Discovery of Unexpected Recurrence Behaviors in Sequence Databases

Haoyuan Li, Anne Laurent, Pascal Poncelet

Discovery of Unexpected Fuzzy Recurrence Behaviors in Sequence Databases

Dong (Haoyuan) Li [1], Anne Laurent [2] and Pascal Poncelet [2]

[1]LI, Université François Rabelais de Tours,
3 place Jean Jaurès, 41029 Blois, France
Haoyuan.Li@univ-tours.fr

[2]LIRMM, Université Montpellier 2,
161 rue Ada, 34392 Montpellier, France
laurent@lirmm.fr, poncelet@lirmm.fr

Abstract: **The discovery of unexpected behaviors in databases is an interesting problem for many real-world applications. In previous studies, unexpected behaviors are primarily addressed within the context of patterns, association rules, or sequences. In this paper, we study the unexpectedness with respect to the fuzzy recurrence behaviors contained in sequence databases. We first propose the notion of fuzzy recurrence rule, and then present the problem of mining unexpected sequences that contradict prior fuzzy recurrence rules. We also develop, UFR, an algorithm for discovering the sequences containing unexpected recurrence behaviors. The proposed approach is evaluated with Web access log data.**

Keywords: Data mining, sequence database, fuzzy recurrence rule, unexpectedness.

I. Introduction

During the past years, as very important models of data mining, association rules (frequent patterns) [1] and sequential patterns [2] have received much attention, such as the work addressed in [5, 14, 15] and [3, 27, 32, 35, 40, 42].

Association rule mining finds the frequent correlations between attribute sets (a.k.a. *patterns*) as rules in the form "if X then Y", where X and Y are two patterns. An association rule reflects the information typically like "60% of customers who purchase Coca Cola also purchase potato chips (if *Coca Cola* then *potato chips*)". Different from association rules, the goal of mining sequential patterns is to find frequent correlations in sequence data, where a sequential pattern is a frequent sequence depicting that "A then B then C then ...", where A, B, C, ... are patterns. A sequential pattern can help interpreting the information typically like "60% of customers purchase beers, then purchase Sci-Fi movies, and then purchase rock music".

On the other hand, the discovery of unexpected behaviors [33] contradicting prior knowledge (which in general stands for frequent or predefined behaviors) becomes more and more interesting for many real-word applications. In previous studies of discovering unexpected behaviors, unexpectedness is mainly stated in the context of patterns [20, 25], association rules [24, 29, 30, 31, 37, 36, 38, 39], or sequences [22, 34].

In our previous work [22], we proposed a semantics based framework of unexpected sequence mining. For instance, according to the behavior "people purchase Sci-Fi movies, and then purchase rock music", the behavior "people purchase Sci-Fi movies, and then purchase classical music" can be considered as unexpected, if the classical music is considered as semantically opposite to the rock music. This work has been extended with fuzzy methods in [23].

In this paper, we are interested in the unexpectedness stated by *fuzzy recurrence rule*, in the form "if the sequence s_α repeatedly occurs, then the sequence s_β repeatedly occurs". For instance, a fuzzy recurrence rule can be "60% of customers who *often* purchase Sci-Fi books then Sci-Fi movies later, also purchase PC games *often*". This type of rules reflects the associated correlations between repeatedly occurred elements in sequential data. The unexpectedness on recurrence behaviors is determined by the domain-expert-defined semantic oppositions. For instance, if we consider that the classical music is semantically opposite to PC games, then the fact "1% customers who *often* purchase Sci-Fi books then Sci-Fi movies later, *often* purchase classical music" stands for an unexpected recurrence behavior in a customer transaction database. In this case, the unexpectedness can also be determined from the description occurrence, such like the consequence "*rarely* purchase PC games" is opposite to "*often* purchase PC games".

Such unexpected recurrence behaviors can be interesting for many application domains, including marketing analysis, finance fraud detection, DNA segment analysis, Web content personalization, network intrusion detection, weather prediction, and so on.

The remainder of the paper is organized as follows. In Section 2, we introduce the related work. In Section 3, we propose the notions of fuzzy recurrence rules and present a belief-driven approach to unexpected recurrence behavior discovery. In Section 4, we develop an effective algorithm UFR for discovering unexpected recurrence behaviors in a sequence database. Finally, we conclude in Section 5.

II. Related Work

In data mining, fuzzy set theory [41] have been many employed to change the domain of the attributes, employing granules defined by fuzzy sets instead of precise values.
For instance, an association rule $X \rightarrow Y$ depicts the relation "if X then Y" between patterns X and Y. With fuzzy sets, there is a very extended way of considering fuzzy association rules as "if X is A then Y is B" in considering various information of attributes (mostly *quantitative attributes*), such as the type "if `beer` is `lot` then `potato chips` is `lot`" or "if `age` is `old` then `salary` is `high`" [6, 9, 11, 16, 21, 19].
In the same manner, the notion of fuzzy sequential patterns [7, 17, 8, 12, 13] considers the model sequential patterns like "60% of `young people` purchase a `lot` of `soft drinks`, then purchase `few opera movies` later, then purchase `many PC games`", where the sequence represents "people is `young`, then `soft drinks` is `lot`, then `opera movie` is `few`, and then `PC game` is `many`".
Another application of fuzzy set theory is to discovery *gradual* patterns and rules [18, 4, 10, 13]. In this form of fuzziness in quantitative attributes considers the correlations within the gradual trends of the values of attributes, such as the association rule "if `age increases` then `salary increases`", or the sequential pattern "the `more` visits of `search` page, the `more` visits of `KB` articles later, and at the same time the `less` visits of `question` submitting page".
Unexpected behaviors are generally considered within the framework of subjective interestingness measure. The discovery of unexpectedness depends on prior knowledge of data that indicates what users expect. Thus, in comparison with the data mining methods based on statistical frequency of data, the methods to discover *unexpectedness* contained in data can be viewed as a process using user-oriented *subjective measures* instead of using data-oriented *objective measures*.
The notions of objective measure and subjective measure for finding potentially interesting patterns (and sequential patterns) or rules are addressed in terms of *interestingness measures* for data mining. McGarry systematically studied the development of interestingness measures in [28], where objective measures are considered as using the statistical strength (such as *support*) or structure (such as *confidence*) of discovered patterns or rules to assess their degree of interestingness however subjective measures are considered as incorporating users subjective knowledge (such as *belief*) into the assessment.
In the past years, unexpectedness measure has been widely studied in various approaches to pattern and rule discoveries. Liu and Hsu studied the unexpected structures of discovered rules in [24]. In the proposed approach, the existing rules (denoted as E) from prior knowledge are regarded as fuzzy rules by using fuzzy set theory and the newly discovered rules (denoted as B) are matched against the existing fuzzy rules in the post-analysis process. A rule consists of the *condition* and the *consequent*, so that given two rules B_i and E_j, if the conditional parts of B_i and E_j are similar, but the consequents of the two rules are quite different, then it is considered as *unexpected consequent*; the inverse is considered as *unexpected condition*. The computation of the similarity in the matching is based on the attribute name and value. The same techniques are extended to find unexpected patterns in [25]. Moreover, in [26], Liu et al. investigated the problem of finding unexpected information in the context of Web content mining.
Suzuki et al. systematically studied *exception rules* in the context of association rule mining [37, 36, 38]. An association rule can be classified into two categories: a *common sense rule*, which is a description of a regularity for numerous objects, and an *exception rule*, which represents, for a relatively small number of objects, a different, regularity from a common sense rule. The exception rules are considered with respect to the common sense rules within a rule triplet

$$(A_\mu \Rightarrow c, A_\mu \wedge B_\nu \Rightarrow c', B_\nu \not\Rightarrow c'),$$

where A_μ, B_ν are itemsets and c, c' are items. Such a rule triplet can be interpreted as "if A_μ then c, however if A_μ and B_ν then c', and if B_ν then not c'".
Padmanabhan and Tuzhilin proposed a semantics-based belief-driven approach [29, 30, 31] to discover unexpected patterns in the context of association rules, where a rule $A \Rightarrow B$ is *unexpected* with respect to a belief $X \Rightarrow Y$ in a given database $\mathcal{D}$ if: (1) $B \wedge Y \models FALSE$, which means that the two patterns B and Y logically contradict each other (i.e., $\nexists R$ in $\mathcal{D}$ such that $B \cup Y \subseteq R$); (2) $A \wedge X$ holds on a statistically large subset of tuples in $\mathcal{D}$ (e.g., with respect to a given minimum support, the pattern $A \cup X$ is frequent in the database $\mathcal{D}$); (3) the rule $A \wedge X \Rightarrow B$ holds and the rule $A \wedge X \Rightarrow Y$ hoes not hold (e.g., the support and confidence of $A \wedge X \Rightarrow B$ satisfy given minimum support and minimum confidence but those of $A \wedge X \Rightarrow Y$ do not). An example can be that given a belief `professional` $\Rightarrow$ `weekend` (professionals shopped on weekends), if the rule (`professional, December`) $\Rightarrow$ `weekday` (professionals shopped on weekdays in December) holds but the rule (`professional, December`) $\Rightarrow$ `weekend` (professionals shopped on weekends in December) does not, then the rule `December` $\Rightarrow$ `weekday` is unexpected relative to the belief `professional` $\Rightarrow$ `weekend`. Notice that in this approach, the logically contradiction between patterns is defined by domain experts.
In [34], Spiliopoulou proposed an approach for mining unexpectedness with sequence rules transformed from frequent sequences. The sequence rule is built by dividing a sequence into two adjacent parts, which are determined by the support, confidence and improvement from association rule mining. A belief on sequences is constrained by the frequency of the two parts of a rule, so that if a sequence respects a sequence rule but the frequency constraints are broken, then this sequence is unexpected. Although this work considers the unexpected sequences and rules, it is however very different from our problem in the measure and the notion of unexpectedness contained in data.
In [39], Wang et al. studied unexpected association rules with respect to the value of attributes. In [20], Jaroszewicz and Scheffer proposed a Bayesian network based approach to discover unexpected patterns, that is, to find the patterns with the strongest discrepancies between the network and the database. Therefore, this approach can be regarded as frequency based, where unexpectedness is defined from whether itemsets in the database are much more, or much

less frequent than the background knowledge suggests.
In our recent work [23], we proposed a belief-driven approach for recognizing fuzzy unexpected sequences corresponding to sequential implication rules. A *sequential implication rule* is a rule of the form "if the sequence s_α occurs then the sequence s_β occurs latter" so that the beliefs are created with respect to (1) the distance between s_α and s_β; (2) the semantics of the implication between s_α and s_β, i.e., s_β cannot be replaced by another sequence s_γ. The fuzzy sets are considered on the distance between the two sequences.

III. Unexpected Recurrence Behaviors

In this section, we first introduce the data model and formalize the fuzzy recurrence rules, and then we present a belief system based on such fuzzy recurrence rules, with which the unexpected recurrence behaviors are therefore proposed.

A. Data Model

We consider the sequence data that consist in binary-valued attributes. Given a set of a limited number of attributes $R = \{i_1, i_2, \ldots, i_n\}$, each attribute is an *item*. An *itemset* is an unordered collection of items, denoted as $I = \{i_1, i_2, \ldots, i_m\}$, where $i_j \in R$ is an item. We have that $I \subseteq R$. A *sequence* is an ordered list of itemsets, denoted as $s = I_1 I_2 \cdots I_k$, where $I_j \subseteq R$ is an itemset. A *sequence database* is usually a large set of sequences, denoted as D.
Given two sequences $s = I_1 I_2 \ldots I_m$ and $s' = I'_1 I'_2 \ldots I'_n$, if there exist integers $1 \leq i_1 < i_2 < \ldots < i_m \leq n$ such that $I_1 \subseteq I'_{i_1}, I_2 \subseteq I'_{i_2}, \ldots, I_m \subseteq I'_{i_m}$, then the sequence s is a *subsequence* of the sequence s', denoted as $s \sqsubseteq s'$. If $s \sqsubseteq s'$, we say that s *is contained in* s', or s' *supports* s. For example, the sequence $s_1 = (a)(b)$ is contained in the sequence $s_2 = (a)(b)(c)$, but not contained in the sequence $s_3 = (ab)(c)$. In addition, we denote the *concatenation* of n sequences as $s_1 s_2 \cdots s_n$. For example, let $s_1 = (a)(b)$ and $s_2 = (c)(d)$, then we have $s_1 s_1 = (a)(b)(a)(b)$ and $s_1 s_2 = (a)(b)(c)(d)$.
Given a sequence database $\mathcal{D}$, the *support* of a sequence s is the fraction of the total number of sequences in $\mathcal{D}$ that support s, denoted as $supp(s, \mathcal{D})$. Given a user specified threshold of support called *minimum support*, denoted as $supp_{min}$, a sequence s is *frequent* if $supp(s, \mathcal{D}) \geq supp_{min}$.

B. Fuzzy Recurrence Rules

To study the repeatedly occurred elements in sequences, we first propose the notion of *recurrence sequence* in the form $\langle s, \psi \rangle$, where s is a sequence and ψ is a positive integer. If a sequence s' *supports* a recurrence sequence $\langle s, \psi \rangle$, then the sequence s occurs in s' at least ψ times, denoted as $\langle s, \psi \rangle \sqsubseteq s'$, that is,

$$(\langle s, \psi \rangle \sqsubseteq s') \iff (\underbrace{s \cdots s}_{n} \sqsubseteq s') \wedge (n \geq \psi).$$

A recurrence sequence $\langle s, \psi \rangle$ is also called a ψ-*recurrence sequence*. We use the wildcard "$*$" for denoting the general meaning of the support between sequences, that is,

$$(\langle s, * \rangle \sqsubseteq s') \equiv (s \sqsubseteq s').$$

In the remainder of this paper, we use the term *sequence* to describe the notion of *recurrence sequence*.
A *recurrence rule* is a rule on sequences with form $\langle s_\alpha, \psi \rangle \rightarrow \langle s_\beta, \theta \rangle$, where s_α, s_β are two sequences, and ψ, θ are two integers for describing recurrence behaviors in sequence data. A recurrence rule indicates the association relation that given a sequence s, if s_α orderly occurs no less than ψ times within s, then orderly s_β occurs in s no less than θ times, that is,

$$(\underbrace{s_\alpha \cdots s_\alpha}_{n} \sqsubseteq s) \wedge (n \geq \psi) \Rightarrow (\underbrace{s_\beta \cdots s_\beta}_{k} \sqsubseteq s) \wedge (k \geq \theta).$$

Given a sequence s and a recurrence rule $r = \langle s_\alpha, \psi \rangle \rightarrow \langle s_\beta, \theta \rangle$, if $\langle s_\alpha, \psi \rangle \sqsubseteq s$ and $\langle s_\beta, \theta \rangle \sqsubseteq s$, then we say that s *supports* r, denoted as $s \models r$. For instance, the recurrence rule $r = \langle (a)(b), 3 \rangle \rightarrow \langle (c)(d), * \rangle$ depicts that given a sequence s, if $(a)(b)$ is contained repeatedly in s no less 3 times, then $(c)(d)$ is contained in s; in other words, if $(a)(b)(a)(b)(a)(b) \sqsubseteq s$, then $(c)(d) \sqsubseteq s$.
Notice that the occurrences of s_α must be ordered, that is, for example, given a rule $r_1 = \langle (a)(b), 2 \rangle \rightarrow \langle (c), * \rangle$, the sequence $s_1 = \langle (a)(a)(c)(b)(b) \rangle$ does not support r_1, but the sequence $s_2 = \langle (a)(b)(c)(a)(b) \rangle$ supports r_1; however, the sequence s_1 supports the rules $r_2 = \langle (a), 2 \rangle \rightarrow \langle (c), * \rangle$ and $r_3 = \langle (b), 2 \rangle \rightarrow \langle (c), * \rangle$.
Considering the integer ψ, a human-friendly interpretation is more flexible and more relevant to described the recurrence in sequence data. For instance, in market basket analysis, to point out that "the customers who often purchase `action movie DVDs` often purchase `pop music CDs`" is more relevant than the conclusion "the customers who purchase at least 7 times of `action movie DVDs` purchase at least 5 times of `pop music CDs`".
We therefore extend the recurrence rule with fuzzy sets, so called the *fuzzy recurrence rule*, in the form $\langle s_\alpha, \zeta_\alpha \rangle \rightarrow \langle s_\beta, \zeta_\beta \rangle$, where ζ_α and ζ_β are two fuzzy sets for describing s_α and s_β, and the sequences $\langle s_\alpha, \zeta_\alpha \rangle$ and $\langle s_\beta, \zeta_\beta \rangle$ are two fuzzy recurrence sequences. Given a sequence s' and a fuzzy recurrence rule $\langle s, \zeta \rangle$, that s' *supports* $\langle s, \zeta \rangle$ is defined as

$$(\langle s, \zeta \rangle \sqsubseteq s') \iff (\underbrace{s \cdots s}_{n} \sqsubseteq s) \wedge (\mu_\zeta(n) \geq recu_{min}), \tag{1}$$

where the fuzzy degree measured by the membership function $\mu_\zeta(n)$ must be superior or equal to a threshold $recu_{min}$. Let us consider the following example.

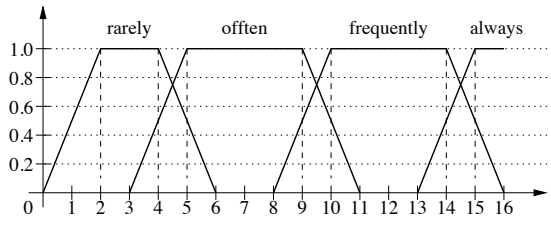


Figure. 1: Fuzzy sets for describing recurrence rules.

Example 1. *Given a set of distinct events $a, b, c, d, \ldots$, an ordered of events can be represented as the data model of sequence. Assuming that given an event sequence s, if s supports the recurrence sequence $\langle (a)(b), 4 \rangle$, then s supports the subsequence $(c)(d)$; if s supports the recurrence*

sequence $\langle(a)(b),9\rangle$*, then* s *supports* (c)*. These behaviors can be described by recurrence rules, such as the rule* $r_1 = \langle(a)(b),4\rangle \rightarrow \langle(c)(d),*\rangle$ *and the rule* $r_2 = \langle(a)(b),9\rangle \rightarrow \langle(c),*\rangle$*. Given a sequence* s_1 *such that* $\langle(a)(b),3\rangle \sqsubseteq s_1$ *and* $\langle(c)(d)\rangle \sqsubseteq s_1$*, a sequence* s_2 *such that* $\langle(a)(b),8\rangle \sqsubseteq s_2$ *and* $\langle(c)\rangle \sqsubseteq s_2$*, we have* $s_1 \not\models r_1$ *and* $s_2 \not\models r_2$*. However, since the recurrence sequences contained in these sequences and rules are close, the sequences* s_1 *and* s_2 *can be still potentially interesting. On the other hand, considering the fuzzy recurrence rules* $r_1' = \langle(a)(b),rarely\rangle \rightarrow \langle(c)(d),*\rangle$ *and* $r_2' = \langle(a)(b),often\rangle \rightarrow \langle(c),*\rangle$*, corresponding to the rules* r_1 *and* r_2 *with respect to the fuzzy partitions shown in Figure 1, let the threshold* $recu_{min} = 0.5$*, then we have* $s_1 \models r_1'$ *and* $s_2 \models r_2'$*. We can further define more partitions, such as "always" or "rarely".* □

In this paper, the fuzzy recurrence rules are considered as having been predefined by domain experts, the discovery of fuzzy recurrence rules will be covered in our future research work.

C. Belief System

We now present the *belief system* on fuzzy recurrence rules with integrating semantic contradiction between sequences.

A *belief* specifies that if a sequence $\langle s_\alpha, \zeta_\alpha\rangle$ occurs, then a sequence $\langle s_\beta, \zeta_\beta\rangle$ occurs; however a sequence $\langle s_\gamma, \zeta_\gamma\rangle$ should not occur at the occurrence position of the sequence $\langle s_\beta, \zeta_\beta\rangle$.

Definition 1 (Semantic contradiction)**.** *Given two sequences* $\langle s_\beta, \zeta_\beta\rangle$ *and* $\langle s_\gamma, \zeta_\gamma\rangle$*, the semantic contradiction between* $\langle s_\beta, \zeta_\beta\rangle$ *and* $\langle s_\gamma, \zeta_\gamma\rangle$ *is a boolean value determined by a predicate* $o(\langle s_\beta, \zeta_\beta\rangle, \langle s_\gamma, \zeta_\gamma\rangle)$*: if* $\langle s_\beta, \zeta_\beta\rangle$ *semantically contradicts* $\langle s_\gamma, \zeta_\gamma\rangle$*, then* $o(\langle s_\beta, \zeta_\beta\rangle, \langle s_\gamma, \zeta_\gamma\rangle)$ *returns 1; otherwise* $o(\langle s_\beta, \zeta_\beta\rangle, \langle s_\gamma, \zeta_\gamma\rangle)$ *returns 0.*

Given two sequences $\langle s_\beta, \zeta_\beta\rangle$ and $\langle s_\gamma, \zeta_\gamma\rangle$, denote by $\langle s_\beta, \zeta_\beta\rangle \not\approx_{sem} \langle s_\gamma, \zeta_\gamma\rangle$ when $o(\langle s_\beta, \zeta_\beta\rangle, \langle s_\gamma, \zeta_\gamma\rangle) = 1$. The semantic contradiction is symmetric but not transitive. We have that $\langle s_\beta, \zeta_\beta\rangle \not\approx_{sem} \langle s_\gamma, \zeta_\gamma\rangle$ is equivalent to $\langle s_\gamma, \zeta_\gamma\rangle \not\approx_{sem} \langle s_\beta, \zeta_\beta\rangle$, however $\langle s_\beta, \zeta_\beta\rangle \not\approx_{sem} \langle s_\gamma, \zeta_\gamma\rangle$ and $\langle s_\gamma, \zeta_\gamma\rangle \not\approx_{sem} \langle s_\alpha, \zeta_\alpha\rangle$ do not imply that $\langle s_\beta, \zeta_\beta\rangle \not\approx_{sem} \langle s_\alpha, \zeta_\alpha\rangle$.

The predicate $o(\langle s_\beta, \zeta_\beta\rangle, \langle s_\gamma, \zeta_\gamma\rangle)$ can be designed to compute the semantic contradiction between the elements $\langle s_\beta, \zeta_\beta\rangle$ and $\langle s_\gamma, \zeta_\gamma\rangle$ in various manners. For instance, given a set $\mathcal{S}$ of sequences, we can build a projection table T of predefined relations on $\mathcal{S} \times \mathcal{S}$, and then the semantic contradiction between any $(\langle s_\beta, \zeta_\beta\rangle, \langle s_\gamma, \zeta_\gamma\rangle) \in \mathcal{S}$ can be returned by $o(\langle s_\beta, \zeta_\beta\rangle, \langle s_\gamma, \zeta_\gamma\rangle)$ with searching the table T; the semantic contradiction can also be determined by the fuzzy sets of the recurrence, i.e., if ζ_β semantically contradicts ζ_β' (e.g., *often* v.s. *rarely*), then $o(\langle s_\beta, \zeta_\beta\rangle, \langle s_\beta, \zeta_\beta'\rangle) = 1$.

Let $\langle s_\alpha, \zeta_\alpha\rangle \rightarrow \langle s_\beta, \zeta_\beta\rangle$ be a fuzzy recurrence rule and $\langle s_\beta, \zeta_\beta\rangle \not\approx_{sem} \langle s_\gamma, \zeta_\gamma\rangle$ be a semantic contradiction. The fuzzy recurrence rule implies an association relation between the sequences $\langle s_\alpha, \zeta_\alpha\rangle$ and $\langle s_\beta, \zeta_\beta\rangle$ that if the recurrence of s_α is ζ_α, then the recurrence of s_β is ζ_β. The semantic contradiction then implies that the recurrence sequences $\langle s_\beta, \zeta_\beta\rangle$ and $\langle s_\gamma, \zeta_\gamma\rangle$ semantically contradict each other.

Definition 2 (Belief)**.** *A belief is a conjunction* $\{\langle s_\alpha, \zeta_\alpha\rangle \rightarrow \langle s_\beta, \zeta_\beta\rangle\} \wedge \{\langle s_\beta, \zeta_\beta\rangle \not\approx_{sem} \langle s_\gamma, \zeta_\gamma\rangle\}$*, where* $\{\langle s_\alpha, \zeta_\alpha\rangle \rightarrow \langle s_\beta, \zeta_\beta\rangle\}$ *is a fuzzy recurrence rule and* $\langle s_\beta, \zeta_\beta\rangle \not\approx_{sem} \langle s_\gamma, \zeta_\gamma\rangle$ *is a semantic contradiction. A belief is denoted as* $[\langle s_\alpha, \zeta_\alpha\rangle ; \langle s_\beta, \zeta_\beta\rangle ; \langle s_\gamma, \zeta_\gamma\rangle]$.

A belief $[\langle s_\alpha, \zeta_\alpha\rangle ; \langle s_\beta, \zeta_\beta\rangle ; \langle s_\gamma, \zeta_\gamma\rangle]$ depicts that given a sequence s, if s supports $\langle s_\alpha, \zeta_\alpha\rangle$, then s supports $\langle s_\beta, \zeta_\beta\rangle$; however s should not support $\langle s_\gamma, \zeta_\gamma\rangle$, that is,

$$(\langle s_\alpha, \zeta_\alpha\rangle \sqsubseteq s) \wedge (\langle s_\beta, \zeta_\beta\rangle \sqsubseteq s) \wedge (\langle s_\gamma, \zeta_\gamma\rangle \not\sqsubseteq s). \quad (2)$$

Example 2. *Assume that the customers who purchase* `movies` *like to play* `games`*. If we consider that* `games` *and* `books` *semantically contradict each other, where the semantic contradiction can be* $\langle(\texttt{game}),often\rangle \not\approx_{sem} \langle(\texttt{book}),often\rangle$*, then a belief can be defined as*

$$[\langle(\texttt{movie}),often\rangle ; \langle(\texttt{game}),often\rangle ; \langle(\texttt{book}),often\rangle] .$$

The fuzzy sets for purchases can also be that shown in Figure 1. The above belief describes that the customers who $often$ *purchase* `movies` *also purchase* `games` $often$*, however do not* $often$ *purchase* `books`*.* □

Given a belief b, if a sequence s satisfies Equation (2), then we say that the sequence s *supports* the belief b, denoted as $s \models b$. A sequence s *unexpected* to a belief b is denoted as $s \not\models b$.

D. Unexpected Sequences

We are considering to discover the sequences contained in a database those semantically contradict a given set of fuzzy recurrence rules. In order to find such sequences, we construct a belief system from given fuzzy recurrence rules with semantic contradictions between fuzzy recurrence sequences, so that each sequence not respecting the belief base is unexpected.

A sequence s is unexpected if (1) the sequence $\langle s_\alpha, \zeta_\alpha\rangle$ occurs and the sequence s_β but the sequence $\langle s_\beta, \zeta_\beta\rangle$ does not occur; or (2) the sequence $\langle s_\alpha, \zeta_\alpha\rangle$ and the sequence $\langle s_\gamma, \zeta_\gamma\rangle$ occurs. Therefore, we consider two forms of unexpectedness in our approach with respect to the occurrence of the sequences $\langle s_\beta, \zeta_\beta\rangle$ and $\langle s_\gamma, \zeta_\gamma\rangle$ contained in a belief.

Definition 3 (Occurrence-unexpectedness)**.** *Given a sequence* s *and a belief* $b = [\langle s_\alpha, \zeta_\alpha\rangle ; \langle s_\beta, \zeta_\beta\rangle ; \langle s_\gamma, \zeta_\gamma\rangle]$*, if* s *supports* $\langle s_\alpha, \zeta_\alpha\rangle$ *and there exist* $s_\beta \sqsubseteq s$ *and* $\langle s_\beta, \zeta_\beta\rangle \not\sqsubseteq s$*, then the sequence* s *is occurrence-unexpected, denoted as* $s \not\models_\beta b$.

The primary factor of the occurrence-unexpectedness in a sequence s is that the recurrence sequence $\langle s_\beta, \zeta_\beta\rangle$ does not occur as expected however at least the sequence s_β occurs in s, so that we also called this form of unexpectedness as β-unexpectedness.

For instance, considering the belief in Example 2, noted as b, let s be a customer transaction sequence, if we have that $\langle(\texttt{movie}),often\rangle \sqsubseteq s$ and $\langle(\texttt{game}),often\rangle \sqsubseteq s$, then s is expected with respect to the fuzzy recurrence rule $\langle(\texttt{movie}),often\rangle \rightarrow \langle(game),often\rangle$; however, if we have $\langle(\texttt{game})\rangle \sqsubseteq s$ but not $\langle(\texttt{game}),often\rangle \sqsubseteq s$, for example, the case $\langle(\texttt{game}),rarely\rangle \sqsubseteq s$, since $\langle(\texttt{game}),rarely\rangle \sqsubseteq s$ implies that $\langle(\texttt{game})\rangle \sqsubseteq s$, then s is a β-unexpected sequence, i.e., $s \not\models_\beta b$.

Definition 4 (Semantics-unexpectedness)**.** *Given a sequence* s *and a belief* $b = [\langle s_\alpha, \zeta_\alpha\rangle ; \langle s_\beta, \zeta_\beta\rangle ; \langle s_\gamma, \zeta_\gamma\rangle]$*, if* s *supports* $\langle s_\alpha, \zeta_\alpha\rangle$ *and there exists* $\langle s_\gamma, \zeta_\gamma\rangle \sqsubseteq s$*, then the sequence* s *is semantics-unexpected, denoted as* $s \not\models_\gamma b$.

Respectively, the primary factor of the semantics-unexpectedness in a sequence s is that the semantic contradiction $\langle s_\beta, \zeta_\beta\rangle \not\approx_{sem} \langle s_\gamma, \zeta_\gamma\rangle$ is broken because the recurrence sequence $\langle s_\gamma, \zeta_\gamma\rangle$ occurs in s, so that we also called this form of unexpectedness as γ-unexpectedness.

Considering again the belief b in Example 2, let s be a customer transaction sequence, if we have that $\langle(\texttt{movie}), often\rangle \sqsubseteq s$ and $\langle(\texttt{book}), often\rangle \not\sqsubseteq s$, then the sequence s is not unexpected with respect to the semantic contradiction $\langle(\texttt{game}), often\rangle \not\approx_{sem} \langle(\texttt{book}), often\rangle$; however, if we have $\langle(\texttt{book}), often\rangle \sqsubseteq s$, then s is a γ-unexpected sequence, i.e., $s \not\models_\gamma b$. Of course, it is not necessary to forbid $(\texttt{book}) \sqsubseteq s$, for example, according to this belief, the occurrence of $\langle(\texttt{book}), rarely\rangle$ does not imply the γ-unexpectedness.

Now we discuss the coherence in a belief system. The coherence in a belief system consists of fuzzy recurrence rules and semantic contradictions on fuzzy recurrence sequences must be considered in sequence inclusions and covers of the fuzzy sets on recurrence. Let $\mathcal{B}$ be a set of beliefs, for any two beliefs $(b, b') \in \mathcal{B}$, where $b = [\langle s_\alpha, \zeta_\alpha\rangle; \langle s_\beta, \zeta_\beta\rangle; \langle s_\gamma, \zeta_\gamma\rangle]$ and $b = [\langle s_\alpha{}', \zeta_\alpha{}'\rangle; \langle s_\beta{}', \zeta_\beta{}'\rangle; \langle s_\gamma{}', \zeta_\gamma{}'\rangle]$, the following condition must be satisfied if the belief b is coherent:

$$(s_\beta \not\sqsubseteq s_\gamma{}') \vee (\zeta_\beta \neq \zeta_\gamma{}')$$

For example, let us consider two fuzzy recurrence rules r_1 and r_2. Let $r_1 = \langle(a), often\rangle \rightarrow \langle(c)(d), often\rangle$ and $r_2 = \langle(a), often\rangle \rightarrow \langle(e), often\rangle$ where $\langle(c)(d), often\rangle \not\approx_{sem} \langle(e)(f), often\rangle$ and $\langle(e), often\rangle \not\approx_{sem} \langle(c), often\rangle$. Then r_1 and r_2 are in conflict because $\langle(e)(f), often\rangle$ implies that $\langle(e), often\rangle$.

Given a belief $b = [\langle s_\alpha, \zeta_\alpha\rangle; \langle s_\beta, \zeta_\beta\rangle; \langle s_\gamma, \zeta_\gamma\rangle]$, an constraint $\tau = [min..max]$ on the occurrence range of the sequences $\langle s_\beta, \zeta_\beta\rangle$ or $\langle s_\gamma, \zeta_\gamma\rangle$ can be further applied, which indicates that the we only take account of the occurrence of $\langle s_\beta, \zeta_\beta\rangle$ or $\langle s_\gamma, \zeta_\gamma\rangle$ within the range $[min..max]$ after the occurrence of $\langle s_\alpha, \zeta_\alpha\rangle$. With the constraint τ, a belief can be written as $[\langle s_\alpha, \zeta_\alpha\rangle; \langle s_\beta, \zeta_\beta\rangle; \langle s_\gamma, \zeta_\gamma\rangle; min..max]$. In this case, we note $\tau = *$ if the occurrence range is not specified.

Example 3. *Let us consider the problem of "expiration" in the instance addressed in Example 2. If we concentrate on the short term customer behaviors, e.g., within 5 to 30 days, the belief proposed in Example 2 can be written with an occurrence constraint* $\tau = [5..30]$

$$[\langle(\texttt{movie}), often\rangle; \langle(\texttt{game}), often\rangle; \langle(\texttt{book}), often\rangle; 5..30],$$

if we count the days without purchase as an empty itemset in customer purchase sequences. □

Given a sequence database $\mathcal{D}$ and a belief base $\mathcal{B}$, the problem of discovering unexpected fuzzy recurrence sequences is therefore to find all sequences $s \in \mathcal{D}$ that contain β-unexpectedness and/or γ-unexpectedness with respect to each belief $b \in \mathcal{B}$ that consist of recurrence rules and semantic contradictions on recurrence sequences.

IV. Approach UFR

In this section we develop the approach UFR, which stands for mining Unexpected Fuzzy Recurrence behaviors.

A. Belief Tree Representation

In this section, we propose a tree representation of a *belief system* consisting of a set of beliefs.

Before constructing the tree representation, we first propose the notions of *premise sequence*, *conclusion sequence set*, and *contradiction set* of a belief system. Given a belief $b = [\langle s_\alpha, \zeta_\alpha\rangle; \langle s_\beta, \zeta_\beta\rangle; \langle s_\gamma, \zeta_\gamma\rangle]$, we call the sequence s_α the *premise sequence* and the sequence s_β the *conclusion sequence*. A belief system can be regrouped by each distinct premise sequence, and each group with the same premise sequence can be regrouped by each distinct conclusion sequence.

Definition 5 (Conclusion sequence set)**.** *Given a belief system $\mathcal{B}$, the conclusion sequence set with respect to a premise sequence $\langle s_\alpha, \zeta_\alpha\rangle$, denoted as $\Delta\langle s_\alpha, \zeta_\alpha\rangle$, is the set of the conclusion sequences all beliefs having the premise sequence $\langle s_\alpha, \zeta_\alpha\rangle$.*

Definition 6 (Contradiction sequence set)**.** *Given a belief system $\mathcal{B}$ and a premise sequence $\langle s_\alpha, \zeta_\alpha\rangle$, let $\langle s_\beta, \zeta_\beta\rangle \in \Delta\langle s_\alpha, \zeta_\alpha\rangle$ be a conclusion sequence. The contradiction sequence set with respect to the sequence $\langle s_\alpha, \zeta_\alpha\rangle$ and $\langle s_\beta, \zeta_\beta\rangle$, denoted as $\Theta\langle s_\alpha, \zeta_\alpha\rangle \,|\, \langle s_\beta, \zeta_\beta\rangle$, is the set of sequences such that for each sequence $\langle s_\gamma, \zeta_\gamma\rangle \in \Theta\langle s_\alpha, \zeta_\alpha\rangle \,|\, \langle s_\beta, \zeta_\beta\rangle$, we have that $\langle s_\beta, \zeta_\beta\rangle \not\approx_{sem} \langle s_\gamma, \zeta_\gamma\rangle$.*

Now we define the data structure of the belief tree representation. A *belief tree*, denoted as T, is a tree representation of a belief. According to the notions defined in above, a belief tree is a tree structure defined as below[1].

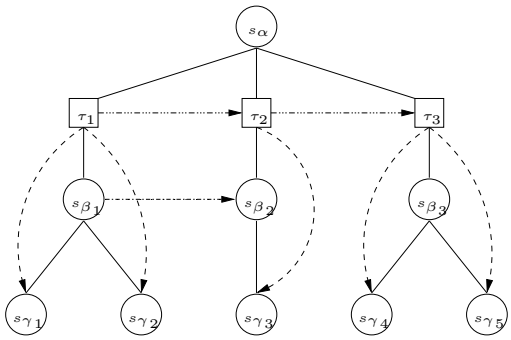


Figure. 2: A belief tree example.

1. A belief tree T corresponding to a set of beliefs b consists of one root node s_α-node for a premise sequence $\langle s_\alpha, \zeta_\alpha\rangle$, a set of τ-nodes as the sub-nodes of the root, and a set of sequence subtrees consisting of s-nodes.

2. The τ-node has two field: min and max corresponding to the occurrence range $[min..max]$. If the occurrence range is not specified, we let $min = -1$.

3. A s-node contains a recurrence sequence. In our implementation, a s-node is a reference (e.g., a *pointer* in C/C++, or originally a *reference* in JAVA) to a sequence stored external to the tree structure.

4. Each τ-node possesses a sequence subtree. The sub-root node of a sequence subtree corresponds to a con-

[1] To respect the space allowed in the figure, a notation like s_α denotes a recurrence sequence $\langle s_\alpha, \zeta_\alpha\rangle$, etc.

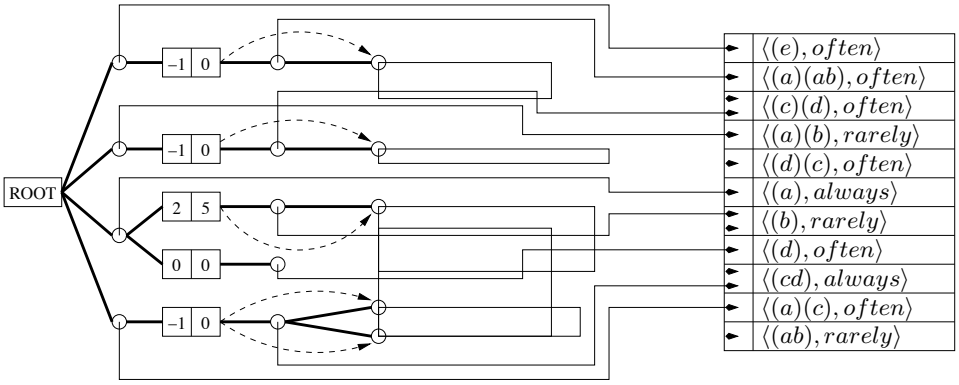


Figure. 3: An example tree presentation of a belief base.

clusion sequence $\langle s_\beta, \zeta_\beta \rangle \in \Delta \langle s_\alpha, \zeta_\alpha \rangle$ and the subnodes correspond to the set of sequences $\langle s_\gamma, \zeta_\gamma \rangle \in \Theta \langle s_\alpha, \zeta_\alpha \rangle \,|\, \langle s_\beta, \zeta_\beta \rangle$. Each τ-node is linked by appending order for optimizing the performance of traversal.

5. A τ-link connects a τ-node and each s-node corresponding to each sequence $\langle s_\gamma, \zeta_\gamma \rangle \in \Theta \langle s_\alpha, \zeta_\alpha \rangle \,|\, \langle s_\beta, \zeta_\beta \rangle$.

6. A s-link connects all $(\langle s_\beta, \zeta_\beta \rangle, \langle s_\beta{}', \zeta_\beta{}' \rangle) \in \Delta \langle s_\alpha, \zeta_\alpha \rangle$ such that $\langle s_\beta, \zeta_\beta \rangle = \langle s_\beta{}', \zeta_\beta{}' \rangle$, with respect to the appending order. For instance, in Figure 2, $\langle s_{\beta_1}, \zeta_{\beta_1} \rangle = \langle s_{\beta_2}, \zeta_{\beta_2} \rangle$.

Example 4 shows a tree representation of a belief base with 6 different beliefs, which are shown in Figure 3.

Example 4. *Given a belief base containing the following 6 beliefs:*

$$b_1 = [\langle (e), often \rangle ; \langle (a)(ab), often \rangle ; \langle (c)(d), often \rangle ; *] ;$$
$$b_2 = [\langle (a)(b), rarely \rangle ; \langle (c)(d), often \rangle ; \langle (d)(c), often \rangle ; *] ;$$
$$b_3 = [\langle (a), always \rangle ; \langle (b)(c), rarely \rangle ; \langle (cd), always \rangle ; 2..5] ;$$
$$b_4 = [\langle (a), always \rangle ; \langle (d), often \rangle ; \langle (cd), always \rangle ; 0..0] ;$$
$$b_5 = [\langle (a)(c), often \rangle ; \langle (cd), always \rangle ; \langle (ab), rarely \rangle ; *] ;$$
$$b_6 = [\langle (a)(c), often \rangle ; \langle (cd), always \rangle ; \langle (b), rarely \rangle ; *] .$$

The corresponded belief base tree is shown in Figure 3. □

B. Algorithms

First, we have the following belief tree construction algorithm `BeliefTree` (Algorithm 1). Given an input belief set B with all beliefs b having the same premise sequence $\langle s_\alpha, \zeta_\alpha \rangle$, the algorithm first creates a belief tree T with the root node $\langle s_\alpha, \zeta_\alpha \rangle$. For each conclusion sequence $\langle s_\beta, \zeta_\beta \rangle \in \Delta \langle s_\alpha, \zeta_\alpha \rangle$, the algorithm appends the occurrence constraint τ as a τ-node to the root node and appends the conclusion sequence s_β as a s-node to the newly appended τ-node. Then, for each contradiction sequence $\langle s_\gamma, \zeta_\gamma \rangle \in \Theta \langle s_\alpha, \zeta_\alpha \rangle \,|\, \langle s_\beta, \zeta_\beta \rangle$, the algorithm finds the location of the s-node of $\langle s_\beta, \zeta_\beta \rangle$ in the tree and appends $\langle s_\gamma, \zeta_\gamma \rangle$ as a s-node to $\langle s_\beta, \zeta_\beta \rangle$. Finally, the algorithm outputs the belief tree T for a belief group where all beliefs have the same premise sequence.

Algorithm 1: `BeliefTree (`b`)` : Belief tree construction.

Input : A set B of beliefs having the same premise sequence.
Output : A belief tree T.

$T := BeliefTree.Create(\langle s_\alpha, \zeta_\alpha \rangle)$;
foreach $b \in B$ **do**
  $n_\tau := T.appendTauNode(r.\tau)$; /* do not create new τ-node if the same τ exists */
  $n_s := T.appendSeqNode(n, \langle s_\beta, \zeta_\beta \rangle)$;
  $n'_s := T.getLastSeqNode(n_s)$; /* find last s-node having the same sequence with n_s */
  $T.linkSeqNode(n'_s, n_s)$;
foreach $b \in B$ **do**
  $n_s := T.getSeqNode(\langle s_\beta, \zeta_\beta \rangle)$;
  $n'_s := T.appendSeqNode(n_s, \langle s_\gamma, \zeta_\gamma \rangle)$;
  $T.linkTauNode(n_s.parent, n'_s)$;
return T;

Algorithm 2: `SeqMatchUfr (`$\langle s, \zeta \rangle$`,` s'`,` $range$`)` : Matching fuzzy recurrence sequence.

Input : A fuzzy recurrence sequence $\langle s, \zeta \rangle$, a sequence s', and a pair $range$.
Output : The occurrence of $\langle s, \zeta \rangle$ in s' with respect to $range$.

$\mu_\zeta := FuzzyMembershipFunction(\zeta)$;
$pos := pair(0, 0)$;
$ran := range$;
$rec := 0$;
$ret := pair(-1, -1)$;
while $pos.first \neq -1$ **do**
  $pos :=$ `SeqMatchFirst`(s, s', ran);
  if $pos.first = -1$ **then**
    break;
  $ran.first := pos.second + 1$;
  $rec := rec + 1$;
  if $ret.first = -1$ **then**
    $ret.first := pos.first$;
  $ret.seconf := pos.second$;
if $\mu_\zeta(rec) \geq recu_{min}$ **then** /* $recu_{min}$ is globally accessible */
  return ret;
return $pair(-1 - 1)$;

The fuzzy recurrence sequence matching routine is therefore the core of the approach UFR, so that we develop the algorithm `SeqMatchUfr` (Algorithm 2), which finds the occurrence of a fuzzy recurrence sequence in a sequence. The algorithm accepts a fuzzy recurrence sequence $\langle s, \zeta \rangle$, a sequence s', and a pair $range$ for bounding the occurrence of $\langle s, \zeta \rangle$ in s' as inputs, and outputs the occurrence of $\langle s, \zeta \rangle$ in s', if s' supports $\langle s, \zeta \rangle$ with respect to Equation (2). The subroutine `SeqMatchFirst` finds the first occurrence of the sequence s in the sequence s'.

Base on the algorithm `SeqMatchUfr`, we develop the β-unexpected fuzzy recurrences as the routine `UfrMatchBeta`, listed in Algorithm 3.

Algorithm 3: `UfrMatchBeta` (T, s, pos) : Matching β-unexpected fuzzy recurrences.

Input : A belief T, a sequence s, and a pair pos indicating the occurrence of the premise sequence s_α contained in the s_α-node of T in s.
Output : The set of all β-unexpected fuzzy recurrences in s with respect to T.

```
1  uxps := TupleSet.Create();
2  n_τ := T.firstTauNode();
3  while n_τ ≠ null and n_τ ∉ N do
4      if n_τ.data.min ≠ −1 then
5          continue; /* recurrence rule is in sequence association
           rule form */
6      n_{s_β} := n_τ.firstSubNode();
7      while n_{s_β} ≠ null do
8          u := SeqMatchFirst(n_{s_β}.data, s, pair(pos.second +
           1, |s| − 1));
9          if u.first ≠ −1 then
10             u := SeqMatchUfr(⟨n_{s_β}.data, n_{s_β}.ζ⟩, s, pair(pos.second+
               1, |s| − 1));
11             if u ≠ −1 then
12                 uxps.add(tuple(s.id, u.first, u.second));
13                 if options | FIRST_UXPS_ONLY then /* use the
                   conclusion of Lemma ?? */
14                     return uxps;
15     n_τ := T.nextTauNode(n_τ);
16 return uxps;
```

The algorithm accepts a belief group T, a sequence s, and a pair pos indicating the occurrence of the premise sequence $\langle s_\alpha, \zeta_\alpha \rangle$ contained in the s_α-node of T in the sequence s as inputs, and outputs all or the first β-unexpected fuzzy recurrence(s) in s. The argument pos is specified with respect to the constraint on occurrence range.

For each conclusion sequence $\langle s_\beta, \zeta_\beta \rangle$ contained in the belief of fuzzy recurrence rules, the algorithm verifies whether s_β is contained in s by the subroutine `SeqMatchFirst`. If $s_\beta \sqsubseteq s$, the subroutine `SeqMatchUfr` matches whether $\langle s_\beta, \zeta_\beta \rangle \not\sqsubseteq s$. Thus, finally algorithm returns all β-unexpected fuzzy recurrences $\langle s_\beta, \zeta_\beta \rangle \not\sqsubseteq s$.

We illustrate in Figure 4 the matching of β-unexpected fuzzy recurrence in a given sequence s with respect to the fuzzy sets shown in Figure 1 and the belief

$$[\langle (a)(ab), often \rangle ; \langle (c)(d), rarely \rangle ; \langle (ef)(g), rarely \rangle ; *] ,$$

where $recu_{min} = 0.6$.

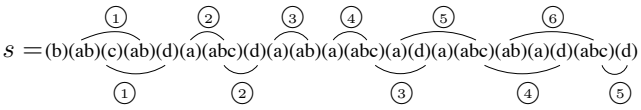


Figure. 4: Matching β-unexpected fuzzy recurrence.

We have that $\langle (a)(ab), often \rangle \sqsubseteq s$ by calling `SeqMatchUfr` before matching β-unexpected fuzzy recurrence (i.e., performed in the main routine of the framework MUSE, where `SeqMatch` is replaced by `SeqMatchUfr`), which is marked as ① to ⑥ above the sequence shown in Figure 4 and satisfies the minimum fuzzy membership degree $recu_{min} = 0.6$. Then, $\langle (c)(d), rarely \rangle \sqsubseteq s$ will be verified, where the recurrence of $\langle (c)(d) \rangle$ is marked as ① to ⑤ under the sequence shown in Figure 4. According to the fuzzy sets shown in Figure 1, we have that $\mu_\zeta(5) = 0.5$ for "rarely", so that we have that $\langle (c)(d), rarely \rangle \not\sqsubseteq s$ and the sequence s is β-unexpected.

With the illustration of matching β-unexpected fuzzy recurrence in a sequence, the matching of γ-unexpected fuzzy recurrences `UfrMatchGamma` is not difficult to understand, which is listed in Algorithm 4.

The algorithm accepts a belief group T, a sequence s, and a pair pos indicating the occurrence of the premise sequence s_α contained in the s_α-node of T in the sequence s as inputs, and outputs all or the first γ-unexpected fuzzy recurrence(s) in s.

Algorithm 4: `UfrMatchGamma` (T, s, pos) : Matching γ-unexpected fuzzy recurrences.

Input : A belief T, a sequence s, and a pair pos indicating the occurrence of the premise sequence s_α contained in the s_α-node of T in s.
Output : The set of all γ-unexpected fuzzy recurrences in s with respect to T.

```
1  uxps := TupleSet.Create();
2  n_τ := T.firstTauNode();
3  while n_τ ≠ null and n_τ ∉ N do
4      if n_τ.data.min ≠ −1 then
5          continue; /* recurrence rule is in sequence association
           rule form */
6      n_{s_γ} := n_τ.firstLinkedNode();
7      while n_{s_γ} ≠ null do
8          u := SeqMatchUfr(⟨n_{s_γ}.data, n_{s_γ}.ζ⟩, s, pair(pos.second +
           1, |s| − 1));
9          if u ≠ −1 then
10             uxps.add(tuple(s.id, u.first, u.second));
11             if options | FIRST_UXPS_ONLY then /* first
               occurrence of γ-unexpectedness */
12                 return uxps;
13     n_τ := T.nextTauNode(n_τ);
14 return uxps;
```

C. Experiments

The approach UFR is evaluated with Web access record data. Two types of Web access log are used in our experiments: one is a large access log file of an online forum site (labeled as `BBS`), and another is a large access log file of a mixed

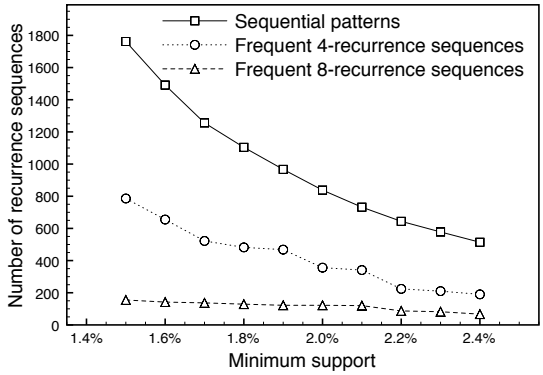
homepage hosting server (labeled as WWW).

Data Set	Size	Distinct Items	Average Length
BBS	135,562	126,383	15.5591
WWW	53,325	85,810	8.3507

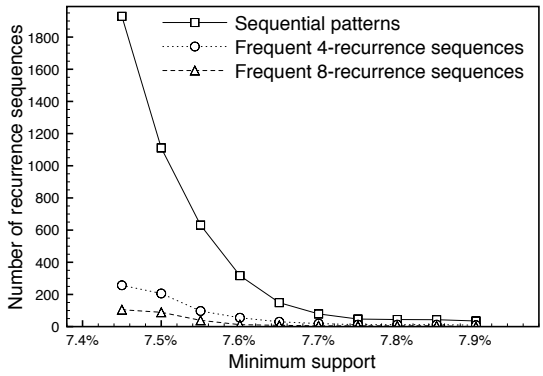
Table 1: Web access logs used for the evaluation.

The composition of the two data sets are listed in Table 1. We first apply a sequential pattern mining algorithm to discover frequent sequences for studying the general behaviors of the data sets. The frequent 4-recurrence sequences and 8-recurrence sequences are shown in Figure 5.

The recurrence sequences in the data sets show that the recurrence behaviors depend on the semantic characteristics of data, for instance, in our experimental data sets, the recurrence behaviors in online forum site are more stronger than those in mixed content Web site.



(a) Data set BBS.



(b) Data set WWW.

Figure. 5: Number of frequent recurrence sequences.

We generate 15 beliefs for each data set after examining the discovered sequential patterns, frequent 4-recurrence and 8-recurrence sequences, which correspond to 3 groups of 5 beliefs: with "rarely", "often" and "frequently", with respect to the fuzzy sets shown in Figure 1.

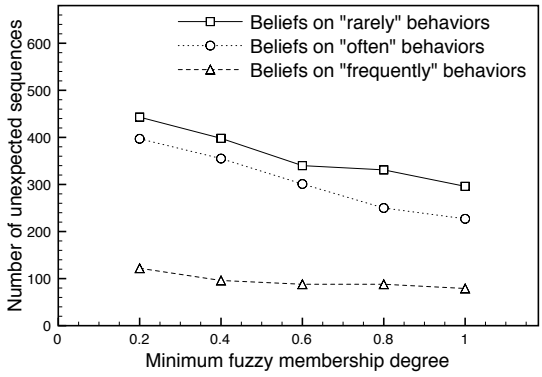
Table 2 lists several sample beliefs in our experiments. For instance, the belief

$$\texttt{BBS}_1 = [\langle(\texttt{f=4}), rarely\rangle ; \langle(\texttt{f=9}), rarely\rangle ; \langle(\texttt{f=9}), often\rangle ; *]$$

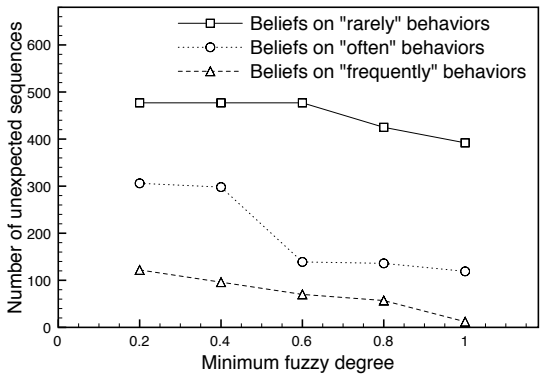
depicts that the forum users who rarely visit the forum No.4 also rarely visit the forum No.9, and that they often visit the forum No.9 is a contradiction; the belief

$$\texttt{WWW}_2 = [\langle(\texttt{/pub/}), often\rangle ; \langle(\texttt{/}), rarely\rangle ; \langle(\texttt{/doc/}), often\rangle ; *]$$

(for respecting the thesis layout, we trim the prefix /~li of the path) depicts that the homepage visitors who often access the publications located in /~li/pub/ rarely access the homepage /~li/, so that they should not often access the documents located in /~li/doc/.



(a) Data set BBS.



(b) Data set WWW.

Figure. 6: Number of sequences with unexpected fuzzy recurrences.

Figure 6 shows our experimental results. With the decrease of the minimum fuzzy degree threshold, the number of unexpected sequences increases. In Figure 6(a), we find that in the "frequently" fuzzy set, the number of unexpected sequences is much less than those in the other two fuzzy sets, because in the data set the number of long recurrence sequences, such as 8-recurrence sequences, is less. We can also find that the unexpected behaviors focus on the recurrences between "rarely" and "often". In Figure 6(b), there is a sharp increase of the number of unexpected sequences in the "often" fuzzy set when the minimum fuzzy membership degree decreases from 0.6 to 0.4, because in the "often" fuzzy set, the fuzzy degree 0.5 corresponds to 4-recurrence sequences, so that a lot of unexpected sequences in the "rarely" fuzzy set are counted as "often".

Belief	Premise $\langle s_\alpha, \zeta_\alpha \rangle$	Conclusion $\langle s_\beta, \zeta_\beta \rangle$	Contradiction $\langle s_\gamma, \zeta_\gamma \rangle$
$\mathtt{BBS}_1$	$(\mathtt{f{=}4}), rarely$	$(\mathtt{f{=}9}), rarely$	$(\mathtt{f{=}9}), often$
$\mathtt{BBS}_2$	$(\mathtt{f{=}0})(\mathtt{f{=}5}), often$	$(\mathtt{f{=}8}), often$	$(\mathtt{f{=}4}), often$
$\mathtt{BBS}_3$	$(\mathtt{f{=}5}), frequently$	$(\mathtt{f{=}4}), rarely$	$(\mathtt{f{=}9}), often$
$\mathtt{WWW}_1$	$(/\mathtt{\tilde{}li/}), rarely$	$(/\mathtt{\tilde{}li/pub/}), often$	$(/\mathtt{\tilde{}li/pub/}), rarely$
$\mathtt{WWW}_2$	$(/\mathtt{\tilde{}li/pub/}), often$	$(/\mathtt{\tilde{}li/}), rarely$	$(/\mathtt{\tilde{}li/doc/}), often$
$\mathtt{WWW}_3$	$(/\mathtt{\tilde{}li/}), frequently$	$(/\mathtt{\tilde{}li/doc/}), rarely$	$(/\mathtt{\tilde{}li/doc/}), often$

Table 2: Sample beliefs of fuzzy recurrence rules.

V. Conclusion

In this paper, we introduce the problem of discovering unexpected recurrence behaviors in sequence databases. We propose a novel notion, the fuzzy recurrence rules, for depicting the recurrence behaviors of the data, where fuzzy set theory is applied to describe the recurrence of sequences. We present a belief-driven approach for modeling two types of unexpectedness in recurrence behaviors, where the belief consists in a fuzzy recurrence rule and a semantic constraint on the rule. We also develop an effective algorithm UFR, which discovers all unexpected sequences in a sequence database with respect to domain expert specified belief base and minimum fuzzy degree threshold. The experimental results on Web access logs show the usefulness of our propositions.

Our future research includes the discovery of fuzzy recurrence rules in sequential data, we believe that our proposal of this novel rule model on sequences can be interesting for many real-word application domains.

References

[1] R. Agrawal, T. Imielinski, and A. N. Swami. Mining association rules between sets of items in large databases. In *SIGMOD*, pages 207–216, 1993.

[2] R. Agrawal and R. Srikant. Mining sequential patterns. In *ICDE*, pages 3–14, 1995.

[3] J. Ayres, J. Flannick, J. Gehrke, and T. Yiu. Sequential PAttern Mining using a bitmap representation. In *KDD*, pages 429–435, 2002.

[4] F. Berzal, J. C. Cubero, D. Sánchez, M. A. V. Miranda, and J.-M. Serrano. An alternative approach to discover gradual dependencies. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 15(559-570), 2007.

[5] T. Calders. Computational complexity of itemset frequency satisfiability. In *PODS*, pages 143–154, 2004.

[6] K. C. C. Chan and W.-H. Au. Mining fuzzy association rules. In *CIKM*, pages 209–215, 1997.

[7] R.-S. Chen, G.-H. Tzeng, C. C. Chen, and Y.-C. Hu. Discovery of fuzzy sequential patterns for fuzzy partitions in quantitative attributes. In *AICCSA*, pages 144–150, 2001.

[8] Y.-L. Chen and T. C. K. Huang. A new approach for discovering fuzzy quantitative sequential patterns in sequence databases. *Fuzzy Sets and Systems*, 157(12):1641–1661, 2006.

[9] M. Delgado, N. Marín, D. Sánchez, and M.-A. Vila. Fuzzy association rules: general model and applications. *IEEE Transactions on Fuzzy Systems*, 11(2):214–225, 2003.

[10] L. Di-Jorio, A. Laurent, and M. Teisseire. Fast extraction of gradual association rules: A heuristic based method. In *CSTST*, pages 205–210, 2008.

[11] D. Dubois and E. H. H. Prade. A systematic approach to the assessment of fuzzy association rules. *Data Mining and Knowledge Discovery*, 13(2):167–192, 2006.

[12] C. Fiot, A. Laurent, and M. Teisseire. From crispness to fuzziness: Three algorithms for soft sequential pattern mining. *IEEE Transactions on Fuzzy Systems*, 15(6):1263–1277, 2007.

[13] C. Fiot, F. Masseglia, A. Laurent, and M. Teisseire. Gradual trends in fuzzy sequential patterns. In *IPMU*, pages 456–463, 2008.

[14] D. Gunopulos, R. Khardon, H. Mannila, S. Saluja, H. Toivonen, and R. S. Sharm. Discovering all most specific sentences. *ACM Transactions on Database Systems*, 28(2):140–174, 2003.

[15] J. Han, J. Pei, Y. Yin, and R. Mao. Mining frequent patterns without candidate generation: a frequent-pattern tree approach. *Data Mining and Knowledge Discovery*, 8(1):53–87, 2004.

[16] T.-P. Hong, K.-Y. Lin, and S.-L. Wang. Fuzzy data mining for interesting generalized association rules. *Fuzzy Sets and Systems*, 138(2):255–269, 2003.

[17] 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 and Knowledge-Based Systems*, 11(2):173–194, 2003.

[18] E. Hüllermeier. Association rules for expressing gradual dependencies. In *PKDD*, pages 200–211, 2002.

[19] J. hyong Lee and H. Lee-kwang. An extension of association rules using fuzzy sets. In *IFSA*, pages 399–402, 1997.

[20] S. Jaroszewicz and T. Scheffer. Fast discovery of unexpected patterns in data, relative to a bayesian network. In *KDD*, pages 118–127, 2005.

[21] C. M. Kuok, A. W.-C. Fu, and M. H. Wong. Mining fuzzy association rules in databases. *SIGMOD Record*, 27(1):41–46, 1998.

[22] D. H. Li, A. Laurent, and P. Poncelet. Mining unexpected sequential patterns and rules. Technical Report RR-07027 (2007), LIRMM, 2007.

[23] D. H. Li, A. Laurent, and P. Poncelet. Discovering fuzzy unexpected sequences with beliefs. In *IPMU*, pages 1709–1716, 2008.

[24] B. Liu and W. Hsu. Post-analysis of learned rules. In *AAAI/IAAI*, pages 828–834, 1996.

[25] B. Liu, W. Hsu, L.-F. Mun, and H.-Y. Lee. Finding interesting patterns using user expectations. *IEEE Transactions on Knowledge and Data Engineering*, 11(6):817–832, 1999.

[26] B. Liu, Y. Ma, and P. S. Yu. Discovering unexpected information from your competitors' web sites. In *KDD*, pages 144–153, 2001.

[27] F. Masseglia, F. Cathala, and P. Poncelet. The PSP approach for mining sequential patterns. In *PKDD*, pages 176–184, 1998.

[28] K. McGarry. A survey of interestingness measures for knowledge discovery. *The Knowledge Engineering Review*, 20(1):39–61, 2005.

[29] B. Padmanabhan and A. Tuzhilin. A belief-driven method for discovering unexpected patterns. In *KDD*, pages 94–100, 1998.

[30] B. Padmanabhan and A. Tuzhilin. Small is beautiful: Discovering the minimal set of unexpected patterns. In *KDD*, pages 54–63, 2000.

[31] B. Padmanabhan and A. Tuzhilin. On characterization and discovery of minimal unexpected patterns in rule discovery. *IEEE Transactions on Knowledge and Data Engineering*, 18(2):202–216, 2006.

[32] J. Pei, J. Han, B. Mortazavi-Asl, J. Wang, H. Pinto, Q. Chen, U. Dayal, and M. Hsu. Mining sequential patterns by pattern-growth: the prefixspan approach. *IEEE Transactions on Knowledge and Data Engineering*, 16(11):1424–1440, 2004.

[33] A. Silberschatz and A. Tuzhilin. On subjective measures of interestingness in knowledge discovery. In *KDD*, pages 275–281, 1995.

[34] M. Spiliopoulou. Managing interesting rules in sequence mining. In *PKDD*, pages 554–560, 1999.

[35] R. Srikant and R. Agrawal. Mining sequential patterns: generalizations and performance improvements. In *EDBT*, pages 3–17, 1996.

[36] E. Suzuki. Autonomous discovery of reliable exception rules. In *KDD*, pages 259–262, 1997.

[37] E. Suzuki and M. Shimura. Exceptional knowledge discovery in databases based on information theory. In *KDD*, pages 275–278, 1996.

[38] E. Suzuki and J. M. Zytkow. Unified algorithm for undirected discovery of exception rules. *International Journal of Intelligent Systems*, 20(7):673–691, 2005.

[39] K. Wang, Y. Jiang, and L. V. S. Lakshmanan. Mining unexpected rules by pushing user dynamics. In *KDD*, pages 246–255, 2003.

[40] X. Yan, J. Han, and R. Afshar. CloSpan: Mining closed sequential patterns in large databases. In *SDM*, pages 166–177, 2003.

[41] L. A. Zadeh. Fuzzy sets. *Information and Control*, 8:338–353, 1965.

[42] M. J. Zaki. SPADE: An efficient algorithm for mining frequent sequences. *Machine Learning*, 42(1-2):31–60, 2001.

Author Biographies

Dong (Haoyuan) Li received his PhD degree in computer science from the University of Montpellier 2, France. He is an associate professor at the University of Tours, France. He is a member of the BDTLN team in the LI Laboratory. His research interests include knowledge based data mining and its applications.

Anne Laurent received her PhD degree in computer science from the University of Paris 6, France. She is an associate professor at the University of Montpellier 2, France. As a member of the TATOO team in the LIRMM Laboratory, she works on data mining, sequential pattern mining, tree mining, both for trends and exceptions detections and is particularly interested in the study of the use of fuzzy logic to provide more valuable results, while remaining scalable.

Pascal Poncelet received his PhD degree in computer science from the Nice Sophia Antipolis University, France. He is a professor at the University of Montpellier 2, France and the head of the TATOO team in the LIRMM Laboratory, France. He was a professor and the head of the data mining research group in the computer science department at the École des Mines d'Alès, France. His research interest can be summarized as advanced data analysis techniques for emerging applications.