Proceedings of the International MultiConference of Engineers and Computer Scientists 2011 Vol I,
IMECS 2011, March 16 - 18, 2011, Hong Kong

Fuzzy Sequence Mining for Similar Mental Concepts

M. Gholizadeh, M. M. Pedram, J. Shanbehzadeh

***Abstract*— Sequence mining, a branch of data mining, is recently an important research area, which recognizes subsequences repeated in a temporal database. Fuzzy sequence mining can express the problem as quality form that leads to more desirable results. Sequence mining algorithms focus on the items with support higher than a specified threshold. Considering items with similar mental concepts lead to general and more compact sequences in database which might be** indistinguishable in situations where the support of individual **items are less than threshold. This paper proposes an algorithm to find sequences with more general concepts by considering mental similarity between items by the use of fuzzy ontology.**

***Index Terms*— sequence mining, subsequence, similar, mental concept, fuzzy ontology.**

I. INTRODUCTION

Sequential data is an important class of data with a wide range of applications in science, medicine, security and commercial activities. The sequential data is a set of sequences or sub-structures in a data set that repeats more than or equal to a known minimum support as a threshold declared by the user. DNA sequence is an example that encodes the generic makeup of humans and all other species; and protein sequence that expresses the information and functions of proteins. Besides, the sequential data is able to describe the individual human behavior such as the history of customers' purchases in a store. There are various procedures to extract data and patterns out of data sets such as time series analyzing, association rules mining and, sequence mining.

Time series is defined as a set of stochastic data gathered within a regular fixed time intervals and, time series analyzing refers to stochastic methods that operate on such data [1, 2]. The diagram of time series could be figured by setting the horizontal axis representing the time and the vertical axis denoting to the desired variable. Fig.1 demonstrates the general form of a diagram of time series.

Manuscript received January 18, 2011;
M. Gholizadeh is student in the Computer Engineering Department at Islamic Azad University Science and Research Branch, Tehran, Iran (e-mail: mhdgholizadeh@ gmail.com).
M. M. Pedram, is with the Computer Engineering Department, Faculty of Engineering, Tarbiat Moallem University, Karaj/Tehran, Iran (e-mail: pedram@tmu.ac.ir).
J. Shanbehzadeh is with the Computer Engineering Department, Faculty of Engineering, Tarbiat Moallem University, Karaj/Tehran, Iran (e-mail: shanbehzadeh@gmail.com).

The association rule based mining algorithms try to find the dependencies and relations between various data in a data base. These algorithms consist of two stages. The first one finds a set of highly repeated items and, the second one extracts some suitable rules from the highly repeated collections. The highly repeated items are collected by methods like the Apriori algorithm based on the number of repetitions [3-6]. Then, the algorithm generates the data and patterns by using the collected items. Sequence mining identifies the repeated sub-sequences in a set of sequential data. The input data in sequence mining is comprised of a list of transactions and their occurrence time. Moreover, each transaction includes a set of items. Sequential patterns are also a set of sequentially happened items. The main purpose of sequence mining is to search and find all the sequential patterns with support values greater than or equal to a minimum support threshold declared by the user [7-9]. Fig.2 shows the classification of frequent pattern mining studies.

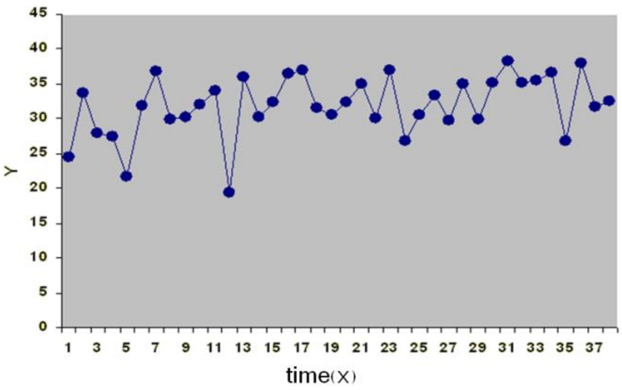

Fig. 1. A diagram of time series

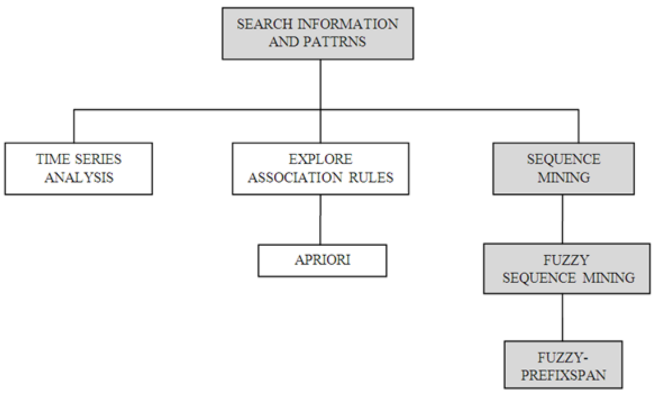

Fig. 2. Frequent pattern mining studies

ISBN: 978-988-18210-3-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

This paper presents a new sequence mining algorithm in which a common item set is used to describe the similar mental concepts. Therefore, we can find more general sequences with higher support values. The rest of this paper has been organized as follows. Section 2 defines the sequence mining by employing an example and introduces two useful sequence mining algorithms. Section 3 presents Fuzzy PrefixSpan sequence mining algorithm. Section 4 introduces the algorithm of sequence mining based on similar mental concepts and investigates the proposed algorithm. In Section 5, the numerical experiments are shown and discussed. Section 6 is the conclusion.

II. FUZZY SEQUENCE MINING

Sequence mining aims to find the sequential patterns with support values greater than or equal to a minimum support threshold (declared by the user). The following sentence is an example of sequential patterns: "Customers who have purchased a printer, are reasonably probable to purchase printer ink, too". In this example, the purchase of printer and printer ink can represent a sequence.

Classic sequence mining algorithms show sequences like (printer, printer ink), but there is no information about the number of purchase of any item. There are two solutions. The first one is to use certain sequence mining algorithms and, the second one is based on the fuzzy sequence mining algorithms. The first group of algorithms can mine the repeated sequences and, has the ability to provide the number of items occurred in the sequences. In this case, the form of output will be such as (printer: 2, printer ink: 5). The second method has the ability to provide the fuzzy term of the number of items occurred in the sequences. In this case, the algorithm has an output like (printer: low, printer ink: medium).

The first method shows the number of each item but, the major problem is the severe decrease in the sequences support values compared to the classic sequence mining. In fact, to find the support value, these algorithms must consider both the number of items' occurrence and their repetition. This will decrease the support value. For example, to find sequences with support threshold equal to 2, in the classical sequence mining method it is just sufficient to see the item at least two times; but in the crisp sequence mining, the item must occur at least two times with the repeat number of 2 for printer and 5 for printer ink.

Fuzzy sequence mining expresses items' repetition in Fuzzy linguistic terms. This method introduces a criterion to determine the number of purchasing each item and moreover, somewhat moderates the problem of the first method because, in this case the supported value of the sequences increase, due to be fuzzy terms. Figure 2 shows that there are Fuzzy-Apriori and Fuzzy-PrefixSpan algorithms to find the repeated or frequent sequences. Comparing the Fuzzy-Apriori algorithm with the Fuzzy-PrefixSpan algorithm, the latter runs faster[12], thus it has been used as the base algorithm in this paper.

In this paper, the fuzzy membership functions shown in Fig.3 have been used to fuzzify crisp values.

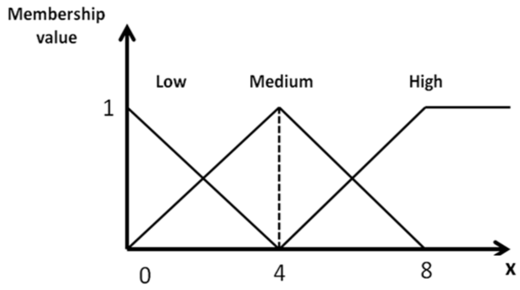

Fig. 3. The fuzzy membership functions

III. FUZZY PREFIXSPAN ALGORITHM[13]

At the first we need to introduce the concepts of *prefix* and *suffix* that are the basic and essential terms in fuzzy PrefixSpan algorithm.

A. Prefix

Suppose that all the items within an element are listed alphabetically. For a given sequence α, where $\alpha=(p_1{:}k_1\ p_2{:}k_2\ \dots\ p_n{:}k_n)$, each $p_i{:}k_i$($1 \le i \le n$) is an element. A sequence $\beta=(p'_1{:}k'_1\ p'_2{:}k'_2\ \dots\ p'_m{:}k'_m)$ ($m \le n$) is called a prefix of α if (1) $p'_i{:}k'_i=p_i{:}k_i$ for $i \le m-1$; (2) $p'_m{:}k'_m \subseteq p_m{:}k_m$; and (3) all items in ($p_m{:}k_m$ - $p'_m{:}k'_m$) are alphabetically after those in $p'_m{:}k'_m$.

For example, consider sequence s=(a:low)(a:low b:medium c:medium)(a:high c:high d:low). Sequences (a:low)(a:low b:medium) and (a:low b:medium c:medium) are prefixes of s, but neither (b:medium a:high) nor (a:low a:low) is a prefix.

B. Suffix

Consider a sequence $\alpha=\langle p_1{:}k_1\ p_2{:}k_2\ \dots\ p_n{:}k_n\rangle$ and each $p_i{:}k_i$ ($1 \le i \le n$) is an element. Let $\beta=\langle p'_1{:}k'_1\ p'_2{:}k'_2\ \dots\ p'_m{:}k'_m\rangle$ ($m \le n$) be a subsequence of α. Sequence $\gamma=\langle p''_l{:}k''_1\ p_{l+1}{:}k_{l+1}\dots p_n{:}k_n\rangle$ is the suffix of α with respect to prefix β, denoted as γ=α/β , if is the suffix of α with respect to prefix β, denoted as γ=α/β , if

1. $l=i_m$ such that there exist $1\le i_1 \le \dots \le i_m$ such that there exist $p'_j{:}k'_j \subseteq p_{ij}{:}k_{ij}$ ($1\le j \le m$) and i_m is minimized. In other words, $p_1{:}k_1 \dots p_l{:}k_l$ is the shortest prefix of α which contains $p'_1{:}k'_1\ p'_2{:}k'_2 \dots p'_{m-1}{:}k'_{m-1}\ p'_m{:}k'_m$ as a subsequence; and;

2. $P''_l{:}k''_l$ is the set of items in $p_l{:}k_l$ - $p'_m{:}k'_m$ that are alphabetically after all items in $p'_m{:}k'_m$.

If $P''_l{:}k''_l$ is not empty, the suffix is also denoted as (-items in $P''_l{:}k''_l$) $p_{l+1}{:}k_{l+1}\dots\ p_n{:}k_n$. Note that if β is not a subsequence of α, the suffix of α with respect to β is empty.

For example, consider sequence s=<(a:low)(a:low b:medium c:medium)(a:medium c:high)(d:high)(c:low f:low)> and (a:low b:medium c:medium)(a:medium c:high)(d:high)(c:low f:low) is the suffix with respect to (a:low), and (c:medium)(a:medium c:high)(d:high)(c:low f:low) is the suffix with respect to (a:low)(b:medium) and (a:medium c:high)(d:high)(c:low f:low)is the suffix with respect to (a:low)(a:low c:medium).

ISBN: 978-988-18210-3-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

The depth first search method is applied on the tree of the Fig. 4. In this tree, sub-trees related to each node indicate all sequence patterns which are prefixes of the node. This tree is called as sequence enumeration tree.

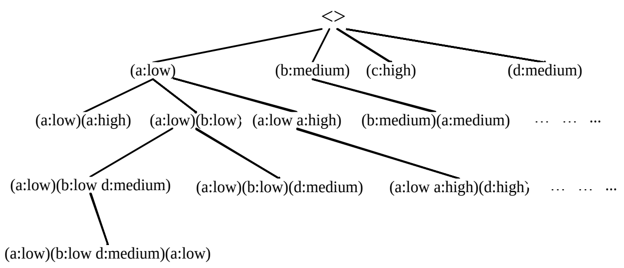

Fig. 4. The fuzzy sequence enumeration tree on the set of items {a, b, c, d, d}

C. Projected database

Let α be a fuzzy sequential pattern in a fuzzy sequence database S. The fuzzy α-projected database, denoted as $S|_\alpha$, is the collection of suffixes of sequences in S with respect to prefix α. Based on the above discussion, we present the algorithm of fuzzy PrefixSpan as follows.

D. Fuzzy PrefixSpan algorithm

Input: A sequence database S, and the minimum support threshold min_support.
Output: The complete set of fuzzy sequential patterns.
Method: Call fuzzy PrefixSpan(Ø, 0, S).
Subroutine fuzzy PrefixSpan(α, l, $S|_\alpha$)
The parameters are :
(1) α is a fuzzy sequential pattern;
(2) l is the i-length of α; and
(3) $S|_\alpha$ is the fuzzy α-projected database if α ≠Ø, otherwise, it is the sequence database S.
Method:
1. Scan $S|_\alpha$ once, find each fuzzy frequent item as (b:k) that leads to face with two states:
1.1) b can assembled to the last element of α as at different times to form a sequential pattern like (α)(b:k);
1.2) b can be append to α as simultaneous to form a sequential pattern like (α b:k).

2. For each fuzzy frequent item (b:k), append it to α to form a sequential pattern α’, and output α’;

3. For each α’, construct fuzzy α’-projected database $S|_{\alpha'}$, and call fuzzy PrefixSpan(α’, l + 1, $S|_{\alpha'}$).

IV. Fuzzy Sequence Mining for Similar Mental Concept

Sequence mining algorithms often work in binary form. In other words, an item is in a desired sequence if its repeatition is more than a Minimum Support. This definition ignores the inter-items’ mental similarity. If we use these similarities, we can achieve more general sequences. In other words, we have to consider the items’ repeatition and, their mental similarity to gather them in a collection and put them under a general concept. This case, rather than studying the items one by one, we calculate the repeatition of the general concepts. This results in sequences with upper support and besides, more general sequences. For this purpose, in addition to the data collection that shows the transactions, there should be another collection which represents the items similarities.

Ontology can be used to show the similar mental concepts. Ontology is a method to represent knowledge in an understandable format for both human and machine and provides the ability to share the information between different programs. All the concepts in the desired range, associated with their hierarchical structure and the existing relations between concepts are defined in ontology. In fuzzy ontology, we can also model and represent the uncertainty of the real world [10, 11].

The proposed algorithm receives two sets as inputs. The first one is a collection including identification number, time, number of items and the items’ repetition. The second data set describes the similarity between each item and each general concept by a membership function, i.e, the fuzzy ontology database. The first dataset is transformed into a new dataset in which items are substituted with general concepts described by the fuzzy ontology; then, the Fuzzy PrefixSpan algorithm is employed on the new dataset and the final results are sequences with more general concepts.

A. Nomenclator

A_i: i-th general concept,

a_j: j-th item which has mental similarity with the i-th general concept,

C_k: Identification number,

t_m: Transaction date,

$n_{aj}(t_m)$: Number of item a_j in the transaction with date t_m,

Similarity (A_i, a_j): The measure describing the similarity of item a_j and the general concept A_i,

Count(A_i, t_m, C_k): Number of times that concept A_i occurred by the identification number C_k at the time t_m,

Fuzzified(n): The fuzzified term for n,

Fuzzy-Count(A_i, t_m, C_k): Fuzzy value of times that concept A_i occurred by the identification number C_k at the time t_m.

B. Algorithm

✓Inputs

1. The dataset including identification number, time, number of items and the number of items happening

2. The dataset containing a list of similar mental concepts by which their similarity is determined via fuzzy ontology.

✓Outputs

General sequences that indicate the items regularity and priority.

ISBN: 978-988-18210-3-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

✓Steps

1. Receive the first and second datasets and build the new one as follows:

   a. The identification number (C_k) and the transaction date (tm) get no change,

   b. The items a_j are replaced with the concepts A_i ,

   c. The number of occurrence of the concept A_i in fuzzy form is calculated as:

$$\mathrm{Count}(A_i, t_m, C_k) = \mathrm{Count}(A_i, t_m, C_k) + \mathrm{Similarity}\ (A_i, a_j) \times n_{aj}(t_m) \quad (1)$$

$$\mathrm{Fuzzy\text{-}\ Count}(A_i, t_m, C_k) = \mathrm{Fuzzified}(\mathrm{Count}(A_i, t_m, C_k)) \quad (2)$$

2. Use the Fuzzy PrefixSpan algorithm for the new dataset.

3. Return the mined general sequences in step 2.

4. End.

V. ILLUSTRATED EXAMPLE

As an example, consider the transactional dataset shown in Table1. Table2 shows the fuzzy ontology, in which general concepts as well as items are shown. In fact, the similarity degree for item aj and general concept Ai is shown by the table.

The original data set (Table1) is transformed into Table3 by (2), in which general concepts are used.

Table3 shows that more general transactions can be mined. This table, unlike Table1, uses more general concepts such as hot drink and fat dairy. Fuzzy PrefixSpan algorithm has been applied on the dataset in Table3 with the minimum support equal to 1. Table4 lists the results and shows each item with its fuzzy values. Table5 presents the results of applying Fuzzy PrefixSpan algorithm on the dataset shown by Table1.

TABLE I.
TRANSACTIONS OF SOME CUSTOMERS

Customer identification number	*Purchase time*	*product*	*Number*
100100002	95/07/22	Tea	1
100100002	95/07/22	Cream	3
100100003	95/07/23	Butter	5
100100003	95/07/23	Coffee	1
100100002	95/07/27	Fruit juice	6
100100003	95/07/29	Fruit juice	2

TABLE I.
AN INSTANT OF ITEMS WITH SIMILAR MENTAL CONCEPT

Product	*Hot drink*	*Fat dairy*
Tea	1	0
Coffee	1	0
Cream	0	0.9
Butter	0	0.9
Fruit juice	0.1	0

The Table4 and Table5 have been mined from the same basic transactions. It is clear that sequences of Table4 are more general with higher support values. In this example if minimum support is equal to 1, then in the first case, the results will be <(Hot drink : Low)> , <(Fat dairy : Low)> , <(Fat dairy : High)> , <(Hot drink : Medium Fat dairy : Low)> , <(Hot drink : Low)(Hot drink : Medium)> , <(Hot drink : Low Fat dairy : Medium)(Hot drink : Low)> but in the second case the result will be <(Fruit juice : Medium)>.

TABLE III.

TRANSACTIONS OF THE CUSTOMERS WITH SIMILAR MENTAL CONCEPT.

Customer identification number	*Purchase time*	*Product*	*Number*	*Fuzzy values*		
				Low	*Medium*	*High*
100100002	95/07/22	Hot drink	1	0.75	0.25	-
100100002	95/07/22	Fat dairy	2.7	0.32	0.68	-
100100003	95/07/23	Fat dairy	4.5	-	0.78	0.13
100100003	95/07/23	Hot drink	1	0.75	0.25	-
100100002	95/07/27	Hot drink	0.6	0.85	0.15	-
100100003	95/07/29	Hot drink	0.2	0.95	0.05	-

ISBN: 978-988-18210-3-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

TABLE V.

OUTPUT SEQUENCES FOUND BY PROPOSED FUZZY METHOD

Sequences	***support***	***Sequences***	***Support***
<(Hot drink : Low)>	1.8	< (Fat dairy : Low) (Hot drink : Medium) >	0.15
< (Hot drink : Medium) >	0.5	< (Fat dairy : Medium) (Hot drink : Low) >	0.68
< (Fat dairy : Low) >	1	< (Fat dairy : Medium) (Hot drink : Medium) >	0.15
< (Fat dairy : Medium) >	0.32	< (Fat dairy : High) (Hot drink : Low) >	0.13
< (Fat dairy : High) >	1.46	< (Fat dairy : High) (Hot drink : Medium) >	0.05
< (Hot drink : Low Fat dairy : Low) >	0.13	< (Hot drink : Low Fat dairy : Low) (Hot drink : Medium) >	0.15
< (Hot drink : Low Fat dairy : Medium) >	0.32	< (Hot drink : Low Fat dairy : Medium) (Hot drink : Medium) >	0.15
< (Hot drink : Medium Fat dairy : Low) >	1.07	< (Hot drink : Medium Fat dairy : Low) (Hot drink : Medium) >	0.2
< (Hot drink : Medium Fat dairy : Medium) >	0.25	< (Hot drink : Medium Fat dairy : Medium) (Hot drink : Medium) >	0.2
< (Hot drink : Low) (Hot drink : Low) >	0.5	< (Hot drink : Low Fat dairy : Low) (Hot drink : Low) >	0.32
< (Hot drink : Low) (Hot drink : Medium) >	1.5	< (Hot drink : Low Fat dairy : Medium) (Hot drink : Low) >	1.48
< (Hot drink : Medium) (Hot drink : Low) >	0.2	< (Hot drink : Medium Fat dairy : Low) (Hot drink : Low) >	0.25
< (Hot drink : Medium) (Hot drink : Medium) >	0.4	< (Hot drink : Medium Fat dairy : Medium) (Hot drink : Low) >	0.5
< (Fat dairy : Low) (Hot drink : Low) >	0.32		

TABLE IV.

OUTPUT SEQUENCES FOUND BY APPLYING THE FUZZY PREFIXSPAN ALGORITHM ON TABLE 1

Sequences	***support***	***Sequences***	***support***
<(Cream : Low)>	0.25	< (Cream: Low) (Fruit juice : High) >	0.25
< (Cream : Medium) >	0.75	< (Cream: Medium) (Fruit juice : Medium) >	0.5
< (Tea : Low) >	0.75	< (Cream: Medium) (Fruit juice : High) >	0.5
< (Tea : Medium) >	0.25	< (Coffee : Low) (Fruit juice : Low) >	0.5
< (Coffee : low) >	0.75	< (Coffee : Low) (Fruit juice : Medium) >	0.5
< (Coffee : Medium) >	0.25	< (Tea : Low Cream : Low) (Fruit juice : Medium) >	0.25
< (Butter : Medium) >	0.75	< (Tea : Low Cream : Low) (Fruit juice : High) >	0.25
< (Butter : High) >	0.25	< (Tea : Low Cream : Medium) (Fruit juice : Medium) >	0.5
< (Fruit juice : Low) >	0.5	< (Tea : Low Cream : Medium) (Fruit juice : High) >	0.5
< (Fruit juice : Medium) >	1	< (Tea : Medium Cream : Low) (Fruit juice : Medium) >	0.25
< (Fruit juice : High) >	0.5	< (Tea : Medium Cream : Low) (Fruit juice : High) >	0.25
< (Tea : Low Cream : Low) >	0.25	< (Tea : Medium Cream : Medium) (Fruit juice : Medium) >	0.25
< (Tea : Low Cream : Medium) >	0.75	< (Tea : Medium Cream : Medium) (Fruit juice : High) >	0.25
< (Tea : Medium Cream : Low) >	0.25	< (Coffee : Low Butter : Medium) (Fruit juice : Low) >	0.5
< (Tea : Medium Cream : Medium) >	0.25	< (Coffee : Low Butter : Medium) (Fruit juice : Medium) >	0.5
< (Coffee : Low Butter : Medium) >	0.75	< (Coffee : Low Butter : High) (Fruit juice : Low) >	0.25
< (Coffee : Low Butter : High) >	0.25	< (Coffee : Low Butter : High) (Fruit juice : Medium) >	0.25
< (Tea : Low) (Fruit juice : Medium) >	0.5	< (Coffee : Medium) (Fruit juice : Low) >	0.5
< (Tea : Low) (Fruit juice : High) >	0.5	< (Coffee : Medium) (Fruit juice : Medium) >	0.5
< (Tea : Medium) (Fruit juice : Medium) >	0.25	< (Coffee : Medium Butter : Medium) (Fruit juice : Low) >	0.25
< (Coffee : Medium, Butter : Medium) >	0.25	< (Coffee : Medium Butter : Medium) (Fruit juice : Medium) >	0.25
< (Coffee : Medium Butter : High) >	0.25	< (Coffee : Medium Butter : High) (Fruit juice : Low) >	0.25
< (Tea : Medium) (Fruit juice : High) >	0.25	< (Coffee : Medium Butter : High) (Fruit juice : Medium) >	0.25
< (Cream: Low) (Fruit juice : Medium) >	0.25		

VI. CONCLUSION

This paper introduced a new algorithm for mining sequences of more general items and concepts. This algorithm works based on the similar mental concepts and uses the Fuzzy PrefixSpan algorithm and gives more general results as output sequences. Moreover, the proposed method was able to find the sequences which might be hidden when no mental similarity was considered.

REFERENCES

[1] C. Faloutsos, M. Ranganathan, Y. Manolopoulos, "Fast Subsequence Matching in Time-Series Databases", Proceedings of the ACM SIGMOD International Conference on Management of Data, Minneapolis, Minnesota,
1994, pp. 505-511.

[2] B. LeBaron, A.S. Weigend, "A Bootstrap Evaluation of The Effect of Data Splitting on Financial Time Series", IEEE Transactions on Neural Networks, vol. 9, no. 1, 1998, pp. 213–220.

[3] Agrawal, R. and Srikant, R, "Fast Algorithms for Mining Association Rules in Large Databases", In 20th International Conference on Very Large Data Bases, Santiago de Chile, Chile, 1994, pp 487–499.

[4] C.Y. Chang, M.S. Chen, C.H. Lee, "Mining General Temporal Association Rules for Items with Different Exhibition Periods", IEEE International Conference on Data Mining, Maebashi City, Japan, 2002, pp.272-279

[5] C.H. Lee, M.S. Chen, C.R. Lin, "Progressive Partition Miner: An Efficient Algorithm for Mining General Temporal Association Rules", IEEE Transactions on Knowledge and Data Engineering, vol. 15, no. 4, 2003, pp. 1004–1017.

[6] Chen, G. and Wei, Q, "Fuzzy association rules and the extended mining algorithms", Fuzzy Sets and Systems, vol. 147, no. 1-4, 2002, pp. 201-228.

[7] Agrawal, R. and Srikant, R, "Mining sequential patterns, in P. S. Yu and A. S. P. Chen, eds, '11th International Conference on Data Engineering (ICDE'95)', IEEE Computer Society Press, Taipei, Taiwan, 1995, pp. 3–14.

[8] R. Srikant, R. Agrawal, "Mining sequential patterns: generalizations and performance improvements", Research Report RJ 9994, IBM Almaden Research Center, San Jose, California, 1995.

[9] R. Srikant, R. Agrawal, "Mining sequential patterns: generalizations and performance improvements", Proceedings of the 5th International Conference on Extending Database Technology, Avignon, France, 1996, pp. 327-332.

[10] Hou, X., Gu, J., Shen, X., and Yan, W, "Application of Data Mining in Fault Diagnosis Based on Ontology", In Third International

Conference on Information Technology and Applications (ICITA'05), Sydney, Australia, 2005, pp. 260–263.
[11] Kuok C.-M. ,Fu A., Wong M. H., "Mining Fuzzy Association Rules in Databases", SIGMOD Record, vol. 27,no. 1, 1998, pp. 41-46.
[12] Dong.G., Pei.J ,"Sequence Data Mining", Springer-New York, 2007.
[13] Nancy P. Lin, Hung-Jen Chen, Wei-Hua Hao, Hao-En Chueh, Chung-I Chang, "Mining negative fuzzy sequential patterns", Proceedings of the 7th WSEAS International Conference on Simulation, Modelling and Optimization, 2007, pp. 111-117.