Fuzzy Sequential Patterns Summarization with Lattice Structure

Bhawna Mallick[1], Deepak Garg[1] and P. S. Grover[2]

[1]Deptt of Computer Science & Engg Thapar University, Patiala, India
[2]Deptt of Computer Science & Engg GTBIT, Delhi, India

Abstract

Data mining is new but an interdisciplinary field utilizing statistics, machine learning, and other methods. In recent years, fuzzy logic has also been applied to augment data mining. The application of fuzzy logics makes the mining results more understandable and interpretable, apart from being useful and informative. Fuzzy rules are useful to summarize large databases. Several studies are done on fuzzy association rules. In our study, we argue that it is important and interesting to mine rules with timing information, as given by fuzzy sequential patterns. We propose the use of lattice structure for the summarization of fuzzy sequential patterns. We show that mining such rules requires managing a lot of information and we propose the use of lattice structure.

Keywords: *Fuzzy sequential patterns, data mining, lattice structure*

1. Introduction

The main challenge for the end users who work with large databases is to retrieve concise and understandable summaries. The use of fuzzy logic can help to extract and maintain the rules and summaries from huge databases. This is achieved by first deriving linguistic summaries for the database and extending it further to fuzzy summaries. The user interaction is also useful to get interesting information along with different methods. Association mining is one such method that is most often used to derive linguistic and fuzzy summaries. However, these extracted rules do not take into account the order aspect of the data, and makes them less informative [1]. In this article, we present the construction of fuzzy rules over time, called as fuzzy sequential patterns. There has been some research work done on fuzzy sequential patterns. However, these studies do not discuss much on the implementation efficiency of the methods used. We focus here on database and rule management in order to mine fuzzy sequential patterns efficiently by making use of lattice structure. In our study we have justified that the lattice structure is as an effective tool and mechanism to represent fuzzy sequential patterns. The different item sets of a sequence can be hierarchically grouped together satisfying the properties of mathematical lattice structure and hence can be used for knowledge extraction.

2. Problem Formulation

We introduce here a short example that is used in the next sections to illustrate the need of fuzzy sequential patterns and use of lattice structure to represent them. The data in Table 1 refers to the customers' transaction from the computer retail shop. It shows eight transactions each having a unique identifier in the transaction database. Each transaction identifier represents the computer related item sets purchased by the customer in a single transaction. It may include the hardware and software items.

Table 1. A Transaction Database ‘S’

Transaction Identifier	Item Sets
T101	PC, Windows 7, Norton Ghost, McAfee antivirus plus, Expansion Card, Image Scanner.
T102	PC, Windows 7, Adobe Reader, Terapad, Norton Antivirus, Printer.
T103	Microsoft Office, Adobe Flash Player, Windows Media Player, Webcams, Video Card, Sound Card.
T104	Mac OS 9, McAfee Internet Security, Acronis Disk Director, LHMelt, Logical Disk Manager, Oracle, RAM. Barcode reader, Printer.
T105	Windows Media Player, Windows Live Movie Maker, RealPlayer, Quick Heal, Internet Security, Webcams. Video Card, Sound Card, Expansion Cards, Digital Camera, Microphones, Printer.
T106	PC, Network Card, CD-ROM, Printer, RAM, Windows 7, McAfee Internet Security, Total Recovery, Norton Ghost, Install Builder.
T107	PC, OS/2.1x, NOD32 Antivirus, Terapad, JExpress, LHMelt, Printer.
T108	Windows 7, Adobe Flash Player, Microsoft Office, Protector Plus Professional, Loudspeaker, CD-ROM, Sound Card.

From the database, the purchasing behavior/ pattern of the customer can be analyzed for different marketing and strategic reasons. For instance, this database can be used to summarize that “Most customers purchase at least one hardware and one software product. If we use association rules to build fuzzy summaries from this example database, we can get frequent correlated product purchases made by the customer like “Many customers purchasing PC would buy some hardware items and software” or “Most customers purchasing Adobe Flash /Windows media player also buy video and sound card”.

However information regarding the ordering of the purchases cannot be summarized using the association rules. In our sample database, using association rules or fuzzy association rules [2, 3] ,we cannot get the knowledge regarding the rules like “Most customers purchase antivirus software after deciding for the operating system software” or “The printer is the last purchase item for any customer”.

The implementation of sequential pattern mining and particularly the fuzzy sequential pattern mining can provide these useful rules. In fact, sequential patterns can be considered to be an extension of association rules with the order aspect of data into account. The use of fuzzy sequential patterns [4] provides more information and knowledge as compared to the sequential patterns. For example, use of sequential patterns mining gives us either the presence or absence of an item in a sequence like “Most customers purchase anti-virus software followed by operating system software”. On the other hand, the approximate number of antivirus software purchase is provided by the fuzzy sequential patterns.

This information given by the fuzzy sequential pattern mining can be used to study and analyze the customers’ purchasing behavior. The appropriate strategies formulated based on above analysis such as product line adjustments, or product shelf-space reallocations, constant promotions, seasonal reason, or the sales of new products and replacements may also influence customers’ purchase behavior in future.

3. Data Summarization using Association Rules and Sequential Patterns

Association rule mining is used for extracting useful and interesting correlations, associations and frequent patterns from the transaction databases or other data warehouses. For the example in Section 2, we can have an association rule like when the software windows media player is purchased by the customer, the hardware components video and sound card are brought along with it 100% times in all transaction. However, it is only 50% of the time the video card is purchased with adobe flash player software. These rules which are discovered from mining the transaction database of the computer retail shop can not only summarize the data but also can be used for various commercial reasons.

However, the sequential pattern mining gives rules with added information of ordering or the timestamp with the databases. In fact, the transaction databases are transformed to sequence databases with the ordering information to generate the sequential pattern rules. Consequently, the sequential patterns mined from sequence databases can be more useful to provide accurate information for business organizations for prediction and further make more sound strategic decisions. Hence, they find application in mining user access patterns for the web sites, predicting a certain kind of disease using the history of symptoms, making efficient inventory control by the retailers.

There is an important difference between the two methods of mining: association rule and sequential pattern. Sequential patterns are generated by finding the correlation between transactions while association rule mining is based on intra transactions. To elaborate it further, the association rules mining results are based on the items that are purchased together frequently and belong to the same transaction [5]. However, the results generated by the sequential pattern mining are based on the items that are brought in a certain order by the same customer and can be from different transactions.

Table 2. A Sequence Database 'D'

Sequence Identifier	Sequences
S100	< PC (Windows Media Player, Video Card, Sound Card)(Barcode reader, Printer) Quick Heal (Loudspeaker, CD-ROM)>
S200	<(PC, Windows 7) Adobe Reader (Logical Disk Manager, Oracle) (Digital Camera, Microphones)>
S300	<(PC, OS/2.1x) (Adobe Flash Player, Sound Card) (Windows Media Player, Video Card) JExpress,Printer >
S400	< PC, Network Card (Windows Media Player, Video Card) McAfee Internet Security, Acronis Disk Director, LHMelt >

Sequential pattern is a sequence of item sets that frequently occurred in a specific order, all items in the same item sets are supposed to have the same transaction-time value or within a time gap [4, 6]. Usually all the transactions of a customer are together viewed as a sequence, usually called customer-sequence, where each transaction is represented as an itemsets in that sequence, all the transactions are list in a certain order with regard to the transaction-time. Table 2, have four sequences of data represented with its own sequence identifier. Every sequence consists of item sets that are purchased in specific order; $<i_1, (i_2, i_3),\ldots..,i_n>$ represents that the item i_1 has been purchased before items i_2 and i_3 and its timestamp value is less than other items in the sequence. However, item i_2 and i_3 have same timestamp value, though more than i_1 and less than i_n.

4. Motivation for the Use of Fuzzy Sequential Patterns

This has been noticeably observed that the real-world databases consist of numerical and time-stamped data. We can get more relevant rules from these databases by making use of fuzzy set theory to minimize the sharp cuts between intervals. The sequential pattern mining is based on binary valued transaction data. This should however be extended to fuzzy sequential pattern mining for quantitative valued data. Fuzzy sequential mining generate simple and practical patterns which are close to human reasoning. For example, the transaction databases of Table 1 in Section 2 can have rules like “75% of people who purchase PC brought few hardware items and much software.”

The term fuzzy was introduced by Zadeh [7] and further studied to state that there could be additional zone apart from only true and false. The concept of fuzzy provides flexibility to model imprecisely defined conditions. It allows for approximate reasoning that is useful for expert systems with powerful reasoning capabilities. In fact, the logic behind any thought process is hardly two valued but based on imprecise and unclear truths and rules of inference. A fuzzy set theory is a logical extension of a crisp set. Crisp sets allow only full membership or no membership at all, and can take either 0 or 1 value. Fuzzy set theory extends this concept by defining partial membership and hence allows for any value between 0 and 1. This is done by introducing gradual memberships for the quantitative data using membership functions.

Therefore, each quantitative item has to be partitioned into several fuzzy sets to mine fuzzy sequential patterns [8]. This redefines the concept of attribute and itemset as compared to classical sequential patterns. The association of one item (attribute) and corresponding fuzzy set formulates the fuzzy item. For example, the software purchased by the customer can be modeled with [number, low] as a fuzzy item where low is a fuzzy set defined by a membership function on the software universe of possible values of the item ‘number’. The list of fuzzy items is called a fuzzy itemset denoted as a pair of sets (set of items, set of fuzzy sets associated to each item) [9]. To elaborate, ([number, low][quantity, small]) is a fuzzy itemset having two fuzzy item ‘number’ and ‘quantity’ for quantitative items ‘software’ and ‘hardware’ respectively. One fuzzy itemset contains only one fuzzy item related to one single attribute. A s-f-sequence [9] $S =< s_1 \cdot \cdot \cdot s_g >$ is a sequence constituted by‘s’ fuzzy itemsets s = (X, A) grouping together ‘f’ fuzzy items [x, a]. The sequence < ([number, low][quantity, small]) ([accessories, many]) > groups together 3 fuzzy items into 2 itemsets. It is a fuzzy 2-3-sequence.

5. Fuzzy Sequential Patterns Summarization

Mining fuzzy rules is one of the best ways to summarize large databases while keeping information as clear and understandable as possible for the end-user. The common approach to express such knowledge consists in deriving linguistic summaries, which can further be extended to fuzzy summaries. Such summarization often requires a user-interaction for quality and validity in order to select interesting and useful knowledge from the huge datasets. There are few methods based on functional dependencies or association rule mining that perform summarization using automatic generation. However these methods are useful for quantifying and reasoning [10]. For applications where it is vital to mine rules that express information about the order fuzzy sequential patterns summarization is helpful.

Initially, the fuzzy partitions are created for each of the numerical attributes in the given crisp dataset. Then, using these fuzzy partitions, the fuzzy version of the dataset are created by converting crisp numerical attributes and its associated numerical values into fuzzy attributes and its associated values/ membership degrees. Fuzzy sequential pattern mining is a significant approach, which deals with temporally annotated numerical data [11]. It allows mining of frequent sequences embedded in the records. However, such fuzzy sequential patterns, in their current form, do not allow extracting temporal tendencies that are typical of sequential data [12]. We can elaborate this by considering the example tables discussed in Section 2 and 3, and deriving the fuzzy sets for them. Table 1 has been parsed to see the frequency of purchases made for the hardware, software and accessories items from a computer retail shop. The results derived from this parsing are being shown in Table 3. The hardware item/s that cannot be placed on the slots of the motherboard has been considered as accessories. Similarly, Table 4 has results derived from the Table 2 parsed for frequency check of purchases made in sequential order.

Table 3. Parsing results of sequence database 'S'

Transaction Identifier	Hardware	Software	Accessories
T101	2	3	1
T102	1	4	1
T103	2	3	1
T104	1	6	2
T105	3	5	4
T106	4	6	1
T107	1	5	1
T108	2	4	1

Table 4. Parsing results of sequence database 'D'

Sequence Identifier	Hardware	Software	Accessories
S100	4	2	3
S200	1	4	4
S300	3	4	1
S400	3	4	0

The reduction of item sets in the mining process is possible due to the role of fuzzy sets that transform quantitative values into linguistic terms. An object or item may either belong to a particular set or not, in case of classical set theory. However, the fuzzy set theory makes it possible that the object can even belong to a set to a certain degree. This is achieved by using the linguistic knowledge for the property that defines the set. Interpretations of membership degrees include similarity, preference, and uncertainty. The quantitative attributes are first divided into fuzzy sets using linguistic terms like small, moderate, medium, large etc.

We further use the membership functions, to get the membership degrees for each attribute and fuzzy set. There are different types of membership functions that can be used like triangular, trapezoidal, generalized bell shaped, Gaussian curves, polynomial curves, sigmoid functions etc. By making use of these membership function and linguistic knowledge, we can get the fuzzy sets for each attribute, are given in Figure1 (a), (b) and (c).

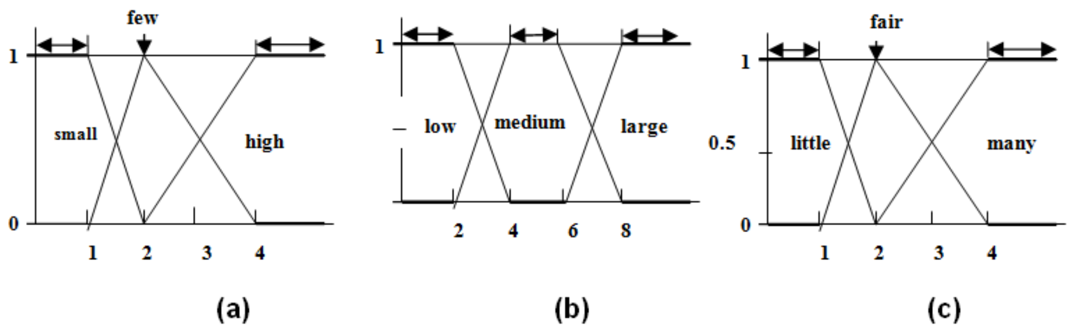

Figure 1. Fuzzy sets for elements hardware, software and accessories

As shown in figure above, a membership function is a curve to represent each point of the input space with a membership value, that is, degree of membership between 0 and 1. As an example, consider fuzzy sets small, few and high for universe of discourse hardware with attribute as ‘quantity’. With a crisp set, hardware quantity of ‘0’, ‘1’ is considered as ‘small’, ‘2’ as ‘few’ and any quantity ‘4’ and ‘above’ is considered as ‘high’. However, it is not clear that the hardware quantity of 3 purchased by customer is to be treated as which type of set. The corresponding fuzzy set with a smooth membership function is shown in Figure 1(a). The curve gives the transition from few to high quantity of hardware with the membership degree. Figure 1(b) and (c) shows the fuzzy sets for ‘number’ attribute of software and accessories.

Table 5. Membership degree of customer sequences for fuzzy sets

Trans-action Identifier	Hardware= quantity			Software= number			Accessories		
	small	few	high	low	medium	large	little	fair	many
T101		1		0.5	0.5		1		
T102	1				1		1		
T103		1		0.5	0.5		1		
T104	1				1			1	
T105		0.5	0.5		1				1
T106			1		1		1		
T107	1				1		1		
T108		1			1		1		

Any given element can be a member of more than one fuzzy set at a time. So, a fuzzy set ‘A’ in universe ‘U’ may be represented as a set of ordered pairs. Each pair consists of a generic element ‘E’ having a linguistic variable ‘x’ and its grade of membership ‘m’. For example, given the transaction database ‘S’ we can have the membership degree as given by the fuzzy sets in Figure 1(a) as {T101. few, 1}. This represents that transaction identifier ‘T101’ has ‘1’ degree membership for fuzzy set ‘few’, and {T105.high, 0.5} represents ‘T105’ has a membership degree of ‘0.5’ for fuzzy set ‘high’ for hardware. The membership degree for all attributes for all transactions is summarized in Table 5.

For given sequential database 'D', {S100. high, 1} shows that sequence identifier 'S100' has '1' degree membership for fuzzy set 'high', and {S300.few, 0.5} represents 'S300' has a membership degree of '0.5' for fuzzy set 'few' for hardware. The membership degrees of all attributes for customer sequences are summarized Table 6.

Table 6. Membership degree of customer sequences for fuzzy sets

Sequence Identifier	Hardware= quantity			Software= number			Accessories		
	small	few	high	low	medium	large	small	few	high
S100			1	1				0.5	0.5
S200	1				1				1
S300		0.5	0.5		1		1		
S400		0.5	0.5		1		1		

6. Role of Lattice structure

Lattice is an important concept of discrete mathematics. The lattice is characterized by specific properties which makes it quite different from other data structures like graph, tree etc. We can make use of this structure to represent and maintain the fuzzy sequential patterns. The typical definition of the lattice as given by mathematical theory is:

Let (P, ⊑) be a partially order set and A ⊆ P a subset of P. An element p Є P is called an upper bound for A if a ⊑ p for all a Є A. It is called a lower bound for A if p ⊑ a for all a Є A. If the set of all upper bounds of A has a smallest element, then this element is called the join or supremum or least upper bound of A. Similarly the largest lower bound of A (if it exists) is called the meet or infimum or greatest lower bound of A. For example, if we consider the set N of natural numbers with order relation |, "is a divisor of". Then the supremum of two elements a, b Є N equals lcm (a, b). The greatest common divisor gcd(a, b) is the infimum of {a, b}.A partially ordered set or poset (P, ⊑) is called a lattice, if for all x, y Є P the subset {x, y}of P has a supremum and an infimum. The supremum of x and y is denoted by x ⊔y and the infimum as x ⊓ y. We can say, (R, ≤) is a lattice, if x, y Є R, then sup{x, y} = max{x, y} and inf{x, y} = min{x, y}. Similarly, if S is a set and P = P(S) the poset of all subsets of S with relation ⊆, then P is a lattice with ⊔= ⋃ and ⊓=⋂.

As stated before, if (P, ⊑) is a lattice, then for all x, y, z Є P has certain properties [13] as;

1. x ⊔ x = x and x ⊓ x = x; (Reflexive)

2. x ⊔ y = y ⊔ x and x ⊓ y = y ⊓ x; (Symmetric)

3. x ⊔ (y ⊔ z) = (x ⊔ y) ⊔ z and x ⊓ (y ⊓ z) = (x ⊓ y) ⊓ z; (Transitive)

A lattice in which every subset has a supremum and infimum, is called a complete lattice. We can represent this visually by means of directed acyclic graph. This graph has nodes which represents elements of the poset and there is directed arc from node y to node x if and only if y ⊑ x. This type of graph is called as Hasse diagram that is used to represent a poset [14]. Usually the direction of arcs in the graph are avoided by showing node x above node y if y ⊏ x. So, the lattice can be used to represent the order relation or the hierarchical order of elements.

This data structure could be useful for the fuzzy summarization of the sequential pattern once the fuzzy set has been formulated using the linguistic knowledge and each data element is associated with the membership degree with these fuzzy sets using the membership function. To explain this let us take the example as discussed in Section 2. Considering a domain of discourse in which each element of a set of sequence of transaction, SID ={s_1, s_2, …, s_n} have one or more attributes ATT= {a_1, a_2, …, a_m}. We have triple F = <SID, ATT, M>, where M is a membership degree attained by a particular sequence identifier for a specific attribute. This membership degree is derived by using some membership function. For example in Table 6, we can define a database D = <P, ⊑ > related to triple F as consisting of a set P, of sequences of transactions which are partially ordered by the relation ⊑. The database is restricted as the attributes (ATT) in F form the maximal elements and the sequences of transactions (SID) in F are the minimal elements. We can represent this database as in Figure 2.

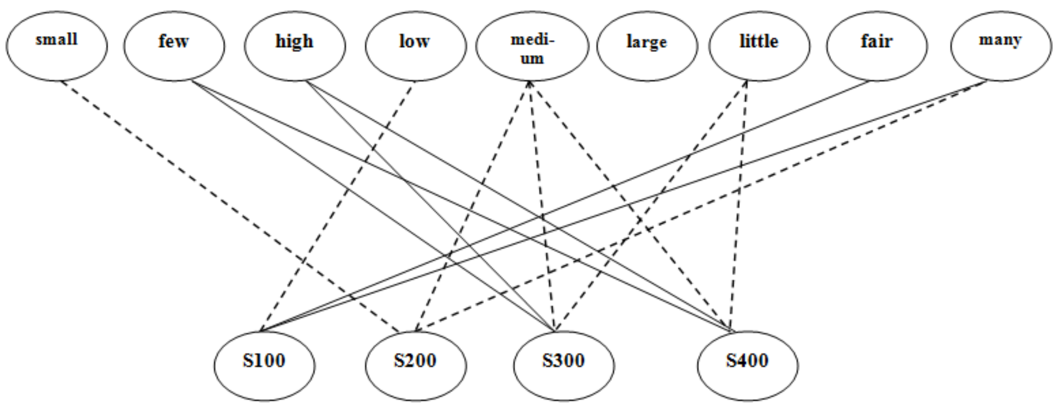

Figure 2. Graph summarizing the sequences with the fuzzy sets

However the above graph has a major drawback that by following the ascending paths of two different nodes we will reach to more than one single common node being at their top. For instance, by ascending path from ‘S300’ and ‘S400’ we reach to node ‘few’ as well as ‘high’. It is very unclear and ambiguous that whether the sequence ‘S300’ has feature of attribute ‘few’ or ‘high’. In fact due to fuzziness it has features of both the attributes that is ‘few’ and ‘high’ to certain membership degree. This is not reflected by the use of the graph.

Making use of lattice structure, P, with a property that every two elements of P have a least upper bound, called join or supremum and greatest lower bound known as meet or infimum we can take care of this problem. This characteristic feature is termed as a closure feature of lattice, where any two nodes must have one single supremum. Forcing the fulfillment of this condition, we may need to add some extra nodes to ensure the consistency of the system. The lattice theory ensures that these extra nodes, added as ‘artificial’ supremum of two closed sets will consist on the union of the maximal sequences that are contained by their immediate predecessors. We can have following lattice structure represented using the Hasse diagram in Figure 3 for the Table 6 with added nodes;

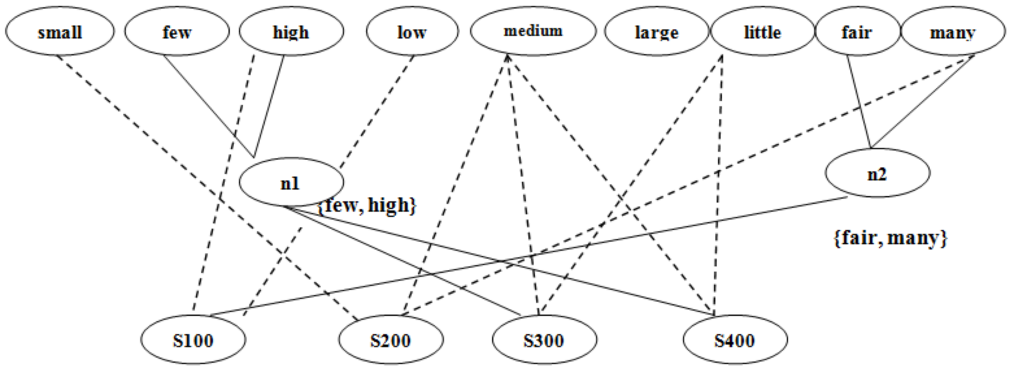

Figure 3 Lattice structure summarizing the sequences with the fuzzy sets

We are able to reflect the fuzziness in the sequences for certain attributes by using the lattice structure. It has good properties involving the completeness of pairs and independence of the order of input variable or the attributes. No doubt, lattice structure is a robust tool that can be utilized for data analysis and knowledge discovery

7. Conclusion

The very first step of knowledge discovery and data mining is to focus on extracting useful data from the huge databases. However, from the user view the non-numeric information usually provide truly useful results. So there is choice between the quantitative methods that give good performance and qualitative methods that are close to human reasoning to retrieve information. In recent times, there is need for user friendly methods that help the user to process, retrieve, exploit and clarify the available knowledge in a simple way. Here, the fuzzy set theory comes to an aid. We may have to compromise in terms of correctness, completeness and efficiency of extracted information using this technique, but they provide simple and understandable results. The mined results are only limited to frequency and association in case if fuzzy association rules are implemented. To get the knowledge regarding the order of the mining results, fuzzy sequential mining has to be used. The sequential mined results need to be represented and maintained with time. This can be done efficiently using the lattice-theoretic framework. The closure and certain properties of lattice model helps in making it a good option to represent the fuzzy data. The lattice can be represented using the undirected graph to get the fuzzy sequential patterns and consequently useful knowledge for business applications and otherwise.

Acknowledgements

The authors would like to thank the Editor and anonymous referees for providing valuable comments and constructive suggestions. We express our gratitude towards the Galgotias College of Engineering and Technology, Greater Noida for the support being provided to complete this research work.

References

[1] R. Srikant and R. Agrawal, "Mining quantitative association rules in large relational tables", Proceedings of ACM SIGMOD, International Conference on Management of Data, **(1996),** pp. 1–12.
[2] C. M. Kuok, A. W. -C. Fu and M. H. Wong, "Mining fuzzy association rules in databases", SIGMOD Record, vol. 27, no. 1, **(1998)**, pp. 41–46.

[3] T. P. Hong, K. Y. Lin and S. L. Wang, "Fuzzy data mining for interesting generalized association rules", Fuzzy Sets System, vol. 138, **(2003)**, pp. 255–269.
[4] R. Srikant and R. Agrawal, "Mining sequential patterns: Generalizations and performance improvements", Proceedings of 5th International Conference Extending Database Technology, **(1996)**, pp. 3–17.
[5] D. Dubois, E. Hüllermeier and H. Prade, "A note on quality measures for fuzzy association rules", Proceedings of 10th IFSA World Congress on Fuzzy Sets System, **(2003)**, pp. 346–353.
[6] T. P. Hong, K. Y. Lin and S. L. Wang, "Mining fuzzy sequential patterns from multiple-items transactions", Proceedings of 20th NAFIPS International Conference, **(2001)**, pp. 1317–1321.
[7] L. A. Zadeh, "Fuzzy sets", Information Control, vol. 8, (1965), pp. 338-353.
[8] R. S. Chen, G. H. Tzeng, C. C. Chen and Y. C. Hu, "Discovery of fuzzy sequential patterns for fuzzy partitions in quantitative attributes", Proceedings of ACS/IEEE International Conference on Computational System Application, **(2001)**, pp. 144–150.
[9] Y. C. Hu, R. S. Chen, G. H. Tzeng and J. H. Shieh, "A fuzzy data mining algorithm for finding sequential patterns", International Journal of Uncertainty Fuzziness Knowledge-Based System, vol. 11, no. 2, **(2003)**, pp. 173–193.
[10] Y. C. Hu, G. H. Tzeng and C. M. Chen, "Deriving two-stage learning sequences from knowledge in fuzzy sequential pattern mining", Information Science, vol. 159, **(2004)**, pp. 69–86.
[11] R. S. Chen and Y. C. Hu, "A novel method for discovering fuzzy sequential pattern using the simple fuzzy partition method", Journal of American Society Information Science, vol. 54, no. 7, **(2002)**, pp. 660–670.
[12] C. Fiot, A. Laurent and M. Teisseire, "Web access log mining with soft sequential patterns", Proceedings of 7th International FLINS Conference on Applied Artificial Intelligence, **(2006)**, pp. 519–524.
[13] N. P. Lin, W. -H. Hao and H. -J. Chen, "Fast Accumulation Lattice Algorithm for Mining Sequential Patterns", Proceedings of the 6th Conference on WSEAS/ International Conference on Applied Computer Science, Wisconsin, USA, vol. 6, **(2007)**.
[14] Z. Zhang, X. Qian and Y. Zhao, "Galois Lattice for Web Sequential Patterns Mining", Proceedings of the Pacific-Asia Workshop on Computational Intelligence and Industrial Application, PACIIA '08. Wuhan, **(2008)**, pp. 102 - 106.

Authors

Bhawna Mallick received the B.Tech in Computer Technology from Nagpur University, Nagpur, India and M.Tech in Information Technology from Punjabi University, Patiala, India. She is currently pursuing the Ph.D. degree and working as Head, Department of Computer Science & Engineering at Galgotias College of Engineering & Technology, Greater Noida affiliated to UP Technical University, India. She has 13 years of industry and academic experience with organizations like Infosys Technologies Ltd, Chandigarh, India and NIIT Technologies Ltd, New Delhi, India. She is member of IEEE (Institute of Electrical and Electronics Engineers), USA. Her area of research is Data Mining focusing on Sequential Mining of Progressive Databases.

Deepak Garg has done his Ph.D. in the area of efficient algorithm design from Thapar University. He is certified on latest technologies from Sun for Java Products, IBM for Web services and Brain bench for Programming concepts. He is Senior Member of IEEE (Institute of Electrical and Electronics Engineers), USA, Executive Member of IEEE Delhi Section and secretary of IEEE Computer Society, Delhi Section. He is Life Member of ISTE, CSI , IETE (Institute of Electronics and telecommunication Engineers), ISC (Indian Science Congress), British Computer Society and ACM, UK. He started his career as a Software Engineer in IBM Corporation Southbury, CT, USA and then with IBM Global Services India Pvt Ltd, Bangalore, India. He is presently working as Professor, Thapar University, Patiala.Deepak has 37 publications in International Journals and conferences. He is on the Editorial Board of seven International Journals. His active research area is Data Structure, Algorithms and Data Mining.

P. S. Grover received his Master's degree and doctorate from Delhi University, Delhi, India. He is widely travelled and delivered invited talks/key note addresses at many National/International Conferences/Seminars and Workshops. He is on the Editorial Board of four International Journals. He has written 9 books and many of his articles have appeared in several books published by IEEE of USA. He has published more than 100 research papers in international and national journals and conferences including published by IEEE, ACM and Springer. He is presently Director General at Guru Tegh Bahadur Institute of Technology, GGS Indraprastha University, Delhi, India. Formerly he was Dean & Head of Computer Science Department, Delhi University, Delhi, India. Prof Grover is a member of IEEE Computer Society. His current research interests are: Component based and Aspect-oriented Software Engineering, Autonomic Embedded Systems.