Dillon, T. S.; and Shiu, S. C. K. 2000. A tutorial on case-based reasoning; in Pal, S. K.; Dillon, T. S.; and Yeung, D. S. eds. 2000. Soft Computing in Case Based Reasoning. London, UK, Springer.

[15] G. Navarro, R. Baeza-Yates, E. Sutinen, and J. Tarhio. Indexing methods for approximate string matching. IEEE Data Engineering Bulletin, 24(4):19--27, 2001.

[16] Pearson, W. R., and Lipman, D. J. (1988) Improved tools for biological sequence comparison, Proceedings of the National Academy of Sciences, Vol. 85, 2444–2448, April, 1988, Biochemistry.

[17] Rafiei, D. (1999) Fourier-Transform Based Techniques in Efficient Retrieval of Similar Time Sequences, Ph.D. thesis, Department of Computer Science, University of Toronto, Ontario, Canada.

[18] Rougegrez, S. (1993) Similarity evaluation between observed behaviours and the prediction of processes. In Wess, S., Althoff, K. D. and Richter, M. (eds.), Topics in Case-Based Reasoning, Proceedings First European Workshop on Case-Based Reasoning, 1993, Springer-Verlag, Berlin, 155–166.

[19] Shivakumar, N., and Garcia-Molina, H. (1995) SCAM: A copy detection mechanism for digital documents, Digital Libraries '95, The Second Annual Conference on the Theory and Practice of Digital Libraries, 155–163.

[20] Xia, B. B. (1997) Similarity Search in Time Series Data Sets, Master of Science thesis, Department of Computer Science, Simon Fraser University, BC, Canada.

[21] Zadeh, L. A. (1965) Fuzzy sets, Information and Control, Vol. 8 (June 1965), 338-353; reprinted in Bezdek, J. C., and Pal, S. K. (eds.) (1992) Fuzzy Models for Pattern Recognition, IEEE Press, 35–45.

[22] Zadeh, L.A. – "Fuzzy Logic = Computing with Words", IEEE Transactions on Fuzzy Systems, Vol. 4, No. 2, May 1996, pp. 103 - 111

[23] Zimmerman, H. J. (1991) Fuzzy Set Theory and its Applications (2nd edition), Kluwer Academic Publishers.