Journal of the Nigerian Association of Mathematical Physics
***Volume* 22 (November, 2012), pp 319 – 326**
© J. of NAMP

A Fuzzified User Constraints Framework for Mining Sequential Patterns

[1]David Allenotor and [2]Carson Kai-Sang Leung

[1]Department of Computer Science,
University of Benin, PMB 1154 Benin City, Nigeria
[2]Department of Computer Science,
University of Manitoba, R3T 2N2 MB, Canada

Abstract

Previous studies propose that the use of a relaxed user-controlled focus (user-controlled constraints) for mining sequential patterns in a large database has a high tendency for generating interesting patterns. However, the extent or degree of relaxation of the user controlled constraints that is capable of generating optimized sequential patterns was not specified. In this study, we propose a grading scheme based on fuzzy logic which is suitable for defining the degree of a user-controlled focus for sequential pattern mining in a large database.

Keywords: Data Mining, Sequential Patterns, Fuzzy logic, User-controlled Focus, Constraints Relaxation.

1.0 Introduction/Background

Sequential pattern mining describes the process of analyzing data collected over time to discover non-trivial, interesting, and meaningful trends [1]. This structure becomes apparent when the data to be mined is characterized by time or sequential attributes. Generally, we can describe the sequential pattern as an approach that is aimed at performing inter-transactional processes that are capable of dealing with sequences of sets of items from a database. It was motivated by the need in the retailing industry to capture regular or predictable trends in the sequences of items and they used such information for decision making as well as other related marketing strategies. Notably, data mining techniques have gained its popularity and wide research as a result of its application [1, 2, 3, 4, 5] in commercial area such as customer relations management, credit card fraud detection, and market basket analysis as well as in other medical (bioinformatics) and scientific fields.

The concept was introduced by Agrawal and Srikant [1] and in the following years, a number of other researches proposed several sequential pattern mining algorithms. Despite these popularity and wide research in data mining algorithms, the algorithms are, in most cases unable to provide results that optimize the various trade-offs associated with minimum support and confidence levels, or degree of the model expressiveness and the algorithmic time complexity [6]. Figure 1 shows a conceptual model for a hypothetical data mining system such as mining association rules and that based on sequential pattern mining. A sequential mining model attempts to bypass the cost intensive feedback loop by introducing a user defined (specified) constraint at the input.

From Figure 1, input to a sequential pattern mining system may consists of a set of sequences such as < a, b, c,…, z >, called data-sequences where each of these data-sequenceis list of transactions, that is ordered by increasing transaction-time tr. Each transaction(element of the sequence), consists of a set of items called item sets. Other inputs to thesequential mining systems may include a value of the predicted minimum support and constraints.The number of elements of a sequence is called the length of a sequence. The sequence of a sequential pattern with k is called k-sequence.

[1]Corresponding author: ***David Allenotor,*** E-mail: dallen@uniben.edu, Tel. +234 7056009584

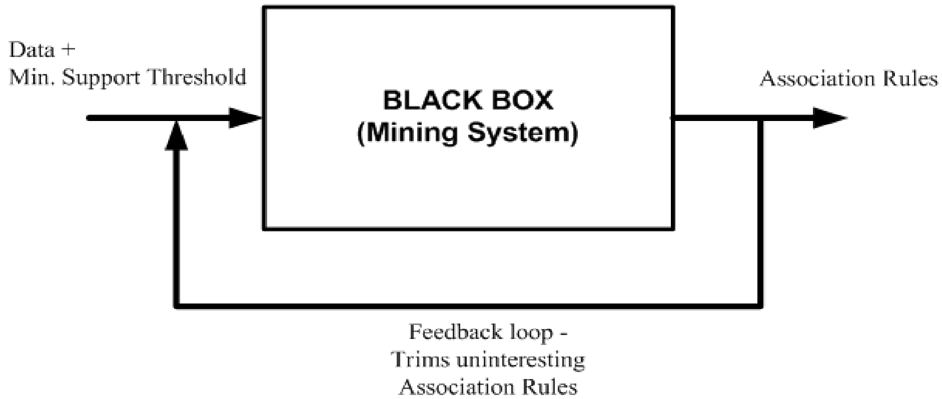

Figure 1: A hypothetical System for Association Rule Mining.

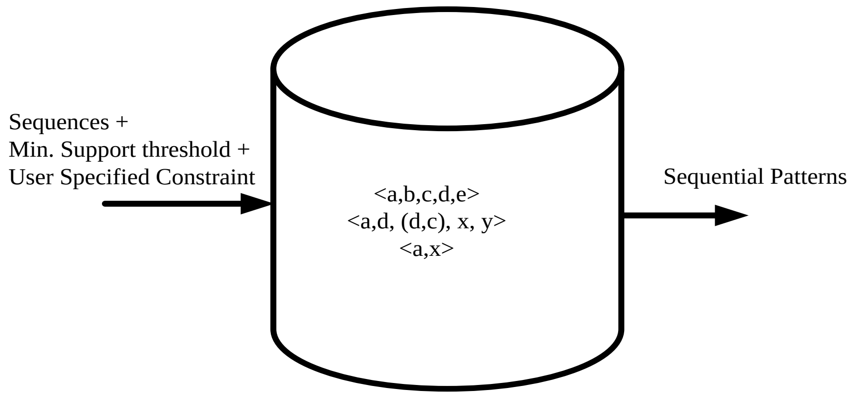

Figure 2: A hypothetical System for Sequential Pattern Mining.

In a k-sequence, the elements maynot necessarily contiguous. A sequence is characterized by its support – the percentage ofdata-sequences that contain this sequence. All sequences that are equal or greater than theuser-specified minimum support are called frequent sequences. A sequence is maximal if itis not contained in any other sequences. The primary goal of sequential pattern mining isthe discovering of maximal sequences among all frequent sequences. This could be achievedthrough a scan of the database and enforcing a more detailed set of rules. However, imposinga more detailed set of rules may require more iteration and thus, more database scans.Multiple database scans is among several factors responsible for the high I/O overheads thatis required to analyze a large number of generated candidate sets. For example, when thedatabase changes, the entire mining process may need to be repeated or some algorithm mustbe used to merge the rules from old and new data.

Particularly, several algorithms for discovery frequent patterns (despite allowing the userto specify contingent rules) often produce larger amount of patterns which may become uninterestingand ineffectual to the end user. In association rule mining such as depicted in Figure 1, a user does not have control over the output generated by the mining system.

The user is limited by sparse information provided by the minimum support. In Figure 2, the user is able to determine the output of the sequential pattern mining system by specifying a value for the constraint in the input. Although users may specify such contingent values, the resultant sequential pattern may not exactly represent the user's what the user wish to extract from the database. This may largely be due to inability of the user to focus the discovery process on required expectations and lack of background experience. To minimize this problem, recent researches had resort to the use of constraints to restrict the number and the span of discovered patterns. The application of user constraints makes it possible to focus attention on patterns or sub-patterns where user-relevant information may be extracted. Also, extracting useful patterns may depend of data mining ability to express the associated constraints [6]. The data miner may choose to relax or tight the bounds on constraints that are necessary to achieve the mining set goals. The question one may want to ask is to what extent is it required relaxing or tight a constraint bound? However, as noted in some researches [7], restricting the search too much may degenerate into an approximation problem which may result to statistical hypothesis and inferences. As application domains and decisions variables differs, it may become necessary to reason that no hard and fast rules could exactly determine the exact extent to restrict a constraint bound. To capture the vagueness in expressing a user constraint, we therefore, introduce fuzzy logic [8].

The rest of this paper is organized as follows. In Section 2 weprovide the relevant related work. Section 3 presents material associated with fuzzy logic. Section 4 provides our proposed fuzzified user-controlled constraint relaxations and the analysis.In Section 5, we draw up the conclusions.

2.0 Related Work

The specification of user constraints is one of the necessary requirements for mining interesting patterns in a database.

In sequential pattern mining algorithms, the focus is on the discovering of maximal frequent sequence(s) in a given database provided that the data to be mined has sequential properties associated with time. In contrast to some existing data mining approaches [1, 9, 10] sequential mining attempts to incorporate a human (user) interface with DBMS to achieve two main objectives; (i) to circumvent the high CPU and I/O cost of generating large candidate sets and (ii) to allow a user-controlled constraints into the mining system which is capable of reducing the volume of uninteresting patterns. The user may need to use associated support and confidence levels that are arbitrary (trial and error) – set high enough to get the algorithm to terminate with reasonable performance. The consequence of arbitrarily selecting the support and confidence levels may include but not limited to a total rejection of useful patterns because the algorithm did not go deep enough – it lack focus on the user-controlled constraints. Lack of focus, a major characteristic of pattern mining in sequential data is one of the limitations of sequential pattern mining. Lack of focus is responsible for the inherently large number of rules that is generated from a proportionately small set of sequential data. A number of researches [11, 12] have proposed the used of user constraints but none of these have consider the performance of the pattern mining algorithm when the user constraint assumes extreme values such as "very close" to the relaxed value or "very far" from the relaxed value.

Essentially, in sequentially pattern mining, minimum support level and confidence level had followed an arbitrary or intuitive selection schemes. Instead of arbitrarily selecting a support and confidence level that may lead to several iterations, Garofalakis et al. [11] propose a family of the SPIRIT algorithms. They show that the application of a relaxed "somewhat too generic" constraint (replacement of C by a weaker C´) C´ was capable of effective and efficient candidate generation and candidate pruning phases of the SPIRIT algorithms. However, Garofalakis et al. [11] are silent about two important issues; (i) the degree of relaxation of C that was necessary to provide user-targeted interesting patterns and (ii) the consequence(s) of over-relaxing or under relaxing the user-constraint C´. The degree of relaxation of C shows that the relaxed value C´ is a set of alternatives such as C´ = c´

The question of how well the constraint may be relaxed may be evaluated by applying fuzzy logic techniques to measure the degree of satisfaction and or, the relative flexibility between C´ (the relaxed value) and C, the user-specified constraint value.

The consequence of arbitrarily selecting the support and confidence levels may include but not limited to a total rejection of useful patterns because the algorithm did not go deep enough – it lack focus on the user-controlled constraints. Lack of focus, a major characteristic of pattern mining in sequential data is one of the limitations of sequential pattern mining. Lack of focus is responsible for the inherently large number of rules that is generated from a proportionately small set of sequential data. A number of researches [11, 12] have proposed the used of user constraints but none of these have consider the performance of the pattern mining algorithm when the user constraint assumes extreme values such as "very close" to the relaxed value or "very far" from the relaxed value. Essentially, in sequentially pattern mining, minimum support level and confidence level had followed an arbitrary or intuitive selection schemes. Instead of arbitrarily selecting a support and confidence level that may lead to several iterations, Garofalakis et al. [11] propose a family of the SPIRIT algorithms. They show that the application of a relaxed "somewhat too generic" constraint (replacement of C by a weaker C´) C´ was capable of effective and efficient candidate generation and candidate pruning phases of the SPIRIT algorithms. However, Garofalakis et al. [11] are silent about two important issues; (i) the degree of relaxation of C that was necessary to provide user-targeted interesting patterns and (ii) the consequence(s)

In contrast to some existing data mining approaches [2, 6, 13] sequential mining attempts to incorporate a human (user) interface with DBMS to achieve two main objectives; (i) to circumvent the high CPU and I/O cost of generating large candidate sets and (ii) to allow a user-controlled constraints into the mining system which is capable of reducing the volume of uninteresting patterns. The user may need to use associated support and confidence levels that are arbitrary (trial and error) – set high enough to get the algorithm to terminate with reasonable performance. The consequence of arbitrarily selecting the support and confidence levels may include but not limited to a total rejection of useful patterns because the algorithm did not go deep enough – it lack focus on the user-controlled constraints. Lack of focus, a major characteristic of unsupervised pattern mining in sequential data is one of the limitations of sequential pattern mining. Lack of focus is responsible for the inherently large number of rules that is generated from a proportionately small set of sequential data. A number of researches such as [11, 12] have proposed the use of constraints but none of these have considerthe performance of the pattern mining algorithm when the user constraint assumes extreme values such as “very close” to the relaxed value or “very far” from the relaxed value.

Essentially, in sequentially pattern mining, minimum support level and confidence level had followed an arbitrary or intuitive selection schemes. Instead of arbitrarily selecting a support and confidence level that may lead to several iterations, Garofalakis et al. [11] propose a family of the SPIRIT algorithms. They show that the application of a relaxed “somewhat too generic” constraint (replacement of C by a weaker C′) C′ was capable of effective and efficient candidate generation and candidate pruning phases of the SPIRIT algorithms. However, Garofalakis et al. [11] are silent about two important issues; (i) the degree of relaxation of C that was necessary to provide user-targeted interesting patterns and (ii) the consequence(s) of over-relaxing or under relaxing the user-constraint C′. The degree of relaxation of C shows that the relaxed value C′ is a set of alternatives such as $C' = c_1', c_2', \ldots, c_n'$ The question of how well the constraint may be relaxed may be evaluated by applying fuzzy logic techniques to measure the degree of satisfaction and or, the relative flexibility between C′ (the relaxed value) and C, the user-specified constraint value. In this paper, we investigate the degree of relaxation that is required to provide effective candidate generation and pruning associated with sequential mining.

Existing research such as SPIRIT [11] presents a family of apriori-based algorithms that uses a regular language to constrain the mining procedure. SPIRIT introduces the notion of constraint relaxation to speed up the SPIRIT algorithm by using a weaker constraint (more relaxed constraint). The SPIRIT and GSP [9] algorithms have similar philosophy, however, a difference exist in the candidate generation step. The candidate generation step creates a set of candidates that potentially satisfy the constraint. Most of the interesting sequence patterns generated are not anti-monotone hence, the candidate generation phase use constraint relaxations to guide the generation of sequences that meets the user specified constraint. Other researches that base sequence generation on the user-specified constraints include PreFixSpan [7] and Srikant and Agrawal [4]. Srikant and Agrawal [4] provide an approach for generalizing a framework for sequential pattern mining [9]. The generalized schemes include; time constraints and more importantly, user defined taxonomy. In these researches [7, 9, 11], the value of relaxed constraints is arbitrarily obtained by a user and the interestingness of the associated generated sequences is based on the intuitive ability of the user's selection and a prior knowledge of the database. In this research, we are interested

In this research, we are interested in providing answers to the following questions:

i. How well does C′ approximate the value of R without adding errors due to approximations?
ii. What parameters determine the choice for selecting C′ that guarantees interesting sequential patterns?
iii. Since user's constraint for mining sequential patterns in a database varies, how far from R can the value of C′ be approximated?
iv. What effects would arbitrarily varying the value of C′ have on the performance issues associated with mining sequential patterns in user constrained environment?

3.0 Basic Fuzzy Logic Concepts

Zadeh [8] introduced fuzzy set theory as a technique for modeling uncertainty (approximation)in natural language. In traditional crisp set theory, an object has two distinct possible membership values. Either the object is a member of the set or it does not belong to the set. Therefore, a logical proposition either holds or does not hold for a traditional set. In fuzzy set theory, a membership function describes the degree to which an object belongs to a fuzzy set. It attaches a numerical magnitude to a fuzzy set and maps crisp inputs from a specified domain to membership grades that lie between 0 and 1. If A is a fuzzy set in domain X, then _x(A) is called the membership function of the fuzzy set A. Unlike traditional set theory, fuzzy set theory extends beyond the boundaries of true or false to the degree of truth or falsity (the partial truth concept) expressed as a membership function in a fuzzy set. Fuzzy techniques, unlike Boolean techniques, therefore, have the potential for handling graded approximations such as the constraint definitions for mining sequential patterns in a large database.

4.0 Fuzzification of User-Controlled Constraints Relaxation

The basic idea of constraints relaxations were first used to improve the performance of algorithms[11]. According to Garofalakis et al. [11], a constraint may be naive, conservative,approximate, or non-accepted. Generally, a constraint may actually be seen as an approximationsuch as C′ to the actual constraint C. While C represents the known informationabout the domain in the problem definition, the relaxed constraint C′ encapsulates a methodthat extends that knowledge that is being represented. Since they do not represent existingknowledge, they cannot be simple constraints. When a relaxed constraint is used in the place of the constraint, there is likelihood that some unknown information about the database willbe discovered. Such discovering of unknown information may match the user expectationssomewhat. If this condition holds, and the relaxed constraint is used in a sequential miningprocess, with a view to discovering patterns that satisfy the imposed constraint, then therelies every possibility that unknown uninteresting patterns may be obtained. Therefore giventhat a user may choose any level of constraints relaxation deserved, it is necessary to obtaina level that ensures that the goals of sequential pattern mining are not compromised. In thefollowing sections, we explore how to represent user constraints using fuzzy logic. First, weemploy the concept of constraints selection and its fuzzification based on [11]. This will be followedby a general constraints selection procedure and then relate it to the concept proposedin SPIRIT[11]. In the SPIRIT procedure, lines 1, 6, and 15 – 17 uses a weaker C´ (given thatevery sequence that satisfy C also satisfies C´) to express the strength of the degree to whichthe user-constraint C may be pushed into the mining system. The set of relaxed constraintsmay be expressed over a scale according to a pre-defined degree to which those constraintsmay be pushed into the algorithm. The depth such degree extends from a value that provides best sequence patterns to the worst case scenario. For simplicity, let the degree we want topush these constraints be denoted by X, the set of alternatives, then a preference relation suchas $\geq X^2 \rightarrow \{0,1\}$ (where $x \geq y$ means "x is at least as good as y") assigns the number 0 or 1 to two alternatives x and y. i.e., the set of x values that weakly relax those of y. The strict preference relation "$\geq$" may be defined by $x \geq y \Leftrightarrow x \geq y$ but not $y \geq x$. Using the same set of alternatives, we can define a standard fuzzy presence relation as follows:

$$R(x,y): X^2 \rightarrow [0,1] \tag{1}$$

This fuzzy relation shows the degree of associating x with y in a range of membership function that exists between 0 and 1 and enables us to define the user-controlled constraints in [14] as a Fuzzy Choice Problem (FCP).FCP is defined by a set of variables $\chi = \{x_1, x_2, \cdots, x_N\}$ in a universe of discourse denoted by a domain $D = \{d_1, d_2, \cdots, d_N\}$ where D_i is a finite domain of X_i, and a set of fuzzy constraints defined by:

$$C = \{c_1, c_2, \cdots, c_K\} \tag{2}$$

where c_k = (V_k), R_k and is the set of variables concerned by the constraint c_k and R_k is a fuzzy set defined over a Cartesian product of the domains of the variables in V_k. Hence, we can associate a membership value to each tuple of V_k values by expressing the membership degree preferences of C. The membership degree preferences of C denoted by μ(C) may be defined using a general form of the trapezoidal membership function is given in Equation (2) as:

$$\mu_C(x) = \begin{cases} 1 & \text{for } x \in [b,c] \\ \dfrac{x-a}{b-a} & \text{for } a \leq x \leq b \\ \dfrac{d-x}{d-c} & \text{for } c < x \leq d \\ 0 & \text{otherwise i.e., for } x \notin [a,b] \end{cases} \tag{3}$$

Figure 2 shows a trapezoidal membership function. The fuzzy mean of the trapezoidal membership function applied as a defuzzification to Equation (3) is given as:

$$[((d-a)+(c-a)+(b-a))/4]+a \tag{4}$$

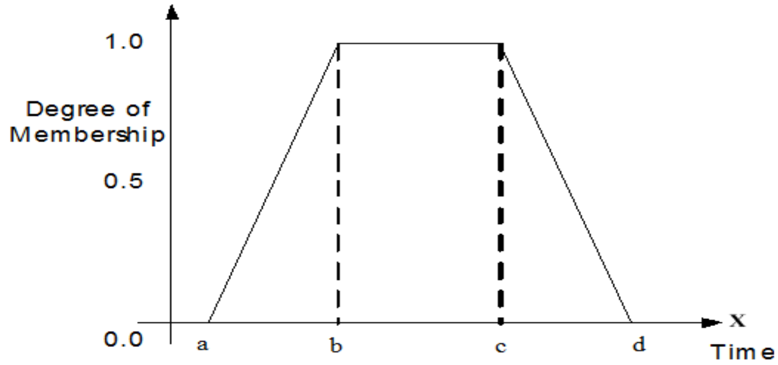

Figure 3: Trapezoidal Membership Function.

For any constraint $C=\{c_1, c_2, \cdots, c_K\}$, each $c_1, c_2, \cdots, c_K$ is associated with [c_{ai}, c_{bi}, c_{ci}, c_{di}].For i = 1,2,…, k, we define a fuzzy constraint C as follows:

$$\mu\ (c_i)=\begin{cases} 1 & \text{for } x\in[c_{bi}, c_{ci}] \\ \dfrac{x-c_{ai}}{c_{bi}-c_{ai}} & \text{for } \mathrm{c}_{ai}\le x\le c_{bi} \\ \dfrac{c_{di}-x}{c_{di}-c_{ci}} & \text{for } \mathrm{c}_{ci}< x\le c_{di} \\ 0 & \text{otherwise i.e., for } x\notin[c_{ai}, c_{bi}] \end{cases} \tag{5}$$

In Figure 2, [a, d] marks the universe of discourse, b and c are the values of X for whichμc(x) = 1. In Equation (2), the values of C = {c_1, c_2, c_3, c_4, c_k} represents the graded userconstraints where $c_1 < c_2 < c_3 < c_4 <$ ckare the graded values of user-controlled constraints.These values are expressed in terms of the relative closeness of C´ from C. The degree ofthe closeness is given in Figure 4 with a graded constraints for k = 5. Consequently, wecan express each value of {c_1, c_2, c_3, c_4, c_k} as a trapezoidal membership function and showthe rated user-constraints values in Table 1. To illustrate this idea, we give a fundamentalexample and assume that we have a minimum of 4 labels [15] in the area of discourse. Ourexample is given as follows:

Table 1.0: Rated Constraint Levels

Fuzzy Constraints	a	b	c	d	Crisp value
c1	0.00	0.00	1.00	3.00	0.125
c2	1.00	3.00	5.00	7.00	0.667
c3	5.00	7.00	9.00	11.00	0.875
c4	9.00	11.00	12.00	14.00	1.90
c5	12.44	14.00	16.00	18.00	2.50

From Table 1.0, we obtain the trapezoidal membership function for {c_1, c_2, c_3, c_4, c_5} as follows:

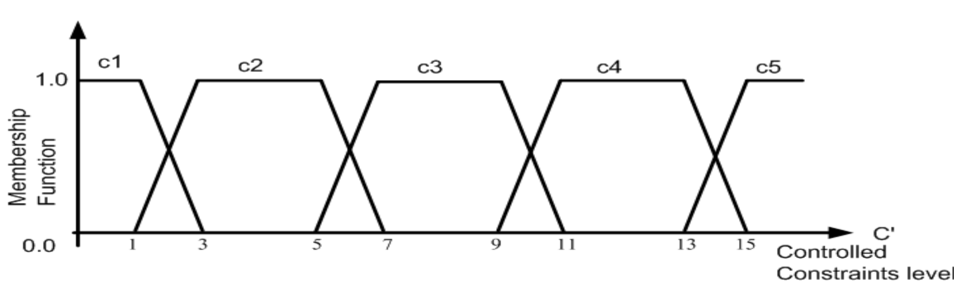

Figure 4: Trapezoidal Membership Function for User-Controlled Constraints.

$$\mu(c_1) = \begin{cases} 1 & \text{for } x \in [0.0, 1.0] \\ undefined & \text{for } 0.00 \le x \le 0.00 \\ \dfrac{3.0 - x}{3.0 - 1.0} & \text{for } 1.0 < x \le 3.0 \\ 0 & \text{otherwise i.e., for } x \notin [0.00, 19.0] \end{cases} \tag{6}$$

$$\mu(c_2) = \begin{cases} 1 & \text{for } x \in [3.0, 5.0] \\ \dfrac{x\text{-}1.0}{3.0\text{-}1.0} & \text{for } 1.0 \le x \le 3.0 \\ \dfrac{7.0 - x}{7.0 - 5.0} & \text{for } 5.0 < x \le 7.0 \\ 0 & \text{otherwise i.e., for } x \notin [0.00, 19.0] \end{cases} \tag{7}$$

$$\mu(c_3) = \begin{cases} 1 & \text{for } x \in [7.0, 9.0] \\ \dfrac{x\text{-}5.0}{7.0\text{-}5.0} & \text{for } 5.0 \le x \le 7.0 \\ \dfrac{11.0 - x}{11.0 - 9.0} & \text{for } 9.0 < x \le 11.0 \\ 0 & \text{otherwise i.e., for } x \notin [0.00, 19.0] \end{cases} \tag{8}$$

$$\mu(c_4) = \begin{cases} 1 & \text{for } x \in [11.0, 13.0] \\ \dfrac{x\text{-}9.0}{11.0\text{-}9.0} & \text{for } 9.0 \le x \le 11.0 \\ \dfrac{15.0 - x}{15.0 - 13.0} & \text{for } 13.0 < x \le 15.0 \\ 0 & \text{otherwise i.e., for } x \notin [0.00, 19.0] \end{cases} \tag{9}$$

$$\mu(c_5) = \begin{cases} 1 & \text{for } x \in [15.0, 17.0] \\ \dfrac{x\text{-}12.0}{15.0\text{-}12.0} & \text{for } 12.0 \le x \le 15.0 \\ \dfrac{19.0 - x}{19.0 - 17.0} & \text{for } 17.0 < x \le 19.0 \\ 0 & \text{otherwise i.e., for } x \notin [0.00, 19.0] \end{cases} \tag{10}$$

From our example, the crisp value for C = {c_1, c_2, c_3, c_4, c_5}= 0.125, 0.667, 0.875, 1.930, 2.580respectively and from Equation (1), the value of a fuzzy membership function must lie in the regionof [0, 1]. Our example show that {c_1 = 0.125, c_2 = 0.667, c_3 = 0.875, c_4 = 1.930, c_5 = 2.580}.These values means that the lowest value C (or the constraint value C´) that could be relaxedis 13%. The highest values C´ could have is 0.875. In our example, C´ may also assume c_4 = 1.930, c_5 = 2.580. However, the user may have over constrained (under relaxed theconstraints) if C´ assumes c_4 or c_5, values that exists outside the fuzzy membership function.Therefore c_4 = 1.930, c_5 = 2.580 may not be a good choice to select a constraint value for auser-constraint.

5.0 Conclusion

In this paper, we have obtained an important, intuitive fuzzification technique of the previouscrisp and weak axiom for user-controlled focus. Our fuzzification technique is suitable forgrading constraints values for mining sequential patterns in database. We have also showthat the degree of constraints relaxation is not trivial or based on arbitrary selection of userconstraintsfor effective mining of interesting sequential patterns. For example our degree ofrelaxation was 5 out of the k graduated scales. This indicates that a degree close to 1 isassumed a worst case while a k degree is assumed the best and closest to the user-specifiedconstraint.

References

[1] Rakesh Agrawal and Ramakrishnan Srikant. Mining Sequential Patterns. In Proceedings of the 11th International Conference on Data Engineering, March 1995.

[2] Heikki Mannila, Hannu Toivonen, and A. Inkeri Verkamo. Discovery of Frequent Episodes in Event Sequences. Data Mining and Knowledge Discovering, 1(3):259-289, 1997.

[3] Jian Pei, Jiawei Han, Behzad Mortazavi-Asl, Helen Pinto, Qiming Chen, Umeshwar Dayal, and Meichun Hsu. PreFixSpan: Mining Sequential Patterns by Prefix-Projected Growth. In Proceedings of the 17th International Conference on Data Engineering, pages 215-224, 2001

[4] Ramakrishnan Srikant and Rakesh Agrawal. Mining Quantitative Association Rules in Large Relational Tables. In SIGMOD '96: Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data, pages 1-12, 1996.

[5] Mohammed J. Zaki. SPADE: An Efficient Algorithm for Mining Frequent Sequences. Mach. Learn., 42(1-2):31-60, 2001.

[6] Bayardo R. J. The Many Roles of Constraints in Data Mining. SIGKDD Explorations. 4(1):i-ii, 2002.

[7] Hipp J. and Guntzer U. Is Pushing Constraints Deeply Into The Mining Algorithms Really What We Want? SIGKDD Explorations., 4(1):50-55, 2002.

[8] Lotfi Zadeh. Fuzzy Sets. Information and Control, 8(2):338-353, March 1965.

[9] Richard Relue, Xindong Wu, and Hao Huang. Efficient Runtime Generation of Association Rules. In CIKM '01: Proceedings of the tenth International Conference on Information and Knowledge Management, pages 466-473, 2001.

[10] Jong Soo Park, Ming-Syan Chen, and Philip S. Yu. An Effective Hash-Based Algorithm for Mining Association Rules. In SIGMOD '95: Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data, pages 175-186, 1995.

[11] Minos N. Garofalakis, Rajeev Rastogi, and Kyuseok Shim. SPIRIT: Sequential Pattern Mining with Regular Expression Constraints. In VLDB 1999: Proceedings of the 25th International Conference on Very Large Data Bases, pages 223-234, 1999.

[12] Jiawei Han and Jian Pei. Mining Frequent Patterns by Pattern-Growth: Methodology and Implications. SIGKDD Explor. Newsletter 2(2):14-20, 2000.

[13] Rakesh Agrawal and Ramakrishnan Srikant. Fast Algorithms for Mining Association Rules in Large Databases. In VLDB 1994: Proceedings of the 20th International Conference on Very Large Data Bases, pages 487-499, 1994.

[14] Asis Banerjee. Fuzzy Choice Functions, Revealed Preference and Rationality. Fuzzy Sets and Systems, 70(1):31-43, 1995.

[15] Sanya Mitaim and Bart Kosko. What is The Best Shape of a Fuzzy Set in Function Approximation?In IEEE International Conference on Fuzzy Systems (FUZZ-96), pages 1237-1243, September 1996.