Decision Support Systems 56 (2013) 234–246

Contents lists available at ScienceDirect

Decision Support Systems

journal homepage: www.elsevier.com/locate/dss

A prediction framework based on contextual data to support Mobile Personalized Marketing

Heng Tang [a,*], Stephen Shaoyi Liao [b], Sherry Xiaoyun Sun [b]

[a] Faculty of Business Administration, University of Macao, Macao, China
[b] Department of Information Systems, City University of Hong Kong, Hong Kong, China

ARTICLE INFO

Article history:
Received 17 December 2010
Received in revised form 8 June 2013
Accepted 10 June 2013
Available online 19 June 2013

Keywords:
Multidimensional rule
Sequential rule
Activity prediction
Mobile Personalized Marketing
Data mining

ABSTRACT

Personalized marketing via mobile devices, also known as Mobile Personalized Marketing (MPM), has become an increasingly important marketing tool because the ubiquity, interactivity and localization of mobile devices offers great potential for understanding customers' preferences and quickly advertising customized products or services. A tremendous challenge in MPM is to factor a mobile user's context into the prediction of the user's preferences. This paper proposes a novel framework with a three-stage procedure to discover the correlation between contexts of mobile users and their activities for better predicting customers' preferences. Our framework helps not only to discover sequential rules from contextual data, but also to overcome a common barrier in mining contextual data, i.e. elimination of redundant rules that occur when multiple dimensions of contextual information are used in the prediction. The effectiveness of our framework is evaluated through experiments conducted on a mobile user's context dataset. The results show that our framework can effectively extract patterns from a mobile customer's context information for improving the prediction of his/her activities.

1. Introduction

Personalized Marketing (PM), also known as one-to-one marketing, is the process of delivering targeted products and services to a customer based on the customer's profile [36,41]. The main objective of PM is to identify the needs of a customer and offer products and services that appeal to that particular customer. Recently, with the dazzling proliferation of mobile commerce, personalized marketing via mobile devices (Mobile Personalized Marketing, MPM) has become an increasingly important marketing tool because the ubiquity, interactivity, and localization of mobile devices offers great potential for collecting customers' information, understanding their preferences and quickly advertising customized products [16,37,60]. Recent studies have predicted that the volume of business transactions associated with MPM will soon become the primary contributor to revenue growth in one-to-one marketing [39].

In personalized marketing, it is important to consider the contextual information, i.e. the environment where a customer is located, in order to understand the needs of the customer. Since it is possible to collect mobile device carriers' geographical positions, value-added services can be delivered via mobile devices, based on the location of a customer, which is often referred to as "Location-based Services", or LBS for short [44]. The correlations between a specific location and the actual activities of a customer can be identified by analysis of his/her short-term location log and then used to predict his/her preferences at certain locations [11,24,44]. However, location is only one aspect of a context [9]. In practice, predicting a mobile user's possible activities simply based on "location" may not achieve satisfactory accuracy in many cases. As empirical studies have shown [40], a view of a customer's activities from multiple perspectives can enhance the predictive accuracy of data-based methods of customer analysis. Thus, dimensions other than location, of contextual information, e.g., time of the day and weather, can also be useful in predicting activities of a mobile user. In order to accurately predict customer preferences, we need to take into account multiple dimensions of a customer's context. Let us look at the following example.

Example 1. When a customer is in a shopping mall, there is a 60% possibility of him/her being interested in redeeming a mobile coupon in a shop. The estimation of this possibility can significantly vary with extra contextual information. When it is a rainy weekend, the possibility that the service is preferred by the customer in a shopping mall can increase to 95%, and in other contexts the possibility can be as low as 15% because people tend to be indoors when the weather is not favorable for outdoor activities.

* Corresponding author at: Av. Padre Tomás Pereira Taipa, Macao, China. Tel.: +853 83974172; fax: +853 83978320.
E-mail addresses: hengtang@umac.mo (H. Tang), issliao@cityu.edu.hk (S.S. Liao), sherry.sun@cityu.edu.hk (S.X. Sun).

0167-9236/$ – see front matter
http://dx.doi.org/10.1016/j.dss.2013.06.004

In the example above, multiple dimensions of the contextual information provide important clues to the customer's preferences under a more specific circumstance (e.g., location, weather and time). As such, accuracy of the prediction whether a customer will likely accept an offer of a service can be improved when such multidimensional information about the customer's context is incorporated into a prediction method for personalized marketing.

The estimation of a customer's preference for a service can be regarded as a mapping from the customer, context, and service to a probability, i.e. $p = f\ (Customer, Context, Service)$. In recommender systems, the extent to which a customer prefers a service is reflected by the "User Rating" [2]. In this study, the probability of a customer preferring a service is obtained through analyzing the correlation between a sequence of contexts and the activities of a customer based on historical data. Then, for a given context, the service or product with the highest probability of being preferred can be proactively offered to the customer. The correlations between a series of contexts and activities of a customer are represented as sequential rules, often represented as "*x* leads to *y*" indicating *y* happens after *x* has happened [5]. Users' activities, notably, can also be viewed as an important type of contextual information, since they can offer valuable clues for predicting future moves. This sequential rule based solution enables the service provider to not only tailor services for customers, but also deliver the services in advance.

Example 2. A simple sequential rule is given as follows, showing the correlation between the contexts and the activities of the customer in Example 1.

{office, afternoon}, {shopping mall, night} *leads to* "redeeming a mobile coupon for the food court" (probability = 70%).

This sequential rule indicates that the customer is likely to accept a mobile coupon when going from office to shopping mall at night. Location and time are the involved dimensions of the two contexts in sequence. As this rule has a comparatively high probability, it can be used to make predictions. Then, whenever the antecedent, i.e. the contexts {office, afternoon} and {shopping mall, night}, occur again, a prediction can be made that a mobile coupon for the food court will be preferred by the customer.

Given that in practice, a huge amount of contextual information with various dimensions can be collected using mobile devices, it is a great challenge to effectively identify the sequential rules that are most useful for prediction of customer preferences. Moreover, in order to proactively address customers' needs, it is also critical to quickly identify situations where a sequential rule is applicable.

As different combinations of dimensions of a context can be used for prediction of customer preferences, accuracy of the prediction undoubtedly depends on the set of dimensions for various contexts. Rather than enhancing the predictiveness, incorporation of additional contextual dimensions sometimes results in redundancy [63], which is generally known as a phenomenon wherein "parts of knowledge are in fact corollaries of other parts of knowledge" [38].

Example 3. Seven sequential rules shown as follows are derived from the sequential rule in Example 2 but incorporate one new dimension, i.e. day of the week:

(1) {office, afternoon, Monday}, {shopping mall, night, Monday} *leads to* "redeeming a mobile coupon", and
(2) {office, afternoon, Tuesday}, {shopping mall, night, Tuesday} *leads to* "redeeming a mobile coupon".
…
(7) {office, afternoon, Sunday}, {shopping mall, night, Sunday} *leads to* "redeeming a mobile coupon".

The probabilities of these rules are found to be very close to each other. Thus, the additional dimension, day of the week, does not introduce new knowledge to the original rule.

As illustrated in Example 3, redundancy of sequential rules needs to be taken into account. It is worth noting that the number of sequential rules may increase dramatically after adding a new dimension. Thus, in order to reduce the complexity of the rule base and optimize prediction efficiency, the redundancy issue needs to be addressed.

In summary, the following problems are important and need to be addressed when mining multidimensional contextual data; these problems have motivated the work reported in this paper. First, how can we efficiently discover sequential rules that can achieve high prediction accuracy from multidimensional data? Second, how can we reduce knowledge redundancy in identified rules and, moreover, using those rules, how can a prediction be made based on the context about a customer?

In this study, we propose a data mining based framework to extract and apply sequential rules for a proactive MPM solution. This framework enables incorporation of multidimensional contextual information into sequential rule mining; a new concept, i.e. snapshot, is proposed to capture contextual information. Under our framework, the existing Apriori-like mining methods [4] can be easily applied to predict the activities of mobile users. Moreover, we propose a post-pruning method to help reduce rule redundancy, based on the multidimensional nature of contextual information. In addition, we propose an online mining algorithm that detects the situations where certain services can be delivered according to the probability of being preferred by a customer, thus enabling real-time predictions based on the extracted rules. The proposed framework follows a 3-stage process comprising rule learning, selection and matching, as summarized in Fig. 1.

The learning stage starts with analysis of contextual data to extract sequential rules. The proposed rule-learning algorithm is underpinned by the classical Apriori method [5], which generates candidate rules in a level-wise manner and then eliminates unqualified candidates using "support" as the criterion. The purpose of the rule reduction stage is to screen out rules conveying redundant dimensional knowledge from the generated rule base. This stage helps diminish the number of rules so as to optimize the efficiency of the matching process. The rule matching process monitors the ongoing context changes, evaluates the probability of a user event to occur, based on previously extracted rules, and identifies events with a high probability of being preferred.

The multidimensionality of contextual data has posed many challenges for mining useful rules. The first challenge is to consider the multidimensional setting in the mining algorithms [25]. In this paper, in order to handle the multidimensional data, we propose to take "snapshots" along a continuous dimension (such as *time*), and then identify the co-occurrence relation between the snapshots and the user's actions. With our formulation of the problem, the data mining algorithm handling single-dimension mining, i.e. WINEPI [33], is extended for multidimensional sequential rule mining. Another challenge is to alleviate the rule base complexity, for which we propose an information-entropy-based post-pruning method to identify redundant rules. Our framework can be applied in proactive MPM, and throughout the rest of the paper, we use the MPM scenario as a running example to demonstrate the efficacy of our proposed methods. This framework, however, is generalizable to many other business applications characterized by multidimensional data.

Overall, the main contributions of this paper include:

- Presenting a generic framework with detailed procedures to take into account contextual information in predicting activities of customers;

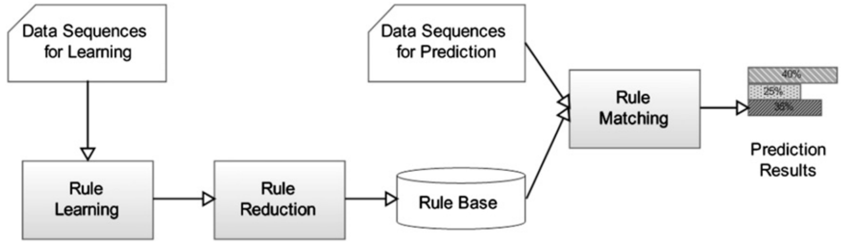

Fig. 1. A 3-stage framework (learning, selection, and matching).

- Proposing the concepts of snapshot and event to deal with multidimensionality of contextual information using the existing rule learning approaches with extensions;
- Proposing a reduction method along with a novel redundancy measure to tackle the challenge of information redundancy, which is inherently caused by the multidimensional nature of contextual information, and demonstrating the synergetic effect of combining of different reduction methods; and
- Demonstrating that contextual information other than location also matters in predicting activities of a mobile user, i.e. effectively using multidimensional contexts can outperform location-based predictions.

The remainder of the paper is structured as follows. In Section 2, we provide a brief review of the relevant literature. In Section 3, we formulate the rule-learning problem and outline the learning algorithm. The rule reduction method is introduced in Section 4. In Section 5, we describe the matching algorithm. The experiments are presented in Section 6. Section 7 summarizes the paper and outlines future research directions. The symbols and denotations used in more than one place in the paper are summarized in Appendix A.

2. Literature review

This section provides a review of related works in several areas, including sequential pattern mining, multidimensional sequence, rule reduction and association rule based prediction.

2.1. Sequential pattern mining

Data sequence, a set of data records generated sequentially, has found many applications in different business areas, such as investment, auctions and banking. Pattern mining from data sequences has aroused consistent interest in the data mining community [52]. Agrawal et al. [5] addressed the problem of discovering frequent sequential patterns, and the approach they proposed was further improved in [45]. Thereafter, research in this area gained momentum, with studies falling into two broad streams, based on the forms of input dataset. The first stream focuses on developing effective algorithms to detect sequential patterns from transactional or sequence databases. Major studies in this stream include [35,43,50,59,61]. The second stream of research focuses on mining one sequence, 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 [21]. Mining from a transactional database and mining from a sequence are different. The former aims to identify patterns from multiple sequence segments to predict the preference of a customer based on what other customers with similar preferences have done while the latter intends to discover recurring patterns from a single sequence to predict the activities of a customer. Predicting the activities of a customer is unique in that people's activities are more closely related to their personal schedule, pattern of life, places of living (home, office and entertainment, etc.), which can be very special from one individual to another.

The mining algorithm we propose in this paper attempts to deal with the second type of data format, i.e. a single sequence, which is more applicable for context-specific MPM. In this category, Mannila and Toivonen [32] use “Episode” to describe frequently recurring subsequences and propose two efficient algorithms, WINEPI and MINEPI. Many others have also focused on episode mining, including [8,10,27,33]. For example, Bettini et al. [10] address the problem of mining event structures with multiple time granularity; Laxman et al. [27] extend the episode mining approach by explicitly bringing event duration constraints into the concept of episode. The difference between episode mining techniques and ours is that the former are not directly applicable to multidimensional sequence mining.

2.2. Multidimensional sequence

The term “multidimensional” used in this paper comes originally from the multidimensional data model used for data warehousing and On-Line Analytical Processing (OLAP) [13]. The problem of discovering multidimensional sequential rules for prediction studied in this paper is a new issue. To the best of our knowledge, no related work has directly addressed this. However, the general concept of mining multidimensional sequential rules has been addressed in several studies and “dimension” is also referred to as “attribute” [47]. Yu and Chen [59] investigate the episode mining problem for a multidimensional sequence. However, the term “multidimensional” used in their work refers to multiple granularities in terms of the time dimension of occurrence of events. It is thus a concept different from the way the term is used in our study. Attempts to detect sequential patterns from a multidimensional transactional database have been made by Pinto et al. [42], in which “sequence” refers to purchase sequence segments of a certain customer. The research problem discussed in this paper is different in that our approaches aim to extract patterns from an entire sequence rather than from a database of short sequence segments.

2.3. Rule reduction

Knowledge redundancy is known as a common problem of knowledge-based systems, as it reduces maintainability and efficiency of the knowledge base [49]. In this paper, we concentrate on the redundancy problem associated with rule-based systems.

In general, a rule base with problematic rules, including redundant ones, can be validated via conducting post-analysis by either domain experts or through an automatic process. The approach of involving domain experts is known to be flexible and highly applicable. For example, Adomavicius and Tuzhilin [3] propose an expert-driven framework for validation of a given rule base. The automatic process for redundancy check normally relies on precisely defining the measure of redundancy. For instance, Zaki [62] proposes the concept of “Closed Itemset” based on Formal Concept Analysis, and proves that a closed itemset can be used to capture all information about a conventional frequent itemset. Moreover, a redundant rule is defined to be a super rule with the same frequency and confidence as its sub-rules [62]. Similar redundancy definitions have also been proposed in some other studies [31,55]. In Ashrafi et al.'s approach [6,7], given a rule r, if r's sub-rules with a higher confidence are found in the rule base, then the rule r should be regarded as redundant. [14,15] define a δ-tolerance association rule (δ-TAR) mining task. The itemset identified using the δ-TAR method only includes items with dramatic frequency changes, and rules excluded in the δ-TAR are viewed as redundant.

Despite these prior efforts on redundancy reduction, to the best of our knowledge, few works have addressed this issue in multidimensional settings. Specifically, our research investigates the redundancy problem introduced by the multidimensionality of sequential rules.

2.4. Association rule based prediction

The proposed matching approach is based on the n-gram method in which an n-gram refers to a succession of n items from a given sequence [28]. n-gram has been widely used in statistical natural language processing [34] and genetic sequence analysis [54]. Many works attempt to build n-gram models using association mining techniques. For example, Yang et al. [56] attempt to discover association rules from web user sessions to estimate conditional probability of accessing web documents for caching optimization. Similarly, the WhatNext system developed in [46] generates simple n-grams through association mining. We extend the n-gram based prediction by borrowing the concept of "alignment" from string comparison algorithms [16], such that "gaps" in the input sequence are allowed in matching. In addition, time constraints are also taken into account.

The framework proposed in this paper differs from time series forecasting [12,53] in two regards. First, forecasting in the time series area mainly studies the prediction problem with continuous data whereas the problem to be solved in this paper is a prediction of categorical data across multiple dimensions. Second, to apply time series forecasting approaches, an appropriate multivariate model needs to be determined in advance [12]. In contrast, data mining approaches such as ours are essentially problem-oriented, aiming at exploring a large amount of data without many restrictions associated with the preset model [12].

3. Learning rules from multidimensional data sequence

We will first introduce the relevant definitions and formulate the problem of learning sequential rules in Section 3.1. The learning algorithm will be described in Section 3.2.

3.1. Problem statement

The input contextual data are considered as a sequence of data items with multiple dimensions. We use the term "snapshot" to describe a mapping from time domain to context, in order that we can apply conventional rule mining methods in a multidimensional setting for rule extraction.

Definition 1. Snapshot

Given a function mapping $S: T \rightarrow \mathcal{D}$ from a discrete time domain $T = \{t_1, t_2, \ldots, t_n\}$ to an $(m+1)$-dimensional state space $\mathcal{D} = D_1 \times D_2 \times \ldots \times D_{m+1}$, the mapping $S(t) = \{v_1, \ldots, v_{m+1}\}$, $t \in T$ is called a *snapshot*, where v_j, $j = 1, \ldots, m+1$ indicates the state in the j-th dimension.

In particular, D_1, D_2, …, D_m are the dimensions of context called *contextual dimensions*, whereas D_{m+1} is the dimension of customer action, referred to as *actional dimension*. In this research, domains in all dimensions are required to be categorical in order to form a discrete state space, therefore, a discretization method needs to be applied in advance in the case of continuous state space.

A snapshot describes the contextual state in every dimension. In practice, though, the recurrence of context can only be found in some dimensions, while the states in other dimensions are random. For example, a user shows up in office in most weekdays, but during which the temperature could be nearly random. We hence define the concept of event, which can be viewed as the "template" of state vectors, as follows.

Definition 2. Event

An *event* e is a subset of the state space, denoted e $\mathcal{D}$.

For instance, $e_1 = \{(v_1, \ldots, v_{m+1}) | v_1 =$ "*Street*", $v_2 =$ "*Morning*", $v_3 =$ "*Raining*"$\}$ is an event characterized by the context of location, time, and weather, and e_1. $dim = \{D_1, D_2, D_3\}$ is called the (restricted) *dimension set* of e_1. As another example, the event $e_2 = \{(v_1, \ldots, v_{m+1}) | v_{m+1} =$ "*Purchase*"$\}$ describes a purchase action of a customer, of which e_2. $dim = \{D_{m+1}\}$. For simplicity, the above two events are also written as $e_1 = \{v_1=$"*Street*", $v_2=$"*Morning*", $v_3=$ "*Raining*"$\}$ and $e_2 = \{v_{m+1} =$ "*Purchase*"$\}$, respectively.

A customer's context recorded in a stream of state vectors forms a sequence, which is defined as follows.

Definition 3. Sequence

Given $T = \{t_1, t_2 \ldots, t_n\}$, a *sequence* is a list $seq_T = \{S(t), t \in T\}$ whose elements are ordered ascendingly by t. Assuming $t_1 < t_2 < \ldots < t_n$, the overall *time span* of seq_T is $span(seq_T) = t_n - t_1$.

We say an event e *occurs* in sequence seq_T at time t, denoted $e \dashv_t seq_T$, if there exists a time point t such that $S(t) \in e$. For brevity it is also denoted $e \dashv seq_T$ in the case that occurring time does not matter. For example, given $S(t_1) = \{$" *Street* ", " *Morning* ", " *Raining* ", " *Purchase* "$\}$ defined on a 4-dimensional state space $\mathcal{D}$ and $e_2 = \{v_4 =$ " *Purchase* "$\}$, since $S(t_1) \in e_2$, we say e_2 occurs in seq_T, denoted $e_2 \dashv seq_T$. Note that there can be, in a general case, more than one event occurring at the same time point.

To simplify the notation in the paper, we use a tuple (e, t) to represent an event occurring in a sequence at a specific time point, where e is the event label taking value from a finite alphabet and t is the occurring time of e. Thus a sequence could be denoted by, for example, $seq_{T1} = <$(x,11), (x,12), (x,13), (w,14), (y,15), (x,16), (z,17), (x,18)$>$, where w, x, y and z are events occurring in seq_T.

A rule is a type of pattern representing the hidden correlation among events in a sequence defined as follows.

Definition 4. Rule

A *rule* is a list of events denoted by $r = \langle e_1, e_2, \ldots, e_l \rangle$, where $e_1, \ldots, e_l$ $\mathcal{D}$.

Having introduced the notion of a rule, we are now able to define what is meant by its occurrence in a sequence so as to formulate its significance. Shortly, an occurrence of a rule r is considered as a series of time points recording the occurring time of the corresponding events of r in a sequence. For practical purposes, we should allow some time gap between the adjacent events in an occurrence, hence two thresholds g and w are introduced into our formulation: g (or *gap*) is the maximum allowed time difference between the occurring time of any two neighboring event types, while w (or *width*) is the maximum allowed time difference between the occurring time of the first and the last event types. The rigorous definition is provided as follows.

Definition 5. Occurrence

Given a sequence seq_T and a rule $r = \langle e_1, e_2, \ldots, e_l\rangle$, a list of time points $o_r = \langle t_1^o, t_2^o, \ldots, t_l^o\rangle$, $t_i^o \in T$, is called a (g,w)-occurrence of r in seq_T, if and only if: (1) $e_i \dashv_{t_i^o} seq_T$ for $\forall\ i = 1 \ldots l$, (2) $t_1^o \le t_2^o \le \ldots \le t_l^o$, (3) $t_{i+1}^o - t_i^o \le g$ for $\forall\ i = 1 \ldots l-1$, and (4) $t_l^o - t_1^o \le w$. The set of all (g,w)-occurrences of r in seq_T is denoted $Occr(r, seq_T, g, w) = \{o_r | o_r = \langle t_1^o, t_2^o, \ldots, t_l^o\rangle$ *is a* (g,w)*-occurrence of* r *in* seq_T for $\forall\ t_1^o, t_2^o, \ldots, t_l^o \in T\}$.

For instance, given seq_T in the previous example, if we consider a rule $r = \langle x,y,z\rangle$ and thresholds $g = 2$ and $w = 5$, then $< 13, 15, 17 > \in Occr(r, seq_{T1}, 2, 5)$ is an occurrence of r, while $< 12, 15, 17 > \notin Occr(r, seq_{T1}, 2, 5)$, since it violates constraint (3) in the above definition. Likewise, $< 11, 15, 17 > \notin Occr(r, seq_{T1}, 2, 5)$ as it violates both constraints (3) and (4).

We adopt the time window concept introduced in [33] to measure the significance of a rule. A window win is a half-open time interval in the span of seq_T, denoted $win_T = [t_i, t_j)$, if $t_j > t_1$ and $t_i < t_n$. The *width* of the window is $|win_T| = t_j - t_i$. Let $W(seq_T, w)$ be the set of all windows with width w in the span of seq_T, where w is the aforementioned time threshold, i.e., $W(seq_T, w) = \{win_T = [t_i, t_j) |\ \ |win_T| = w\ for\ \forall\ t_i, t_j \in T\}$. We assume, without loss of generality, that time points in T are consecutive integers, $W(seq_T, w)$ thus has the cardinality $||W(seq_T, w)|| = span(seq_T) + w - 1$. For example, consider the same seq_{T1}, we have $||W(seq_{T1}, 5)|| = 7 + 5 - 1 = 11$, where $W(seq_{T1}, 5) = \{[7,12), [8,13), [9,14), [10,15), [11,16), [12,17), [13,18), [14,19), [15,20), [16,21), [17,22)\}$. Notice that windows defined on seq_{T1} can extend out of its span.

The number of windows containing a rule's occurrences can be used to gauge its frequency. A window $win = [t_i, t_j)$ *contains* an occurrence $o_r = \langle t_1^o, t_2^o, \ldots, t_l^o\rangle$ if and only if $t_i \le t_1^o$ and $t_l^o < t_j$, denoted $o_r \sqsubset win$. Of all windows in $W(seq_T, w)$, those containing any occurrence of r is defined as $W_r(seq_T, g, w) = \{win | win\ \ W(seq_T, w)$ and $o_r \sqsubset win$ and $o_r \in Occr(r, seq_T, g, w)\}$.

Let rule $r = \langle x,y,z\rangle$, according to the definition of W_r, we have $W_r(seq_{T1}, 2, 5) = \{[13,18)\}$. The cardinality of the set W_r can be used to measure the frequency of rule r, thereby the frequency of $r = \langle x,y,z\rangle$, subject to g and w, is calculated by $||W_r(seq_{T1}, 2, 5)|| = 1$.

We adopt the classical support-confidence framework [4,33] to quantify the significance of a rule, in which for a rule r: $X \to Y$, $supp(r) = P(XY)$ and $conf(r) = P(XY) / P(X)$ are two key measures. The support of a sequential rule $r = \langle e_1, e_2, \ldots, e_l\rangle$ is defined as the ratio $supp(r, seq_T, g, w) = \frac{W_r(seq_T, g, w)}{W(seq_T, w)}$, implying the probability that the rule may occur in any window. Note that, the numerator $||W_r(seq_T, g, w)||$ only counts the number of windows containing r's occurrences, regardless of how many of them are found in the same window. Accordingly, the confidence of rule r is thus the support of the entire rule r over that of the antecedent, namely, $conf(r, seq_T, g, w) = \frac{supp(r, seq_T, g, w)}{supp(\langle e_1, e_2, \ldots, e_{l-1}\rangle, seq_T, g, w)}$.

All in all, the problem of mining a sequential rule with length l from seq_T, subject to time thresholds g and w, is to identify all rules satisfying the following two conditions:

1)

$$supp(r, seq_T, g, w) \ge \min\_supp \tag{1}$$

2)

$$conf(r, seq_T, g, w) \ge \min\_conf. \tag{2}$$

The above conditions define the minimum required support and confidence of qualified rules. In addition, for any rule r, we say that r is a *frequent* rule if and only if condition (1) is satisfied.

3.2. The mining algorithm

Since the ultimate goal of rule mining is to anticipate the occurrence of customers' actions (rather than that of other contextual events), we only need to consider rules whose end event is actional. Such kind of rule is called an *actional rule*, denoted $r.tail.dim = \{D_{m+1}\}$, where D_{m+1} is a dimension of customer action. An actional rule can also be written in the implication form as $r : \langle e_1, e_2, \ldots, e_{l-1}\rangle \to e_l$. Note that the definition of rule so far does not prohibit distinct dimensions in different events in a rule. Such flexibility allows to express some general rules like <"weather is good", "called shopping buddy", "shopping">. However, this flexibility in dimensions imposes great challenge to the rule learning phase of an MPM system. Because in addition to the massive searching space to be dealt with when enumerating nearby events and growing candidate rules, meanwhile we have to consider a vast number of dimension combinations for each event, leading to a synergetically combinatorial explosion in searching space. Hence, in this paper, we only consider a restricted version of actional rule whose antecedent is with the identical dimension set only.

Definition 6. Sequential rule

A rule $r : \langle e_1, e_2, \ldots, e_{l-1}\rangle \to e_l$ is called a *sequential rule* if it is an actional rule and $e_1.\ dim = e_2.\ dim = \ldots = e_{l-1}.\ dim$. The common dimension set of the antecedent is denoted $r.dim$. Rule r is said to be k-dimensional if $||r.\ dim|| = k$.

The problem of mining sequential rules addressed in this paper is similar to the Episode Mining problem studied in [8,10,27,32,33]. In this paper, we modify the WINEPI algorithm proposed in [33] by incorporating thresholds g and w in order to enhance the pruning process for reducing the search space.

The algorithm is outlined in Fig. 2 (Algorithm 1). It adopts a level-wise strategy used in the Apriori algorithm in that the rules with length k are generated in the k-th iteration (level). Initially, in the level 1 procedure, rules with length 1 are counted and stored (line 1 in Fig. 2). Each new rule in the level k candidate set is generated by concatenating two *concatenatable* rules with length $k-1$ (line 5 in Fig. 2). Hence, the length of the rule will grow by one in each iteration. Concatenation is the basic operation in the rule-growing process formulated as follows.

Definition 7. Concatenation of overlapping rules

Given two rules $r_1 = \langle e_{1,1}, e_{1,2}, \ldots, e_{1,l}\rangle$ and $r_2 = \langle e_{2,1}, e_{2,2}, \ldots, e_{2,l}\rangle$, where $l \ge 2$ is their length. If for any $i = 2 \ldots l$ we have $e_{1,i} = e_{2,i-1}$, we say that r_1 and r_2 are *concatenatable*. The outcome of concatenation is $concat(r_1, r_2) = \langle e_{1,1}, e_{1,2}, \ldots, e_{1,l}, e_{2,l}\rangle$ which has the length $l + 1$. r_1 and r_2 are referred to as the left and right rules of $concat(r_1, r_2)$, respectively.

The concatenation operation generates a new rule with length $l + 1$ by "stitching" together two rules with length l and hence prevents extensive combinatorial explosion. This operation conforms to the downward-closure property [4,33], it therefore exhausts all frequent rules with length $l + 1$. As a special case, initially two events (i.e., rules with length 1) can be directly concatenated together to form a new rule with length 2. The concatenation operation is performed iteratively to generate all possible rules and stores them in the candidate set (line 5 in Fig. 2).

The algorithm in Fig. 2 is explained as follows. The dimension set of the sequential rules to be extracted, denoted *dimset*, needs to be specified in advance. Given a data sequence seq_T, the algorithm extracts rules with maximum length l and dimension set *dimset*. The aforementioned thresholds *min_supp*, *min_conf*, g, and w are parameters. $Cand_i$ is used to temporarily store candidate rules with length i that are not yet pruned. In line 1 $Cand_1$ is initialized with occurred

1 Initialize $Cand_1 = \{r \mid r = \langle e_j \rangle \wedge e_j \dashv seq_T \wedge (e_j.dim = dimset \vee e_j.dim = \{D_{m+1}\})\}$
2 FOR $i = 2$ TO l-1 DO
3 Compute $Freq_i = \{r \mid r \in Cand_i \wedge supp(r, seq_T, g, w) \geq min\_supp\}$
4 Compute $Freq_i = prune(Freq_i, g, w)$
5 Compute $Cand_{i+1} = \{r = concat(r_1, r_2) \mid r_1, r_2 \in Freq_i$ are concatenatable $\wedge r_1.tail.dim \neq \{D_{m+1}\}\}$
6 END FOR
7 Compute $Freq_l = \{r \mid r \in Cand_l \wedge supp(r, seq_T, g, w) \geq min\_supp\}$
8 Compute $Freq_l = prune(Freq_l, g, w)$
9 Compute the confidence for all frequent rules in $R = \{r \in Freq_i \mid r.tail.dim = \{D_{m+1}\}, i = 2 \ldots l\}$,
then output $r \in R$ if $conf(r, seq_T, g, w) \geq \min\_conf$

Fig. 2. Algorithm 1—Extracting rules from a data sequence.

events that are either actional or with the dimension set *dimset*. In the loop between lines 2 and 6, $Freq_i$ is used to store frequent rules with length i, and the pruning method (denoted *prune*() in line 4) is then applied to eliminate rules violating either of the thresholds g or w. Each concatenatable rule pair is then merged to generate a new candidate with length $i + 1$. The above procedure is iterated until all frequent rules with length l are identified. In addition, since the goal is to extract sequential rules in which actional events do not appear in the antecedent, rules ending with actional event thus will not be selected as the left rule when conducting concatenation (line 5, denoted $r_1.tail.dim \neq \{D_{m+1}\}$). This restriction on rule pair selection is another effective pruning in the algorithm. Subsequently, lines 7 and 8 compute $Freq_1$ so as to finalize the loop.

Ultimately, only sequential rules with the minimum length of 2 are chosen (line 9 in Fig. 2). Since the support of all rules have been calculated and stored, computing their confidence is straightforward. Rules with confidence greater than *min_conf* are considered as valid rules to output.

4. Rule reduction

We now consider the redundancy problem motivated in the examples in Section 1. By specifying different dimension sets, the rule learning algorithm can identify sequential rules with various configurations of contextual dimensions. Rules with high dimensionality could be of great interest because they offer more specific and accurate description of context, they may nevertheless also be considered redundant if they do not carry additional knowledge than their lower dimensional variations. Traditionally, redundant dimensions can be spotted by various dimension (feature) selection methods, such as applying heuristics [19,26], so that they can be excluded from rule extraction in the first place. However, conventional dimension selection methods consider redundancy problem from the dimension level, but ignore the fact that a "valueless" contextual dimension in terms of some rules could possibly be valuable for some others. To this end, this study focuses on the redundancy problem on rule level.

In the research of association rule mining, a number of criteria are proposed to determine whether a rule $r_1 : XY \rightarrow e$ is redundant with regard to its "closure" $r_2 : X \rightarrow e$, where X and Y are items. The most representative criteria fall into two categories, that is: Association rule r_1 is redundant in terms of r_2 if and only if:

(1) $conf(r_1) \leq conf(r_2)$ [6,7], or
(2) $conf(r_1) = conf(r_2)$ and $supp(r_1) \leq supp(r_2)$ [31,55,62]

The above criteria can be straightforwardly applied in multi-dimensional settings if states in different dimensions are considered as items. In other words, to determine whether a state $Y \in D_k$ is necessary for a given sequential rule r_2, we can simply compare the confidence and support between r_2 and its specialization on Y, i.e. r_1. These criteria consider the relationship between r_2 and r_1 but ignore the overall effect of dimension D_k where Y comes from. As shown in Example 3 in Section 1, the set of sequential rules derived using the newly added dimension "Day of the week" together, rather than as individual sequential rules, should be considered in order to infer whether the new dimension brings additional knowledge into the rule base. As a remedy, we have developed a new measure based on the concept of "specialization" defined as below. Note that since the consequent of a sequential rule is single dimensional, only the antecedent part needs to be examined. Hence for convenience, we here assume that the length of a sequential rule is $l + 1$, such that l is the length of the antecedent. For simplicity of writing, the term "rule" specifically refers to "sequential rule" hereafter in this paper.

Definition 8. Specialization

Let $r : \langle e_1, e_2, \ldots, e_l \rangle \rightarrow e_{l+1}$ be a sequential rule with dimension set $r.dim = \{D_1, D_2, \ldots D_{m-1}\}$ where each $e_i = \{v_1 = p_{i1}, v_2 = p_{i2}, \ldots, v_{m-1} = p_{i(m-1)}\}$, $i = 1 \ldots l$, is an event with $m - 1$ restricted dimensions. Given $r' : \langle e_1', e_2', \ldots, e_l' \rangle \rightarrow e_{l+1}$, a rule with $r'.dim = \{D_1, D_2, \ldots, D_m\}$, where each $e_i' = \{v_1 = q_{i1}, v_2 = q_{i2}, \ldots, v_m = q_{im}\}$, $i = 1 \ldots l$ is an event with m restricted dimensions. If $p_{ij} = q_{ij}$ for any $i = 1 \ldots l$ and $j = 1 \ldots m - 1$, we say that r' is a *specialization* of r on dimension D_m. The set of all possible specializations of r on D_m is called the *specialization set* of r on D_m, denoted $spec(r, D_m)$. Or, conversely, rule r is called the *generalization* of $spec(r, D_m)$.

The level of redundancy of $spec(r, D_m)$ can be gauged by the uniform extent of frequency distribution of the rules in $spec(r, D_m)$. Information entropy-based measure, which is widely used to quantify the diversity of probability distribution and information amount [30], is adopted in this paper.

The entropy of a specialization set $spec(r, D_m)$ is the summation of two parts, i.e. the entropy of frequent specializations, which can be directly calculated, and the entropy of infrequent specializations, which is unavailable because infrequent rules are pruned in the rule-learning phase. Suppose that with the inclusion of dimension D_m, the set of m-dimensional frequent rules with respect to r is denoted $R(r, D_m) = \{r_j \mid r_j \in spec(r, D_m)$ and r_j is frequent$\}$, and thus the information amount of frequent specializations can be calculated by

$$I_{freq}(r, D_m) = -\sum_{r_j \in R(r, D_m)} \frac{supp(r_j)}{supp(r)} \cdot \log\left(\frac{supp(r_j)}{supp(r)}\right). \quad (4)$$

On the other hand, based on the assumption of the Principle of Indifference [20,23], we assume that the remaining infrequent specializations are equally probable. Thus the probability of each infrequent specialization is the average of the remaining possibility, which can be calculated by $p_{inf} = \left(1 - \sum_{r_j \in R(r,D_m)} \frac{supp(r_j)}{supp(r)}\right) / (spec(r,D_m) - R(r,D_m))$, where the denominator is the estimated number of infrequent specializations of r. Because $spec(r,D_m)$ is the set of all combinations of m-dimensional specializations with length l, we can calculate $||spec(r,D_m)|| = ||dom(D_m)||^l$, where $dom(D_m)$ is the value domain of D_m.

Therefore, the total information amount of all infrequent specializations can be estimated by

$$I_{inf}(r,D_m) = -p_{inf} \cdot \log p_{inf} \cdot (||spec(r,D_m)|| - ||R(r,D_m)||). \tag{5}$$

Using bounds [0, log(||$spec(r,D_m)$||)], we normalize the entropy-based redundancy degree into [0,1], which is the ratio of the overall entropy of the specializations over the upper bound, that is,

$$Redun(r,D_m) = \frac{I_{freq}(r,D_m) + I_{inf}(r,D_m)}{\log(||spec(r,D_m)||)}. \tag{6}$$

A large value of the above measure (i.e., close to 1) is undesirable because it implies that $spec(r,D_m)$ conveys little additional knowledge than that implied in r.

We use the scenario in the three examples discussed in Section 1 as an illustration of the redundancy calculation in Eqs. (4), (5), and (6). To avoid lengthy calculation, we start from a one-dimensional frequent rule r with length 2 as $\langle\{v_1 = \text{“ office ”}\}\rangle \to e$, and assume that the support of r is: $supp(r) = 0.2$.

Suppose the frequent specializations on the dimension “Day of the week” are the following:

$$supp(\langle\{v_1 = \text{“office”}, v_2 = \text{“Sat”}\}\rangle \to e) = 0.08$$
$$supp(\langle\{v_1 = \text{“office”}, v_2 = \text{“Sun”}\}\rangle \to e) = 0.09$$
$$supp(\langle\{v_1 = \text{“office”}, v_2 = \text{“Mon”}\}\rangle \to e) = 0.02.$$

Thus, we have $|spec(r,C_m)| = 7^1 = 7$ (i.e., number of days in a week), and

$$P_{inf} = \left(1 - \left(\frac{0.08}{0.2} + \frac{0.09}{0.2} + \frac{0.02}{0.2}\right)\right)/(7-3) = 0.0125$$
$$I_{inf} = -0.0125 \cdot \log 0.0125 \cdot (7-3) = 0.316$$
$$I_{freq} = -\left(\frac{0.08}{0.2}\log\frac{0.08}{0.2}\right) - \left(\frac{0.09}{0.2}\log\frac{0.09}{0.2}\right) - \left(\frac{0.02}{0.2}\log\frac{0.02}{0.2}\right) = 1.379.$$

Then the redundancy of the dimension “Day of the week” with regard to r is calculated as: $Redun(r,D_m) = (0.316 + 1.379)/\log 7 \approx 0.604$.

From the calculation above, we find that the new dimension “Day of the week” carries extra knowledge for the given rule r because its redundancy level is significantly smaller than 1 (based on our trial-and-error simulations, when $dom(D_m)$ is not a large set, *Redun* value greater than 0.9 could be considered significantly redundant). In fact, all supports given above have already intuitively shown the sequential occurrence of Office and actional event e is significantly more frequent on 2 days (Sun and Sat) than on other week days. Additionally, a byproduct of the above process is the generalization of context in the conceptual hierarchy, i.e. some values in a contextual dimension could be generalized (e.g., Monday… Friday to Weekdays) if they result in high information amount. This topic is out of the scope of the current paper and will be discussed in our other works.

Based on the definition of the redundancy measure, the algorithm for rule reduction is sketched in Fig. 3 as follows. Various rules with different dimension sets are stored in the original rule base. First, each rule r and its specializations in the original rule base are

Method:

1 $RB_r = \varnothing$
2 FOR EACH $r \in RB$
3 FOR EACH $D_m \in D \setminus (r.\text{dim})$
4 IF $Redun(r, D_m) \le \theta$ THEN
5 $RB_r = RB_r \cup R(r, D_m)$
6 ELSE
7 $RB_r = RB_r \cup \{r\}$
8 END IF
9 END FOR
10 END FOR
11 Output RB_r

Fig. 3. Algorithm 2—Reducing a rule base.

examined (line 3), where $D \setminus (r.\text{dim})$ is the set of dimensions not involved in rule r. If the redundancy level of the specialization being tested does not exceed a given threshold, frequent rules in this specialization set are added into RB_r (line 5 in Fig. 3); otherwise rule r is incorporated while its specialization set is discarded (line 7 in Fig. 3). The threshold θ in line 4 can be determined by trial-and-error through simulations.

5. Generating prediction by rule matching

In order to predict an actional event, we need to match the data stream with previously identified rules relevant to the actional event. In this study, we have developed a generic approach to allow an inconsecutive matching while factoring the time thresholds (g and w). This approach is underpinned by the n-gram prediction mechanism originally used for sequence comparison and has been applied to predict Web requests on the basis of the historical “access patterns” [46,58]. In order to allow neighboring events in a rule to find their corresponding occurrences inconsecutively in the incoming sequence, approximate matching [17] is implemented in the matching process. The rule matching process is sketched in Algorithm 3 in Fig. 4, which is an infinite procedure that keeps monitoring and processing the next incoming event. Suppose that *RB* is the rule base with k rules, in which all rules are with sufficiently high support and confidence. Three tables are used to maintain the current matching status of each rule: *matched* [1…k] maintains the last matched event of each rule, *time_last*[1…k] maintains the time point of the last matched event of each rule, and *time_first*[1…k] records the time point of the first matched event of each rule.

Assume that $r: \langle e_1, e_2, \ldots, e_l\rangle \to e_{l+1}$ is the rule to be compared with, and $r[k]$ is used to denote the k-th event e_k in rule r. Given a matched portion $\tilde{r} = \langle e_1, e_2, \ldots, e_i\rangle, i = 1\ldots l$ and an incoming snapshot $S(t)$, if $S(t) \in e_{i+1}$, the current input state $S(t)$ is considered matching $\tilde{r}$. Consequently, the conditional probability $p(e|\langle e_1, e_2, \ldots, e_{i+1}\rangle)$ is examined: if it is no less than the predefined threshold φ, the entire rule r is considered a “confident” match hence its corresponding actional event e can be triggered (line 7–9 in Fig. 4). Otherwise, we record the current match and wait for the next incoming state (line 12 in Fig. 4), since a longer matching is expected to achieve higher confidence [56]. Whenever either of the time thresholds g or w is violated, the current matching with rule r is considered a failure (line 4 in Fig. 4), we then can start over again by resetting the matching status maintained in the three tables and wait for the new incoming state (line 5 in Fig. 4). In particular, owing to the level-wise nature of the rule extraction algorithm during the learning stage, the conditional probabilities associated with all prefixes of a rule are readily available thus there is no need to calculate during

Input: Rule Base RB ($\|RB\|=k$), thresholds g and w, confidence threshold φ

Output: Predicted actional event e and its corresponding probability $p(e)$

Method:

1 Initialize: *matched*[1..*k*] = 0; *time_last*[1..*k*] = 0; *time_first*[1..*k*] = 0
2 Repeat: Wait for next update snapshot $S(t)$ where t is the time point
3 FOR EACH rule $r_i \in RB$, where $r_i : \langle e_1, e_2, \ldots, e_l \rangle \rightarrow e_{l+1}$
4   IF t – *time_last*[*i*] >*g* OR t – *time_first*[*i*] >*w* THEN
5   *matched*[*i*] = 0; *time_first*[*i*] = 0;
6   ELSE
7   IF $S(t) \in r_i\left[matched[i]+1\right]$ THEN
8   IF $p\left(e_{l+1} \mid \langle e_1, e_2, \ldots, e_{i+1} \rangle\right) \geq \varphi$ THEN
9   Output e_{l+1};
10   *matched*[*i*] = 0; *time_first*[*i*] = 0; *time_last*[i] = 0
11   ELSE
12   *matched*[*i*] ++; *time_last*[*i*] = *t*
16   IF *matched*[*i*] = 1 THEN *time_first*[*i*] = *t*
17 GOTO Repeat:

Fig. 4. Algorithm 3—Matching input sequence to generate prediction.

the matching process. For each new state, the algorithm scans all rules in the rule base and attempts to find a match from the current matching position of each rule stored in the table *matched*.

Note that multiple rules can be applicable simultaneously since more than one rule can have a conditional probability no less than the threshold φ in an iteration of scan. While the algorithm in Fig. 4 only chooses the first matched rule (line 9 in Fig. 4), an extension can be easily made to adopt different criteria, such as the shortest rule, the largest probability, and a predefined profit/cost matrix [1,57], to determine the most applicable rules.

For real-time MPM applications, such on-line processing method serially handling the input piece-by-piece as depicted in Algorithm 3 is preferable. In traditional string comparison algorithms, inconsecutively comparing elements in two strings can be solved by using Dynamic Programming [17] within the time complexity O(MN), where M and N are lengths of the two strings. Algorithm 3 attempts to align an input sequence with a fixed rule incrementally, hence the maximum time complexity is reduced to O(N) where N is the length of input sequence. As such, the time complexity of comparing one incoming state with a rule becomes constant, and comparing a state with all rules in a rule base has complexity $O(\|RB\|)$ where $\|RB\|$ is the size of the rule base RB.

6. Evaluation

To validate the methods proposed for extracting, selecting and ranking sequential rules, we conduct experiments on a context database referred to as "Nokia Context Data" [18]. The dataset consists of a sequence of contextual data. In addition to validating the framework proposed in this study, the dataset is used as a manifestation of the MPM scenario mentioned in Section 1. The evaluation encompasses two experiments:

Experiment I. In order to examine the ability of the proposed framework to identify effective multi-dimensional rules, we apply the learning and matching algorithms to the "Nokia Context Data" to generate rule bases and make predictions. The experiment shows that by taking additional contextual dimensions into consideration, the yielded rule bases may outperform the location-only rule base in predictiveness. However, the actual performance results depend largely upon contextual dimensions that have been used.

Experiment II. We apply several rule-level reduction methods to a rule base produced in Experiment I and compare the newly generated rule bases with the original one. To examine how the predictiveness of a rule base is affected after the reduction, we apply the matching algorithm to the reduced rule bases and compare their prediction performance with that of the original one.

The paper below will introduce the dataset and the adopted prediction measures, the experiment setup and evaluation results will also be reported. In this paper we use bar charts instead of lines when reporting the performance since many series are close in value.

6.1. Data description and measures

The Nokia context dataset [18] consists of a set of feature files for 43 different recording sessions. In each session, the same user carrying a number of devices is going from home to the workplace or vice versa, during which he may choose different means of transportation. Portable sensors were used to record the carrier's contextual information, such as atmospheric pressure and temperature, etc. GSM positioning technology was used to locate the user's current geographical position. There are a total of 15 dimensions in this dataset, of which 14 are contextual dimensions and one records actions, including interactions with mobile phone such as calls, short messages and Web pages accessed. The interactions in this dataset are viewed as the actional events in our experiments. The summary of the dataset with exemplar dimensions is presented in Table 1.

The measures of prediction effectiveness used in this research are "precision", "recall" and F1, which have been adopted widely in previous studies [22,56]. Precision represents the probability that a

Table 1
Data samples from the Nokia context dataset.

Record type	Exemplar dimensions	Value domains/format	Example
Context	v_1: Location	"Area Code, Cell ID"	"1,3"
	v_2: Day name	1 ~ 7 (Mon,…,Sun)	1
	v_3: Day period	1 ~ 4 (Night, Morning, Afternoon, Evening)	1
v_4: Actional Event	Launch an application	a,(0–31)	"a,2"
	Access a Web page	b,(0–13)	"b,13"
	Initiate communication	c,(0–4)	"c,3"

predicted actional event actually occurs, whereas recall represents the probability that the actional events will be predicted. Let *TP* be the number of actually occurred actional events that were predicted, *FP* be the number of actional events that were predicted but did not actually occur, and *FN* be the number of actional events that were not predicted but actually occurred; thus, $Precision = TP / (TP + FP)$, and $Recall = TP / (TP + FN)$. Another widely used measure that combines precision and recall into a single metric is F1, which can be calculated by $F1 = 2 * Precision * Recall / (Precision + Recall)$ [48]. In all experiments, 6-fold cross-validation is adopted.

6.2. Experiment I

We employ the rule learning algorithm on various dimension sets. The identified rules are denoted in the format of, for example, $\{v_1 = \text{“1, 1”}, v_2 = 1, v_3 = 4\}\{v_1 = \text{“1, 3”}, v_2 = 1, v_3 = 4\}\{v_1 = \text{“1, 4”}, v_2 = 1, v_3 = 4\} \rightarrow \{v_4 = \text{“}c\text{, 1”}\}$, in which $\{v_4 = \text{“}c\text{, 1”}\}$ is an actional event meaning "making a phone call to a certain number". This rule implies that if on Monday evening the mobile device carrier visits three places "1,1","1,3" and "1,4" in turns, it is very likely that s/he will call a number afterwards (in the same time window). Using the matching algorithm, prediction of an actional event (i.e. call a number) is done by comparing the sequence of context that has been received with the extracted rules. As another example, extracted rule $\{v_1 = \text{“3, 46”}, v_3 = 2\} \rightarrow \{v_4 = \text{“}a\text{, 12”}\}$ implies that if the device carrier visits location "3,46" in the morning, s/he tends to launch a certain application afterwards, denoted "a,12".

The setting of thresholds is specific to different applications and datasets. The two time thresholds in the learning process of our experiments are set to $g = 36{,}000$ s and $w = 70{,}000$ s (approximately 10 and 20 h, respectively), such that we will not miss the co-occurrence of two neighboring events occurring in different periods on a single day (e.g., morning, afternoon, etc.). Notice that the effect of window size on the performance of the rule mining algorithm has been discussed in [33]. In order to compare the results of multi-context prediction and the location-based prediction, we have conducted this experiment in the following three steps.

Step 1: we have compared predictiveness of different rule bases, each of which is generated using one contextual dimension available in the dataset. It has been confirmed that the dimension of location is indeed an effective dimension, which can be used to generate a single-dimensional rule base with the best precision, recall and F1 among all available contextual dimensions.

Step 2: since location is the context adopted extensively by various LBS applications [44], location-based prediction is used as the baseline for comparison. We thus need to examine whether the use of multi-dimensional contextual information does achieve better predictiveness than the baseline. Specifically, we have compared the performance of rule bases generated by "Location" (denoted "L"), "Location + Day of a Week" (denoted "LD"), "Location + Period of a Day" (denoted "LP"), and the mix of all

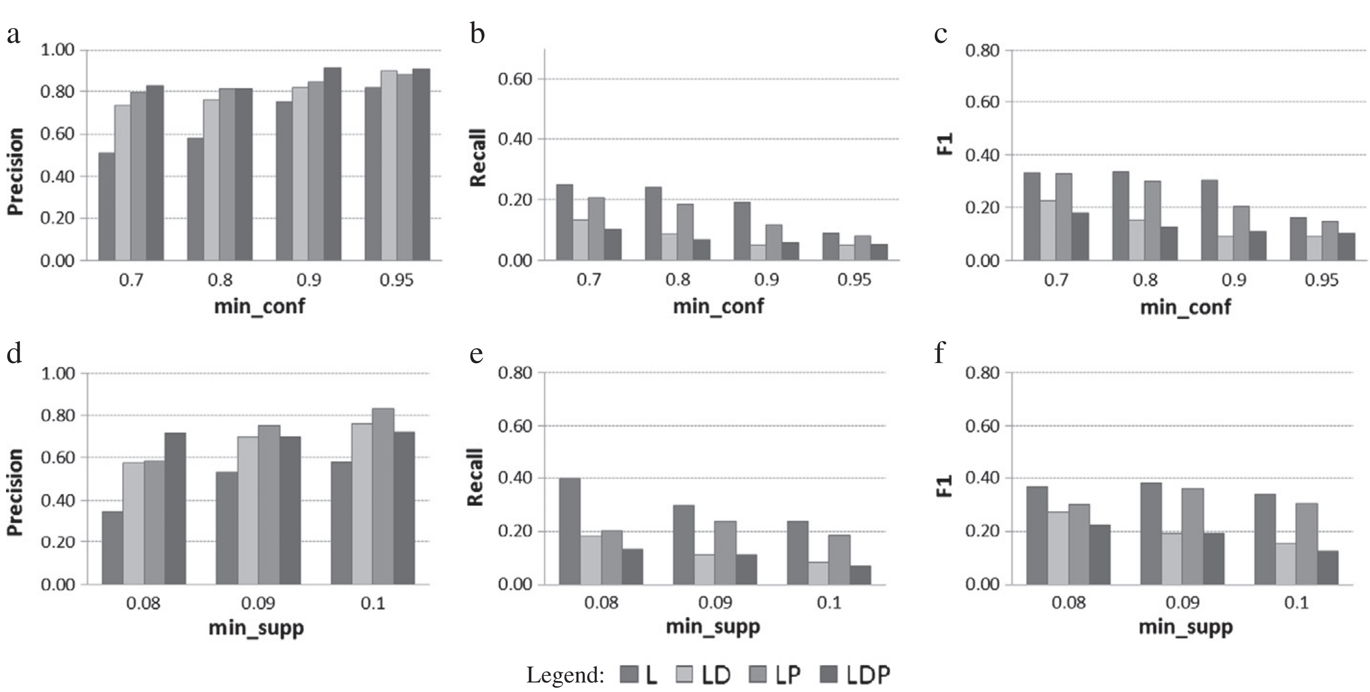

Fig. 5. Performance of rule bases versus different *min_conf* thresholds (in a, b, and c) and *min_supp* thresholds (in d, e, and f).

Table 2
Number of rules in rule bases at different *min_supp* levels (when *min_conf* = 0.8).

min_supp	# of rules			
	L	LD	LP	LDP
0.08	191	150	209	67
0.09	98	70	139	25
0.1	39	31	83	3

the 3 aforementioned dimensions (denoted "LDP"). Fig. 5(a)–(c) shows the performance (precision, recall and F1) of the five rule bases versus different confidence thresholds (0.7, 0.8, 0.9, and 0.95 respectively, while *min_supp* is fixed to 0.1). It can be seen that the precisions of rule bases generated by multi-dimensions, i.e., LP, LD, and LDP, are higher than that of "L" at all *min_conf* levels. It is also shown in the figure that, the recall of "L", on the contrary, is significantly better than that of the other 3 rule bases. Actually it is natural to expect such a result since a predictive model with more "variables" normally outperforms that with one in prediction precision; on the other hand, given the same training dataset, the latter generally have fewer generated rules, hence may result in lower coverage of all actional events (lower recall). Dragged down by low recall, the overall performance (measured by F1) of LD and LDP is thus worse than the baseline. Fig. 5(d)–(f) depicts the performance versus different *min_supp* thresholds (0.08, 0.09, 0.1, while *min_conf* is fixed to 0.8), in which a similar trend can be observed.

In order to investigate the impact of support threshold on the number of extracted rules, we have inspected the size of rule bases at different *min_supp* levels (the confidence threshold is set to 0.8, as shown in Table 2). As expected, the number of rules in each rule base increases dramatically when support threshold goes down. The number of rules in the 2-dimensional rule base "LP", interestingly, is larger than that of the single-dimensional rule base "L". The reason is that, while high-dimensional rules are rarer than low-dimensional rules, rules with dimensions "Location + Period" are generally with higher confidence, hence fewer of them are screened out by the specified confidence threshold. This finding implies that, when used in conjunction with "Location", "Period" is indeed an informative contextual dimension.

Step 3: while it has been shown in step 2 that multi-context prediction does not necessarily outperform location-based prediction in overall predictiveness, using extra contextual dimensions could still be beneficial. When various contextual dimensions are available, a "compound" rule base can be utilized. That is, a rule base can be constructed by mixing single dimensional rules and multidimensional rules so as to achieve better performance. Here we have examined the *additional* improvement in predictiveness caused by introducing extra contextual dimensions into the baseline rule base. For instance, if we consider the predictiveness of "Day" along with "Location" in addition to "Location" only, the overall performance of the combinations "Day", "Location", and "Location + Day" should be considered as a whole. In the experiment we have observed the performance of 5 rule bases with a variety of dimension combinations: RB1 (Location), RB2 (Location, Location + Day), RB3 (Location, Location + Period), RB4 (Location, Location + Day + Period) and RB5 (Location, Location + Period, Location + Day, Location + Period + Day), so that the additional improvement introduced by "Period" and "Day" can be scrutinized. Note that time-related dimensions are with regularity per se, hence single dimensional rules using either "Day" or "Period" are spurious thus should not be considered.

Fig. 6(a)–(c) depict the performance of compound rule bases RB1–RB5 versus different *min_conf* thresholds (0.7, 0.8, 0.9, and 0.95 respectively, while *min_supp* is fixed to 0.1). As shown in Fig. 6(a) and (b), while rule bases RB3 and RB5, involving the combination of Location and Period, do not improve dramatically from RB1 in terms of precision, they achieve significantly better recall, indicating that a larger portion of actional events can be predicted when additional contextual dimensions are utilized. Consequently, better overall effectiveness measured by F1 is observed from RB3 and RB5 in Fig. 6(c).

The result leads to the following implications. First, contextual dimensions other than Location matter in predicting the activities of a customer. Therefore, by utilizing extra informative contexts, the overall predictiveness of various existing LBS systems can be improved. Second, incorporating more contextual dimensions does not necessarily lead to a better result. For instance, though RB4 uses one more dimension ("Day") than RB3, F1 of RB3 significantly exceeds that of RB4 (Fig. 6(c)), since the dimension "Day" does not contribute as much additional information as "Period" when used in conjunction with "Location". This also explains the lack of difference between RB1 and RB2. A plausible reason is that the user's activity patterns on different days of a week do not have much difference.

Overall, the findings in this experiment support that given a multidimensional sequence of contextual data, the proposed framework and the associated algorithms are able to effectively identify rules with improved predictiveness, compared with using the location dimension alone.

6.3. Experiment II

In this experiment, we compare the proposed entropy-based method (referred to as *ENT*) for rule reduction with the two methods

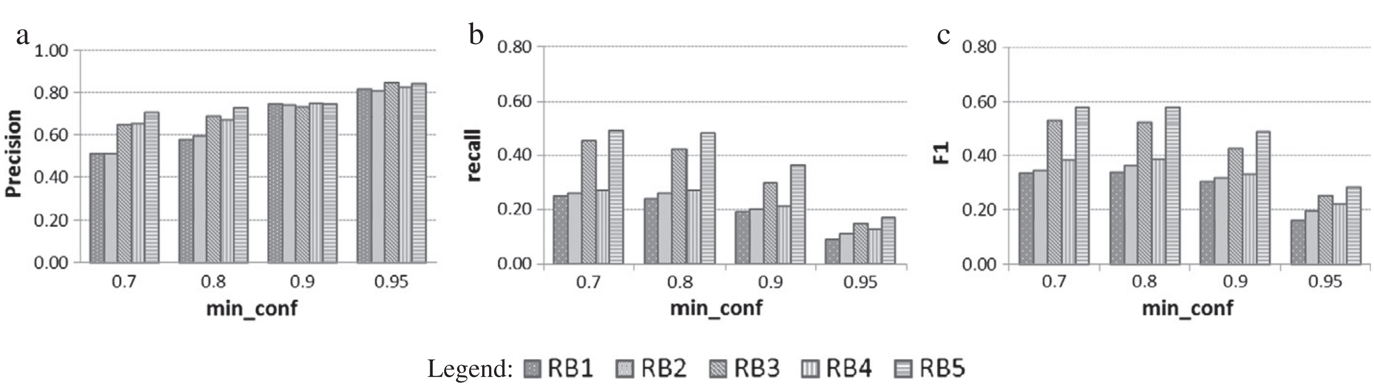

Fig. 6. Performance of compound rule bases versus different *min_conf* thresholds.

Table 3
Reduction results after applying different approaches on various rule bases*.

Method	L	LD	LP	LDP	Overall
ORIGINAL	39	31	83	3	156
RM1	39	**28 (−9.68%)**	**79 (−4.82%)**	3 (0.00%)	**149 (−4.49%)**
RM2	39	31 (0.00%)	83 (0.00%)	3 (0.00%)	156 (0.00%)
ENT	39	**29 (−6.45%)**	**74 (−10.84%)**	**2 (−33.33%)**	**144 (−7.69%)**
RM1 + ENT	39	**26 (−16.13%)**	**73 (−12.05%)**	**2 (−33.33%)**	**140 (−10.26%)**

* Note: The integers indicate the number of rules remaining after reduction and percentages in parentheses indicate the ratio of reduction compared with the original rule base. Labels L, LD, LP, and LDP are the same meaning as those described in Experiment I. Numbers in bold indicate the size of rule bases being actually reduced.

discussed in Section 4. One is referred to as *RM1* [6,7] and the other *RM2* [31,62]. The three reduction methods are applied to rule base RB5 only; reduction results are reported in Table 3. Note that *RM2* has no impact on the rule base at all because the condition $supp(r_1) < supp(r_2)$ almost consistently holds in datasets with large number of records, making the reduction criterion too loose to be applicable in screening multidimensional sequential rules.

We have scrutinized the results and found that some rules which survive the screening of *RM1* are filtered by *ENT*, for example, a rule $\{v_1 = \text{“2,12”}\}\{v_1 = \text{“2,13”}\} \rightarrow \{v_4 = \text{“}a\text{,12”}\}$ has two frequent specializations $\{v_1 = \text{“2,12”}, v_3 = 2\}\{v_1 = \text{“2,13”}, v_3 = 2\} \rightarrow \{v_4 = \text{“}a\text{,12”}\}$ and $\{v_1 = \text{“2,12”}, v_3 = 3\}\{v_1 = \text{“2,13”}, v_3 = 3\} \rightarrow \{v_4 = \text{“}a\text{,12”}\}$ in the rule base. Although the two specializations are retained in *RM1* due to their confidence being higher than that of their generalization, they are filtered by the *ENT* reduction method.

Experiment results in Table 3 also lead to an important implication in practice: the two different reduction methods, *ENT* and *RM1*, complement each other and can generate synergistic reduction results when used together. This is because they define the concept of redundancy from different perspectives. *RM1* views the higher-dimensional rule with lower confidence as a redundant rule, whereas *ENT* determines redundancy by examining the frequency distribution of the rules in a specialization set.

To study whether the reduction has affected prediction effectiveness, we examine the prediction results using the reduced rule base. Fig. 7 depicts precision, recall and F1 scores at different confidence levels for both before and after the rule reduction process.

Fig. 7(a) shows that at almost all confidence levels, the redundancy-removed rule base has higher precision than the original one, especially when both methods (*RM1* + *ENT*) are used. Fig. 7(b) shows that the recall is reduced after some rules are removed from the original rule base, which reveals a side-effect of redundancy elimination: a relatively smaller rule base tends to cover a smaller portion of all the involved actional events. As such, rule reduction has a negative effect on recall. The overall prediction performance (F1) is compared in Fig. 7(c), in which the improvement of precision is neutralized, in large part, by the decrease of recall. Nevertheless, as shown in Fig. 7(c), the rule reduction does not deteriorate the overall prediction performance (F1) while the size of the rule base is reduced. On the other hand, a quality rule base, in which redundancy is minimized and precision is guaranteed, can indeed help detect the rare regularity of mobile customers. Furthermore, at almost all confidence levels, the combination of *ENT* and *RM1* outperforms *ENT* or *RM1* in precision, implying that using the two methods together can produce a smaller rule base with better precision than using them separately.

7. Conclusion

This research aims to provide a novel solution to enable context-dependent MPM. We propose a prediction framework in order to detect temporal correlations between the mobile user's context and his/her preferences. This framework consists of a series of algorithms that process a multidimensional contextual data sequence. As many challenges have to be addressed before multidimensional sequential rules can be utilized to its full potential, this paper describes not only ways to discover multidimensional sequential rules, but also ways to overcome the common barrier in mining multidimensional sequential rules, i.e. eliminating rules with redundant dimensional information. We first formulate the task of multidimensional sequential rule mining into the classical association rule-mining problem by introducing the concept of snapshot and event, and then develop the corresponding rule-learning algorithm based on Apriori-like method to identify multidimensional sequential rules. Furthermore, we develop an entropy-based measure to quantify the redundancy level of multidimensional rules in order to select rules effective for prediction. The approximate string matching method is altered to allow online prediction.

To evaluate our framework and demonstrate its applicability, we validate the proposed algorithms using the Nokia context dataset that contains mobile device carriers' contextual information along with their interactions with mobile devices. The experimental results indicate that the proposed approach can effectively identify multidimensional sequential rules. Moreover, rule reduction coupled with redundancy measures can effectively remove redundant rules while preserving predictiveness of the rule base. Additionally, although it has been widely recognized intuitively that multidimensional rules can predict with better accuracy than single-dimensional rules, we have empirically proven that the actual performance depends on the dimensions used.

Other than the MPM domain, the proposed framework for mining multidimensional data sequences may have extensive applications in other areas where multidimensionality needs to be incorporated in the mining process to enhance the prediction of future events.

Our approach does not attempt to make precise predictions about people's daily activities, which could be extremely difficult due to the inherent randomness of human behavior. Instead, we seek to discover significant patterns from a customer's contextual information, based on which we hope some interesting actional events can be anticipated.

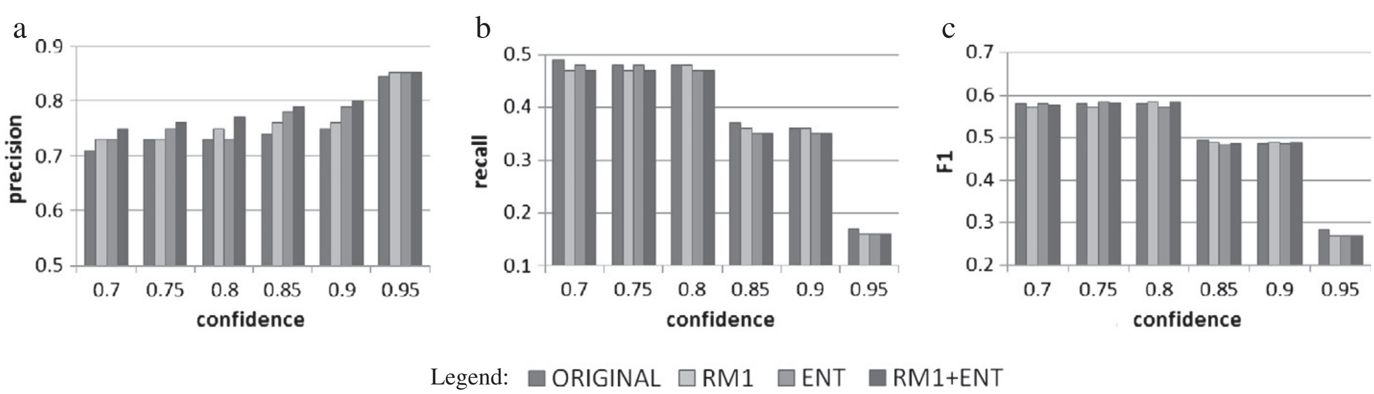

Fig. 7. Comparison of Precision, Recall, and F1 in reduced rule bases.

Hence, the effectiveness of the prediction model, in large part, relies on the extent of regularity of a customer's daily activities.

We plan to further enhance this study as follows. The rule-learning method used in this research is based on the Apriori support-confidence filters. This mechanism, however, cannot spot rules with low support but high confidence for sparse patterns with high business value. A solution to address this problem is to filter rules based on only the confidence constraint. This idea is used in traditional association mining studies [29,51], but has not attracted much attention from the perspective of sequential rule mining. Additionally, with Apriori rule generation mechanism, if most of the rules in a specialization set are infrequent, calculation of its redundancy can be inaccurate because infrequent rules will not be generated in the first place and thus their information amount has to be estimated by assuming that they are equally probable. A confidence-based filter, consequently, may also help address this problem and thereby further enhance the proposed reduction method.

Acknowledgment

We would like to thank the anonymous reviewers for the constructive review. The comments are of great help in improving the manuscript. This research was mainly supported by grant numbered MYRG005(Y1-L1)-FBA12-TH of University of Macau. It is also supported by grants from National Natural Science Foundation of China (71090402 & 71002064).

Appendix A.

Table of symbols and denotations

Symbol	Definition
$T = \{t_1, t_2, \ldots, t_n\}$	A discrete time domain
$\mathcal{D} = D_1 \times D_2 \times \ldots \times D_{m+1}$	An $(m+1)$-dimensional state space
$S(t) = \{v_1, \ldots, v_{m+1}\}$, $t \in T$	An snapshot mapping $S: T \rightarrow \mathcal{D}$
e	Event, a subset of the state space $\mathcal{D}$
$e.\ dim$	The dimensions set of e
seq_T	A sequence defined on T: $seq_T = \{S(t), t \in T\}$
$r = < e_1, e_2, \ldots, e_l >$	A rule in which $e_1, \ldots, e_l$ $\mathcal{D}$, also written as $< e_1, e_2, \ldots, e_{l-1} > \rightarrow e_l$
$r.\ dim$	The common dimension set of the antecedent of r
g	The maximum time difference allowed between the occurring time of any two neighboring events (*gap*)
w	The maximum time difference allowed between the occurring time of the first and the last events (width)
$o_r = < t_1^o, t_2^o, \ldots, t_l^o >$	An occurrence of rule r subject to g and w, also called a (g,w)-occurrence
$Occr(r, seq_T, g, w)$	The set of all (g,w)-occurrences of r in seq_T
$win = [t_i, t_j)$	A time window defined by the time points at borders t_i and t_j
$W(seq_T, w)$	The set of all windows with width w in the span of seq_T
$W_r(seq_T, g, w)$	The set of windows in $W(seq_T, w)$ that contain any occurrence of r
$concat(r_1, r_2)$	Concatenation of rules r_1 and r_2
$Cand_i$	The set of candidate rules with length i
$Freq_i$	The set of frequent rules with length i
$supp(r, seq_T, g, w)$	Support of rule r
min_supp	The threshold of minimum support
$conf(r, seq_T, g, w)$	Confidence of rule r
min_conf	The threshold of minimum confidence
$spec(r, D_m)$	The set of all possible specialization of r on D_m
$R(r, D_m)$	The set of m-dimensional frequent rules with respect to r
$Redun(r, D_m)$	The redundancy of $spec(r, D_m)$
matched[]	The table to maintain the last matched snapshot of each rule
time_last[]	The table to maintain the time point of the last matched snapshot of each rule
time_first[]	The table to maintain the time point of the first matched snapshot of each rule

References

[1] A.S. Abrahams, A. Becker, D. Fleder, I.C. MacMillan, Handling generalized cost functions in the partitioning optimization problem through sequential binary programming, Proceedings of the 5th IEEE International Conference on Data Mining, IEEE Computer Society, Houston, Texas, 2005, pp. 3–9.
[2] G. Adomavicius, R. Sankaranarayanan, S. Sen, A. Tuzhilin, Incorporating contextual information in recommender systems using a multidimensional approach, ACM Transactions on Information Systems 23 (1) (2005) 103–145.
[3] G. Adomavicius, A. Tuzhilin, Expert-driven validation of rule-based user models in personalization applications, Data Mining and Knowledge Discovery 5 (1) (2001) 33–58.
[4] R. Agrawal, R. Srikant, Fast algorithms for mining association rules in large databases, Proc. of the 20th Internat. Conf. on Very Large Data Bases, Morgan Kaufmann Publishers Inc., Santiago, Chile, 1994, pp. 487–499.
[5] R. Agrawal, R. Srikant, Mining sequential patterns, Proc. of the 11th Internat. Conf. on Data Engrg, 1995, pp. 3–14.
[6] M.Z. Ashrafi, D. Taniar, K. Smith, A new approach of eliminating redundant association rules, Database And Expert Systems Appl.: 15th Internat. Conf, Springer, Zaragoza, Spain, 2004, pp. 465–474.
[7] M.Z. Ashrafi, D. Taniar, K. Smith, Redundant association rules reduction techniques, AI 2005: Adv. in Artificial Intelligence: 18th Australian Joint Conf. on Artificial Intelligence, Springer, Sydney, Australia, 2005, pp. 29–63.
[8] M. Atallah, R. Gwadera, W. Szpankowski, Detection of significant sets of episodes in event sequences, Proc. of the 4th Internat. Conf. on Data Mining, 2004, pp. 3–10.
[9] M. Baldauf, S. Dustdar, F. Rosenberg, A survey on context-aware systems, International Journal of Ad Hoc and Ubiquitous Computing 2 (4) (2007) 263–277.
[10] C. Bettini, X.S. Wang, S. Jajodia, J.L. Lin, Discovering frequent event patterns with multiple granularities in time sequences, IEEE Transactions on Knowledge and Data Engineering 10 (2) (1998) 222–237.
[11] I. Bose, X. Chen, A framework for context sensitive services: a knowledge discovery based approach, Decision Support Systems 48 (1) (2009) 158–168.
[12] C. Chatfield, Time-series Forecasting, Chapman & Hall/CRC, 2001.
[13] S. Chaudhuri, U. Dayal, An overview of data warehousing and OLAP technology, ACM Sigmod Record 26 (1) (1997) 65–74.
[14] J. Cheng, Y. Ke, W. Ng, Delta-tolerance closed frequent itemsets, Proc. of the 6th Internat. Conf. on Data Mining, IEEE Computer Society, Hong Kong, China, 2006, pp. 139–148.
[15] J. Cheng, Y. Ke, W. Ng, Effective elimination of redundant association rules, Data Mining and Knowledge Discovery 16 (2) (2008) 221–249.
[16] D.-Y. Choi, Personalized local internet in the location-based mobile web search, Decision Support Systems 43 (1) (2000) 31–45.
[17] M. Crochemore, C. Hancart, T. Lecroq, Algorithms on Strings, Cambridge University Press, 2007.
[18] A. Flanagan, Nokia Context Data. , December 13, 2010 http://www.pervasive.jku.at/Research/Context_Database/index.php2004, (retrieved).
[19] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, Journal of Machine Learning Research 3 (2003) 1157–1182.
[20] I.A.N. Hacking, Equipossibility theories of probability, The British Journal for the Philosophy of Science 22 (4) (2004) 339–355.
[21] J. Han, M. Kamber, Data Mining: Concepts and Techniques, 2nd ed. Morgan Kaufmann, 2006.
[22] J.L. Herlocker, J.A. Konstan, L.G. Terveen, J.T. Riedl, Evaluating collaborative filtering recommender systems, ACM Transactions on Information Systems (TOIS) 22 (1) (2004) 5–53.
[23] E.T. Jaynes, Probability Theory: The Logic of Science, Cambridge University Press, 2003.
[24] H.A. Karimi, X. Liu, A predictive location model for location-based services, Proc. of the 11th ACM Internat. symposium on Adv. in geographic information systems, 2003, pp. 126–133, , (ACM, New Orleans, Louisiana, USA).
[25] J. Kogan, C. Nicholas, M. Teboulle, Grouping multidimensional data: recent advances in clustering, Springer-Verlag, New York, 2006.
[26] R. Kohavi, G.H. John, Wrappers for feature subset selection, Artificial Intelligence 97 (1–2) (1997) 273–324.
[27] S. Laxman, P.S. Sastry, K.P. Unnikrishnan, Discovering frequent generalized episodes when events persist for different durations, IEEE Transactions on Knowledge and Data Engineering 19 (9) (2007) 1188–1201.
[28] K.F. Lee, H.W. Hon, R. Reddy, An overview of the SPHINX speech recognition system, IEEE Transactions on Acoustics, Speech, and Signal Processing 38 (1) (1990) 35–45.
[29] J. Li, X. Zhang, G. Dong, K. Ramamohanarao, Q. Sun, Efficient mining of high confidence association rules without support thresholds, Lecture Notes in Computer Science 1704/1999 (1999) 406–411.
[30] J. Lin, Divergence measures based on the Shannon entropy, IEEE Transactions on Information Theory 37 (1) (1991) 145–151.
[31] C. Loglisci, D. Malerba, Mining multiple level non-redundant association rules through two-fold pruning of redundancies, Proc. of the 6th Internat. Conf. on Machine Learn. and Data Mining in Pattern Recognition, Springer, 2009, p. 265.
[32] H. Mannila, H. Toivonen, Discovering generalized episodes using minimal occurrences, Proc. of the 2nd Internat. Conf. on Knowledge Discovery in Databases and Data Mining, 1996, pp. 146–151.
[33] H. Mannila, H. Toivonen, A. Inkeri Verkamo, Discovery of frequent episodes in event sequences, Data Mining and Knowledge Discovery 1 (3) (1997) 259–289.
[34] D. Mladenic, M. Grobelnik, Feature selection on hierarchy of web documents, Decision Support Systems 35 (1) (1994) 45–87.

[35] B. Mortazavi-Asl, J. Wang, H. Pinto, Q. Chen, M.C. Hsu, Mining sequential patterns by pattern-growth: the PrefixSpan Approach, IEEE Transactions on Knowledge and Data Engineering 16 (11) (2004) 1424–1440.
[36] B.P.S. Murthi, S. Sarkar, The role of the management sciences in research on personalization, Management Science 49 (10) (2003) 1344–1362.
[37] E.W.T. Ngai, A. Gunasekaran, A review for mobile commerce research and applications, Decision Support Systems 43 (1) (2007) 3–15.
[38] T.A. Nguyen, W.A. Perkins, T.J. Laffey, D. Pecora, Knowledge-base verification, AI Magazine 8 (2) (1987) 69–75.
[39] D. O'Shea, Small screen for rent, Telephony Online, 2007. (December 13, 2010 http://telephonyonline.com/mag/telecom_small_screen_rent/, retrieved).
[40] B. Padmanabhan, Z. Zheng, S. Kimbrough, An empirical analysis of the value of complete information for eCRM models, MIS Quarterly 30 (2) (2006) 247–267.
[41] D. Peppers, M. Rogers, Enterprise One to One: Tools for Competing in the Interactive Age, Currency-Doubleday, New York, 1997.
[42] H. Pinto, J. Han, J. Pei, K. Wang, Q. Chen, U. Dayal, Multi-dimensional sequential pattern mining, Proc. of the 10th Internat. Conf. on Inform. and Knowledge Management, ACM Press, New York, 2001, pp. 81–88.
[43] M. Plantevit, T. Charnois, J. Klema, C. Rigotti, B. Cremilleux, Combining sequence and itemset mining to discover named entities in biomedical texts: a new type of pattern, International Journal of Data Mining, Modelling and Management 1 (2) (2009) 119–148.
[44] J.H. Schiller, A. Voisard, Location-based Services, Morgan Kaufmann, San Francisco, 2004.
[45] R. Srikant, R. Agrawal, Mining sequential patterns: generalizations and performance improvements, Proc. of the 5th Internat. Conf. on Extending Database Tech, Springer, Avignon, France, 1996, pp. 3–17.
[46] Z. Su, Q. Yang, Y. Lu, H. Zhang, WhatNext: a prediction system for Web requests using n-gram sequence models, Proc. of the 1st Internat. Conf. on Web Inform. Systems Engrg., 2000, pp. 214–221.
[47] P.N. Tan, M. Steinbach, V. Kumar, Introduction to Data Mining, Addison-Wesley, 2005.
[48] C.J. Van Rijsbergen, Information Retrieval, Butterworth-Heinemann Newton, MA, USA, 1979.
[49] A.I. Vermesan, Quality assessment of knowledge-based software: some certification considerations, Proc. of the 3rd Internat. Software Engrg. Standards Sympos, 1997, pp. 1–6.
[50] J. Wang, J. Han, C. Li, Frequent closed sequence mining without candidate maintenance, IEEE Transactions on Knowledge and Data Engineering 19 (8) (2007) 1042–1056.
[51] K. Wang, Y. He, D.W. Cheung, Mining confident rules without support requirement, Proc. of the 10th Internat. Conf. on Inform. and Knowledge Management, ACM, Atlanta, Georgia, 2001, pp. 89–96.
[52] W. Wang, J. Yang, Mining Sequential Patterns From Large Data Sets, Springer, 2005.
[53] W.W.S. Wei, Time Series Analysis: Univariate and Multivariate Methods, Addison Wesley, 1989.
[54] O. White, T. Dunning, G. Sutton, M. Adams, J.C. Venter, C. Fields, A quality control algorithm for DNA sequencing projects, Nucleic Acids Research 21 (16) (1993) 3829–3838.
[55] X. Yan, J. Han, R. Afshar, CloSpan: mining closed sequential patterns in large datasets, SIAM Internat. Conf. on Data Mining, Society for Industrial and Applied Mathematics, San Francisco, CA, 2003.
[56] Q. Yang, T. Li, K. Wang, Building association-rule based sequential classifiers for Web-document prediction, Data Mining and Knowledge Discovery 8 (3) (2004) 253–273.
[57] Q. Yang, J. Yin, C. Ling, R. Pan, Extracting actionable knowledge from decision trees, IEEE Transactions on Knowledge and Data Engineering 18 (16) (2006).
[58] Q. Yang, H.H. Zhang, Web-log mining for predictive Web caching, IEEE Transactions on Knowledge and Data Engineering 15 (4) (2003) 1050–1053.
[59] C.C. Yu, Y.L. Chen, Mining sequential patterns from multidimensional sequence data, IEEE Transactions on Knowledge and Data Engineering 17 (1) (2005) 136–140.
[60] S.T. Yuan, C. Cheng, Ontology-based personalized couple clustering for heterogeneous product recommendation in mobile marketing, Expert Systems with Applications 26 (4) (2004) 461–476.
[61] M.J. Zaki, SPADE: an efficient algorithm for mining frequent sequences, Machine Learning 42 (1) (2001) 31–60.
[62] M.J. Zaki, Mining non-redundant association rules, Data Mining and Knowledge Discovery 9 (3) (2004) 223–248.
[63] Y. Zhao, S. Zhang, Generalized dimension-reduction framework for recent-biased time series analysis, IEEE Transactions on Knowledge and Data Engineering 18 (2) (2006) 231–244.

Heng Tang received his Ph.D degree in the Department of Information Systems, City University of Hong Kong. He is currently an assistant professor at the Department of Accounting and Information Management, University of Macau. His research interests are in the areas of Data Mining, Recommender Systems, and Mobile Computing.

Stephen Shaoyi Liao is currently a professor at the Department of Information Systems, City University of Hong Kong. He obtained his Ph.D degree in Information Systems from Aix-Marseille University, France. His main research areas include mobile applications and Intelligent Transportation Systems.

Sherry Xiaoyun Sun received M.S. and Ph.D degrees in Management Information Systems from Eller College of Management, The University of Arizona, Tucson, Arizona, USA. She is currently an assistant professor at the Department of Information Systems, City University of Hong Kong. Her research mainly focuses on business process management, electronic commerce, and service oriented computing. Her research work has been published in Information Systems Research, IEEE Transactions on Systems, Man, and Cybernetics, Journal of Systems and Software, Information Systems Frontier, Knowledge-Based Systems and others.