2008

Mining Sequential Relations from Multidimensional Data Sequence for Prediction

Heng Tang
City University of Hong Kong, hengtang@umac.mo

Stephen Shaoyi Liao
City University of Hong Kong, issliao@cityu.edu.hk

Sherry Xiaoyun Sun
City University of Hong Kong, xiaoysun@cityu.edu

MINING SEQUENTIAL RELATIONS FROM MULTIDIMENSIONAL DATA SEQUENCE FOR PREDICTION

Completed Research Paper

Heng Tang
Dept. of Information Systems, City University of Hong Kong
Faculty of Business Administration, University of Macao, Postal 3001 Macao
hengtang@umac.mo

Stephen Shaoyi Liao
Dept. of Information Systems, City University of Hong Kong
83 Tat Chee Av., Kowloon, Hong Kong
issliao@cityu.edu.hk

Sherry Xiaoyun Sun
Dept. of Information Systems, City University of Hong Kong
83 Tat Chee Av., Kowloon, Hong Kong
xiaoysun@cityu.edu.hk

Abstract

By analyzing historical data sequences and identifying relations between the occurring of data items and certain types of business events we have opportunities to gain insights into future status and thereby take action proactively. This paper proposes a new approach to cope with the problem of prediction on data sequence characterized by multiple dimensions. The proposed relation mining approach improves the existing sequential pattern mining algorithm by considering multidimensional data sequences and incorporating time constraints. We demonstrate that multidimensional relations extracted by our approach are an enhancement of single dimensional relations by showing significantly stronger prediction capability, despite of the substantial work done in the latter area. In addition, matching algorithm based on the obtained relations is proposed to make prediction. The effectiveness of the proposed methods is validated by experiments conducted on a mobile user context dataset.

Keywords: Sequential Rule Mining, Multidimensional Data Sequence, Event prediction

Extraire des relations séquentielles à partir de séquences de données multidimensionnelles dans un but de prévision

Résumé

Cet article propose une nouvelle approche pour faire face au problème de la prévision sur des séquences de données caractérisées par des dimensions multiples. L'approche proposée d'extraction de relation améliore l'algorithme existant d'extraction de cadres séquentiels en prenant en considération les séquences de données multidimensionnelles et en incorporant des contraintes de temps.

Introduction

Background and Motivation

Prediction, the process of forecasting future trends based on historical information, has found many applications in various areas such as marketing, manufacturing, banking, etc. Prediction is performed by inferring on a set of observations collected over a period of time, which is referred to as a data sequence or sequence for short (Han and Kamber 2006). Observations in a sequence are often characterized with the feature of multidimensionality. For example, observations of a customer's motion in a shopping mall can be recorded along two dimensions, i.e. location and speed. Sequential relations, which are the frequent in-order occurrences of data values, thus can be discovered from data sequence and stored as knowledge. The sequential relations extracted from a set of multidimensional data sequences have been widely applied in the business world. Let us consider the following two scenarios:

i) Location Based Services (LBS) launched by mobile service providers enable the delivery of value-added services based on customers' location (Schiller and Voisard 2004) through the so called "Location Intelligence", which can help discover the correlation between mobile user's historical locations and their activities in order to support proactive service (Karimi and Liu 2003; Schiller and Voisard 2004). However, in many cases predicting mobile user's possible activity simply based on "location" may not be able to achieve satisfactory accuracy. The opportunity to make better prediction can be increased by taking account of multiple dimensions of the user's context, such as time and weather. For example, the analysis of historical data may suggest that when a customer is in a shopping mall, s/he might be looking for a certain service with a possibility of 60%. This possibility can significantly vary with extra contextual information: when it is a weekend and raining, the possibility of a service purchase by a customer in a shopping mall can increase to 95% and otherwise in other contexts the possibility is actually as low as 15%. As such, multidimensional contextual information provides important clues to the customer's preference under a specific circumstance through frequently occurring sequential relations.

ii) Another scenario involves a driver status early warning system. After a long distance's driving on the highway, the fatigue level of the drivers may increase, along with the lack of alertness and the diminishing ability to control the vehicle (Grace et al. 1998). An early warning system may help to predict the driver's fatigue level by collecting and analyzing driver's movement features (Horerry and Hartley 2001) and the sequential relations identified from the data collected along such dimensions as the frequency of wink, the movement characteristics of head, neck and shoulders, can be used to predict the fatigue level of the driver. Thus the system is able to foresee possible danger and to generate early warning accordingly via monitoring the multidimensional movement features of a driver.

From above two scenarios we can find the following in common in terms of a prediction problem:

First, sequential relations may indicate the occurrence of the target events that we are concerned about. As a matter of fact, a number of researches have demonstrated that sequential relations play a very important role in prediction (Dunning 1994; Su et al. 2000; Zaki et al. 1998) in a variety of application areas including web access behavior modeling, nature language processing, plan failure prediction, etc. The specific objective of both the two scenarios is to identify the sequential relations for prediction so that appropriate action can be taken accordingly. The form of such relation can be "*x leads to y*" where *x* is a sequence and *y* is the target business event. Take an example from the LBS scenario, a sequential relation can be (referred as "m-coupon1" hereafter):

[office, afternoon], [shopping mall, night] *leads* to "mobile coupon for the food court"

in which location and time are two involved dimensions ("dimension" and "attribute" are interchangeable in this paper). This relation indicates that the sequential occurring of the multidimensional context (i.e. [office, afternoon] and [shopping mall, night]) implies that the food court coupon is favorable to that specific customer. Therefore by discovering those relations and monitoring their occurrences, proactive and personalized mobile services can be provided.

Second, data sequences are characterized by multiple dimensions. The LBS scenario suggests that, to some extent, relations discovered from multidimensional sequences can be more accurate for prediction in many cases. Empirical evidences have been given by Padmanabhan et al. that rich and complete customer information rather than a modest number of customer attributes can benefit the analytic Customer Relationship Management (Padmanabhan et al. 2006).

In view of the indicated situations shown above, we propose an approach to discover predictive relations from multi-dimensional data sequences in order to enhance the prediction accuracy. Although substantial work has been done in sequential pattern mining, the approach presented in this paper is unique in that we focus on extracting multidimensional patterns as opposed to single-dimensional patterns identified via other approaches. Moreover, we present an approach to make prediction based on the extracted relations by comparing the relations with incoming sequence and estimating the corresponding probability.

Note, to avoid confusion in this paper, we adopt the same term "rule" as other studies to refer to the forenamed "relation" hereafter. Additionally, the term *frequent rules* used in this paper conforms to its original meaning in data mining literatures, which refers to rules that have the number of occurrences higher than a given threshold in the training dataset.

Contributions and Organization

The intended contributions of this research are the mining and matching algorithms to deal with prediction using multidimensional rules. The multidimensionality of data has posed many challenges for mining useful rules. The first challenge is to take consideration of the multidimensional setting in the mining algorithms (Kogan et al. 2006). In this paper, in order to handle the multidimensional data, we propose to take "snapshots" along one continuous dimension (such as time), and then to identify the co-occurrence relation between the snapshots and the target events. With our formulation of the problem, the data mining algorithms handling single dimension mining, i.e., MINEPI, is extended for multidimensional sequential rule mining.

Our approach can be applied in many areas such as the context-dependent mobile service providing, which strives to deliver relevant, targeted, and timely information to mobile customers in coherence to their context (Rao and Minakakis 2003). Hence throughout the rest of the paper, we use the context-dependent prediction scenario as a running example to demonstrate our methods. This approach, however, is generalizable to many other business applications characterized with multidimensional data.

The remainder of the paper is structured as follows. In Section 2 we provide a brief review of the relevant literatures. In Section 3 we formulate the rule mining problem and outline the mining algorithm. The matching process will be given in Section 4. The experiments will be presented in Section 5 to validate the effectiveness of the proposed approaches. We conclude the paper in Section 6 by summarizing the contribution of this paper and outlining future research directions.

Literature Review

This section consists of the review of several areas of related works, including sequential pattern mining, multidimensional sequence, and association rule based prediction.

Sequential Pattern Mining

Data sequence, as an important data format, has a wide range of business applications, such as investment, auction, workflow, and banking. Pattern mining from data sequences has aroused the consistent research interest the data mining community (Wang and Yang 2005). (Agrawal and Srikant 1995) is one of the earliest studies addressing the problem of discovering frequent sequential patterns and the proposed approach is improved in (Srikant and Agrawal 1996). Thereafter research in this area becomes increasingly active, which in general fall into two categories based on the forms of input dataset to be mined: The first category focuses on developing effective algorithms to detect sequential patterns from transactional database. Many studies contribute to this topic, including (Mortazavi-Asl et al. 2004; Wang et al. 2007; Yu and Chen 2005; Zaki 2001). The second category of research focuses on mining the sequence database which stores the succession of data items, with or without a concrete notion of time. Examples include customer shopping sequences, Web click streams, and biological sequences (Han and Kamber 2006). Mining tasks rely on transactional database is different from that on a sequence because the former is to identify frequent patterns from multiple sequence segments whereas the latter is to discover recurring patterns from a single sequence. The mining algorithm we propose in this paper attempts to deal with the second type of data format. In this area Mannila and Toivonen (1996) use "Episode" to describe frequently recurring subsequences and proposed two efficient algorithms, *WINEPI* and *MINEPI*. Many other substantial works have been done on Episode mining, including (Atallah et al. 2004; Bettini et al. 1998; Laxman et al. 2007; Mannila et al. 1997). For example, Bettini et al. (1998) address the problem of mining event structures with multiple time granularity; Laxman, Sastry et al. (2007) extend the episode mining approach by bringing into event duration constraints explicitly into the episode. However, the difference between the above episode mining techniques and ours is that, the formers are not directly applicable to multidimensional sequence which can improve prediction accuracy in many cases.

Multidimensional Sequence

The problem of discovering multidimensional sequential rules for prediction studied in this paper is a new problem and to our knowledge, no related work has directly tackled this problem. Note that the general concept of mining multidimensional sequential rules is addressed in several articles. Yu and Chen's (2005) work studies the episode mining problem in multidimensional sequence, however, the term "multidimensional" used in the paper refers to the multiple granularity in terms of the time dimension of the events' occurrence, which is a different concept to that discussed in this article. Attempts to detect sequential patterns from multidimensional transactional database are made by Pinto et al. (2001). Nonetheless, as previously discussed mining transactional database differs from mining a sequence, and sequential patterns generated from transactional database cannot be directly used for prediction.

Association Rules Based Prediction

The proposed matching approach is an extension of conventional *n*-gram model in which an *n*-gram refers to a subsequence of n items from a given sequence (Lee et al. 1990). As one of the most well-known probabilistic models coping with sequential rule based prediction, *n*-gram has been widely used in statistical natural language processing (Dunning 1994) and genetic sequence analysis (White et al. 1993). Many works attempt to build up *n*-gram models using association mining techniques. For example, Yang et al. 's (2004) research mines association rules from web user sessions to build conditional probability of future web documents for caching optimization. Similarly, the *WhatNext* model developed in (Su et al. 2000) generates simple *n*-grams through mining associations. As a step forward, our rule prediction algorithm extends *n*-grams by taking the time constraints into account when matching grams with the input sequence.

Note that although another extensively studied area - time series forecasting (Chatfield 2001; Wei 1989) focuses on the general prediction problem as well, the framework introduced in this article primarily differs in the following two regards. First, the research topics to be coped with are different. Forecasting in time series area mainly studies the prediction problem with continuous data whereas the problem to be solved in this paper is the prediction of categorical data across multiple attributes. Second, to apply time series forecasting approaches to multivariate prediction problems, an appropriate multivariate models should be determined in advance (Chatfield 2001). In contrast, such data mining based framework as introduced in this article is in essential a problem-oriented solution (Cui et al. 2006). It aims to explore a large amount of data, removing many of the restrictions associated with model-based methods and perform computation based on the statistic of discrete events' frequencies thus to certain extent, can be viewed as a model-free approach.

Mining Rules from Multidimensional Sequence

In this section we first formulate the problem of mining sequential rules, and then the associated algorithm is presented.

Problem Formulation

The input data are considered as a sequence of data items with multiple attributes or dimensions. In the paper we use the term "Snapshot" to describe a data item at a time point in a generalized form:

DEFINITION 1 (Snapshot). A snapshot $s \in D_1 \times D_2 \times \cdots \times D_m$ is a sample value vector defined on *m* dimensions. It is denoted as $s = [v_1, ..., v_j, ..., v_m]$ representing the values in all dimensions at a given time point, where v_j represents the observation of the *j*-th dimension, $j = 1...m$.

For instance, a snapshot $s = ["\text{office}", "15:00\text{PM}", "\text{Raining}"]$ can be used to represent the value reading of location, time, and weather at a particular time point.

Data collected from heterogeneous sources can help to trace any changes of the status of an object of our concern. Back to the LBS scenario, a customer's current contextual information can be captured and recognized through a variety of sources. For instance, location information can be obtained via positioning devices, and weather related information is available at web services offered by various providers. Notably, although time information is in general regarded as the abscissa axis of a data sequence, it is also an essential type of contextual information because many kinds of time granularities are closely associated with people's activities (e.g., working hours, day of the week, etc).

In the learning and matching process we need to know whether two snapshots being compared are equal or not. In general, since the value domains of all dimensions are required to be categorical in this research (except the time axis), two snapshots are regarded as equivalent if their values in corresponding dimensions are always equal, that is:

DEFINITION 2 (Equivalent Snapshots). For any two *m*-dimensional snapshots $s_p = \lfloor v_{p1}, v_{p2}, ..., v_{pm} \rfloor$ and $s_q = \lfloor v_{q1}, v_{q2}, ..., v_{qm} \rfloor$, s_p and s_q are Equivalent Snapshots to each other if and only if $v_{pj} = v_{qj}$ for all $j = 1...m$, denoted as $s_p = s_q$.

We use Γ to denote the set of all snapshots in the domain. Target events are a specific type of snapshots that we attempt to predict, denoted as $EVT \subset \Gamma$. For example, a mobile phones user's activity, such as redeeming a mobile coupon, can be regarded as a target event.

A sequence is the input multidimensional succession of snapshots which is defined as:

DEFINITION 3 (Sequence). A *data sequence* (or *sequence* for short) is a list of elements occurring in order, denoted as $S = \langle s_1, s_2, ..., s_n \rangle$, where for each $i = 1...n$, $s_i \in \Gamma$ can be either an ordinary snapshot vector or a target event. The set of time points that s_i occurs in *S* is denoted as $t(s_i)$.

The prediction method is proposed in this way: the temporal correlations between snapshots and target events can be detected from a historical snapshot sequence and stored as rules, thus a prediction regarding the target event can be made in the future. That is, once the antecedent of a rule appears in an incoming sequence, the target event indicated by the corresponding consequent is possibly to occur within a certain time period. The similar idea is used in many other approaches to prediction, such as *n*-gram (Dunning 1994; Lee et al. 1990).

A sequential rule can be regarded as a subsequence of the original data sequence *S*, where a subsequence is defined as a new sequence which is formed from *S* by deleting some of the elements without disturbing the relative positions of the remaining elements. The learning process employs the associations mining technique to identify the frequent sequential rules. The set of all possible sequential rules is called the "Preliminary Rule Set":

DEFINITION 4 (Preliminary Rule Set). Let $S =< s_1, s_2, ..., s_n >$ be an input sequence with length *n*. Suppose $S' =< s_{p1}, s_{p2}, ..., s_{pl} >$ is a subsequence of *S* with length *l*. We say *S'* is a preliminary rule if $s_{pi} \notin EVT$ for any $i \in [1, l-1]$ and $s_{pl} \in EVT$. The set of all preliminary rules of *S* is denoted as $PR(S)$.

A sequential rule may be written in the implication form $P \rightarrow A$, where *P* is a sequence and *A* is a target event. For convenience, hereafter in this paper a sequential rule can be either written as implication ($P \rightarrow A$) or sequence ($< s_{p1}, s_{p2}, ..., s_{pl} >$), where $< s_{p1}, s_{p2}, ..., s_{pl-1} >= P$ is the sequence of snapshots and $s_{pl} = A$ is a target event.

Given the importance of the time span of the snapshots in a sequential rule, we use two temporal constraints to restrain the relative occurring time of snapshots in a sequential rule. That is, σ_1, the maximum allowed time difference between any two neighboring snapshots and σ_2, the maximum allowed time difference between the first snapshots and the target event. Based on these two constraints we define the concept of occurrence, which is the sequence of time points recording when each snapshot of *S*' in *S* occurs, subject to σ_1 and σ_2.

DEFINITION 5 (Occurrence). Let *S* be the original data sequence. Suppose $S' = < s_{p1}, s_{p2}, ..., s_{pl} >$, $S' \in PR(S)$. Let $< t_1, t_2, ..., t_l >$ be the sequence of time points of *S'*. If for any $i = 1...l-1$, we have $0 < t_{i+1} - t_i \le \sigma_1$ and $t_l - t_1 \le \sigma_2$, then we say that $< t_1, t_2, ..., t_l >$ is an occurrence of *S'* in *S*, denoted as $o(S', S, \sigma_1, \sigma_2)$, and the overall time span of *o* is denoted as $span(o) = [t_1, t_l]$.

Let $occr(S', S, \sigma_1, \sigma_2)$ be the set of all occurrences of S' in *S* subject to σ_1 and σ_2. We use Agrawal and Srikant's (1994) classical support-confidence framework to extract significant rules, in which for a rule *r*: *X*→*Y*, *supp*(*r*) = *P*(*XY*) and *conf*(*r*) = *P*(*XY*) /*P*(*X*) are used as the criteria of a qualified rule in the mining process. In the definition below we use the occurring frequency instead of the probability *supp*(*r*) for simplicity.

Let $\|\cdot\|$ be the number of elements in a set. The problem of mining sequential rule with length *l*, therefore, is defined as to identify $R_l = \underset{R_l \subseteq PR(S)}{\arg\max} \|R_l\|$, such that for any $r = < s_{p1}, s_{p2}, ..., s_{pl} >$, $r \in R$, the following three conditions are satisfied:

1) Given any $o, o' \in occr(S', S, \sigma_1, \sigma_2)$, $o \ne o'$, then $span(o) \bigcap span(o') = \varnothing$ (1)

2) $$freq(r) = \left\| occr(< s_{p1}, s_{p_2}, ...s_{p_l} >, S, \sigma_1, \sigma_2) \right\| \ge min\_freq \quad (2)$$

3) $$conf(r) = \frac{\left\| occr(< s_{p1}, s_{p_2}, ...s_{p_l} >, S, \sigma_1, \sigma_2) \right\|}{\left\| occr(< s_{p1}, s_{p_2}, ...s_{p_{l-1}} >, S, \sigma_1, \sigma_2) \right\|} \ge min\_conf \quad (3)$$

Condition 1 requires the time spans of any two occurrences of a rule should not be overlapped. Condition 2 and 3 define the minimum frequency and confidence of the discovered rules. In this case rule *r* can be written as $r :< s_{p1}, s_{p2}, ..., s_{pl-1} > \rightarrow s_{pl}$.

For any rule *r*, if equation 2 is satisfied, we say *r* is a frequent rule.

An illustrative example of the above definitions is given in figure 1. Suppose the mobile user's contextual information in the form of snapshot sequence (*S*) is shown in the grid and the time difference between two adjacent cells is 1 time unit. In this example s_i and a_1 represent the context snapshots and a target event (user's activity), respectively. Before the event a_1, there are snapshots s_2 and s_3 that tend to occur sequentially (marked with shading cells), regardless the occurring of other snapshots. Thus the frequent occurring of the rule $r :< s_1, s_2 > \rightarrow a_1$ implies the association between of a succession of occurrences of snapshots and an event. Let σ_1 = 3 and σ_2 = 5 time units, then in this example the occurrence frequency (e.g $freq(r)$) of snapshot sequences $< s_2, s_3 >$ and $< s_2, s_3, a_1 >$ are 5 and 4, respectively. Thus $conf(r) = \frac{\left\| occr(< s_1, s_2, a_1 >, S, \sigma_1, \sigma_2) \right\|}{\left\| occr(< s_1, s_2 >, S, \sigma_1, \sigma_2) \right\|} = {}^4\!/_5$.

Thus if above calculated $freq(r)$ and $conf(r)$ are greater than the predefined thresholds *min_freq* and *min_conf*, the rule $r :< s_1, s_2 > \rightarrow a_1$ is identified as an element of R_3.

s6	s2	s3	s4	s1	a1	s6	s9	s10	s2	s15	s3	s19	a1

a2	s8	s7	s9	s6	s2	s16	s3	s20	s17	s15	s14	s2	s19

s5	s3	s16	a1	s11	s3	s2	s3	s19	s10	a1	s6	s8	s9

Figure 1. An example of sequential rule

The mining process with respect to this specific mobile user, therefore, can be treated as the process of identifying rules with regard to given minimal support and confidence.

The mining algorithm

The problem of mining sequential rules addressed in this paper is similar to the Episode mining problem studied in (Atallah et al. 2004; Bettini et al. 1998; Laxman et al. 2007; Mannila and Toivonen 1996; Mannila et al. 1997). In this paper we use a modified version of the MINEPI proposed in (Mannila et al. 1997) by incorporating time constraints σ_1 and σ_2 in order to enhance the pruning process for reducing searching space.

This algorithm adopts a level-wise strategy used in the Apriori Algorithm, namely, we generate the subsequences with length *k* in the *k*-th iteration. Initially in level 1 procedure, frequent subsequences with length 1 are counted and stored. Then new subsequences in level *k* candidate set are generated by concatenating two overlapping subsequences with length *k*-1. Hence the length of rule will grow by one in each iteration/level. Concatenation is the basic operation in the rule growing process which is formulated below.

DEFINITION 6 (Concatenation of Overlapping Subsequences). Suppose *S* is the input snapshot sequence. Given two subsequences of *S* with length m, say $S_1 =< s_{1,1}, s_{1,2},...s_{1,m} >$ and $S_2 =< s_{2,1}, s_{2,2},...s_{2,m} >$, $m \geq 2$. If for any $i = 2...m$ we have $s_{1,i} = s_{2,i-1}$, then we use $concat(S_1, S_2) =< s_{1,1}, s_{1,2},...s_{1,m}, s_{2,m} >$ to denote the concatenation of S_1 and S_2 which has the length of *m*+1.

This operation generates a new subsequence with length *m*+1 by combining two sequences with length *m*.

Specifically, any two subsequences with length 1 can be directly concatenated together to form a new sequence with length 2.

Frequent subsequence set will be scanned and infrequent new entries will be eliminated. Invalid entries will also be removed using pruning strategies. That is, we compute the gap between any two neighboring snapshots and the time span of a whole sequence, and if the two time constraints σ_1 and σ_2 are not satisfied, the current rule will be pruned from the frequent item set (the *prune*() function in figure 2). The above procedures are executed iteratively until all frequent subsequences with length *L* are identified.

Finally, all frequent sequences that end with an event are regarded as potential rules. Calculating the confidence of a rule set is straightforward with equation (3). All rules with confidence larger than *min_conf* are collected into the validate rule set for further process.

The algorithm is outlined in figure 2 (algorithm1).

```
Input: original mining sequence S, min_freq, min_conf, $\sigma_1$, $\sigma_2$, the maximum rule length L
Output: the set of all frequent rules R
Method:
i =1      // i is the level indicator, and Cand_i is the set to store candidate rules in level i
Generate $Cand_i = \{S' \mid length(S') = 1\}$ where S' is a subsequence of S
For i = 2 to L do
```

```
    Compute $Freq_i = \{S' | S' \in Cand_i \wedge freq(S') \geq \min\_freq\}$
    Compute $Freq_i = prune(Freq_i, \sigma_1, \sigma_2)$
    Compute $Cand_{i+1} = \{S' | length(S') = i + 1 \wedge S_1 \in Cand_i \wedge S_2 \in Cand_i \wedge S' = concat(S_1, S_2)\}$
End For
Compute the confidence for all $S' \in Freq_i$
For i = 1 to L do
    For any $S' = < s_1, s_2, ..., s_i >$, $S' \in Freq_i$, if $s_l \in EVT$ and $conf(S') \geq \min\_conf$, output $S'$
End For
```

Figure 2. Algorithm 1 – Mining of sequential rules from snapshot sequence

Generating Prediction by Rule Matching

In order to predict a target event, we need to compare incoming data sequence with previously identified rules relevant to the target event. We extend the *n*-gram based prediction method, which has been applied to predict web request on the basis of the historical Access Patterns (Su et al. 2000; Yang and Zhang 2003), and develop a generalized rule matching approach, which allows inconsecutive match in the rule matching process and factors the time constraints (σ_1 and σ_2) into the matching process. Our approach looses the requirement for the lengths of rule. i.e., the lengths of rules do not have to be predefined and the lengths of rules can vary in the same rule base.

Our procedure for rule matching is outlined in Figure 3 as algorithm 2, which generates predictions as follows. Suppose *RB* is the rule base with *k* rules, three tables are used to maintain the current matching status of each rule: *matched*[1..*k*] maintains the last matched snapshot of each rule, *time_last*[1..*k*] maintains the time point of the last matched snapshot of each rule, and *time_first*[1..*k*] records the time point of the first matched snapshot of each rule, respectively.

Assume that $r :< s_1, s_2, ..., s_n > \rightarrow a$ is the rule to be compared with. Given a matched portion $\tilde{r} = < s_1, s_2, ..., s_i >$ and an incoming snapshot *s*, if $s = s_{i+1}$ (the next snapshot after s_i), which means the current input snapshot *s* is found matching with the current portion $\tilde{r}$, if the conditional probability $p(a | < s_1, s_2, ..., s_{i+1} >)$ is no less than a predefined threshold ϕ, then the entire rule *r* can be regarded as matched and then the corresponding target event *a* is triggered; otherwise the current *s* is received and we wait for the next incoming snapshot because the confidence of the occurring of *a* is not adequate. In the whole process whenever the matching time spans recorded in status tables violate two time constraints σ_1 and σ_2, current matching of rule *r* is considered as failed. Hence we start over by resetting the matching status maintained in the three tables and waiting for the next incoming snapshots. In particular, owing to the level-wise nature of the rule mining algorithm during the learning stage, the conditional probabilities associated with all prefixes of a rule are already available and do not need to be recalculated during the matching process. At each update of the incoming snapshots, the algorithm scans all rules in the base and attempt to find a validate match, hence the time complexity at each update is $O(|RB|)$, where $|RB|$ is the size of the rule base *RB*.

```
Input: Rule Base RB ($|RB| = k$), time constraints $\sigma_1$ and $\sigma_2$, confidence threshold $\phi$
Output: Predicted target events a and corresponding probability p(a)
Method:
Initialize: matched[1..k] = 0; time_last[1..k] = 0; time_first[1..k] = 0
Repeat: Wait for next update snapshot s
For each rule $r_i \in RB$, where $r :< s_1, s_2, ..., s_n > \rightarrow a$ and n is the length of the antecedent
If s.time – time_last[i] > $\sigma_1$ OR s.time – time_first[i] > $\sigma_2$ Then
```

```
            $matched[i] = 0$; $time\_first[i] = 0$;   //expire and reset rule $r_i$
        Else
          If $r_i[match[i]+1] = s$ Then          //matched: $s = s_{i+1}$
            If $p(a \mid < s_1, s_2, ..., s >) \geq \phi$ Then
              Output $a$                          //generate the target event of $r$
            Else                                  //receive and wait for next match
              $matched[i]$ ++; $time\_last[i] = s.time$
              If $matched[i] = 1$ Then $time\_first[i] = s.time$
            End If
          End If
        End If
    Loop
    Goto Repeat:
```

Figure 3. Algorithm 2 – Matching snapshot sequence for prediction

Evaluation

To validate the proposed mining algorithm for the prediction of events, we conduct experiments on a context database referred to as “Nokia Context Data” available at (Flanagan 2004). The data consist of contextual data with multiple dimensions. In addition to validating our framework in this study, the data set is used also as a manifestation for the running example – the proactive LBS scenario.

Data Description

The Nokia Context Dataset consists of a set of feature files for 43 different recording sessions and one session for each day (Flanagan 2004). In each recording session the same user carried a mobile phone, sensor box and laptop PC, going from home to the workplace or vice-versa. The user may choose to walk, drive a car, or take a bus or Metro. Furthermore all user interactions with the mobile phone such as calls, SMS, WEB pages accessed etc. were also logged and time stamped. All recorded signals were processed and features of each context attribute/dimension are extracted. Attributes in numerical format are discretisized into levels.

The original data files are combined into one source data file consisting of 210,000 rows ordered by the time stamp. The summary of the dataset is presented in Table 1.

Table 1. A Summary of the Nokia Context Dataset

Record Type	Dimensions	Value Domains/ Formats	Example
Snapshot	Location	Area Code – Cell ID	“1-3”
	Day Name	1~7 (Mon.,...,Sun.)	“1” (Mon.)
	Day Period	1~4 (Night, Morning, Afternoon, Evening)	“1” (Night)
Target Event/ Interaction	Launch an application	a[0-31]	“a2”
	Access a WEB page	b[0-13]	“b13”
	Initiate communication	c[0-4]	“c3”

Measures and Experiment Setup

The measures of prediction quality used in this research are *precision*, *recall* and *F1* which are adopted in a wide range of studies with regard to classification and prediction (Herlocker et al. 2004; Yang et al. 2004). Precision represents the probability that predicted interactions will occur whereas Recall represents the probability that interaction item will be predicted. They can be calculated using the formulas (4) and (5) below where TP is the number of interactions predicted to occur and actually occur, FP is the number of interactions predicted to occur but do not occur, and FN is the number of interactions predicted not to occur but actually occur (subject to σ_1 and σ_2).

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \qquad (4)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \qquad (5)$$

Precision represents the probability that predicted interactions will occur whereas Recall represents the probability that interaction item will be predicted. Another widely used measure that combines Precision and Recall into a single metric is F_1 (Van Rijsbergen 1979).

$$\text{F1} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \qquad (11)$$

In terms of evaluation, the rules learning and validation steps should be conducted on different datasets (Cui et al. 2006; Han and Kamber 2006). Hence in all the three experiments, the Nokia Context Dataset is separated into two parts: a learning set consisting of 33 days of data and a validation set consisting of 10 days of data.

We run the rule mining algorithm (algorithm1) on the mining set and then evaluate the discovered rules on the evaluation set in the experiment. We implement a matching component which reads the snapshots one by one from the validation set and compares them with the rules in the rule base using the matching algorithm (algorithm2), and invokes a predicted target event when the corresponding probability is no less than the given confidence threshold ϕ.

Experiments

The objective of the experiment is to examine the predictive ability of multidimensional sequential rules, specifically, we compare the prediction results produced by one-dimensional rules and multidimensional rules. We employ the mining algorithm on different combination of context attributes. Identified rules are in the forms as "[1-1,1,4][1-3,1,4][1-4,1,4][c,1], in which the square brackets are used to separate each snapshots and the ending element is an interaction (i.e., [c,1]). In each snapshot different dimensions are separated by commas. Some of the identified 2-dimensional rules (Location+Period) are shown in table 2.

Table 2. Part of the sequential rules discovered from the mining dataset

Freq	Conf	Rule
10	0.91	[1-3,4][a,12]
10	0.91	[1-3,4][c,3]
10	1	[1-4,4][a,11]
10	1	[1-1,4][1-3,4][b,2]
10	1	[1-1,4][1-3,4][b,3]
10	1	[1-1,4][1-3,4][a,12]

10	1	[1-1,4][1-3,4][1-4,4][b,1]
10	1	[1-1,4][1-3,4][1-4,4][c,1]
		…

We test the different combination of context attributes: Location, Location+Day, Location+Period, and all 3 attributes, and the number of rules generated are presented in the table 3 (min_freq = 10, min_conf=0.90, σ_1 = 36000 sec, and σ_2 = 70,000 sec). Note that the setting of σ_1 and σ_2 relates to the specific problem. We set the values of σ_1 and σ_2 to approximately 10 and 20 hours respectively such that the discovered rules could capture the co-occurrence of two neighboring activities or snapshots happening in different periods of a day (e.g., morning, afternoon, etc.). It also means that we ignore the rules across two different days. The "Length" column in this table represents for the rule length. The minimum lengths are 2 because a rule consists of at least one snapshot and one interaction.

Table 3. Statistic of a group of the mining results (186 rules in total)

Length	Location	Location+Day	Location+Period	All 3 attributes
2	2	6	5	1
3	18	6	49	1
4	25	16	40	0
5	4	9	4	0

We examine the predicting results using different combinations of context dimensions, including, "location", "location+period", "location+day", "location+period+day", in which "location" is included in all combinations because location information is the most informative dimension regarding people's activity. Four rule bases using the rules extracted along different combinations of dimensions are RB1 (location); RB2 (location, location+day), RB3 (location, location+period), and RB4 (location+period+day, location+period, location+day, location) respectively. The corresponding measure labels in the chart are denoted as prec*, recall*, and F1_*, where the asterisk stands for any rule base from 1 to 4.

Figure 4(a) plots the precisions of different dimension combinations at different levels of confidence threshold. Figure 4(b) plots the recalls of different dimension combinations at different levels of confidence threshold. It can be observed from Figure 4(a) that several dimension combinations, especially those with the combination of location and period (pre3 and pre4), show better prediction when the confidence threshold is high, which conforms the rationale we explained previously, that is, the use of additional contextual dimensional introduces finer granularity to the snapshots, and might consequently increase the prediction by increasing TP and decreasing FP. This effect is suppressed when the confidence threshold is low because in this case, the false positive (FP) predictions made by low confidence rules are significantly increased.

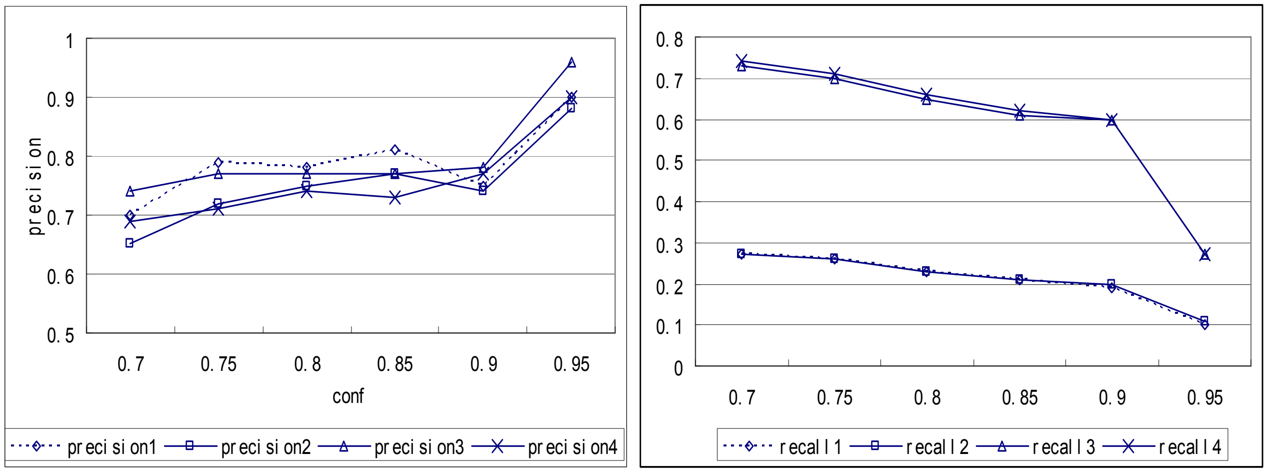

Figure 4. Precisions (a) and Recalls (b) at different confidence levels

Figure 4(b) shows that the significant increase of recall, because with the introducing of multiple dimension combinations, the size of the rule base is increased and consequently the predictions cover a much larger portion of all incoming target events, which conforms to intuition well. It is worth notice that, because the current prediction task is to forecast the human activities which are highly random in general and independent to the regularity of the device carrier, the level precision and recall are acceptable.

To compare the enhancement to the overall performance, the corresponding F1 measure is plotted in Figure 5. It can be seen that the overall effectiveness measured by F1 is higher in two of the multidimensional rule bases (F1_3 and F1_4).

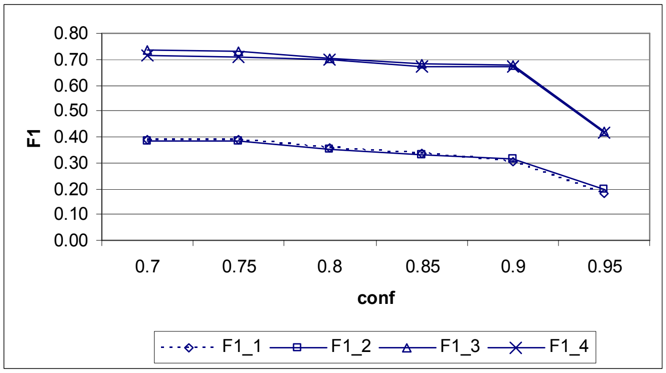

Figure 5. F1 at different confidence levels

Discussions

It is worth noting that, data preparation, which is one of the most important processes in a data mining based solution (Han and Kamber 2006), is also essential to the proposed approach. The specific data preparation problems to be solved before applying this approach are as follows.

i) Due to the settings of the data collecting process, the original snapshot sequence sometimes can be problematic if directly used in the mining and matching without preprocessing. For example, for the one dimensional sequence <AAAABBBBCCC>, local repetition of the same snapshots can be observed which are mainly caused by the setting of sampling rate. If it is used by the rule mining process as input, local repetitions will be identified as frequent patterns although they are actually spurious ones. A solution proved effective in this research is to intervalize the original sequence. That is, we group the repeating identical snapshots into the same interval and record its starting time and ending time (e.g. <A[1,4], B[5,8], C[9,11]> in last example). Consequently, in this case the occurring time of a snapshot becomes an interval rather than time point, thus we need to define the calculation of time difference in order to calculate the time constraints used in the mining and matching process.

Given two time intervals $a = [a_1, a_2]$ and $b = [b_1, b_2]$, the time difference between *a* and *b* is defined as:

$$diff(a,b) = \begin{cases} 0 & (b_1 - a_2) * (b_2 - a_1) \le 0 \\ \min(|b_1 - a_2|, |b_2 - a_1|) & otherwise \end{cases} \quad (11)$$

That is, the difference is zero when two intervals are overlapped or adjacent, otherwise it is calculated by the distance between the closest two boundaries of two respective intervals.

ii) Another problematic phenomenon appears in the primitive sequence is what we called "shuffle region", an example is the following segment regarding the location dimension from the Nokia dataset (in the form of *Area Code* - *Cell ID*):

<…2-8, 2-8, 2-6, 2-8, 2-6, 2-8, 2-6, 2-8, 2-6, 2-8…>

In this segment two locations "2-6" and "2-8" frequently appear alternately for a long duration with the sampling rate of 1 Hz. A possible reason is that the moving route of the mobile device carrier happens to have the same distance to two GSM towers, thus the signal strength detected from two towers tend to be close in value. The similar phenomenon occurs in other dimensions such as activity, noise level which may be caused by the inappropriate granularity generated by the data discretization. Regarding this problem we employ a clustering-like algorithm to identify shuffle regions and substitute the values into a new labels, (e.g., "2-6|2-8" in the above example), which implied for a "tangled" state of the associated values.

In either of the above two situations spurious local patterns will be detected, normally with high support, which will overwhelm the authentically valuable rules. In the experiments on Nokia Context Dataset spurious patterns may account for over 80% of the total identified frequent patterns without preprocess. Thus in the experiment we applied data preprocess procedures on both the training dataset and evaluation dataset in order to eliminate spurious patterns. It is worth mentioning that, in some situations, although additional information may help to tackle the above problems for some special types of data (e.g., information provided by GIS may help to cope with "shuffle regions" more easily), we intend to discuss the above problems in a general way and give an efficient and straightforward solution.

Conclusion

This study attempts to tackle the problem of mining multidimensional sequential rules from data sequence for event prediction. This is a significant problem, because sequential rules with multiple dimensions outperform with single dimensions in predictiveness. First, we formulate the rule mining task on the basis of the classical support-confidence framework. Further, our solution includes two main algorithms for rule mining and matching. We validate the proposed algorithms using a dataset of a mobile device carrier's contextual information along with interactions of the carrier with the mobile device. The experimental results show that the recall of multidimensional rules is substantially improved comparing with that of single-dimensional rules, while the precision remains the same level. The overall performance measured by F1 is also enhanced.

As a summary, this paper demonstrates the potential of the proposed approach to enhance the conventional single dimensional prediction. From a practical perspective, we present the way to identify multidimensional sequential rules and to use them for prediction. We believe that the proposed solution has extensive applications, as long as the domain is related to analyzing multidimensional categorical data sequences for the purpose of predicting future events.

In this research multiple dimensions are considered in order to discover effective rules unveiled by mining single dimensional sequences, thereby the prediction quality can be improved. However, knowledge redundancy is imported with the including of extra dimensions. Thus an issue worth further study is to determine which rule to choose if both a rule and its higher-dimensional derivation are discovered.

Another area deserves exploration is the rule ranking measure besides the conventional support and confidence. An effective ranking measure on rule prediction ability can help to choose high-predictiveness rules thus the size of the rule set can be reduced and the efficiency of the matching algorithm can be consequently increased.

Acknowledgements

This research is fully supported by a CERG grant (CityU 147407) of The Research Grant Commission of the Hong Kong SAR government.

References

Agrawal, R., and Srikant, R. 1995. "Mining Sequential Patterns," *Proceedings of the Eleventh International Conference on Data Engineering*, pp. 3-14.

Atallah, M., Gwadera, R., and Szpankowski, W. 2004. "Detection of Significant Sets of Episodes in Event Sequences," *Proceedings of the 4th International Conference on Data Mining*, pp. 3–10.

Bettini, C., Wang, X.S., Jajodia, S., and Lin, J.L. 1998. "Discovering Frequent Event Patterns with Multiple Granularities in Time Sequences," *IEEE Transactions on Knowledge and Data Engineering* (10:2), pp 222-237.

Chatfield, C. 2001. *Time-Series Forecasting*. Chapman & Hall/CRC.

Cui, G., Wong, M.L., and Lui, H.K. 2006. "Machine Learning for Direct Marketing Response Models: Bayesian Networks with Evolutionary Programming," *Management Science* (52:4), pp 597-612.

Dunning, T. 1994. *Statistical Identification of Language*. Computing Research Laboratory, New Mexico State University.

Flanagan, A. 2004. "Nokia Context Data." from http://www.pervasive.jku.at/Research/Context_Database/index.php

Grace, R., Byrne, V.E., Bierman, D.M., Legrand, J.M., Gricourt, D., Davis, B.K., Staszewski, J.J., and Carnahan, B. 1998. "A Drowsy Driver Detection System for Heavy Vehicles," *Proceedings of the Digital Avionics Systems Conference (DASC)*, Bellevue, WA, USA, pp. I36/31-I36/38.

Han, J., and Kamber, M. 2006. *Data Mining: Concepts and Techniques*, (2 ed.). Morgan Kaufmann.

Herlocker, J.L., Konstan, J.A., Terveen, L.G., and Riedl, J.T. 2004. "Evaluating Collaborative Filtering Recommender Systems," *ACM Transactions on Information Systems (TOIS)* (22:1), pp 5-53.

Horerry, T., and Hartley, L. 2001. "Fatigue Detection Technologies for Drivers: A Review of Existing Operator-Centred System," in: *IEEE Conference on Human Interfaces in Control Rooms*. Manchester, UK: pp. 321-326.

Karimi, H.A., and Liu, X. 2003. "A Predictive Location Model for Location-Based Services," *Proceedings of the 11th ACM international symposium on Advances in geographic information systems*, New Orleans, Louisiana, USA, pp. 126-133.

Kogan, J., Nicholas, C., and Teboulle, M. 2006. *Grouping Multidimensional Data*.

Laxman, S., Sastry, P.S., and Unnikrishnan, K.P. 2007. "Discovering Frequent Generalized Episodes When Events Persist for Different Durations," *IEEE Transactions on Knowledge and Data Engineering* (19:9), pp 1188-1201.

Lee, K.F., Hon, H.W., and Reddy, R. 1990. "An Overview of the Sphinx Speech Recognition System," *IEEE Transactions on Acoustics, Speech, and Signal Processing* (38:1), pp 35-45.

Mannila, H., and Toivonen, H. 1996. "Discovering Generalized Episodes Using Minimal Occurrences," in: *Proceedings of the 2nd International Conference on Knowledge Discovery in Databases and Data Mining*. pp. 146-151.

Mannila, H., Toivonen, H., and Inkeri Verkamo, A. 1997. "Discovery of Frequent Episodes in Event Sequences," *Data Mining and Knowledge Discovery* (1:3), pp 259-289.

Mortazavi-Asl, B., Wang, J., Pinto, H., Chen, Q., and Hsu, M.C. 2004. "Mining Sequential Patterns by Pattern-Growth: The Prefixspan Approach," *IEEE Transactions on Knowledge and Data Engineering* (16:11), pp 1424-1440.

Padmanabhan, B., Zheng, Z., and Kimbrough, S. 2006. "An Empirical Analysis of the Value of Complete Information for Ecrm Models," *MIS Quarterly* (30:2), pp 247-267.

Pinto, H., Han, J., Pei, J., Wang, K., Chen, Q., and Dayal, U. 2001. "Multi-Dimensional Sequential Pattern Mining," *Proceedings of the tenth international conference on Information and knowledge management*: ACM Press New York, NY, USA, pp. 81-88.

R. Agrawal, and Srikant, R. 1994. "Fast Algorithms for Mining Association Rules in Large Databases," *20th International Conference on Very Large Data Bases*, pp. 487-499.

Rao, B., and Minakakis, L. 2003. "Evolution of Mobile Location-Based Services," *Communications of the ACM* (46:12), pp 61-65.

Schiller, J., and Voisard, A. 2004. *Location-Based Services*. San Francisco: Morgan Kaufmann.

Srikant, R., and Agrawal, R. 1996. "Mining Sequential Patterns: Generalizations and Performance Improvements," *Proceedings of the 5th International Conference on Extending Database Technology*, Avignon, France: Springer, pp. 3-17.

Su, Z., Yang, Q., Lu, Y., and Zhang, H. 2000. "Whatnext: A Prediction System for Web Requests Using N-Gram Sequence Models," in: *Proceedings of the First International Conference on Web Information Systems Engineering*. pp. 214-221.

Van Rijsbergen, C.J. 1979. *Information Retrieval*. Butterworth-Heinemann Newton, MA, USA.

Wang, J., Han, J., and Li, C. 2007. "Frequent Closed Sequence Mining without Candidate Maintenance," *IEEE Transactions on Knowledge and Data Engineering* (19:8), pp 1042-1056.

Wang, W., and Yang, J. 2005. *Mining Sequential Patterns from Large Data Sets*. Springer.

Wei, W.W.S. 1989. *Time Series Analysis: Univariate and Multivariate Methods*. Addison Wesley.

White, O., Dunning, T., Sutton, G., Adams, M., Venter, J.C., and Fields, C. 1993. "A Quality Control Algorithm for DNA Sequencing Projects," *Nucleic Acids Research* (21:16), pp 3829-3838.

Yang, Q., Li, T., and Wang, K. 2004. "Building Association-Rule Based Sequential Classifiers for Web-Document Prediction," *Data Mining and Knowledge Discovery* (8:3), pp 253-273.

Yang, Q., and Zhang, H.H. 2003. "Web-Log Mining for Predictive Web Caching," *IEEE Transactions on Knowledge and Data Engineering* (15:4), pp 1050-1053.

Yu, C.C., and Chen, Y.L. 2005. "Mining Sequential Patterns from Multidimensional Sequence Data," *IEEE Transactions on Knowledge and Data Engineering* (17:1), pp 136-140.

Zaki, M.J. 2001. "Spade: An Efficient Algorithm for Mining Frequent Sequences," *Machine Learning* (42:1), pp 31-60.

Zaki, M.J., Lesh, N., and Ogihara, M. 1998. "Planmine: Sequence Mining for Plan Failures," *4th International Conference on Knowledge Discovery and Data Mining*, pp. 369-374.