Efficient constraint-based Sequential Pattern Mining (SPM) algorithm to understand customers’ buying behaviour from time stamp-based sequence dataset

Niti Ashish Kumar Desai* and Amit Ganatra

Abstract: Business Strategies are formulated based on an understanding of customer needs. This requires development of a strategy to understand customer behaviour and buying patterns, both current and future. This involves understanding, first how an organization currently understands customer needs and second predicting future trends to drive growth. This article focuses on purchase trend of customer, where timing of purchase is more important than association of item to be purchased, and which can be found out with Sequential Pattern Mining (SPM) methods. Conventional SPM algorithms worked purely on frequency identifying patterns that were more frequent but suffering from challenges like generation of huge number of uninteresting patterns, lack of user’s interested patterns, rare item problem, etc. Article attempts a solution through development of a SPM algorithm based on various constraints like Gap, Compactness, Item, Recency, Profitability and Length along with Frequency constraint. Incorporation of six additional constraints is as well to ensure that all patterns are recently active (Recency), active for certain time span (Compactness), profitable and indicative of next timeline for purchase (Length—Item—Gap). The article also attempts to throw light on how proposed...
Constraint-based Prefix Span algorithm is helpful to understand buying behaviour of customer which is in formative stage.

Subjects: Computer Engineering; Data Preparation & Mining; I.T. Research

Keywords: constraints; sequential pattern mining; constraint-based Prefix Span

1. Literature survey
To mine important information and mould it into proper knowledge is important task from past few decades, known as data mining. (Chen, Han, & Philip, 1996). Data mining uses to extract knowledge from huge extent of data (Mary & Iyengar, 2010; Raza, 2010). There are certain activity which is necessary to happen in sequence to mine such sequence are important data mining activity known as sequential pattern Mining (SPM), was introduced in (Agrawal & Srikant, 1995; Srikant & Agrawal, 1996). It is useful in various data mining applications, to discover useful customer and market information from the data, such as product recommendation (Lawrence et al., 2001), e-retailing, customer profiling (Hu & Chen, 2008; Mahdavi, Cho, Shirazi, & Saheljannia, 2008). Current SPM techniques are mainly differentiate by \textit{a priori} based (Agrawal & Srikant, 1994) and FP-Growth-based (Han, Pei, & Yin, 2000) techniques. Both \textit{a priori}-based GSP (Srikant & Agrawal, 1996), SPADE (Zaki, 2001), SPAM (Ayres, Flannick, Gehrke, & Yiu, 2002), SPIRIT (Garofalakis, Rastogi, & Shim, 1999) and FP-Growth-based Freespan (Han, Dong et al., 2000), Prefix span (Pei, Han, Mortazavi, & Pino, 2001) are purely worked on sole parameter frequency, known as support threshold. Conventional SPM techniques often take substantial computational time and space for mining the complete set of sequential patterns in a large sequence database. Introduction of constraint may open a new opportunity for performance improvement: “Can we improve the efficiency of sequential pattern mining by focusing only on interesting patterns?” Constraint-based mining may improve or overcome both the difficulties: efficiency and effectiveness. Conventional SPM techniques are not successful to extract those patterns which are having more potential in terms of Monitory value and Recency. (Chen, Kuo, Wu, & Tong, 2009) has taken care of these parameters by introducing Recency, Frequency and Monitory (RFM) parameters in SPM.

The algorithm RFM-A \textit{a priori} is helpful to users for identification of those patterns which are more frequent, active recently and have high monetary value (Chen et al., 2009). RFM is used in many data mining activities likes, association rule mining, clustering, classification, etc. (Chen, Wu, & Chen, 2005; Mohammad, Hosseini, Maleki, & Mohammad, 2009; Qiasi, Dehnavi, Minaei-Bidgoli, & Amooee, 2012). (Goodman, 1992) has applied RFM model to find valuable customers. RFM is used to specify loyal and profitable customers on based of clustering. Classification algorithms are useful to obtain rules for implementing effective customer relationship management. Combinations of behavioural and demographical characteristics are used to estimate loyalty (Qiasi et al., 2012). Extended RFM model called Weighted RFM (WRFM) and K-means algorithm applied classify customer product loyalty in under B2B concept. Customer loyalty is used in marketing strategy (Mohammad et al., 2009). In most of the researches, two methods are common for identifying loyal customers, one of them is in terms of demographic variables (such as age, gender, etc.) and the other is in terms of interactive behaviours of customers that are expressed with the so-called RFM. RFM model is proposed by (Hughes, 1994) and has been used in direct marketing for several decades (Chen et al., 2005). (Chen et al., 2009) focused on effective discovery of current spending pattern of customers and trends of behavioural change using classification-based clustering on bases of RFM model to identify high-profit, gold customers. It helps to identifies customer preference, and provide desired service according to customer need which prevent customer attrition (Chen et al., 2005). (Shaw, Subramaniam, Tan, & Welge, 2001) mainly describes how to incorporate data mining into the framework of marketing knowledge management. (Song, Kim, & Kim, 2001) depicts a method to detect changes of customer behaviour at different time snapshots from customer profiles and sales data. (Shen & Chuang, 2009) used RFM model-based cluster analysis for customers, which can evaluate high value customers in terms of high loyalty, high interest, and a high amount of purchase. On bases of results company can apply appropriate target marketing to enhance customers’ lifetime value (CLV) (Shen & Chuang, 2009). The problem of sequential pattern mining in B2C (Business to customer) environment and in B2B (Business to Business) environment has been investigated
by Chen.\textsuperscript{1} Importance of Compact, Frequent, and Recent sequential patterns (CFR-patterns) in (Business to customer) B2C environment and Compact, Frequent, Recent, and Repeated sequential patterns (CFR2-patterns) in B2B (Business to Business) environment has been described (see Note 1). The algorithm CFM-PREFIX SPAN has incorporated compactness and Monitory constraint in pattern growth-based Prefix Span which give precise patterns in context of high-valued patterns and pattern active during precise time span (Mallick, Garg, & Grover, 2012). The Algorithm C-Prefix Span has incorporated Gap, Compactness, Recency along with conventional Frequency (Bhensdadia & Kosta, 2012). Following advantages can be achieved by constraint-based sequential mining:

(i) Enhance performance of algorithm by reducing computation cost of uninteresting patterns.
(ii) Along with frequency other important parameter are focused.
(iii) User can focused on only those patterns which are really of his/her interest.
(iv) User can get goal or application-oriented result.

1.1. Major research findings and basic outline of proposed research

(I) Most of the existing SPM methods work purely on frequency parameter, formally known as support threshold. Support is important to distinguish if patterns appear repeatedly or not. On the other hand, the proposed Constraint-based Prefix Span algorithm would be concerned about decision maker’s perception. Proposed algorithm through use of Recency constraint determines current buying patterns and through use of profit constraint determines the more profitable buying patterns. Further compactness constraint can be used to identify buying behaviour of customer during specific time span including seasonal patterns. Item and length constraint will help to give detailed buying behaviour of customer. Further length constraint in the algorithm can be used to understand customers’ buying preferences for identifying influential items of purchase leading or indicating higher probability for next purchase.

(II) Almost all the existing SPM methods are focused on the current scenario. There are some patterns which have potential to become strong in future, but are suffering from slightly less support. Minor reduction in support values may bring such patterns into focus, which can be potential buying patterns for future. Our research highlights such potential patterns using Boundary value reduction and Recency-based Emerging Patterns (EPs) algorithm.

(III) Extraction of sequential patterns works on objective measures like support and confidence. SPM with subjective measures is an unchartered area that needs to be explored. Our research includes subjective analysis for purchase such as profitability, loyalty and influential purchase using amalgamation of various constraints. Customers who are frequent, profitable and recent are identified as loyal customers. Segmentation of customers based on such subjective survey criteria can be done using the proposed method.

(IV) Our research provides an algorithm to study customers’ current buying behaviour and identification of patterns that are likely to become obsolete in future and those that are at their formative stage.

2. System framework

Theoretical physicist Albert Einstein once said: “Everything that can be counted does not necessarily count; everything that counts cannot necessarily be counted.” Since marketers can’t measure everything, the challenge is to focus on those parameters that truly matter to enhance business performance.\textsuperscript{2} Article has adopted this philosophy. Article focused on those constraints which are useful to track customer’s present as well future buying behaviour. Future buying trend of purchase can be discovered by following:

(i) Emerging Patterns (to highlight buying habits of customers which will useful for future).
(ii) High-influenced item (on bases of customer buying preferences).
(iii) Buying habit of customer on forming stage.
(iv) Seasonal buying habit of customer.

Customers’ behavioural study provides current as well future buying snapshot which will be useful for expansion of business.

2.1. Analysis on customers’ buying behaviour based on constraints

2.1.1. Emerging patterns (EPs)

Almost all the existing SPM techniques are concentrated on the present buying behaviour of customer but there are some patterns which are having sufficient potential to become strong in future, which are suffering from slightly less support. Slight reduction in support boundary can laid such pattern in consideration, which can be emerging purchase patterns for tomorrow. Such Emerging Patterns (EPs) are considered in proposed approach here.

In Equation 1, SD is Sequential Database with timestamp. FP is Frequent Patterns, which are generated for support values. x is support value which generates frequent patterns (FP) \( \text{Support} = x\% \). Support \( y \) is generated by reduction of 2% in support value x. (FP) \( \text{Support} = y\% \) is frequent patterns generated by slight reduction of support value which high lights those patterns which are facing problem of little low frequency. SEQ EP1 is emerging patterns generated by reduced boundary value, recent patterns from those patterns might be potential patterns of tomorrow.

\[
\begin{align*}
(\text{SD})_{\text{Support} = x\%} \rightarrow (\text{FP})_{\text{Support} = x\%} \\
(\text{support} = y\%) = (\text{support} = x\%) - (0.002 - (\text{support} = x\%)) \\
\text{SEQ EP1} = (\text{FP})_{\text{Support} = y\%} - (\text{FP})_{\text{Support} = x\%}
\end{align*}
\]  

(1)

Where, \( |Y - X| \) = Boundary value > 0, SD = Sequential Database, FP = frequent pattern, \( \text{SEQ EP1} \) = Sequential Pattern for reduced boundary value; \( X, Y > 0 \) (refer Equation 2)

\[
\text{SEQ recency EP} = \text{SEQ EP1} \text{ satisfied } C_{\text{recency}} \text{ EP} = \text{SEQ recency EP}
\]  

(2)

Where, \( C_{\text{recency}} \) is recency timestamp constraint. \( \text{SEQ recency EP} \) is emerging sequential patterns generated by recency constraint and EP is Emerging Patterns which are recent. Generated patterns are suffering from low support but not old. Old patterns are eliminated by inclusion of recency constraint. For example, \(<\text{Computer} > \text{ Floppy}>\) was earlier famous buying pattern. Suppose, it’s having 18% support value. It was rejected for support value 20%. Reduction of boundary value high lights this pattern though it is old (SEQ EP1 = (FP) Support = y% - (FP) Support = x%). Another buying pattern \(<\text{Laptop} > \text{ Blue ray disk}>\) is recent but not too frequent as compared to complete transactional database. Because of less little value of support, it is rejected. Pattern \(<\text{Laptop} > \text{Blue ray disk}>\) is selected by slight reduction of boundary. The patterns which are on its forming stage can be easily detected.

2.1.2. Profitable pattern

It is important for any business to understand profitable purchase. Profit is indirectly derived from Monitory constraint. Profit is depends upon two valuable parameters: purchase price and sold price. Profitability is one of the important parameter of customer segregation and retention. The Profitable constraint define item in a sequence must be more than the defined threshold value. The Profitable constraint is formally represented as following:

\[
C_{\text{Profit}} \equiv \text{Sold} - \text{Purchase}(\alpha) \omega \Delta T
\]  

(3)

Where, \( \omega \in (2), \Delta T \) integer value/ Profitable threshold. \( \alpha \) is sequence.
A sequence $S = \langle(q_1, \text{qty}_1), t_1, M_{\text{Sold}_1}, M_{\text{Pur}_1}, \ldots, (q_m, \text{qty}_m), M_{\text{Sold}_m}, M_{\text{Pur}_m} \rangle$ is said to be a subsequence of $S$ only if, (1) item set $S_2$ is a subsequence of $S$, $S_2, S \subseteq S$ and (2) the number of items in $S$ should satisfy (refer Equations 3 and 4)

$$\sum_{i=1}^{m} (M_{\text{Sold}_i} - M_{\text{Pur}_i}) \geq T_{\text{Profit}}$$  \hspace{1cm} (4)$$

### 2.1.3. Influential item

It is important for decision maker to understand purchase of which item leads to another purchase(s) after some time period. For example, TV $\rightarrow$ DVD $\rightarrow$ (CD, CD Box). TV is highly influenced item which leads to second or third purchase(s). High association of such chain purchase leads to loyal customer, with the help of length constraint such chain purchase can be identified. Amalgamation of length and profitability can highlight those patterns which are building strong relationship with customer and lead to sequence purchase.

### 2.1.4. Seasonal buying patterns

Most of the customer moving for purchase during festival seasons like Diwali and Christmas. Purchase of garment and house hold is drastically increased during these time span. In India, purchase of gold and diamond are increased during Akshay Trity, Lakshmi Pujan and Pushya Nakshtra every year. Compactness constraint which used to represent duration is helpful parameter to understand active purchase during particular time span. Patterns which are highly purchased can be extracted by compactness constraint; business maker can change sale value of such items which leads to profitability (Pei, Han, & Wang, 2007). (Refer Equation 5)

$$C_{\text{Comp}} \equiv \text{Comp}(\alpha) \cdot \Delta t \hspace{1cm} (5)$$

### 2.2. Constraints to understand proposed algorithm

Traditional sequential pattern mining only is distinguishes whether a pattern appears or not (Ayres et al., 2002; Garofalakis et al., 1999; Zaki, 2001). The original Prefix span algorithm only worked on frequency constraint to discover sequential patterns from sequence database (Pei et al., 2001). RFM pattern mining approach not only determines the existence of a pattern but also checks whether it satisfies the recency and the monetary constraints (Chen et al., 2009). Proposed approach worked on seven constraints—Frequency constraint, Item constraint, Length constraint, Gap constraint, Compactness constraint, Recency constraint, Quantity constraint and profitable constraint. Above constraints are incorporated in original Prefix span algorithm. Following key concepts are required to understand the proposed algorithm:

#### 2.2.1. Frequency constraint

The frequency constraint is defined as each discovered pattern must satisfy minimum support. That is, the support of an item is the percentage of transaction in which that item occurs (Refer Equation 6).

$$C_{\text{Freq}} \equiv P(\alpha) = (\alpha)/N \cdot \omega \cdot \Delta T$$  \hspace{1cm} (6)$$

Where, $\omega \in (2), \Delta T$ integer value/ Frequency threshold. $\alpha$ is sequence. $N$ Number of transaction of Sequence Database SDB (Pei et al., 2007).

#### 2.2.2. Recency constraint

Recency constraint is specified by giving a recency minimum support ($r_{\text{mins}}$). In time stamped-based dataset minimum Recency time stamp is given. The patterns having more time stamp can be selected. For example, $((a)<1>(b,c)<2>)$ and $((a,b)<3>(b,c)<5>)$ are frequent patterns. Suppose, $r_{\text{mins}} = 3$ than only $((a,b)<2>(b,c)<5>)$ is selected because pattern is generated at later time stamped value, which are considered as recent pattern.
“after buying item a and b, the customer moves to buy item b and item c.” Then, the transaction in the sequence that buys item b and item c must satisfy recency constraint. Time stamp of last buying item (b, c) should ≥ \( r_{\text{minsup}} = 3 \). Formally, Recency constraint is define in Equation 7 (Pei et al., 2007).

\[
C_{\text{recency}} \equiv \text{recency}(\alpha) \theta \Delta t
\]  

(7)

where \( \theta \in \{\leq, \geq\} \) and \( \Delta t \) is a given integer. A sequence \( \alpha \) satisfies the constraint if \(| \beta \in \text{SDB}| \exists 1 \leq i < \cdots \leq \text{ilen}(\alpha) \leq \text{len}(\beta) \text{ s.t. } \alpha[1] \sqsubseteq \beta[i], ..., \alpha[\text{ilen}(\alpha)] \sqsubseteq \beta[\text{ilen}(\alpha)], \text{ and } (\beta[i] \text{. time } \theta \Delta t) | \geq \text{min\_sup} \).

2.2.3. Gap constraint
Gap constraint is specified by giving gap range (\( \text{min\_gap} \) and \( \text{max\_gap} \)). In sequence databases each transaction in every sequence has a timestamp. The time-stamp difference (difference of days) between every two adjacent transactions in a discovered sequential pattern must not be greater than given range of gap. \( \text{min\_gap} \) and \( \text{max\_gap} \) time-stamp by 2 and 4 means \((4 - 2 = 2)\). so, pattern \(<t2:(a), t4:(bc), t7:(ac)><\) can’t be discovered because of more time gap occur at \(<t4:(bc), t7:(ac)>, (7 - 4 = 3)\) though time-stamp \((4 - 2 = 2)\) is satisfied for sub-pattern \(<t1:(a), t4:(bc)><\). Formally, a gap constraint is in the form of (Refer Equation 8), (Pei et al., 2007).

\[
C_{\text{gap}} \equiv \text{Gap}(\alpha) \theta \Delta t
\]  

(8)

where \( \theta \in \{\leq, \geq\} \) and \( \Delta t \) is a given integer. A sequence \( \alpha \) satisfies the constraint if and only if

\[|(\beta \in \text{SDB}| \exists 1 \leq i < \cdots \leq \text{ilen}(\alpha) \leq \text{len}(\beta) \text{ s.t. } \alpha[1] \sqsubseteq \beta[i], ..., \alpha[\text{ilen}(\alpha)] \sqsubseteq \beta[\text{ilen}(\alpha)], \text{ and for all } 1 < j \leq \text{ilen}(\alpha), (\beta[j] \text{. time} - \beta[j-1] \text{. time}) \theta \Delta t) | \geq \text{min\_sup}.\]

2.2.4. Compactness constraint
Compactness constraint is specified by giving compactness range (\( \text{min\_compactness} \), \( \text{max\_compactness} \)). In sequence databases each transaction in every sequence has a timestamp. The time-stamp difference (difference of days) between the first and the last transactions in a discovered sequential pattern must not be greater than given period. (Refer Equation 9) (Pei et al., 2007).

\[
C_{\text{Comp}} \equiv \text{Comp}(\alpha) \theta \Delta t,
\]  

(9)

where \( \theta \in \{\leq, \geq\} \) and \( \Delta t \) is a given integer. A sequence \( \alpha \) satisfies the constraint if and only if

\[|(\beta \in \text{SDB}| \exists 1 \leq i < \cdots \leq \text{icomp}(\alpha) \leq \text{comp}(\beta) \text{ s.t. } \alpha[1] \sqsubseteq \beta[i], ..., \alpha[\text{comp}(\alpha)] \sqsubseteq \beta[\text{comp}(\alpha)], \text{ and for all } 1 < j \leq \text{comp}(\alpha), (\beta[j] \text{. time} - \beta[j-1] \text{. time}) \theta \Delta t) | \geq \text{min\_sup}.\]

2.2.5. Quantity constraint
Quantity constraint is specified by giving minimum quantity support (\( q_{\text{minsup}} \)). Total quantity value of the discovered pattern must be greater than \( q_{\text{minsup}} \). Suppose the pattern is \(<(a(10)),(b(5)c(7))><\) and \( m_{\text{minsup}} = 15 \). The sequence contains total quantity value = 10 + 5 + 2 =17 ≥ 15, which satisfied quantity constraint.

A sequence \( S_k = <(q_1(qty_1)), (q_2(qty_2)), ..., (q_m(qty_m))> \) is said to be a subsequence of \( S \) only if, (1) itemset \( S_k \) is a subsequence of \( S, S_k \subseteq S \) and (2) the number of items in \( S \) should satisfied \( \sum_{i=1}^{m}(qty_i) \geq T_{\text{qty}} \)

Where, \( T_{\text{qty}} \) is quantity constraint.

2.2.6. Monitory constraint
Monitory constraint is specified by giving minimum monitory support (\( m_{\text{minsup}} \)). Total monetary value of the discovered pattern must be greater than \( m_{\text{minsup}} \). Suppose the pattern is \(<(a(100)),(b(50)c(70))><\) and \( m_{\text{minsup}} = 130 \). The sequence contains total monitory value = 100 + 50 + 70 = 220 ≥ 130, which satisfied monitory constraint.

Total Monitory: The Total Monitory (TM) constraint defines item in a sequence must be more than the defined threshold value. The Total Monitory constraint is formally represented in Equation 10.
Where, \( \omega \in \{\geq\} \), \( \Delta T \) integer value/ Profitable threshold. \( \alpha \) is sequence.

A sequence \( S_s = \langle (q_1(qty_1), t_1, M_{Sold_1}, M_{Pur_1}), (q_2(qty_2), M_{Sold_2}, M_{Pur_2}), \ldots, (q_m(qty_m), M_{Sold_m}, M_{Pur_m}) \rangle \) is said to be a subsequence of \( S \) if, (1) itemset \( S_s \) is a subsequence of \( S \) and (2) the number of items in \( S \) should satisfied Equation 11.

\[
\sum_{i=1}^{m}(qty_i * M_{Pur_i}) \geq T_{TM}
\]  

2.2.7. Length constraint
Length constraint is specified by giving minimum length support (l_minsup). Length of the discovered pattern must be greater than \( l_{\text{minsup}} \). Suppose the pattern is \( <(a,b)(b,c)> \) and \( l_{\text{minsup}} = 2 \). The sequence is selected because length of the sequence = 2 ≥ \( l_{\text{minsup}}(=2) \). (refer Equation 12) (Pei et al., 2007).

\[
C_{\text{Length}} \equiv len(\alpha) \omega l
\]  

Where, \( \omega \in \{\geq\} \), \( l \) integer value/length threshold. \( \alpha \) is sequence.

2.2.8. Item constraint
An item constraint specifies subset of items that should or should not be present in the patterns. Suppose the pattern is \( <(a),(b,c)> \). If the item constraint is \( b \) than it is satisfied by above pattern. It is in the form of Equation 13 (Pei et al., 2007).

\[
C_{\text{item}}(\alpha) \equiv (\phi: 1 \leq i \leq len(\alpha), \ a[i] \theta V)
\] OR \[
C_{\text{item}}(\alpha) \equiv (\phi: 1 \leq i \leq len(\alpha), \ a[i] \cap V \neq \phi)
\]  

Where \( V \) is a subset of items, \( \phi \in \{V, \emptyset\} \) and \( \theta \in \{\subseteq, \supseteq, \in, \notin\} \). For the sake of brevity, we omit the strict operators (e.g., \( \subset, \supset \)) in our discussion here. However, the same principles can be applied to them.

2.2.9. Representation of data sequence
Data-sequence A is represented as \( <(a1, t1, m1), (a2, t2, m2), \ldots, (an, tn, mn)> \), where \( (aj, tj, mj) \) means that item \( aj \) is purchased at time \( tj \) with total value \( mj \), \( 1 \leq j \leq n \), and \( tj-1 \leq tj \) for \( 2 \leq j \leq n \). In the data-sequence, if items occur at the same time, they are ordered alphabetically.

Dataset Format (Sequence Database.txt)

\( <1> \) 1 -1 <2> 1 2 3 -1 <3> 1 2 -1 -2
\( <1> \) 1 -1 <2> 1 2 -1 <3> 1 2 -1 -2
\( <1> \) 1 2 -1 <2> 1 2 -1 -2
\( <1> \) 2 -1 <2> 3 -1 -2
\( <1> \), <2> ..... – Time Stamp (day at which transaction occurred)
-1 - End of Transaction
-2 - End of Sequence
1, 2, 3 - Actual Items

3. Proposed system framework

3.1. Strength and working of proposed algorithm
(i) The proposed algorithm being FP-Growth based, Constraint-based prefix span reduces candidate generation and works on projected prefix database.
(ii) First the algorithm scans the database and identifies frequent items. It recursively finds the prefix not only on frequent items but also with consideration of gap constraint which takes care of two adjacent time stamps \((\text{max} \_\text{gap}, \text{min} \_\text{gap})\), with first and last time stamp of prefix sequence having to satisfy compactness constraint.

(iii) The sequence which does not follow such constraints can be pruned at pseudo projection level. This process reduces the database projection cost and search space as compared to sole parameter support threshold thereby increasing efficiency of proposed algorithm.

(iv) Incorporation of Length constraint limits the generation of sequences by pruning the sequences having more length at projection level. Item and Recency act as post processing parameters. Incorporation of such constraints in conventional Prefix Span gives more effective results as per user’s interest.

(v) Proposed Emerging Pattern mining algorithm is identifying those patterns which are not in limelight but have potential to become strong in near future. Such hidden patterns can be highlighted using slight reduction of boundary value of support threshold and inclusion of recency constraint.

3.2. Constraint-based sequential pattern generation

Figure 1 shows three stages of sequential pattern generation. Dotted line and straight line denoted backward flow and forward flow. Figure 2 shows detail of Pre-processing of input data. After generation of sequential input file in format, file is used for sequence generation. Figure 3 and Figure 4 shows flow to generate sequences which satisfied frequency, compactness and gap constraints (FCG—sequences).

Figure 5 shows complete set of FCGLIRM (Frequent, Compact, Gap, Length, Item, Recent, Monitory) —constraint-based sequential patterns. Subjective analysis can be done from different amalgamation of constraints. Knowledge can be extracted from constraint-based patterns. Following thing can be derived from knowledge:

- Loyal customer
- Profitable customer or Profitable purchase
- Simplicity of patterns
- Chain purchase or Influence purchase
- Life style-driven shopping
- Trend analysis

3.3. Proposed algorithm

I. Proposed constraint-based prefix span for SPM

Input: Sequence Database SDB

Values of following Constraints:

Support threshold : \( f \_\text{minsup} \)

Recency support : \( r \_\text{min} \)

Compactness : \((\text{min} \_\text{compact}, \text{max} \_\text{compact})\)

Gap : \((\text{min} \_\text{gap}, \text{max} \_\text{gap})\)

Length : \( l \_\text{min} \)

Quantity : \( q \_\text{min} \)

Item : \( i \_\text{constraint} \)
Output:
(i) The complete set of constraint-based sequential patterns.

Procedure:
Step 1: Find length-1 patterns and remove irrelevant sequences.
(i) Scan the sequence database SDB once to count f-support for each item set.
(ii) Identify patterns as length-1 patterns.
(iii) Infrequent items are removed and generate pseudo-sequence database

Step 2: Divide the set of sequential patterns into subsets
Without considering constraint C, the complete set of sequential patterns should be divided into subsets without overlap according to the set of length-1 sequential patterns (prefix).

Step 3: Construct projected database and mine subsets recursively
(i) Construct projected database for each prefix.
(ii) Recursively generate projected database for each new prefix and mine it to find local frequent patterns.

Step 4: Mine constraint-based sequential Pattern from Projected database and mine subset recursively.
(i) Apply gap and compactness constraint.
   (a) In each projected database, frequent adjacent item i and j, t_i and t_j are time stamped, respectively where, sequence <(i, t_i), (j, t_j)>.
   \[ | t_i - t_j | \leq | \text{min_gap} - \text{max_gap} | \]
(b) In each projected database, time stamp difference of first and last frequent items i and k, $t_i$ and $t_k$ are time stamped respectively, where sequence $\{(i, t_i), (j, t_j), (k, t_k)\}$.
\[ | t_i - t_k | \geq | \text{min}\_\text{compact} - \text{max}\_\text{compact} | \]

(ii) For each frequent item i append it to prefix to generate new prefix in such a way that
a) i can be assembled to the last element of prefix to form a sequential pattern
or
b) $<i>$ can be appended to prefix to form a sequential pattern.

(iii) Recursively generate projected database for each new prefix which satisfies $C_{\text{Compactness}}$ and
$C_{\text{Gap}}$ constraints to find local frequent patterns.
Where, $C_{\text{Compactness}}$ and $C_{\text{Gap}}$ are compactness and Gap constraint.
Repeat step 3 and step 4 recursively.

Step -5: Complete set of sequences $\text{SEQ}_{\text{CFG}}$ satisfied Frequency ($f_{\text{min}}$), Compactness ($\text{min}_\text{compact, max}_\text{compact}$) and Gap ($\text{min}_\text{gap, max}_\text{gap}$)

Step 6: Apply post mining constraints Recency, Length, Quantity and Item to local frequent Patterns.

i. Merge local frequent patterns which satisfy $r_{\text{min}}, l_{\text{min}}, q_{\text{min}}$ to generate global frequent patterns.
   a) Time stamp of last frequent item k is $t_k$. Where, sequence $\langle(i, t_i), (j, t_j), (k, t_k)\rangle$.
   $t_k \geq r_{\text{min}}$
   b) Sequence $\text{SEQ}_{\text{CFG}} = \langle(i, t_i), (j, t_j), (k, t_k)\rangle$
   Length ($\text{SEQ}_{\text{CFG}}$) $\geq l_{\text{min}}$
   c) Sequence $\text{SEQ}_{\text{CFG}} = \langle(i, t_i, \text{qty}_i), (j, t_j, \text{qty}_j), (k, t_k, \text{qty}_k)\rangle$ where, $\text{qty}_i$ is quantity of item i is purchased. $\forall i \in \text{SEQ}$, where $i = \{1, 2 \ldots n\}$
   ($\text{qty}_1 + \text{qty}_2 + \ldots + \text{qty}_n \geq c$

ii. Generate complete set of Constraint-based Sequences $\text{SEQ}_{\text{CFGRLQ}}$
   $\text{SEQ}_{\text{CFGRLQ}} = \text{SEQ}_{\text{CFG}} \cap \text{SEQ}_{\text{CFGL}} \cap \text{SEQ}_{\text{CFGQ}}$
   where, $\text{SEQ}_{\text{CFG}}, \text{SEQ}_{\text{CFGL}}, \text{SEQ}_{\text{CFGQ}}$ satisfied $r_{\text{min}}, l_{\text{min}}, q_{\text{min}}$ respectively.

iii. Apply item constraint to frequent patterns: Check item i is present in global frequent patterns.

Once frequent patterns are generated its check the prescribed item is present in the sequence or not, which should satisfied $i_{\text{constraint}}$.

Sequence $\text{SEQ}_{\text{CFGRLQ}} = \langle(i_1, t_1), (i_2, t_2), \ldots (i_n, t_n)\rangle$, where items are $\{i_1, i_2 \ldots i_n\}$ and $\{t_1, t_2 \ldots t_n\}$ are respective timestamp. Suppose, $i_{\text{constraint}} = i_k$ than,

$i_k \in \text{SEQ}_{\text{CFGRLQ}}$.item where $\text{SEQ}_{\text{CFGRLQ}}$.item $= \{i_1, i_2 \ldots i_n\}$

Step 7: Output the complete set of Constraint-based sequential patterns $\text{SEQ}_{\text{CFGRLQ}}$.

II. Proposed reduced boundary value and recency-based Emerging Patterns (EPs)

Input: Sequence Database $\text{SDB}$

Values of following Constraints:
- Support threshold : $f_{\text{min}}$sup
- Recency support : $r_{\text{min}}$
Output:

(1) Emerging patterns (EPs)

Step 1 to 7

\[ \text{EPGen} (SD, f_{\text{minsup}}, r_{\text{minsup}}) \] //Emerging Pattern generation

Supply the sequence database SD

//sequential database with timestamp

\[ \text{SEQ}_{f_{\text{minsup}}} = \text{Call SequenceGen}(<>, 0, S) \]

\[ \text{boundary}_{\text{sup}} = \text{BoundaryCal} (f_{\text{minsup}}) \]

\[ \text{SEQ}_{\text{boundary}_{\text{sup}}} = \text{Call SequenceGen}(<>, 0, S) \] (Refer Section 2.1, Equation 1)

\[ \text{SEQ}_{\text{boundary}} = \text{SEQ}_{f_{\text{minsup}}} - \text{SEQ}_{\text{boundary}_{\text{sup}}} \]

\[ \text{EP} = \text{SEQ}_{\text{boundary}} \text{.lasttimestamp} \geq r_{\text{minsup}} \]

return EP // Emerging patterns

BoundaryCal \( (f_{\text{minsup}}) \) // Boundary value calculation

\[ \text{boundary}_{\text{sup}} = f_{\text{minsup}} - (0.02 \times f_{\text{minsup}}) \] (Refer Section 2.1, Equation 2)

return boundary_{sup}

3.4. Challenges of proposed Algorithm

I. There is no scientific survey about boundary value reduction in proposed Emerging Patterns mining algorithm. Only domain experts can decide how much reduction is required for a particular application.

II. Reduction of support threshold boundary generates enormous numbers patterns.

4. Simulation study and data analysis

4.1. Justification for choosing the prefix span to modify

Following graph shows the execution time and memory usage of various SPM algorithms namely GSP (Srikant & Agrawal, 1996), SPADE (Zaki, 2001), SPAM (Ayres et al., 2002), SPIRIT (Garofalakis et al., 1999) and FP-Growth-based Freespan (Han, Dong et al., 2000), Prefixspan (Pei et al., 2001). Experiment conducted on four real-time datasets namely Breast cancer, Mushroom, Leviathan, MSNBC and six synthetic dataset generated by IBM generator. IBM generated datasets are described in Table 1 and Table 2. Statistical Description of Breast Cancer Dataset is described in Table 3. Figures 6, 7 and 8 describe performance of SPM algorithms on real-time Breast Cancer Dataset.

4.1.1. Working environment

Algorithms worked on Java environment and tested on an Intel Core Duo Processor with 2GB main memory under Windows operating system.

Following observations can be made from above experiments:

• Fifty-eight percent more time is taken by SPADE algorithm as compared to Prefix span, which is gradually reducing with respect to increasing support values. Approximately 18%–19% more execution time is taken by Prefix Span as compared to SPAM. Almost 68.6% more execution time is taken by GSP as compared by Prefix Span.

• Prefix Span and SPAM are generating same number of frequent sequences. Same way the same number of sequences are generated by SPADE and GSP. Ninety percent less sequences are generated by Prefix Span and SPAM as compare to SPADE and GSP.

• Memory consumption is less by Prefix Span as compare to other algorithms because of generation of pseudo database.
4.2. Comparative study of proposed algorithm with the traditional SPM algorithms

This section empirically compared the proposed Constraint-based Prefix Span algorithm with the traditional SPM algorithm Prefix Span (Zaki, 2001) and constraint-based SPM method RFM (Chen et al., 2009). Experimental study performed on synthetic datasets (describe in Table 4). Table 5, describes parameters of dataset. Time stamp is ignored for Prefix Span because it is design for SPM without timestamp. For ease of comparison, some parameters are kept fixed: Confidence=30% Compactness = (1,5), Gap = (1,3), Recency =2, Length = 3. Seven tests are performed to evaluate the proposed algorithm, which are described in Table 4.

The first test is designed for efficiency and effectiveness analysis in form of, execution time and number of pattern generation respectively. Test II emphasis of scalability analysis. Test III evaluates importance of individual constraint in form of pattern generation. Test IV tries to find out buying behaviour of customer using length constraint. Test V, VI and VII emphasis on Recency and Gap constraints, how number of pattern generation is changing for wider range to smaller range of Gap and for increment of Recency time stamp value.

Test I executed based on six synthetic datasets. Comparison made for run times and pattern generation of three algorithms: proposed constraint based Prefix Span (RFCGL) with RFM and algorithm Prefix Span. Here, result of C10S4T2.5N10 and C50S4T2.5N10 are shown. Rest of the results (for remaining four datasets) are same, (Refer Figure 9).

In Test II, six synthetic datasets were used to perform scalability analysis, which varied the value of |C| (from 10 K to 50 K) for support range (0.010%–0.025%) for 30% confidence value. (Refer Figure 10)

Test III explored how the recency, frequency, compactness, gap and length constraints influence the generation of sequential patterns. CFRGL patterns, CFRG* patterns, CFR** patterns, *FR** patterns and *F*** patterns, where *F*** patterns are the traditional sequential patterns. Table 5 lists the amounts of these five patterns and their corresponding percentages with respect to the traditional pattern (*F***). (Refer Figure 11)

Test VI shows, only one-fourth of patterns are recent and compact. Three-fourth patterns are traditional in C10S4T2.5N10 and C20S4T2.5N10 datasets, which is decreasing by more 5%–7% in remaining datasets. Inclusion of all the constraints reduced the patterns drastically. Only 3%–5% patterns are having length more than 3 means 95% times of purchase happen only ones or twice in sequence. Average 36% of time purchase happens only one time. People are not moving for further purchase. Average 45.25% of the purchase happen two times is moving for third or more time. (Refer Figure 12)

Test V shows, Numbers of sequential patterns are reducing gradually for older to later timestamp. But execution time remains same for confidence and support values. Execution time is getting reduced by reducing support count. Same way number of generated sequential patterns are also reduced. (Refer Figures 13, 14, 15)

Test VI, shows number of patterns generated and execution time for various supports and confidence. More no of patterns are generated for larger gap and it’s reduced for smaller gap. Execution time remains same. Increases of confidence value also generate more number of patterns by 11% and 10% wrt. 20%–40%, 40%–60% confidence value. Decreasing rate of support also decrease execution time by 75%. (Refer Figure 16)

| Table 1. Parameter description of synthetic data-set |
|---------------------------------------------------|
| **Parameter** | **Description** |
| [C] | Number of Customers (1000’s) |
| [S] | Average number of transactions per sequence |
| [T] | Average number of items per transaction |
| [N] | Number of distinct items (1000’s) |
Table 2. Parameter setting of synthetic data generation

| Dataset       | |C| | |S| | |T| | |N| |
|---------------|---|---|---|---|---|---|---|---|
| C10S2T2.5N10  | 10,000 | 4 | 2.5 | 10,000 |
| C20S2T2.5N10  | 20,000 | 4 | 2.5 | 10,000 |
| C30S2T2.5N10  | 30,000 | 4 | 2.5 | 10,000 |
| C40S2T2.5N10  | 40,000 | 4 | 2.5 | 10,000 |
| C50S2T2.5N10  | 50,000 | 4 | 2.5 | 10,000 |
| C10S6T2.5N10  | 10,000 | 6 | 2.5 | 10,000 |

Table 3. Statistical description of Breast Cancer dataset

- Number of distinct items: 16
- Largest item id: 20
- Average number of itemsets per sequence: 9.977110157367669
- Average number of distinct item per sequence: 9.977110157367669
- Average number of occurrences in a sequence for each item appearing in a sequence: 1.0
- Average number of items per itemset: 1.0

Figure 6. Execution time (ms) vs. Support for real-time dataset Breast Cancer.

Figure 7. Number of frequent sequence vs. Support for real-time dataset Breast Cancer.

Figure 8. Memory (mb) vs. Support for real-time dataset Breast Cancer.
In Test VII, we have taken support, confidence as a fix value 0.001 (0.1%) and 0.5 (50%) respectively and gap = <1,3>. More number of patterns are generated for wider range. One percent more number of patterns are generated as compared to smaller range of timestamp e.g. <min,max>:<2,3>–<2,4> and <3,3>–<3,4>. (Refer Figure 17)

4.3. Experiment of emerging patterns (EPs)

Second experiment discovers such patterns which are lies on boundary but does not discover because of little low support value. Such patterns are not in lime light now but might be in high light for future. It’s having potential to become strong in future known as Emerging Patterns (EPs). EPs based on customers’ buying behaviour captured at various support values: 0.1%, 0.08% and 0.05%. Almost 10%–12% patterns are known as Emerging Patterns (EPs), are generated after reduction of boundary value for IBM generated synthetic dataset.

Table 4. Parameter setting for different test

| Test | Dataset | Support       | Confidence | Recency | Compactness | Gap                        | Length |
|------|---------|---------------|------------|---------|-------------|----------------------------|--------|
| I    | C10S4T2.5N10C20S4T2.5N10C30S4T2.5N10C40S4T2.5N10C50S4T2.5N10C | 0.01%–0.025% | 30%        | 2       | (min_comp,max_comp)=(1, 5) | (min_gap,max_gap)=(2, 3) | 3      |
| II   | C10S4T2.5N10C20S4T2.5N10C30S4T2.5N10C40S4T2.5N10C50S4T2.5N10C | 0.01%–0.025% | 30%        | 2       | (min_comp,max_comp)=(1, 5) | (min_gap,max_gap)=(2, 3) | —      |
| III  | C10S4T2.5N10C20S4T2.5N10C30S4T2.5N10C40S4T2.5N10C50S4T2.5N10C | 0.01%–0.025% | 30%        | 2       | (min_comp,max_comp)=(1, 5) | (min_gap,max_gap)=(2, 3) | 2, 3   |
| IV   | C10S4T2.5N10C20S4T2.5N10C30S4T2.5N10C40S4T2.5N10C50S4T2.5N10C | 0.01%–0.025% | 30%        | 2       | (min_comp,max_comp)=(1, 5) | (min_gap,max_gap)=(2, 3) | 0, 2, 3 |
| V    | C20S4T2.5N10C | 0.1%–0.5%     | 30%        | 1, 2, 3 | —           | (min_gap,max_gap)=(2, 3) | —      |
| VI   | C20S4T2.5N10C | 0.1%–0.25%    | 30%        | 1, 2, 3 | —           | (min_gap,max_gap)=(2, 3) | —      |
| VII  | C20S4T2.5N10C | 0.1%          | 50%        | —       | —           | (min_gap,max_gap)=(2, 3) | —      |

Table 5. Number of pattern generation (%)

| Synthetic data-set | CFRGL | CFRG* | CFR** | *FR** | *F*** |
|--------------------|-------|-------|-------|-------|-------|
| Name               | Number of patterns | %     | Number of patterns | %     | Number of patterns | %     | Number of patterns | %     |
| C10S4T2.5N10       | 196   | 4.63  | 1078  | 25.49 | 1116  | 26.38 | 1174  | 27.76 | 4229 | 100   |
| C20S4T2.5N10       | 123   | 3.64  | 761   | 22.54 | 786   | 23.28 | 804   | 23.82 | 3375 | 100   |
| C30S4T2.5N10       | 103   | 2.84  | 594   | 16.41 | 623   | 17.21 | 661   | 18.26 | 3619 | 100   |
| C40S4T2.5N10       | 85    | 2.43  | 547   | 15.52 | 547   | 15.52 | 557   | 15.95 | 3491 | 100   |
| C50S4T2.5N10       | 84    | 2.41  | 541   | 15.57 | 541   | 15.57 | 549   | 15.8  | 3473 | 100   |
| C10S6T2.5N10       | 499   | 8.06  | 1646  | 26.59 | 1796  | 29.01 | 1996  | 32.25 | 6189 | 100   |
After reduction of boundary value more patterns are generated, out of them 36% (C10s4T2.5N10), 18% (C30s4T2.5N10) and 23% (C50s4T2.5N10) new patterns are generated which are known as forming stage patterns. Patterns which are already generated for normal support threshold are discovered at reduced boundary support with change support value. Sixty-four percent (C10s4T2.5N10), 82% (C30s4T2.5N10) and 77% (C50s4T2.5N10) patterns are older patterns. (Refer Figure 18).

5. Conclusion and future direction

Proposed Constraint-based Prefix Span algorithm is not restricted to conventional Sequential Pattern Mining (SPM) parameter frequency but incorporates six more important parameters like Gap, Recency, Compactness/Duration, Profitability, Item and Length. Incorporation of these constraints in FP-growth based—Prefix Span leads to more efficient and effective results by reduction of patterns. Concise patterns present relevant and precise results in terms of users’ interest. Seven different experiments are performed on IBM generated six synthetic datasets. Comparison made for run times and pattern generation of three algorithms: proposed constraint-based Prefix Span with RFM and Prefix Span. Proposed constraint-based Prefix Span algorithm is more efficient and effective in terms of reduction of patterns generation of interesting patterns for user. Experiment studies also reveal that less number of patterns is generated by Duration/Compactness constraint with emphasis on patterns that are active for certain time span. Simulation study of Gap states that: “less number of

Figure 9. (A) number of patterns vs. support (C10s4T2.5N10) (B) Execution time of patterns vs. support (C10s4T2.5N10) (C) number of patterns vs. support (C50s4T2.5N10) (D) Execution time of patterns vs. support (C50s4T2.5N10).

Figure 10. (A) Runtime Vs. |C| (B) number of pattern Vs. |C|.
Figure 11. Number of patterns generated by different datasets

Figure 12. Number of patterns for length constraint.

Figure 13. Number of sequential pattern wrt. recency constraint.

Figure 14. Execution time for various support count for wrt. recency time stamp = 1.
Figure 15. No. of sequential pattern for various support count w.r.t. recency time stamp = 1 (C20S4T2.5N10).

Figure 16. No of patterns for various confidence and confidence for gap constraints.

Figure 17. Number of pattern generated wrt. compactness constraint (C20S4T2.5N10).

people is moving towards another purchase within a short time period but increase in time gap will give more number of purchases.” Recency and Profitability are important parameters to formulate marketing strategies and the proposed algorithm addresses both parameters. Proposed Constrained-based PrefixSpan is scalable in terms of generation of patterns and execution time by varying range of customer from 10K to 50K. Simulation also captures Emerging Patterns (EPs) by reduction of support boundary and recency parameter. Algorithm is also able to identify pattern which are at formative stage. Sequential pattern-based clustering can be helpful for customer segmentation for enabling marketers to target loyal customer groups.
Figure 18. (A) EPs (C10s4T2.5N10) (B) EPs (C30s4T2.5N10) (C) EPs (C50s4T2.5N10) (D) EPs in detail (C10s4T2.5N10) (E) EPs in detail (C30s4T2.5N10) (F) EPs in detail (C50s4T2.5N10).

Supplementary material of this article can be accessed here http://dx.doi.org/10.1080/23311916.2015.1072292.

Funding
The authors received no direct funding for this research.

Author details
Niti Ashish Kumar Desai1
E-mail: nitiadesai@gmail.com
ORCID ID: http://orcid.org/0000-0001-7906-2236

Amit Ganatra2
E-mail: amitganatra.ce@ecchanga.ac.in
ORCID ID: http://orcid.org/0000-0001-7906-2236

1 Department of Computer Engineering, Uka Tarsadia University, Bardoli, Surat, Gujarat, India.
2 U and P. U. Patel Department of Computer Engineering, Charotar University of Science and Technology, Changa, Anand, Gujarat 388421, India.

Citation information
Cite this article as: Efficient constraint-based Sequential Pattern Mining (SPM) algorithm to understand customers’ buying behaviour from time stamp-based sequence dataset, Niti Ashish Kumar Desai & Amit Ganatra, Cogent Engineering (2015), 2: 1072292.

Notes
1. PhD Dissertation report on “The research of customer purchase behaviour using constraint-based SPM Approach,” National Central University, Taiwan by Yen-Liang Chen Department of Information Management. (http://www.ncu.edu.tw)
2. Weinstein and Shane S., Game Plan: How can marketers face the challenge of managing customer value metrics. This article is adapted from Dr. Weinstein’s new book (2012), Superior Customer Value: Strategies for Winning and Retaining Customers (3rd ed.), CRC Press.
3. Akshay Tritiya – It also known as Akha Teej, Hindus considers Akshay Tritya to be a holy and supremely auspicious day. Starting a new activity or buying valuables on Akshay Tritiya is considered to bring luck and success. So, many people buy new gold jewellery on Akshay Tritiya.

Lakshmi Pujan – Lakshmi Puja is third day of Diwali. Worship of the goddess Lakshmi is the main event on Diwali in India. People buy gold and silver, precious gemstones, new utensils of copper, brass and bronze as a sign of good luck, prosperity, money and wealth.

Pushya Nakshtra – Moon transiting in pushya nakshtra is considered as auspicious duration in India. It is believed to be the best time of the year for buying assets in India. Experts from the industry believe that there is a saying
that whoever buys gold on Pushya Nakshatra, survives for generations.

4. Originality in Algorithm I and II are written in bold.

5. IBM Quest Synthetic Data Generator is used to generate synthetic datasets.

http://www.philippe-fournier-viger.com/spmf/datasets/IBM_Quest_data_generator.zip

References

Agrawal, R., & Srikant, R. (1994, September). Fast algorithms for mining association rules. In Proceedings of the International Conference of Very Large Data Bases (pp. 487–499). Santiago.

Agrawal, R., & Srikant, R. (1995, March). Mining sequential patterns. In Proceedings of the 11th International Conference on Data Engineering. Taipei.

Ayres, J., Flannick, J., Gehrke, J., & Yu, T. (2003). Sequential pattern mining using a bitmap representation. In Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Krakow.

Bhensdadia, C. K., & Kosta, Y. P. (2011). Discovering active and profitable patterns with RFM (recency, frequency and monetary) sequential pattern mining: A constraint based approach. International Journal of Information Technology and Knowledge Management, 4, 27–32.

Chen, M., Han, J., & Philip, S. Y. (1996, December). Data mining: An overview from a database perspective. IEEE Transactions on Knowledge and Data Engineering, 8, 866–883.

Chen, R.-S., Wu, R.-C., & Chen, J. Y. (2005). In Proceedings of the 29th Annual International Computer Software and Applications Conference (COMPSAC’05) 0730-3157/05 $20.00 © 2005 IEEE. Washington, DC.

Chen, Y.-L., Kuo, M.-H., Wu, S.-Y., & Tang, K. (2009). Discovering recency, frequency, and monetary (RFM) sequential patterns from customers’ purchasing data. Electronic Commerce Research and Applications, 8, 241–251.

Garofalakis, M., Rastogi, R., & Shim K. (1999). SPIRIT: Sequential pattern mining with regular expression constraints. In VLDB’99. Edinburgh.

Goodman, J. (1990). Leveraging the customer database to your competitive advantage. Journal of Direct Marketing, 55, 26–27.

Han, J., Dong, G., Mortazavi-Asl, B., Chen, Q., Dayal, U., & Hsu, M.-C. (2000). Freespan: Frequent pattern-projected sequential pattern mining. In Proceedings of International Conference of Knowledge Discovery and Data Mining (pp. 355–359). New York, NY.

Han, J., Pei, J., & Yin, Y. (2000, May). Mining frequent patterns without candidate generation. In Proceeding of ACM-SIGMOD. International Conference of Management of Data (pp. 1–12). Dallas.

Hu, H. L. & Chen, Y. L. (2008). Mining typical patterns from databases. Information Sciences, 19, 3683–3696.

Hughes, A. M. (1994). Strategic database marketing. Chicago, IL: Probus Publishing.

Lawrence, R. D., Almasi, G. S., Kotlyar, V., Viveros, M. S., & Duri, S. S. (2001). Personalization of supermarket product recommendations. Data Mining and Knowledge Discovery, 5, 11–32.

Mahdavi, I., Cho, N., Shirazi, B., & Sahebjamnia, N. (2008). Designing evolving user profile in e-CRM with dynamic clustering of Web documents. Data and Knowledge Engineering, 65, 355–372.

Mollick, B., Garg, D., & Grover, B. (2012, July). CFM-prefixspan: A pattern growth algorithm incorporating compactness and monotony. International Journal of Innovative Computing, Information and Control, 8, 1349–1418.

Mary, G., & Iyengar, S. N. (2010, April). A frame work for ontological privacy preserved mining. International Journal of Network Security, 2, 9–20.

Mohammad, S., Hosseini, S., Maleki, A., & Mohammad, R. (2009). Cluster analysis using data mining approach to development CRM methodology to assess the customer loyalty. Expert System with Applications, 37, 5259–5264.

Pei, J., Han, J., Mortazavi, B., Pino, H. (2001). Prefix span: Mining sequential patterns efficiently by prefix-projected pattern growth. In International Conference of Data and Engineering. Heidelberg.

Pei, J., Han, J., & Wang, W. (2007). Constraint-based sequential pattern mining: The pattern growth methods. Journal of Intelligent Information System, 28, 133–160.

Qiao, D., Minai-Bidgoli, B., & Amooee, G. (2012). Developing a model for measuring customer’s loyalty and value with RFM technique and clustering algorithms. The Journal of Mathematics and Computer Science, 4, 172–181.

Raza, K. (2010). Application of data mining in bioinformatics. Indian Journal of Computer Science and Engineering, 1, 114–118.

Shaw, M. J., Subramaniam, C., Tan, G.W., & Weigel, M. E. (2001). Knowledge management and data mining for marketing. Decision Support Systems, 31, 127–137.

Shen, C., & Chuang, H. (2009). A study on the applications of data mining techniques to enhance customer lifetime value. WSEAS Transactions on Information Science and Applications ISSN, 6, 1790–1832.

Song, H. S., Kim, J. K., & Kim, S. H. (2001). Mining the change of customer behaviour in an Internet shopping mall. Expert Systems with Applications, 21, 157–168.

Srikant, R., & Agrawal, R. (1996). Mining sequential patterns: Generalizations and performance improvements. In Proceedings of the 5th International Conference Extending Database Technology (pp. 3–17). London.

Zaki, M. (2001). SPADE: An efficient algorithm for mining frequent sequences. Machine Learning, 40, 31–60.