

Evolving temporal association rules in recommender system

Huiyu Zhou¹  · Kotaro Hirasawa²

Received: 23 December 2016 / Accepted: 4 October 2017 / Published online: 22 October 2017
© The Author(s) 2017. This article is an open access publication

Abstract This research involves implementation of genetic network programming (GNP) and ant colony optimization (ACO) to solve the sequential rule mining problem for commercial recommendations in time-related transaction databases. Excellent recommender systems should be capable of detecting the customers' preference in a proactive and efficient manner, which requires exploring customers' potential needs with an accurate and timely approach. Due to the changing nature of customers' preferences and the differences with the traditional find-all-then-prune approach, the interesting temporal association rules are extracted by the metaheuristics, genetic algorithms-based method of GNP. Additionally, a useful model is constructed using the obtained rules to forecast future customer needs and an ACO approach to evolve the online recommender system continuously. The methodology is experimentally evaluated in a real-world application by analysing the customer database of an online supermarket.

Keywords Data mining · Temporal association rule mining · Genetic network programming (GNP) · Ant colony optimization (ACO) · Recommender system

1 Introduction

An electronic commercial recommendation system [2] attempts to deliver meaningful recommendations to customers regarding items of personal interest (e.g. goods and products). Currently, we are in an era of information explosion due to the rapid development of Internet technology. The vast amount of data not only provides statistics but also offers information and knowledge resources. Recommender system technology can provide information based on customers' personal preferences, previously hidden in databases, and thus has drawn substantial attention.

Data mining and analysis of a large number of transaction data sets can reveal significant patterns and rules underlying these sets. This knowledge can be used by the marketing system to make recommendations to customers and promote sales. A variety of methods exist for recommender systems, which can be generally classified into four main categories [16]: collaborative filtering (CF), content-based (CB), computational intelligence (CI) and hybrid algorithms.

Collaborative filtering methods rely on the similarities of users [7]. In these methods, the similarity between users is computed based on their rating profiles, and then the method recommends items that are highly rated by similar users [1, 29]. In content-based approaches, items are recommended based on items previously preferred by a specific user [24]. For example, as a special case of content-based systems, knowledge-based recommender systems directly compute users' favourite items based on their historic profile [11]. The computational intelligence methods apply intelligence techniques to construct proper recommendation models, including artificial neural networks [30], clustering techniques [9, 31], evolutionary algorithms [17, 28] and fuzzy set techniques [18]. Hybrid

✉ Huiyu Zhou
hyzhou@bjtu.edu.cn

¹ SEM, Beijing Jiaotong University, Haidian District, Beijing 100044, China

² Research Center of IPS, Kitakyushu 808-0135, Japan

technologies integrate CF, CB or CI methods with the goal of developing better recommendations for items [4].

Lots of customers found that frequent pop-up recommendations are annoying. Therefore, only the right in time recommendations are truly valuable. However, a number of studies in traditional recommender systems focus on the problem of which item to recommend [1, 9, 11, 24], while relatively few studies also consider the best time to recommend an item.

Excellent recommenders should be capable of identifying customers' preferences in a proactive and efficient manner, which means exploring customers' potential needs accurately and in a timely manner. For example, if product A_1 has been purchased and after 3 months product A_2 is purchased, then the frequency of this event sequence will be reviewed to determine if the occurrence is over the importance threshold, for example, over 90%, then, it becomes a sequential pattern. If a customer purchased product I_1 , the recommender system will send a promotion of product I_2 3 months later, which is the appropriate timing. Since customers have generally short memories, a recommendation that is provided too early will be useless and annoying, and if the promotion comes too late, customers may have already chosen product I_2 from another vendor.

Additionally, the preferences of customers change over time; thus, it is more appropriate to treat the recommendation problem as a dynamic sequential decision process. Most traditional recommender systems are static [6, 12] since they only constitute information that considers whether a customer purchases a product during a certain period. However, customer demands are dynamic and previously preferred products may no longer be of interest after a certain period, while new demands are gradually developing. Therefore, the quality of recommendations can be improved by using the dynamic sequential information in a continuously evolving approach.

1.1 Challenges and literature review

Association rules without temporal information will be insufficient to predict future customer demand. Additionally, the data have a natural sequential form in historical purchasing databases. Therefore, how the customer interest changes over time among purchasing behaviours is one of the problems that must be a focus of the proposed method.

It is difficult to discover sequential associations from a historical database. As an important area in data mining, sequential rule mining has attracted a great deal of attention and many interesting methods have been proposed. Several of the frequently used methods are as follows: association rule mining [10, 15], sequential pattern discovery [20, 26, 27, 32], inter-transactional mining [8], and

periodic pattern and episode mining [3, 14, 33]. However, few of the existing algorithms consider mining the sequential patterns with concrete time information and they either only incorporate the event sequence one after another or mine with limited time tag information. For example, pattern mining only provides the order for the majority of the purchase sequences with vague time limits. It does not provide information about how long it will be before another purchase occurs after a current event; therefore, we cannot know when the recommender system should recommend another product to promote sales.

For mining sequential patterns, how historical events change in temporal sequences are studied closely. Zaki [34] proposed an approach called SPADE which uses an efficient lattice search technique to transform the database into sub-sequence-list databases. Later, the PrefixSpan model was proposed, which can extract rules by searching the original database only twice [25]. These methods extract sequential rules efficiently; however, they do not attempt to find time intervals among event sequences.

Methods such as the I-PrefixSpan algorithm were proposed to generate time-interval-based sequential patterns [5]. However, the time intervals must be specified by users in advance, and the length of the extracted sequential patterns is limited by the time intervals. The time-gap mining method [21] has also been proposed to obtain time-related sequential patterns. However, this method cannot process repeating items among transaction time sequences.

Possible combinations of events with unlimited time intervals suggest an infinite search space. Unlike the conventional *find-all-then-prune* approach, only a sufficient number of rules will be obtained in the rule generation phase of the proposed methods. Therefore, the proposed method solves this NP-hard problem with a metaheuristics evolutionary-based [36] method, GNP, which only obtains a sufficient set of interesting rules as a solution.

Meanwhile, customer preference is ever-changing, especially for e-commerce businesses. The initial rules can be obtained by offline evaluation of a historical purchasing database; however, since customers' desires change rapidly (possibly caused by fashion, seasonal trends, panic buying, etc.), during the online application of association rules, a dynamic design is necessary.

Recently, hybrid models are proposed in existing studies to deal with online process, for example, [23] captures user's changing preferences over the broadcasting time of TV shows as staying preferences at each interval of the broadcasting. For example, a study [35] of the development of vocabulary for capturing a customer's affective experiences to apparel under design has been proposed. Generic words for affect or emotion can be extracted using effective data mining process. Kim et al. [13] inferred user preferences on genres from implicit online feedback system,

clustered users into user groups on genres and recommended for the group to which each user belongs. A cluster ensembles approach, Hypergraph Partitioning Algorithm [22], is applied on SOM and EM clustering results, and with a multi-criteria CF to improve the online recommendation accuracy.

In another study [38], when new data arrived online, they are used to update the concept lattice and the behaviour of target users, and then, the concept lattice and the rule library are used to calculate the ordered recommendation results, which are returned to the user with a feedback mechanism.

1.2 Features of the proposed method

The proposed method in this paper constructs a hybrid recommendation model using the CI and CB approaches. Similar to the CB methods, items previously purchased by a specific user construct the basic item association profile, and only items matching the profile will be recommended. Important associations among items will be represented as association rules [37], which have the form of {diaper, beer} \rightarrow {milk}, where {diaper, beer} is the antecedent of the rule and {milk} is its consequent.

Firstly, different from other studies, this paper developed a temporal rule mining model to extract important sequential patterns containing not only the recommended items but also the time gaps among them. Sequential patterns are extracted effectively with an evolutionary-based method, Genetic Network Programming (GNP) [36]. The extracted temporal association rules in the rule base implicate the sequential associations among items that construct the profile that guides the recommender system.

Secondly, the proposed method implements a unique optimization method in cope with changing customer preference. The if–then type connection of time-related association rules has features that are similar to the directed connection structure of the ant colony optimization (ACO) method proposed by Merkle [19]. The ACO algorithm is inspired by the behaviour of ant colonies in nature, where a pheromone is accumulated to obtain a better solution and evaporates as time passes. The proposed method incorporates the ACO algorithm to maintain an up-to-date association connection model of the recommender system.

In nature, ants that successfully find food return to their colony while leaving pheromone trails. Other ants tend to follow the trail if it is found. If the followers also obtain food rewards, they reinforce the same trail. Therefore, historical successful search information will pass to other search agents and accumulate over time. Additionally, the pheromone trail will evaporate over time, thus reducing its attractive strength. If there are no longer rewards for following a trail, it will eventually disappear. Thus, ants

(optimization agents) can quickly adapt to the changing environment while maintaining guided information. Pheromone evaporation not only improves adaptivity but also avoids convergence to a local optimal solution.

The idea of the proposed method is to mimic this population behaviour with “simulated ants” that search the temporal association connection among recommended items. The association rules represent important correlations among recommended items, similar to the ant trails. Each time the classification uses the association rules is similar to one instance of an ant searching. A successful recommendation prediction will give positive rewards to the related association connections and strengthen the pheromone trails.

Association connections without successful classification rules will eventually disappear since they no longer fit the changing environment, which in this case, means they are no longer adapted to the changing customer preference. Only continuous useful association connections can survive in this evolving process.

However, the GNP rule mining process continuously runs to replenish new rules whenever new customer information is available, which guarantees that the association connection model has sufficient information to evolve and refine.

The objective of the proposed method is to generate a proper number of rules for the objective item (attribute) during the GNP evolution; thus, the fitness functions are defined to reward the individuals who generate more and better rules. The moment that the temporal association rules are obtained, they are ready to construct an analysis model for the recommendations.

The item association analysis model for a user is constituted by the temporal rules generated in the extraction phase for customer interest forecasting. The rules stored in the rule base are continuously evolving online with the ant colony optimization to adapt to customers’ changing demands.

There are two main features of the proposed mechanism:

1. The method processes customers’ temporal behaviour information in the historical purchasing database. The extracted association rule represents the purchasing behaviour corresponding to products with time tags, which show the correlations among events in concrete time sequences.
2. Offline evaluation of the historical purchasing database extracts rules that only constitute a fundamental item association model for customer behaviour analysis. Due to the ever-changing preference of customers, the ACO and continuing mining process of the GNP evolution in the online application guarantees the

adaptivity and flexibility of the proposed model by pheromone trials and rule replenishment.

The concept definitions and criteria calculations are introduced in Sect. 2. Section 3 is devoted to describing the proposed analysis model constituted by the temporal association rules. The simulation results are discussed in Sect. 4. Finally, conclusions and future work are provided in Sect. 5.

2 Equations and definition

Correlations among purchase events are represented as event sequences in the proposed model, and each time sequence has its corresponding association rules, which are clear and directly understood.

Suppose that $I = I_1, I_2, \dots, I_n$ is a set of event attributes, where n represents the total number of attributes in item set I . They represent the behaviour of purchasing different items. Item set size n can change dynamically if needed, for example, if a supermarket got new supplies, item set size should increase correspondingly. If a customer purchased the corresponding product, the attribute will be 1, otherwise it will be 0.

Let H represent the temporal historical purchasing database. Each time unit represents a user-defined time episode and will be arranged in an ascending order in the database pre-processing phase. Let $T = 0, t_2, \dots, t_k$ be the set of time tags of the events. Then, if item I_i was purchased 5 time units later than the original time point 0, t_i is denoted by $t_i = 5$. How the purchasing behaviour changes in sequence is implied by the ascending order of the time-related database.

Definition: Event Sequence: The temporal correlated event sequence is defined as $Es = \{I_{i(0)}, I_{j(t_1)}, \dots, I_{k(t_k)}\}$, where t_1, \dots, t_k defines how many time units after the original time point, $I_i, I_j, I_k \in I$ and $t_1, \dots, t_k \in T$, and the first time unit is established at 0 to represent the original time point of the sequence.

For instance, suppose event sequence $ESeq_x = \{I_{1(0)}, I_{2(3)}, I_{3(5)}\}$, which indicates that product I_1 was purchased at original time unit 0, product I_2 will be purchased 3 time units later, and the customer will purchase product I_3 an additional 2 time units later, which is 5 time units from the original point. Correspondingly, the time sequence for $ESeq_x$ is $\{0, 3, 5\}$, the event set is $\{I_1, I_2, I_3\}$, and the corresponding event sequence is represented as $\{I_{1(0)}, I_{2(2)}, I_{3(3)}\}$. As a result, the event sequence is defined to study how historical purchasing behaviours influence each other sequentially. The definition implies that multiple items could be purchased at the same time unit, even for the 0

time point. For example, in the event sequence $ESeq_y = \{I_{1(0)}, I_{2(0)}, I_{3(4)}, I_{1(5)}, I_{4(5)}\}$, I_1 and I_2 are purchased at original time point, followed by I_3 at time unit 4, and another 1 time unit later, I_1 and I_4 are purchased together at time unit 5.

Let $I_x (t = t_x)$ represent that attribute I_x occurs at t_x time units later. Therefore, the following association rule can be extracted: $I_i(t = 0) \wedge \dots \wedge I_j(t = t_j) \rightarrow I_c(t = t_c)$, where the consequent part of the rule indicates that the objective item I_c is purchased t_c time units later. Here, $t_c \geq t_j - \dots \geq 0$, the original time unit is 0, and the remainder of the time units in the rule shows their relative time difference from the original point.

The criteria of the rules are executed using support, confidence and Chi-square values. The support criteria guarantee the appearance of frequent sequential patterns and are defined as the proportion of rule r in the database with N time units, i.e. if the occurrence of rule r is θ , then $\text{Support}(r) = \theta/N$. The correlations of attributes in the antecedent and consequent part of rule r are calculated using the confidence value. Confidence is defined as the proportion between the occurrence count γ of the antecedent part and the occurrence count θ of whole rule r , i.e. $\text{Confidence}(r) = \theta/\gamma$.

The Chi-square statistical test is applied in rule evaluation to deny the independency assumption of the antecedent and consequent part (X and Y part) of the rules. If the number of time units in the corresponding database is N , the support values of X , Y and $X \cup Y$ are calculated as x , y and z , respectively. However, if X and Y are independent, we can obtain the support value of $X \cup Y$ as xy . Consequently, we obtain the following:

$$\chi^2 = \frac{N(z - xy)^2}{2xy(1 - x)(1 - y)} \quad (1)$$

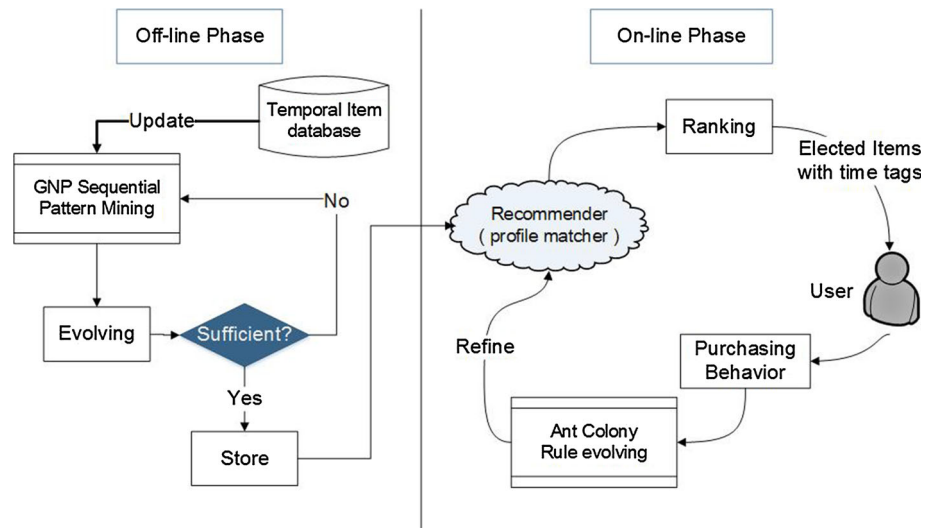
If the Chi-square value of a rule shows a high significance level (99% significance level suggests a Chi-square value higher than 6.63), we can deny the independent assumption of X and Y , and the corresponding rule can be considered as an interesting rule [37].

If the significance level of the event sequence in the rule is important enough, which implies a strong correlation between X and Y , then it will be accepted as an important interesting rule. That is, the event sequence can be considered a time-related association rule if its corresponding three criteria are over the thresholds.

3 Prediction models

Figure 1 describes the general procedure of the proposed method. The first phase occurs offline, where temporal association rules are extracted from the historical customer

Fig. 1 Framework



behaviour database in the rule extraction phase using the GNP-directed graph. New rules will be continuously obtained and recorded in the rule pool for all customers. The new data that arrives will be used to predict the future purchasing behaviour by treating new data as the current situation and matching it with the prediction model constructed by the accumulated rules.

In the online phase, the temporal rules can be provided to constitute the time sequence model that predicts future customer purchasing behaviour and guides the recommender systems to recommend better items for customers at appropriate times. The generated temporal rules establish an important profile for the users, with which the user purchasing behaviour can be estimated and provided to the recommenders.

Although the initial rule extraction phase is executed offline, the stored rules can be applied to analyse customer behaviours after sufficient rules are extracted. The recommendation model adapts to the changing demands of customers by incorporating the ant colony optimization method, in which weak patterns will fade out and strong patterns will be strengthened by the pheromone accumulation process. As a result, the recommendation analysis phase is a continuous online process, as shown in Fig. 1.

3.1 Event sequence and GNP structure

The GNP [37] is an evolutionary algorithm that encodes its individuals as compact directed graphs. Extended from GA and GP, the number of nodes is fixed in GNP; thus, it has a compact structure without bloating problems. The proposed method uses the directed graph structure of GNP to constitute candidate rules. Different from the traditional find-all-then-prune approach, the proposed method does not extract all of the rules; thus, a proper number of rules

will be obtained and applied to effectively constitute the analysis model.

Three types of nodes are included in GNP individuals: the start node, processing nodes and judgement nodes. Every individual starts the calculation process from the start node. Judgement nodes connected by directed connections serve as if-then-else judgement sequences, while processing nodes calculate and record criteria. It is efficient to use the compact structure of GNP to extract association rules. Important building blocks [36] of GNP individuals can be preserved as important associations in the rule extraction process. Meanwhile, effective associations can be reproduced and reconstructed in the subsequent generations with the genetic operations of selection, crossover and mutation.

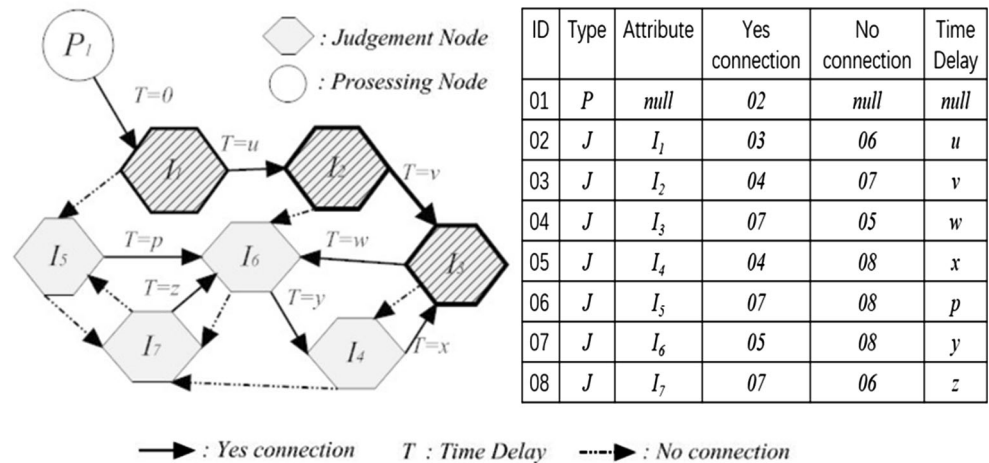
The if-then judgement process in the judgement nodes indicates possible correlations among events. GNP has a directed graph structure that is encoded as a two-dimensional array, as shown on the right side of Fig. 2. Every judgement node has Yes and No connections with different time delays attached to each Yes connection. A judgement node will serve as an if-then judgement process which examines the data in the database.

As shown in Fig. 2, the shaded portion of the GNP individual will be used as the possible X portion of candidate rule r . Combined with the Y portion, which is decided by object class I_c (purchase or not purchase of a certain object or product), the candidate rule can be generated using the GNP structure directly.

Candidate rules are generated from the GNP individuals. In the Initial generation, it will be randomly established, and afterwards, evolutionary algorithm will evolve the GNP structure to obtain more interesting rules.

Generally speaking, each processing node represents the start point of a candidate rule. Since Judgment node of

Fig. 2 GNP encoding



GNP naturally has Yes- and No-side connection based on the judgment result, candidate rules are generated by the Yes-side connection of Judgment nodes. On the other side, the No-side connection of judgement nodes simply denotes a stop sign to candidate rules. During the rule extraction and classification process, whenever judgment node reaches a No connection, its current iteration will be ended. An iteration here is defined as one cycle of executing the classification model to predict customer purchase behaviours.

For instance, the Yes connections of the judgement nodes represent the customers’ purchase behaviour of products, and the time delays among judgement nodes represent the time intervals with respect to these purchasing events. Assuming that the objective item is I_c , start from the Processing node P_1 , the following candidate rule could be generated from the left side of Fig. 2: $I_1(t = 0) \wedge I_2(t = u) \wedge I_3(t = u + v) \rightarrow I_c(t = u + v + w)$. Notice that u , v and w represent the direct time delays among events; therefore, the actual time sequence should be calculated cumulatively.

After a satisfied (occurrence) count search of the database, the criteria can be calculated for all of the candidate rules. If the candidate rule has criteria values higher than the threshold values, it will become an important temporal association rule. Important interesting rules in the proposed methods are stored in the rule pool and will later constitute the analysis model for customer behaviour predictions.

As one of the strengths of GNP evolution, the judgement nodes can be incorporated repetitively to construct candidate rules. Beginning at other processing nodes, even the same directed graph structure will result in different time-related event sequences. Candidate rules can thus be obtained by each GNP individual during the evolution process. There is no need to sort to constitute a sequential candidate rule since the time delays among judgement

nodes in GNP structure are bigger than or equal to 0, thus the candidate rule constructed by GNP is inherently with ascending time tags.

3.2 Rule extraction mechanism

Computing the satisfaction counts is the most essential process for every association rule mechanism. A candidate rule can become an important rule if its corresponding criteria of support, confidence and Chi-square values exceed the given thresholds based on the calculated satisfaction counts.

The satisfaction counts [36] should be examined and calculated by a GNP-directed graph structure based on the training database to judge the importance of the candidate rules. The proposed method introduces a temporal examination mechanism to extract a repeated event sequence in the database in both time and attribute (item) dimensions to address the behaviour prediction problem. Thus, the examination mechanism should consider the time delays and corresponding item concurrently.

Database pre-processing is performed to arrange the database according to different customers to maintain the consistency of event sequences. The purchase behaviours of different customers are meaningless in the sequential database case; thus, the examination process should confirm that the purchase sequences are accomplished by the same customer. This condition is guaranteed by database pre-processing and the mechanism that arranges the purchase events of each customer together and sorts the sequences according to time units, as shown in Fig. 3.

For every customer’s database, the examination mechanism is executed in a similar way. Starting from each time unit, the event sequence is evaluated based on the attributes and corresponding time delays. For instance, to verify the event sequence $ESeq_1$, begin at 0000 of customer 1 in

Fig. 3. I_1 is matched, and 1 time delay later, I_2 is also matched; thus, the corresponding count of a is increased by 1, since it is satisfied by the event sequence beginning at 0000. Therefore, the counts of b and c are also similarly increased. There is no fuzzy matching; thus, if one of the former attributes is not matched at a certain time unit, the event sequence will be considered to be a mismatch, and there is no need to examine the remainder of the event sequence, as shown in the example of the event sequence beginning at 0002 of customer 1. The mechanism will advance to the next time unit and begin a similar examination. Notice that all of the customer databases will be processed; thus, the same event sequence will have different counts in different customer databases, as shown in Fig. 3.

As shown in the example in Fig. 3, two customers are processed simultaneously for the satisfaction counts. Different customers have different purchasing behaviours; therefore, their matching counts should be archived separately during the process and the corresponding rules that are generated should be stored by different customers in the rule base.

The event sequence has a length of four attributes, as shown in Fig. 3; however, since time intervals are attached to each attribute, visiting the same judgement node recursively represents a reasonable behaviour in the time sequences database, e.g. purchasing the same product repeatedly. Therefore, recursive connections are allowed, and there is no limit to the length of possible rules, which indicates infinite possible combinations. After the counts for one event sequence are computed, the counts for all of its sub-sequences are automatically computed; thus, sub-

association rules could be generated conveniently without additional examination.

After calculating the satisfaction counts, the proposed method computes the criteria for each candidate temporal association rule and if all of its criteria are higher than a certain threshold, the candidate rule is guaranteed to be important enough as a temporal association rule to be stored in the rule base.

3.3 Fitness functions of evolution

Since the GNP structure is incorporated to generate association rules instead of finding optimal solutions, the fitness function shows unique characteristics in GNP mining. All GNP individuals will contribute to the rule extraction; therefore, rules can be obtained and recorded beginning at the first generation of evolution. As a result, GNP data mining does not attempt to obtain elite individuals or prediction models during evolution. The evolution goal in the proposed method is to extract a maximum number of rules.

The goal is to generate as many qualified rules as possible and store all the extracted rules in the rule pool through the evolution process. We considered Chi-squared value as the most important criteria since it calculates the correlations. Therefore, we choose to use Chi-squared value instead of support in the fitness function.

Same rule may be extracted repeatedly during the evolution, especially for those short rules. However, each rule should be new only one time, thus the a_{nova} reward is added for individuals actually generated new rules. What's more, long rules are preferred with better ability to predict purchasing behaviours. In the interest of keeping diversity of

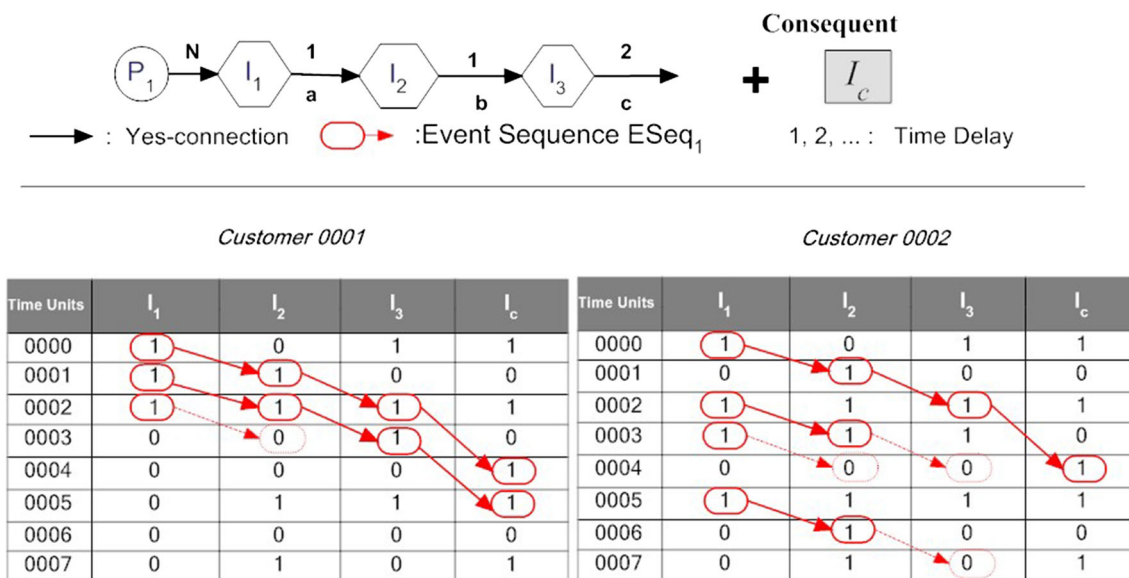


Fig. 3 Examination process

the evolution, rules that contain more than the N_a attribute are defined as multiple rules, we give multiple rule reward α_v , and individuals generate multiple rules will be preferred during the evolution process.

Thus, individuals with the ability to generate many new and diversified rules will survive to reproduce descendants. Then, each individual is evaluated as follows:

$$\text{Fitness} = \sum_{i \in \text{RS}} (\eta \times \chi^2(r) + \lambda \times N_a(r) + \alpha_{\text{nova}}(r) + \alpha_v(r)) \quad (2)$$

where RS: set of interesting rules that satisfy the importance thresholds, $\chi^2(r)$: Chi-square value of the extracted rule r , η : Chi-square coefficient, λ : rule length coefficient, $N_a(r)$: rule length of the extracted rule r , $\alpha_{\text{nova}}(r)$: rewards, which are defined as $\alpha_{\text{nova}}(r) = \begin{cases} \alpha_{\text{nova}} & \text{if rule } r \text{ is a new rule} \\ 0 & \text{otherwise} \end{cases}$, $\alpha_v(r)$: rewards, which are defined as $\alpha_v(r) = \begin{cases} \alpha_v & \text{if rule } r \text{ is a multiple rule} \\ 0 & \text{otherwise} \end{cases}$.

Initial individuals are randomly generated; however, compact structure of GNP individual makes them possible to generate number of rules. The attributes constitute the rules will naturally be frequent items, for example, given the rule of $I_1(t=0) \wedge I_2(t=\tau) \wedge I_3(t=\tau+\varphi) \rightarrow I_c(t=\tau+\varphi+\varepsilon)$, for the rule satisfy criteria thus the item I_1, I_2, I_3 must also be frequent, then in the next generation, when constructing GNP individuals during mutation, I_1, I_2, I_3 will have higher probability to be chosen. The possibility of choosing attributes for judgement nodes is updated each generation based on the attributes' ability to constitute new rules; this is the attribute accumulation mechanism(AAM) to improve the evolution efficiency, for the more details of AAM please refers to [37].

By AAM algorithm, the attributes constitute more rules will have higher probability to construct individuals in GNP evolution. Therefore, the better attributes, which consistently have high correlation with other attributes and have the penitential to constitute more rules, are accumulated through evolution and consequently have more chance to generate new rules. As a result, important building blocks will be preserved generation by generation.

Individuals who failed to obtain enough new rules will be immediately eliminated, and the population is updated without retaining any elite individuals (elite refers to several of the best individuals which live through generations prevented from death in some evolutionary algorithms). In the proposed method, all individuals will perish each generation, but only the best individuals can reproduce offspring to inherit important building blocks (e.g. important combination of items who has potential to generate more new rules). Therefore, the attributes(items) that no

longer generate new rules will also naturally vanish from the population since they can only generate the same old rule.

AAM is applied to exploit already accumulated information and increase extraction efficiency, on the other hand, in order to constantly find new rules, α_{nova} reward, multiple rule reward and no elite evolution process is implemented to strengthen the exploration. Exploration and exploitation are carefully balanced in the proposed method.

Four types of genetic operators are used in this process, including the conventional crossover, judgement function mutation, directed graph connection mutation and the time delays of judgement nodes mutation. The time delays of judgement nodes mutation are designed to evolve the time units among attributes which imitate the event sequence in the historical purchasing databases.

3.4 Evolving rules and ant colony optimization

Generally, association rule mining is an offline approach as all of the rules are extracted from a training database and separately applied to the testing database after the extraction phase is complete. However, customer preference is continuously changing and, with the evolutionary rule extraction process, new time-related rules can always be explored since the possible combinations are infinite. The proposed approach evolves the rule base with ant colony optimization, while new rules are continuously being generated during the rule extraction phase; thus, rule extraction and rule evolution are performed simultaneously, which is an online approach.

All of the rules are stored together after extraction and they constitute the prediction models for each class (purchase or not purchase certain goods or products) in the database and undergo the ant colony optimization evolution process. Recommendations can be made any time after the rules are extracted and the rules are adapted and evolved continuously based on the changing environments, which in this case is the shifting demand of customers.

As a result, the prediction models are composed of three stages: in the first stage, all of the rules related to each class build an event sequence classification model, as shown in Fig. 4. The consequent part of each extracted rule is the corresponding objective class, which can be treated as the prediction result. The second stage is to build the classification model, and third is the prediction stage which will be introduced in detail later.

If the antecedent attributes of a rule satisfy the occurrence of an event sequence in the testing data, it is called a satisfied rule. Then, the satisfaction situation of extracted rules in each class will be examined using testing data. According to the temporal rule, if a rule is satisfied, it will

sequentially classify the prediction result to its corresponding consequent class.

Since multiple rules can satisfy the same testing data, it is difficult to determine which consequent class of satisfied rules should be selected. Unlike other methods that try to prune less important association rules, the proposed method utilizes all of the extracted rules in the classification model. Each rule represents an event sequence. The model of Iteration 1 in upper left side of Fig. 4 is constructed by two event sequences: $\{I_{1(0)}, I_{2(2)}, I_{3(5)}, I_{4(6)}, I_{c(7)}\}$ and $\{I_{1(0)}, I_{4(1)}, I_{c(2)}\}$.

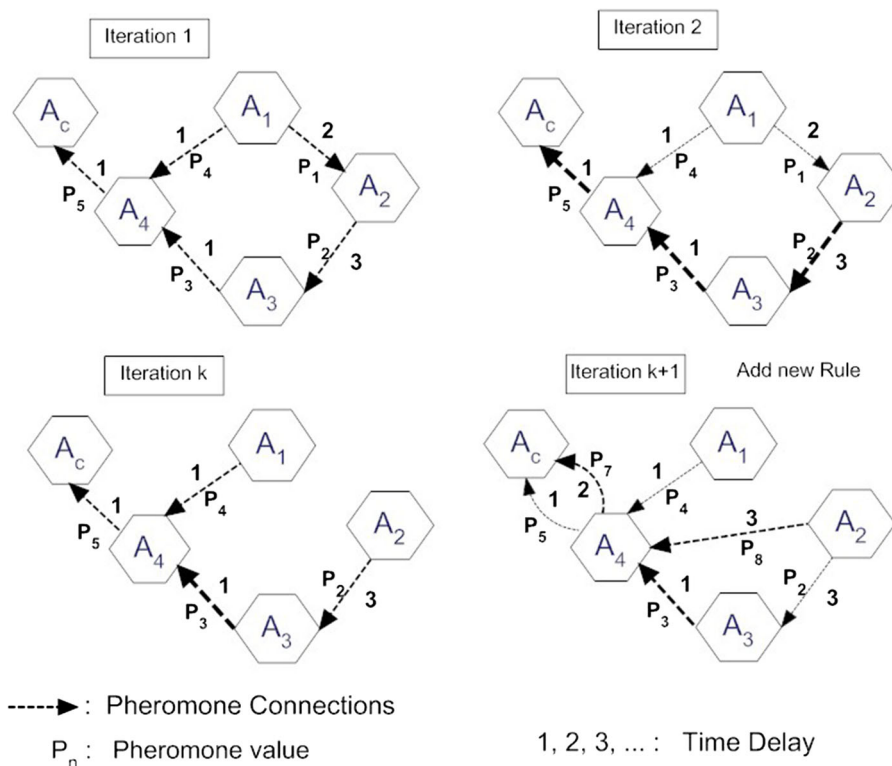
Second, the system uses the classification model to predict customer behaviours and to obtain prediction feedback, where the performance of the classification model is measured through analysis of the results of applying the model to the testing database. The event sequences that contribute to the accurate prediction results are similar to successful ants in a colony and will leave pheromone trails to guide the subsequent prediction process. Consequently, as shown in Iteration 2 in the upper right side of Fig. 4, the pheromones of successful

classification will be strengthened and will influence the subsequent classification process. P_n shows how many pheromones are accumulated in a certain event sequence, which is vividly represented by the thickness of the connections in Fig. 4. An iteration is defined as one cycle of executing the classification model to predict customer purchase behaviours.

After k iterations, due to pheromone evaporation and good event sequence accumulation, the structure of the analysis model can change. To avoid the disappearance of all of the event sequences while maintaining the adaptivity of the model, when new rules are extracted, they can be added to the model immediately. For example, a new rule of $I_2(t=0) \wedge I_4(t=3) \rightarrow I_c(t=5)$ emerged in the analysis model in Iteration $k+1$, as shown in Fig. 4.

This process is described more clearly in Algorithm 1. Each rule is rated by its confidence value and pheromone as calculated by $Score[c]$ and contributes to the final classification result. The system will choose the highest w classes and recommend the corresponding goods and products to customers.

Fig. 4 Event sequence classification model and ant colony optimization



Algorithm 1 Ant Colony Evolution and Classification (RuleBase [][])

```

1: long Score[ ];
2: double pheromone[ ][ ], normalized pheromone[][]; //definition
3: initialize pheromone[][] and normalized pheromone[][] to all 1 // initialization
4: int c=0; // c: counter for classes
5: while (classes!= C) do // C: represents the number of all of the objective classes
6:     int total=0;
7:     while (total!= rule number in current class) do //total is the count of rules in current class
8:         Rule sequences= RuleBase[c ][ rule]; // for all the rules of class c
9:         if (rule sequence is satisfied) // add coefficient of the rule to the score
10:            // coefficient is constituted by confidence and pheromone
11:            Score[c]+=association.confidence*normalized pheromone[c][count];
12:            Set flag of the rule to true; //of class c and set used flag
13:        end if
14:        total++;
15:        Score[c]=Score[c]/C; // Score is calculated for current class c
16:    end while
17:    c++; // move to process the next item
18: end while
19: sort(Score[ ]); // sort scores from high to low
20: Choose the class with highest score values as results;
21: if classification results is true
22:     for all the rules in rule pool RuleBase [ ][ ];
23:         if (flag==true)
24:             pheromone[ ][ ]+=incremental; // good information is enhanced
25:         end if
26:     end for
27: end if
28: evaporate pheromone for all the rules; //all of the rules evaporate their pheromones
29: normalizing pheromone and obtain normalized pheromone[][]; // normalization

```

Meanwhile, all of the rules also evaporate slowly using ant colony optimization; thus, the model avoids becoming trapped in the local optimal of the optimization process. The association rules that are no longer useful due to changing customer preferences will eventually fade out, as shown in Iteration k of Fig. 4, where the sub-sequence of $\{I_{1(0)}, I_{2(2)}\}$ gradually disappears in the prediction model. Notice that although it takes all of the sequences of $\{I_{1(0)}, I_{2(2)}, I_{3(5)}, I_{4(6)}, I_{c(7)}\}$ to construct the classification model, part of the sequence can disappear without removing the entire sequence due to different behaviours in the testing databases.

Third, the testing databases become the new training databases over time, and new association rules are extracted. Those rules can be stored and used for the online prediction model; thus, the proposed process returns to stage one, where the original testing database becomes the training database and fresh new data becomes the new

testing database. This is a closed circle process with a continuous evolution that ensures the online feature of the proposed method.

4 Empirical study

The proposed prediction method attempts to predict changing customer purchasing behaviours; thus, it not only provides prediction results on purchasing behaviour but also provides an insight analysis model for future possible precision marketing. A real-world database is studied in this section to evaluate the performance of the application of the proposed method.

4.1 Database and parameter descriptions

The database used in the empirical study is collected from a small online supermarket that tracked its customers'

purchasing behaviours for more than 9 years. The databases in this application contain purchasing behaviours of 784 loyal customers and 241,382 purchase transactions from January 2001 to June 2009, including 99 different types of groceries. Closer examination of the database suggests that customers usually purchase goods once per week or twice per month, which is consistent with the normal glossary purchasing pattern. Thus, in this simulation, the proposed method establishes the time unit at 2 weeks, which guarantees at least one purchase in each time unit (occasionally empty time units are allowed). Eighty per cent of the database is used for training, and 20% is used for continuous testing.

The entire duration is 8.5 years, and the time unit in the experiment is 2 weeks; thus, there are 204 time units for 784 customers. In other words, a total of $204 \times 784 = 159,936$ time units are used in this simulation. Table 1 provides the corresponding parameters for the evolutionary process and the database characteristics. Here, the support value is set to 0.03 because the database is large; thus, the optimal value is obtained by experimental trials. Individual and population parameters are established by the general case of GNP evolution, which shows the proper performance.

4.2 Results

Since the goal of GNP data mining is not to find the optimal model, the fitness values of individuals do not convergence as with other evolutionary methods and new association rules can be continuously extracted to the overall rule pool. Consequently, the effectiveness of the evolution is assessed by the number of rules it generated.

As shown in Fig. 5, a sufficient number of rules are extracted and stored. The rules are achieved via a weighted average method to constitute temporal classification models. Thus, all of the obtained rules will contribute to the final classification results with their corresponding coefficients of importance levels.

Using the historical purchasing records, the antecedent portion of each association rule will be matched to the testing database time unit by time unit. If the rules are satisfied, the weighted scores will be added to their corresponding consequent portion, which produces the final

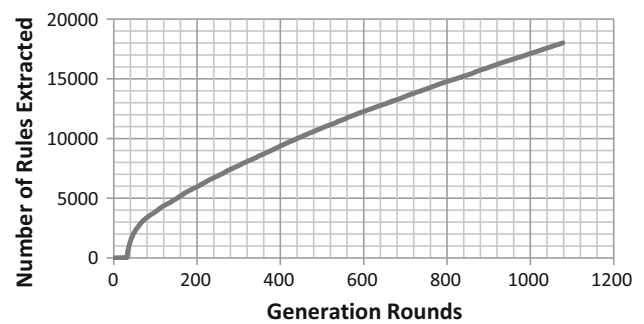


Fig. 5 Extracted association rules for the generation rounds

prediction results, and pheromone is accumulating and evaporating using ACO algorithm as described in Sect. 3.4. The continuously evolving process is executed over 800 times and the possible combination of rules is infinite; thus, the number of extracted rules never reaches convergence.

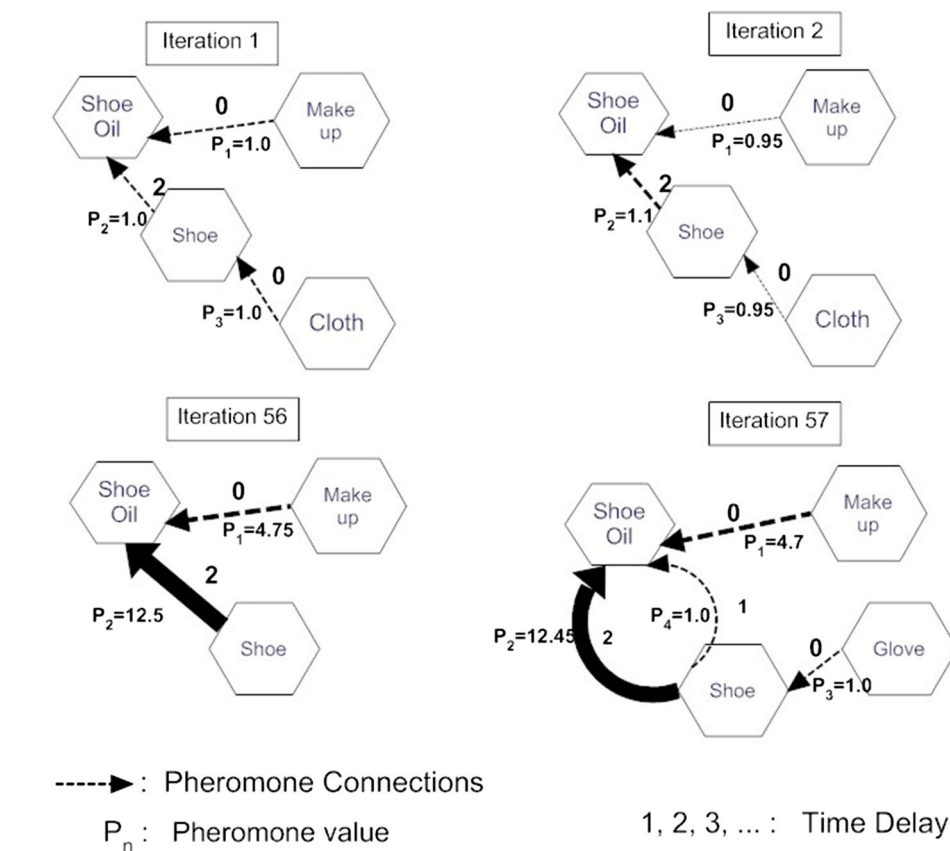
For a concrete example, a shoe oil purchasing situation is considered. The satisfied association rules for the objective item, shoe oil, constituting the analysis model are shown in Fig. 6. There are a total of 89 association rules for the shoe oil purchasing situation for all customers; however, only 2 of these rules are satisfied according to the current iteration in Iteration 1, which is $Cloth(t = 0) \wedge Shoe(t = 0) \rightarrow Shoe\ oil(t = 2)$ and $Make\ up(t = 0) \rightarrow Shoe\ oil(t = 0)$. The first rule suggests that the customer usually buys a cloth and shoes together and purchases shoe oil after 1 month. The second rule suggests that the customer usually buys make-up and shoe oil together. Notice that the second rule is similar to the conventional recommendation system, which actually shows how goods are purchased together without the time consideration. Thus, the proposed method is capable of both a conventional recommendation and a recommendation that considers the time aspect. Satisfied rules are similar to successful ant agents that found food in an ant colony; they leave pheromones to guide the subsequent agents. Thus, P_1, P_2 and P_3 are set to an initial value of 1.0.

In Iteration 2, only a portion of the second rule is satisfied as follows: $Shoe(t = 0) \rightarrow Shoe\ oil(t = 2)$; thus, only a portion of the event sequence has its pheromone of P_2 strengthened, as shown in Fig. 6. The pheromones corresponding to unsatisfied event sequences evaporate gradually. Therefore, after several iterations, the analysis

Table 1 Parametre settings for application

Parameters for evolution		Parameters for database	
Judgement nodes in individual	100	Time unit length	2 weeks
Processing nodes in individual	10	Number of time units	159,936
Population	100	Number of customers	784
Threshold of Chi-square test	6.63	Number of transactions	241,382
Threshold of support	0.03	Number of products	99

Fig. 6 Example of an evolving analysis model



model changes to the shape shown in Iteration 56, where strong time-related associations are strengthened with high pheromone values of 12.5 and 4.75. In Iteration 57, the new rule of $Glove(t=0) \wedge Shoe(t=0) \rightarrow Shoe\ oil(t=1)$ is extracted and satisfied since this new behaviour can be observed from the customer behaviours. Therefore, the manner in which important event sequences (correlations with time tags) change over time can be examined intuitively in the model.

Figure 6 concretely shows how the behaviour of products purchased is extracted by the association rules. The proposed method applies the rules to build analysis models and vividly shows where the event sequence begins, how it changed based on the changing database, and finally, in which order. This provides assistance to punctual recommendation systems and precision marketing.

The training process using GNP evolution to obtain original rule pool from the first 80% of database, which is relatively time-consuming, thus each generation cost around 250 ms as shown in Table 2, after the 1500 rounds of generations, enough number of rules have been obtained. The offline training process has ended.

Table 2 further analysed the result of offline and online time complexity. As for Simple GNP method without continuous GNP and ACO, it directly applies the obtained rules for online prediction. While continuous GNP and

continuous GNP + ACO method both have online training and prediction process, the rest 20% of the database will be used for continuous GNP evolving and cost around 250 ms per generation which is acceptable for online requirement. At least after one online generation, newly generated rules will be immediately added to the rule pool, and the prediction process can be executed which cost almost the same execution time for three of the compared methods within 30–40 ms for each prediction. Since online process predicts the last 20% of the database which has 159,936 time units as shown in Table 1. Therefore, a total of $159,936 \times 20\% = 31,987$ predictions are produced.

Table 3 shows that the proposed method obtains proper prediction accuracy based on different confidence values. Accuracy is calculated as follows: if the recommender system recommends a certain item to the user at the corresponding time point and the user actually purchases the item, then the prediction accuracy is 100%; otherwise it is 0%. The calculation of accuracy is performed for all of the users' behaviours at all time units; thus, it is an averaged performance criterion.

Different than the support and Chi-square criteria described in Sect. 2, when extracting temporal association rules, the confidence value will be the key factor that influences the accuracy of the user behaviour prediction. The support criteria guarantee the frequency of the

Table 2 Offline and online time complexity (unit: ms)

Offline training		Online training or/and prediction		
Simple GNP	245.2 per generation	Simple GNP	Online training	None
			Prediction	30.2
Continuous GNP	246.13 per generation	Continuous GNP	Online training	245.78 per generation
			Prediction	31.2
Continuous GNP + ACO	246.95 per generation	Continuous GNP + ACO	Online training	247.23 per generation
			Prediction	40.69

Offline training		Online training or/and prediction		
Simple GNP	245.2 per generation	Simple GNP	Online training	None
			Prediction	30.2
Continuous GNP	246.13 per generation	Continuous GNP	Online training	245.78 per generation
			Prediction	31.2
Continuous GNP + ACO	246.95 per generation	Continuous GNP + ACO	Online training	247.23 per generation
			Prediction	40.69

Table 3 Average prediction accuracy

	Average prediction accuracy			
Confidence	0.6	0.7	0.8	0.9
Training	71.85	76.37	81.45	89.56
Testing				
Simple testing	61.71	69.20	73.35	80.29
Continuous GNP	63.32	71.21	75.52	82.45
Continuous GNP + ACO	70.3	75.69	80.11	87.05

temporal rule, the Chi-square value ensures that the items in the rule are positively related, and finally, the confidence value evaluates the extracted rule’s precision. Therefore, confidence thresholds of 0.6, 0.7, 0.8 and 0.9 are studied separately. The rest 20% of the database which was tested for online prediction demonstrate that due to the changing preference of customers, the rules adapted to the former 80% of the database are no longer suitable for the remaining 20% of the database, with a decrease in the prediction results. From Table 3, we see that the continuously evolving GNP with newly extracted rules can improve the performance and that the proposed method, which combines continuous GNP with Ant Colony Optimization, can further improve the overall performance of the prediction.

Increasing the confidence threshold with other fixed criteria results in a different accuracy performance, as seen in Table 3. The GNP method without continuous evolution shows good performance in the training phase, but not as precise a performance in testing. Continuous rule mining shows an improvement in prediction accuracy for the testing database, where the extracted rules are a larger portion of the database and testing occurs with the remainder. Continuous GNP combined with ACO shows the best prediction accuracy in the testing phase.

5 Conclusions

In this paper, the continuous online recommendation model has been proposed using association rules with concrete time tags. Sufficient important temporal association rules that consider time-related intervals can be extracted efficiently using an evolutionary-based GNP mining method.

After rule extraction, the stored rules are used to build a sequential prediction model to analyse which class (item) customers will purchase in time sequence order. The real-world data application section has verified that the proposed methods can be used for the customer behaviour analysis. Changing customer behaviour can be predicted accurately by building the proposed analysis model. Applying the proposed method, we not only know which items to sell to customers but also when to use strategies

for these sales. As a result, precision marketing can be executed at a specific time to avoid inefficient promotions.

Ant colony optimization has been used to maintain the adaptivity of the prediction model to the changing demand of customers. Even after rules are extracted, they are still evolving using ACO online. By combining the continuous GNP and ACO, the proposed prediction model maintains up-to-date accuracy in the recommendation system.

Future works of the proposed method include clustering the obtained time-related rules to extract similar patterns among customers, and then using these information for future recommendations. Another idea is to denote certain products as signals to life events (such as purchasing diapers is a likely indicator of a birth of a baby). Once a single or a sequence of signal products have been purchased, based on rules from other customers, recommendations can be made to the customer.

Acknowledgements This work was supported by Beijing philosophy and Social Science Foundation [Grant Nos. 15JGC166, 15JGA016]; the Fundamental Research Funds for the Central Universities [Grant No. 2015jbwy007]; and Chinese National Natural Science Foundation [Grant No. 61602028].

Compliance with ethical standards

Conflict of interest We declare that we have no financial and personal relationships with other people or organizations that can inappropriately influence our work, there is no professional or other personal interest of any nature or kind in any product, service and/or company that could be construed as influencing the position presented in, or the review of, the manuscript entitled, “Evolving Temporal Association Rules in Recommender System”.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

- Bobadilla J, Ortega F, Antonio H, Bernal J (2012) Generalization of recommender systems: collaborative filtering extended to groups of users and restricted to groups of items. *Expert Syst Appl* 39(1):172–186
- Bobadilla J, Ortega F, Hernando A, Gutiérrez A (2013) Recommender systems survey. *Knowl Based Syst* 46:109–132
- Cao H, Mamoulis N, Cheung DW (2007) Discovery of periodic patterns in spatiotemporal sequences. *IEEE Trans Knowl Data Eng* 19(4):453–467. doi:10.1109/TKDE.2007.1002
- Chandak M, Girase S, Mukhopadhyay D (2015) Introducing hybrid technique for optimization of book recommender system. *Procedia Comput Sci* 45(1):23–31
- Chen YL, Huang TCK (2005) Discovering fuzzy time-interval sequential patterns in sequence databases. *IEEE Trans Syst Man Cybern B Cybern* 35(5):959–972
- Cohen WW, Fan W (2000) Web-collaborative filtering: recommending music by crawling the Web. *Comput Netw* 33(1):685–698. doi:10.1016/S1389-1286(00)00057-8
- Deshpande M, Karypis G (2004) Item-based top-N recommendation algorithms. *ACM Trans Inf Syst* 22(1):143–177
- Feng L, Li Q, Wong A (2001) Mining inter-transactional association rules: generalization and empirical evaluation. *Data warehousing and knowledge discovery*. Springer, Berlin
- Ghazanfar MA, Prügel-Bennett A (2014) Leveraging clustering approaches to solve the gray-sheep users problem in recommender systems. *Expert Syst Appl* 41(7):3261–3272
- Harms SK, Deogun JS (2004) Sequential association rule mining with time lags. *J Intell Inf Syst* 22(1):7–22
- Hsu S, Wen M, Lin H, Lee C, Lee C (2007) AIMED-A personalized TV recommendation system. In: *Interactive TV: a shared experience*, pp 166–174
- Huang Z, Chen H, Zeng D (2004) Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering. *ACM Trans Inf Syst* 22(1):116–142. doi:10.1145/963770.963775
- Kim E, Pyo S, Park E, Kim M (2011) An automatic recommendation scheme of tv program contents for (ip) tv personalization. *IEEE Trans Broadcast* 57(3):674–684
- Laxman S, Sastry PS, Unnikrishnan KP (2007) Discovering frequent generalized episodes when events persist for different durations. *IEEE Trans Knowl Data Eng* 19(9):1188–1201
- Lin W, Alvarez SA, Ruiz C (2002) Efficient adaptive-support association rule mining for recommender systems. *Data Min Knowl Discov* 6(1):83–105
- Lu J, Wu D, Mao M, Wang W, Zhang G (2015) Recommender system application developments: a survey. *Decis Support Syst* 74:12–32
- Lv G, Hu C, Chen S (2016) Research on recommender system based on ontology and genetic algorithm. *Neurocomputing* 187:92–97
- Martinez-Cruz C, Porcel C, Bernabé-Moreno J, Herrera-Viedma E (2015) A model to represent users trust in recommender systems using ontologies and fuzzy linguistic modeling. *Inf Sci* 311(March):102–118
- Merkle D, Middendorf M, Schmeck H (2002) Ant colony optimization for resource-constrained project scheduling. *IEEE Trans Evol Comput* 6(4):333–346. doi:10.1109/TEVC.2002.802450
- Nakamura S, Nozaki K, Morimoto Y, Miyadera Y (2014) Sequential pattern mining method for analysis of programming learning history based on the learning process, pp 55–60
- Nguyen TTS, Lu HY, Tran TP, Lu J (2012) Investigation of sequential pattern mining techniques for web recommendation. *Int J Inf Decis Sci* 4(4):293–312
- Nilashi M, Fard KB, Rahmani M, Rafe V (2017) A recommender system for tourism industry using cluster ensemble and prediction machine learning techniques. *Comput Ind Eng* 109:357–368
- Park Y, Oh J, Yu H (2017) Rectime: real-time recommender system for online broadcasting. *Inf Sci* 409–410:1–16
- Pazzani MJ, Billsus D (2007) Content-based recommendation systems. *Adapt Web* 4321:325–341. doi:10.1007/978-3-540-72079-9
- Pei J, Han J, Mortazavi-Asl B, Wang J, Dinto H, Chen Q, Dayal U, Hsu M (2004) Mining sequential patterns by pattern-growth: the prefixspan approach. *IEEE Trans Knowl Data Eng* 16(11):1424–1440
- Pei J, Han J, Wang W (2007) Constraint-based sequential pattern mining: the pattern-growth methods. *J Intell Inf Syst* 28(2):133–160
- Perera D, Kay J, Koprinska I, Yacef K, Zaiane OR (2008) Clustering and sequential pattern mining of online collaborative learning data. *IEEE Trans Knowl Data Eng* 21(6):759–772

28. Salehi M, Pourzaferani M, Razavi SA (2013) Hybrid attribute-based recommender system for learning material using genetic algorithm and a multidimensional information model. *Egypt Inform J* 14(1):67–78
29. Schafer JB, Dan F, Herlocker J, Sen S (2004) Collaborative filtering recommender systems. *Acm Trans Inf Sys* 22(1):5–53
30. Shambour Q, Lu J (2011) A hybrid trust-enhanced collaborative filtering recommendation approach for personalized government-to-business e-services. *Int J Intell Syst* 26(9):814–843
31. Shinde SK, Kulkarni U (2012) Hybrid personalized recommender system using centering-bunching based clustering algorithm. *Expert Syst Appl* 39(1):1381–1387. doi:[10.1016/j.eswa.2011.08.020](https://doi.org/10.1016/j.eswa.2011.08.020)
32. Srikant R, Agrawal E (1996) Mining sequential patterns: generalization and performance improvements. In: 5th international conference on Extending Database Technology (EDBT'96), pp 3–17
33. Tzima FA, Mitkas PA, Voukantsis D, Karatzas K (2011) Sparse episode identification in environmental datasets: the case of air quality assessment. *Expert Syst Appl* 38(5):5019–5027
34. Zaki MJ (2001) SPADE: an efficient algorithm for mining frequent sequences. *Mach Learn* 42(1–2):31–60. doi:[10.1023/A:1007652502315](https://doi.org/10.1023/A:1007652502315)
35. Zhao J, Song J, Montazeri A, Gupta MM, Lin Y, Wang C, Zhang WJ (2017) Mining affective words to capture customer's affective response to apparel products. *Text Res J*. doi:[10.1177/0040517517712092](https://doi.org/10.1177/0040517517712092)
36. Zhou H, Hirasawa K (2014) Traffic conduction analysis model with time series rule mining. *Expert Syst Appl* 41(14):6524–6535
37. Zhou H, Mabu S, Shimada K, Hirasawa K (2011) MBFP generalized association rule mining and classification in traffic volume prediction. *IEEJ Trans Electr Electron Eng* 6(5):457–467
38. Zou C, Zhang D, Wan J, Hassan MM, Lloret J (2017) Using concept lattice for personalized recommendation system design. *IEEE Syst J* 11(1):305–314