Multimedia Systems (2013) 19:527–542
DOI 10.1007/s00530-013-0311-7

REGULAR PAPER

Automatic and personalized recommendation of TV program contents using sequential pattern mining for smart TV user interaction

Shinjee Pyo · Eunhui Kim · Munchurl Kim

Published online: 19 February 2013

Abstract Due to the excessive number of TV program contents available at user's side, efficient access to the preferred TV program content becomes a critical issue for smart TV user interaction. In this paper, we propose an automatic recommendation scheme of TV program contents in sequence using sequential pattern mining (SPM). Motivation of sequential TV program recommendation is based on TV viewer's behaviors for watching multiple TV program contents in a row. A sequence of TV program contents for recommendation to a target user is constructed based on the features such as an occurrence and net occurrence of frequently watched TV program contents from the similar user group to which the target user belongs. Three types of SPM methods are presented—offline, online and hybrid SPM. To extract sequential patterns of preferably watched TV program contents, we propose a preference weighted normalized modified retrieval rank (PW-NMRR) metric for similar user clustering. In the offline SPM method, we effectively construct the sequential patterns for recommendation using a projection method, which yields good performance for relatively longer sequential patterns. The online SPM method mines sequential patterns online by effectively reflecting the recent preference characteristics of users for TV program contents, which is effective for short-sequence recommendation. The hybrid SPM method combines the offline and online SPM methods. The maximum precisions of 0.877, 0.793 and 0.619 for length-1, -2 and -3 sequence recommendations are obtained from the online, hybrid and offline SPM methods, respectively.

Keywords Recommendation · TV Personalization · Sequential pattern mining · Data mining · Intelligent TV user interfaces

S. Pyo · M. Kim (✉)
Department of Information and Communications Engineering, Korea Advanced Institute Science and Technology, Daejeon, Republic of Korea
e-mail: mkim@ee.kaist.ac.kr

S. Pyo
e-mail: sjpyo@kaist.ac.kr

E. Kim · M. Kim
Department of Electrical Engineering, Korea Advanced Institute Science and Technology, Daejeon, Republic of Korea
e-mail: eunhuikim@kaist.ac.kr

1 Introduction

TV broadcasting services become more diverse and abundant with the availability of internet, digital TV and web TV which are combined in web services. Due to the increased number of TV channels and the advent of IPTV services with new media services, TV viewers (users) are exposed to excessive amounts of TV program contents available at TV terminal sides. Under such a TV environment, TV viewers can access TV program contents via many TV channels of terrestrials, satellites and cables, and via the TV program content repositories of IPTV services. However, such excessive amounts of TV program contents can burden the TV viewers, because it takes long time to search their preferred TV program contents. Therefore, the automatic recommendation of TV program contents is beneficial to the TV viewers for easy and effective access to their preferred TV program contents [1]. For general recommendation systems, there have been various approaches such as content-based recommendation, collaborative filtering and hybrid recommendation [2, 3]. However, those approaches can hardly be applied for

recommendation of TV program contents in sequences, that is, in a time-ordered manner.

In e-commerce, when a user purchases a certain item in a web site, the web site may recommend more items related to the just purchased item. For item recommendation, a reasoning engine runs at the server side to find more items that might be further needed to the user after purchasing a particular item. Similarly, in IPTV services, an IPTV server may recommend a sequence of TV program contents (items) to the set-top-boxes of TV viewers. Therefore, purchasing items (or watching TV program contents) can be assisted with the recommended items (or TV program contents) in sequence. This alleviates user's burden of finding preferred items (or TV program contents). In this paper, we use SPM to find a sequence of preferred TV program contents for recommendation to a TV viewer in a time-ordered manner. Here, a sequential pattern means a sequence of preferably watched TV program contents which is extracted in a chronological order from the usage history database by SPM. SPM is a technique that finds sequential patterns from the frequently purchased items made during item transactions [4]. In the SPM for e-commerce, an item may be interpreted as one single purchased product, and an element consists of one or more products (items) purchased at a time. And, a sequence is a set of ordered elements in purchased times [4, 15–17]. When applying the SPM to our TV program recommendation domain in this paper, a watched TV program content is regarded as an item or an element having one item and a sequence is then defined as a set of the watched TV program contents in chronological order by a user per day. Also, in our work, any semantic relation between the attributes of TV programs and the users has not been considered for recommendation, which may require additional analysis schemes into the proposed models. The results of SPM can be used in recommendation services which suggest items to be further purchased to users. So, SPM can be a useful solution to create sequences of preferred TV program contents for recommendation to TV viewers and the TV viewers can easily access their preferred TV program contents sequentially. In this paper, we extend our previous work [5] with a similar user grouping method for effective SPM and with three methods of online, offline and hybrid SPM to recommend TV program contents in sequence by mining the sequential patterns from the history database of watched TV program contents.

To extract the sequential patterns more effectively, we firstly perform the similar user grouping by which more relevant sequential patterns can be extracted, thus recommending sequential patterns of more preferable TV program contents to target users. For similar user grouping, PW-NMRR metric is also proposed for the similarity measure between a target user and non-target user, which considers both the preference values and the watching orders of preferred TV program contents.

The rest of this paper is organized as follows: Sect. 2 discusses the related previous works; Sect. 3 presents our proposed PW-NMRR metric for similar user grouping and the offline, online and hybrid SPM methods for sequential pattern extractions; Sect. 4 presents the experimental results to show the effectiveness of our proposed methods and Sect. 5 concludes the paper.

2 Related work

The algorithms for SPM are widely used in many applications which manage various kinds of sequence data such as bioinformatics, medicine prescription, e-commerce, mobile-commerce and users' behavior analysis. In this paper, we focus on analyzing users' behavior (that is, finding sequential patterns of preferable watched TV program contents) based on SPM. Previous works which are related with analyzing user's behavior are mostly applied to users' item purchase data or webpage access log data [6, 7]. Also, Tseng et al. [8] proposed a data mining method named two-dimensional multilevel (2-DML) association rule for mining the location-based service patterns in a mobile web service environment. The extracted patterns were used for providing the location-based mobile web services and the result of analyzing users' service request patterns can be used to predict user's next behaviors so as to recommend items or webpages for further purchasing or visiting.

The PrefixSpan algorithm has been proposed to find sequential patterns of purchased items in transaction databases and to recommend new items for further purchase to users [9, 10]. It only uses the occurrences of items as a criterion for selecting frequent items, which cannot be effectively applied for the recommendation of TV program contents in sequence because both the watching times and the watching lengths are important in TV program content recommendation.

Zhou et al. [11] studied an intelligent recommender system using sequential web access patterns. Their intelligent recommender system is known as SWARS (Sequential Web Access-based Recommender System) which predicts the user's web access behaviors using SPM. The application environment of this system is the web, and the system mostly focuses on analyzing the users' web page entering and staying behaviors. They called as the pattern the chronologically visited web sites that a user surfed around. The proposed system performs mining for extracting the sequential web access. Also, it controls a log data of users' behavior which is used for SPM, and constructs pattern trees to extract sequential web access

patterns. That is, the system first constructs tree structures using the users' web access log data, and selects the sequential web access patterns based on the minimum support. According to the experimental results, their recommender system showed overall good performance in the web environment.

Huang [12] proposed a general SPM model for progressively increasing database where the users' transaction database is mined to extract the sequential patterns of user's actions. When sequential patterns are generated, newly arriving patterns may not be identified as frequent sequential patterns due to the dominance of sequential patterns in the old usage history data, and the obsolete sequential patterns that have frequently been occurred in the past may stay in the current recommendation results. To remedy this issue in the progressive database, Huang [12] suggested a progressive SPM method which makes it possible to discover the frequent sequential patterns and to remove the obsolete sequential patterns from the extracted sequential patterns. In [12], PISA (Progressive mIning of Sequential pAtterns) is proposed to progressively discover sequential patterns in a time period, that is, POI (period of interest). The PISA algorithm is based on a progressive sequential tree which not only contains the information of all sequences in a progressive database but also helps the PISA algorithm generate frequent sequential patterns in each POI. Using the progressive sequential tree, they can construct and delete the sequential patterns dynamically by shifting the POI. The experimental results of the PISA algorithm showed much better performance in execution time and memory usage by comparing other SPM algorithms, such as SPAM [13], DirApp [12] and GSP [14]. However, when the number of items which constitute a sequence database increases, the size and depth of the progressive sequential tree dramatically increases per POI so the computational complexity of the PISA increases. To overcome the limitation of [12], we propose *P–S* relation with the update process of the occurrence and net occurrence values, which is not affected by the number of POIs. We define *P–S* relation (Program-Sequence relation) as representation for the relation of watching orders between TV programs in a watched TV program sequence. These works [8–12] have some limitations when they are directly applied for recommending TV program contents in sequence from the users' history database of watched TV program contents. The characteristics of watched TV program contents are quite different from the cases of purchasing items, accessing web sites and so on. Therefore, based on a basic concept of SPM and a method for dealing with progressive database, we propose automatic recommendation of preferred TV program contents in sequence using the offline, online and hybrid SPM methods. In the following section, we will introduce the three methods in detail.

3 Proposed SPM-based TV program recommendation schemes

Figure 1 shows a conceptual diagram of the proposed automatic recommendation of sequential TV program contents. The proposed system consists of a web server, a content streaming and archiving server, a recommendation engine, a usage history DB and user terminals. To recommend a sequence of preferred TV program contents for a target user, the recommendation engine performs similar user grouping, reasoning, and decision processes based on SPM.

Similar user grouping is the process of clustering similar taste users in perspective of their preferred TV program contents with similar watching orders and preferences for effective SPM. The similar user grouping is explained in Sect. 3.1. For the results of similar user grouping, we proceed to the SPM process which is performed in offline, online and hybrid manners. These types of SPM methods are explained in Sects. 3.2, 3.3, and 3.4. Also, the watching history data of TV program contents for the proposed SPM methods is divided in each day of the week. The TV program contents that are broadcast on specific days are properly handled by the proposed recommendation schemes. Among the results of offline, online and hybrid SPM methods, the final recommendation is made by the length of sequences of the recommended TV program contents.

3.1 Similar user grouping

In general, the database is composed of various TV program contents that were watched by different users (TV

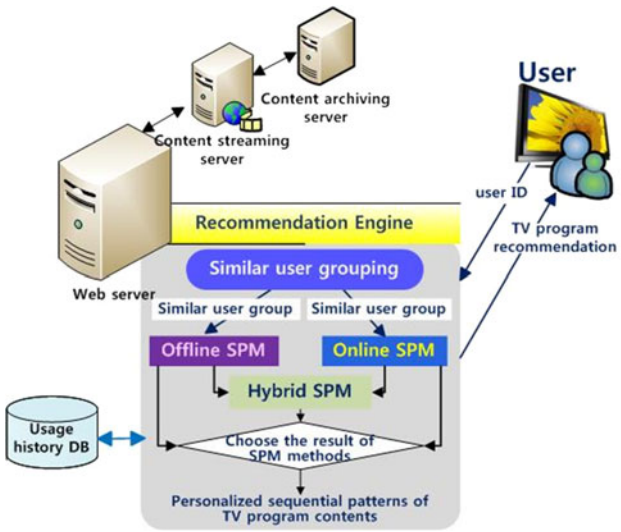

Fig. 1 A conceptual diagram of the proposed automatic recommendation of sequential TV program contents

viewers). Therefore, the extracted sequential patterns from the whole data would not be meaningful because they cannot reflect the characteristics of individual users in personalized manners. If a target user has his/her own similar user group in which the group users have similar patterns in their watched TV program contents, then the extracted sequential patterns from the watched TV program history of the similar user group can be good representatives to the TV watching characteristics of the target user. The evidence of advantage of similar user grouping before processing SPM is introduced in Sect. 4.1 by comparing the result of precisions based on offline SPM with similar user grouping and offline SPM without similar user grouping. To cluster similar users for a target user, we consider the watching orders and preferences of TV program contents by the users. That is, the similar users are clustered into a group for the target user in that they have similar histories of watched TV program contents with watching orders and preferences. For similar user grouping, the normalized modified retrieval rank (NMRR) metric can be considered, which reflects the ranks of retrieved images into similarity measure in MPEG-7 [18–21]. Originally, the NMRR metric was developed to verify the retrieval results by comparing the ground truth images with the resulting retrieved results for each query, so it measures the ranks of the retrieved images. Since the NMRR metric only considers the retrieved ranks in measuring the similarity, we extend the NMRR by additionally taking into account the preference on the watched TV program contents, which is called the preference weighted NMRR (PW-NMRR) as a new similarity metric. Therefore, the PW-NMRR metric measures the similarity between a target user and another user by considering not only the rank values but also the preference values of the watched TV program contents. To compute the PW-NMRR values, we first calculate the preference values of all the watched TV program contents by each user. The preference $pref_{u_j}(P)$ of a TV program content P watched by user u_j is computed as

$$pref_{u_j}(P) = \mathrm{WL_P}/\mathrm{TL_P} \tag{1}$$

where $\mathrm{WL_P}$ and $\mathrm{TL_P}$ are the watching length and the total length of P within the period that a training data have been collected, respectively. Based on $pref_{u_j}(P)$, we select user u_j's one or more preferred TV program contents $P_{u_j}^{pref}$ that their preference values exceed a predefined threshold value. Then, we sort the selected preferred TV program contents in watching orders for each user. Here the watching orders become the corresponding ranks for the preferred TV program contents. So, for each user's preferred TV program contents with their watching orders and the preference values, we can compute the PW-NMRR values of user u_j against target user u_T. The PW-NMRR is a normalized preference weighted average rank (PWAVR) between 0 and 1, and is computed by

$$\text{PW-NMRR}_{u_\mathrm{T}}(u_j) = \frac{\mathrm{PWAVR}_{u_\mathrm{T}}(u_j) - \mathrm{MinPWAVR}_{u_\mathrm{T}}(u_j)}{\mathrm{MaxPWAVR}_{u_\mathrm{T}}(u_j) - \mathrm{MinPWAVR}_{u_\mathrm{T}}(u_j)} \tag{2}$$

where $\mathrm{PWAVR}_{u_\mathrm{T}}(u_j)$ indicates the PWAVR value of u_j against u_T. $\mathrm{MinPWAVR}_{u_\mathrm{T}}(u_j)$ and $\mathrm{MaxPWAVR}_{u_\mathrm{T}}(u_j)$ are the minimum and maximum of $\mathrm{PWAVR}_{u_\mathrm{T}}(u_j)$. $\mathrm{PWAVR}_{u_\mathrm{T}}(u_j)$ in (2) is computed as

$$\mathrm{PWAVR}_{u_T}(u_j) = \frac{\sum_{i=1}^{\mathrm{NG}(u_\mathrm{T})} \left[1 + \Delta pref_{u_j}\left(P_{i,u_\mathrm{T}}^{pref}\right)\right] Rank_{u_j}\left(P_{i,u_\mathrm{T}}^{pref}\right)}{\mathrm{NG}(u_\mathrm{T})} \tag{3}$$

where $\mathrm{NG}(u_\mathrm{T})$ is the number of preferred TV program contents for u_T, $P_{i,u_\mathrm{T}}^{pref}$ indicates the preferred TV program content watched in the ith order by u_T, and $\Delta pref_{u_j}(P_{i,u_\mathrm{T}}^{pref})$ is the difference between the preference values of $P_{i,u_\mathrm{T}}^{pref}$ watched by both u_T and u_j. In (3), $Rank_{u_j}(P_{i,u_\mathrm{T}}^{pref})$ is the rank of $P_{i,u_\mathrm{T}}^{pref}$ that is also watched by u_j. Also, the set of the preferred TV program contents by user u_j is represented as S^{u_j}. If $P_{i,u_\mathrm{T}}^{pref} \in S^{u_j}$ and $P_{i,u_\mathrm{T}}^{pref} = P_{k,u_j}^{pref}$, then $Rank_{u_j}(P_{i,u_\mathrm{T}}^{pref})$ is calculated as

$$Rank_{u_j}\left(P_{i,u_\mathrm{T}}^{pref}\right) = 1 + R_{u_j}\left(P_{k,u_j}^{pref}\right) - offset_{u_j} \tag{4}$$

Otherwise, $Rank_{u_j}(P_{i,u_\mathrm{T}}^{pref}) = K + 1$ where $K = \mathrm{Max}\left(\mathrm{NG}(u_\mathrm{T}), \mathrm{NG}(u_j)\right)$. In (4), $R_{u_j}(P_{k,u_j}^{pref})$ is the watched rank (order) of a preferred TV program content P_{k,u_j}^{pref} watched by user u_j, and $offset_{u_j} = \mathrm{Min}_{P_{k,u_j}^{pref} \in S^{u_\mathrm{T}}}\left(R_{u_j}\left(P_{k,u_j}^{pref}\right)\right)$ by which $R_{u_j}(P_{k,u_j}^{pref})$ is adjusted to $Rank_{u_j}(P_{i,u_\mathrm{T}}^{pref})$. Notice here that $offset_{u_j}$ becomes the smallest rank of the preferred TV program content in the set S^{u_j} where the preferred TV program content also belongs to the set S^{u_T} of the preferred TV program contents by target user u_T. Figure 2 illustrates an example of rank adjustment for PW-NMRR computation. In Fig. 2, the user u_j's preferred TV program contents D, E, F and G in the middle table are also found in the set S^{u_T} of the target user's preferred TV program contents from the most left table.

Here, the ranks for the user u_j's preferred TV program contents are adjusted such that the first matched TV program content D is top ranked and its following ones are ranked afterward. The preceding preferred TV program contents, S and K, prior to the preferred TV program D are ignored in rank adjustment as shown in the most right table of Fig. 2.

The rank computation in (4) is explained as follows: $offset_{u_j}$ is the minimum value of $R_{u_j}(P_{k,u_j}^{pref})$'s where P_{k,u_j}^{pref}'s

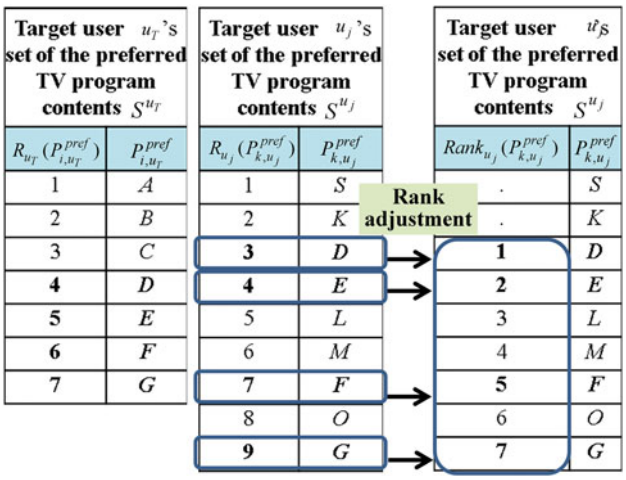

Fig. 2 An example of rank adjustment for PW-NMRR computation

belong to the set S^{u_T} of the preferred TV program contents by target user u_T; If P^{pref}_{i,u_T} exists in S^{u_j} and P^{pref}_{i,u_T} is equal to P^{pref}_{k,u_j}, then $Rank_{u_j}(P^{pref}_{i,u_T})$ becomes the value of $R_{u_j}(P^{pref}_{k,u_j})$ subtracted by $offset_{u_j}$ value; if P^{pref}_{i,u_T} does not exist in S^{u_j}, we put penalty on $Rank_{u_j}(P^{pref}_{i,u_T})$. In this case, $Rank_{u_j}(P^{pref}_{i,u_T})$ is $K+1$. In (3), if $P^{pref}_{i,u_T} \in S^{u_j}$ and $P^{pref}_{i,u_T} = P^{pref}_{k,u_j}$, $\Delta pref_{u_j}(P^{pref}_{i,u_T})$ is computed by

$$\Delta pref_{u_j}\left(P^{pref}_{i,u_T}\right) = \left| pref_{u_T}\left(P^{pref}_{i,u_T}\right) - pref_{u_j}\left(P^{pref}_{k,u_j}\right)\right| \tag{5}$$

Otherwise, $\Delta pref_{u_j}(P^{pref}_{i,u_T}) = pref_{u_T}(P^{pref}_{i,u_T})$. According to the definition of $\mathrm{PWAVR}_{u_T}(u_j)$, we can easily calculate $\mathrm{MinPWAVR}_{u_T}(u_j)$ and $\mathrm{MaxPWAVR}_{u_T}(u_j)$. When S^{u_j} contains all the target user's preferred TV program contents, and their ranks and preference values are perfectly equal to those in S^{u_T}, then $\mathrm{PWAVR}_{u_T}(u_j)$ has the minimum value as

$$\mathrm{Min\ PWAVR}_{u_T}(u_j) = \frac{1+\cdots+\mathrm{NG}(u_T)}{\mathrm{NG}(u_T)} = \frac{1+\mathrm{NG}(u_T)}{2} \tag{6}$$

On the other hand, when all the target user's preferred TV program contents do not exist in S^{u_j}, then $\mathrm{PWAVR}_{u_T}(u_j)$ has the maximum value as

$$\mathrm{MaxPWAVR}_{u_T}(u_j) = \frac{\left(\mathrm{NG}(u_T) + \sum_{i=1}^{\mathrm{NG}(u_T)} pref_{u_T}\left(P^{pref}_{i,u_T}\right)\right) \times (K+1)}{\mathrm{NG}(u_T)} \tag{7}$$

Using $\mathrm{PWAVR}_{u_T}(u_j)$, $\mathrm{MinPWAVR}_{u_T}(u_j)$ and $\mathrm{MaxPWAVR}_{u_T}(u_j)$, we compute $\mathrm{PW\text{-}NMRR}_{u_T}(u_j)$ for all users except target user u_T. Then, we order the $\mathrm{PW\text{-}NMRR}_{u_T}(u_j)$ values in the ascending order. If user u_j has the smallest $\mathrm{PW\text{-}NMRR}_{u_T}(u_j)$ value, then user u_j becomes the most similar user to target user u_T. Finally, the similar user group for target user u_T consists of the users who have smaller $\mathrm{PW\text{-}NMRR}_{u_T}(u_j)$ values than a given grouping threshold G_{th}. For the usage history data set of watched TV program contents from the clustered similar user groups, sequential patterns of preferable TV program contents are then extracted by our proposed SPM methods.

3.2 Offline SPM method

The offline SPM method extracts frequently watched TV program contents in a sequential order from the whole database with accumulated TV watching history data. We find sequential patterns based on an extension to the PrefixSpan algorithm [9, 10] which was developed for item recommendation from transaction database in e-commerce. In the PrefixSpan algorithm, constructing a sequential pattern is to select frequently purchased items which exceed a predefined minimum support threshold. For the extraction of sequential patterns, the PrefixSpan algorithm takes into account only the transaction times of purchased items by users. In our cases, both the number of watching times and watching lengths of TV program contents are considered in finding sequential patterns of frequently watched TV program contents. Also, we divide the usage history data in terms of days of the week because the TV program contents are usually broadcast on a weekly basis. Therefore, the sequential patterns are extracted on a weekly basis. In the usage history database of watched TV program contents, the sequence is defined as a series of TV program contents watched by a user per day. Table 1 shows a stitch of usage history data of TV program contents watched on Sunday by users with their IDs, 6006002 and 1125478.

In general, the watched TV program contents are recorded in order as shown in Table 1 where the program sequences are expressed without their watched time lengths which are importantly used for finding more meaningful sequential patterns. So, in this paper, the watched TV program contents are expressed with additional information of their watched time lengths. A user u_j's relatively watched time length q_{i,u_j} for a TV program content P_i with program index i is calculated by

$$q_{i,u_j} = \mathrm{wl}_{P_i,u_j} / L_{P_i} \tag{8}$$

Table 1 Example of sequences of program watching history data

User ID/day of week/date	Program sequences
6006002/Sun/2002.12.1	$<P_{134}, P_{56}, P_{78}, P_{91}>$
6006002/Sun/2002.12.8	$<P_{578}, P_{134}, P_{14}, P_{87}>$
1125478/Sun/2002.12.1	$<P_{134}, P_{87}>$
1125478/Sun/2002.12.15	$<P_{134}, P_{912}, P_{52}, P_{14}>$

where L_{P_i} is the total broadcast time length of P_i and wl_{P_i,u_j} is the watched time length of P_i by user u_j. The proposed offline SPM method is explained in detail in the following steps.

Step 1: Computation of occurrence and net occurrence values of a TV program content

We use the occurrence and net occurrence values of each TV program content to find frequently watched TV program contents. The accumulated occurrence of a TV program content indicates the total number of the watched times for the TV program content by the group users and the accumulated net occurrence of a TV program content implies the accumulation of the relatively watched time lengths for the TV program content by the group users. The accumulated occurrence O_{P_i} of a watched TV program content P_i is calculated by counting the sequences which contain P_i in the sequence database, and is given by

$$O_{P_i} = \frac{1}{B_{P_i}} \sum_{j=1}^{N} \sum_{k=1}^{M^{u_j}} C_{P_i}\left(s_k^{u_j}\right) \tag{9}$$

where N is the total number of users in the similar user group to which the target user belongs, and M^{u_j} is the total number of TV watching days for user u_j. In (9), B_{P_i} is the total number of broadcast times for P_i during the period that the whole usage history data spans, $s_k^{u_j}$ is the sequence that consists of the watched TV program contents by user u_j in the kth day, and $C_{P_i}(s_k^{u_j})$ is an indicator that results in $C_{P_i}(s_k^{u_j}) = 1$ for $P_i \in s_k^{u_j}$ and $C_{P_i}(s_k^{u_j}) = 0$ for $P_i \notin s_k^{u_j}$. As aforementioned, all the TV program contents watched by a user in 1 day constitute one single sequence. The accumulated net occurrence Q_{P_i} of a watched TV program content P_i is computed from the sequence database of the user group to which the target user belongs, and is given by

$$Q_{P_i} = \frac{1}{B_{P_i}} \sum_{j=1}^{N} \sum_{k=1}^{M^{u_j}} Cq_{P_i}\left(s_k^{u_j}\right) \tag{10}$$

where $Cq_{P_i}(s_k^{u_j})$ is an indicator that results in $Cq_{P_i}(s_k^{u_j}) = q_{i,u_j}$ for $P_i \in s_k^{u_j}$ and $Cq_{P_i}(s_k^{u_j}) = 0$ for $P_i \notin s_k^{u_j}$. Q_{P_i} in (10) implies a royalty of all group users to the TV program content P_i. The maximum Q_{P_i} indicates that all the group users have watched P_i in its whole time length whenever P_i was broadcasted, which is, in this case, equivalent to the total number N of the group users and its minimum value is 0 which corresponds to the case that no group users have watched P_i.

Step 2: Selection of frequently watched TV program contents

In this step, we select frequently watched TV program contents for which their occurrence and net occurrence values of the TV program contents exceed the two predefined minimum support thresholds O_{Min_sup} and Q_{Min_sup}, respectively. The watched TV program content P_i is selected when $O_{P_i} \geq O_{Min\_sup}$ and $Q_{P_i} \geq Q_{Min\_sup}$, and the selected frequently watched TV program contents are represented as $FP_1, \ldots, FP_{K^{off}}$ where K^{off} is the total number of selected TV program contents that were frequently watched in the usage history database of offline SPM method.

Step 3: Construction of a projected database for each frequently watched TV program content

Firstly, for each frequently watched TV program content FP_k with $1 \leq k \leq K^{off}$, all the sequences are selected which contain FP_k extracted in Step 2. For the set $\widehat{s}_{FP_k}$ of all the sequences which contain FP_k, we generate a projected database of FP_k by a projection process [9, 10]. The projection process is to cut off the first part of each sequence from its beginning to FP_k, thus yielding a projected sequence which consists of the remaining part. Therefore, the projected database of FP_k is the set of the projected sequences. In this projection process, the frequently watched TV program content FP_k is regarded as a prefix up to which its first part of the sequence is cut off from the beginning. Table 2 shows an example of projected database for a set of sequences $\widehat{s}_{FP_k}$ when FP_k is P_{157}.

Step 4: Generation of sequential patterns

In this step, we compute the occurrence and net occurrence values of the remaining TV program contents for each projected database. Based on the occurrence and net occurrence values, we again select frequently watched TV program contents for each projected database in the same way as Step 2. The selected TV program contents from the projected database of FP_k are represented as $FP_1^k, \ldots, FP_{M^{off}}^k$ where M^{off} is the total number of selected TV program contents that were frequently watched from the projected database of FP_k.

The frequently watched TV program contents selected in Step 4 are connected to the frequently watched TV program contents selected in Step 2. If we assume that P_{68} and P_{304} are found as the frequently watched TV program contents in Step 4 for the projected database in Table 2, then we have two extracted sequential patterns "$P_{157} - P_{68}$" and "$P_{157} - P_{304}$" of length-2. In this manner, we

Table 2 An example of $\widehat{s}_{P_{157}}$ and projected database of P_{157}

User/date/day	$\widehat{s}_{P_{157}}$	Projected database of P_{157}
u_1/02.12.8/Sun	$<P_{134}, P_{56}, P_{157}, P_{68}>$	$<P_{68}>$
u_1/02.12.15/Sun	$<P_{157}, P_{68}, P_{304}>$	$<P_{68}, P_{304}>$
u_2/02.12.1/Sun	$<P_{479}, P_{157}, P_{68}, P_{304}>$	$<P_{68}, P_{304}>$
u_3/02.12.8/Sun	$<P_{157}, P_{68}, P_{81}, P_{712}>$	$<P_{68}, P_{81}, P_{712}>$

can furthermore extract sequential patterns of longer lengths by repeating the Steps 2, 3 and 4.

3.3 Online SPM method

The online SPM method extracts sequential patterns of frequently watched TV program contents from the progressive database by reflecting the recent characteristics of users. The progressive database is the database for which the data increases as time goes on. So, to efficiently deal with the progressive database for extracting sequential patterns online, the processing time and complexity should be importantly considered. Also, we have to efficiently update the occurrence and net occurrence values of each TV program content for the incoming usage data of watched TV program contents. In the online SPM method, we define an observation window in which the occurrence and net occurrence values of watched TV program contents are calculated. The observation window slides in time without overlapping with the previous observation windows. But, to both consider the current characteristics and reflect the previously consumed characteristics of watched TV program contents, the occurrence and net occurrence values in the current observation window are calculated by updating those obtained in the previous observation window. Figure 3 shows an example of computing the occurrence and net occurrence values in the current observation window $\Phi^{\text{on}}(t)$ and the previous observation windows $\Phi^{\text{on}}(t-1)$ and $\Phi^{\text{on}}(t-2)$ by the online SPM method. Based on these occurrence and net occurrence values computed in $\Phi^{\text{on}}(t-1)$ and $\Phi^{\text{on}}(t)$, we can update the occurrence and net occurrence values at time t by an update scheme of the online SPM method. The update scheme is explained in detail in Step 2.

We obtain the occurrence and net occurrence values and, find the sequential patterns of frequently watched TV program contents in the current observation window. Unlike the offline SPM method to extract sequential patterns by projection process, the online SPM method represents the incoming usage history data of watched TV program contents in the *P–S* relation. The representation of the *P–S* relation is effectively used in finding sequential patterns in the online SPM method, which is explained in detail later. The online SPM method is explained in the following steps:

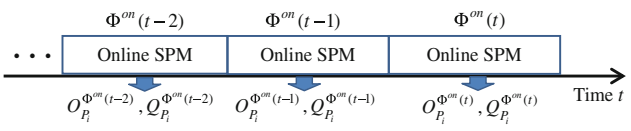

Fig. 3 An example of computing the net occurrence and occurrence values in the online SPM method

Step 1: Representation of P–S relation for watched TV program contents in an observation window

In this step, we represent the watched TV program contents in the current observation window into the *P–S* relation. The *P–S* relation allows for an effective representation of the watched TV program contents in a chronological order with their accumulated occurrence and net occurrence values for online SPM method. Table 3 shows some examples of the sequences of watched TV program contents represented with their occurrence and net occurrence values in the chronological order. The watched TV program contents with their occurrence and net occurrence values are represented in a three-tuple as $(P_i, o_{i,u_j}, q_{i,u_j})$ within a sequence expressed with the bracket ($\langle\rangle$) as shown in Table 3.

o_{i,u_j} is the occurrence of P_i by user u_j in a sequence, so and its value is always 1. In Table 3, the watched TV program contents in the brackets ($\langle\rangle$) are ordered chronologically from the left to the right. For the sequence $\langle \ldots, (P_{i_{m-1}}, o_{i_{m-1}}, q_{i_{m-1}}), (P_{i_m}, o_{i_m}, q_{i_m}), (P_{i_{m+1}}, o_{i_{m+1}}, q_{i_{m+1}}), \ldots \rangle$ of a chronological order, the *P–S* relation for a watched TV program content P_{i_m} is expressed as follow:

$$\underbrace{P_{i_m}^{\hat{o}_{i_m},\hat{q}_{i_m}}}_{\text{DPS}} \Big| \underbrace{P_{i_{m+1}}^{\hat{o}_{i_{m+1}},\hat{q}_{i_{m+1}}} P_{i_{m+2}}^{\hat{o}_{i_{m+2}},\hat{q}_{i_{m+2}}} \cdots P_{i_{m+K}}^{\hat{o}_{i_{m+K}},\hat{q}_{i_{m+K}}}}_{\substack{\text{watched TV program contents} \\ \text{in chronological order after } P_{i_m} - \text{SPS of } P_{i_m}}} \tag{11}$$

where $\hat{o}_{i_m}$ and $\hat{q}_{i_m}$ are the instantaneously accumulated occurrence and net occurrence of P_{i_m}, respectively, which are computed sample-by-sample in the current observation window. In (11), P_{i_m} is called the datum point of the *P–S* relation (DPS) and the following TV program contents are referred to as the subordination of the DPS (SPS). The occurrence and net occurrence values of each DPS are accumulated, and the SPS of the DPS in the *P–S* relation is populated and accumulated for the incoming sequences by the user group to which the target user belongs.

Table 3 Sequence examples of watched TV program contents represented in three-tuple

user/date/day	Sequences of watched TV program contents in chronological order
u_1/02.12.1/Sun	$\langle(P_1,\ 1,\ 0.1), (P_2,\ 1,\ 0.2), (P_3,\ 1,\ 0.1), (P_4,\ 1,\ 0.3)\rangle$
u_1/02.12.8/Sun	$\langle(P_1,\ 1,\ 0.4), (P_3,\ 1,\ 0.4), (P_6,\ 1,\ 0.6), (P_4,\ 1,\ 0.2)\rangle$
u_2/02.12.8/Sun	$\langle(P_3,\ 1,\ 0.4), (P_4,\ 1,\ 0.4)\rangle$
u_3/02.12.15/Sun	$\langle(P_2,\ 1,\ 0.6)\rangle$
u_3/02.12.22/Sun	$\langle(P_3,\ 1,\ 0.7), (P_6,\ 1,\ 0.5)\rangle$

Table 4 Example of *P–S* relations for the sequences in Table 3

Order	Incoming sequences	*P–S* relations for each watched TV program content in each sequence
1	$\langle(P_1,\ 1,\ 0.1),(P_2,\ 1,\ 0.2),(P_3,\ 1,\ 0.1),(P_4,\ 1,\ 0.3)\rangle$	$P_1^{1,0.1}\vert P_2^{1,0.2}P_3^{1,0.1}P_4^{1,0.3}$, $P_2^{1,0.2}\vert P_3^{1,0.1}P_4^{1,0.3}$, $P_3^{1,0.1}\vert P_4^{1,0.3}$, $P_4^{1,0.3}$
2	$\langle(P_1,\ 1,\ 0.4),(P_3,\ 1,\ 0.4),(P_6,\ 1,\ 0.6),(P_4,\ 1,\ 0.2)\rangle$	$P_1^{2,0.5}\vert P_2^{1,0.2}P_3^{2,0.5}P_6^{1,0.6}P_4^{2,0.5}$, $P_3^{2,0.5}\vert P_6^{1,0.6}P_4^{1,0.3}$, $P_6^{1,0.6}\vert P_4^{1,0.2}$, $P_4^{2,0.5}$
3	$\langle(P_3,\ 1,\ 0.4),(P_4,\ 1,\ 0.4)\rangle$	$P_3^{3,0.9}\vert P_6^{1,0.6}P_4^{3,0.9}$, $P_4^{3,0.9}$
4	$\langle(P_2,\ 1,\ 0.6)\rangle$	$P_2^{2,0.8}\vert P_3^{1,0.1}P_4^{1,0.3}$
5	$\langle(P_3,\ 1,\ 0.7),(P_6,\ 1,\ 0.5)\rangle$	$P_3^{4,1.6}\vert P_6^{2,1.1}P_4^{3,0.9}$, $P_6^{2,1.1}\vert P_4^{1,0.2}$

Table 4 shows some examples of populating the *P–S* relation for each watched TV program content shown in Table 3. In Table 4, the incoming sequences in the second column are taken as input in the row order as indicated, and the third column represents the corresponding *P–S* relations with DPS accumulation and SPS population.

The DPS accumulation is performed by accumulating the occurrence and net occurrence values of the current DPS to those of the just previous DPS. For example, the first incoming sequence $\langle(P_1,\ 1,\ 0.1),(P_2,\ 1,\ 0.2),(P_3,\ 1,\ 0.1),(P_4,\ 1,\ 0.3)\rangle$ in Table 4 consists of five watched TV program contents. Therefore, five DPS's can be produced for P_1, P_2, P_3 and P_4. Based on DPS P_1, we have $P_1^{1,0.1}$ for the first time. Then, for the second incoming sequence $\langle(P_1,\ 1,\ 0.4),(P_3,\ 1,\ 0.4),(P_6,\ 1,\ 0.6),(P_4,\ 1,\ 0.2)\rangle$, DPS $P_1^{1,0.1}$ is changed to $P_1^{2,0.5}$ by adding the current occurrence value of 1 and the current net occurrence values of 0.4–1 and 0.1 of $P_1^{1,0.1}$, respectively. Similarly, all the other DPS's can be changed in this manner. For the population of SPS with a given DPS, the occurrence and net occurrence values of each watched TV program content in the SPS are accumulated in the same way as the DPS case if the watched TV program content exists in the SPS. Otherwise, the TV program content is newly added and positioned chronologically in the SPS. For the second incoming sequence $\langle(P_1,\ 1,\ 0.4),(P_3,\ 1,\ 0.4),(P_6,\ 1,\ 0.6),.(P_4,\ 1,\ 0.2)\rangle$ in Table 4, the SPS of DPS P_1 is constructed by adding the occurrence and net occurrence values of P_3 and P_4 to the corresponding ones of the SPS of DPS P_1 for the first sequence $\langle(P_1,\ 1,\ 0.1),(P_2,\ 1,\ 0.2),(P_3,\ 1,\ 0.1),(P_4,\ 1,\ 0.3)\rangle$ as well as by newly positioning P_6 between P_3 and P_4. Thus, we have $P_1^{2,0.5}\vert P_2^{1,0.2}P_3^{2,0.5}P_6^{1,0.6}P_4^{2,0.5}$. as the DPS and SPS of P_1 for the second incoming sequence.

Table 5 shows the final *P–S* relations by DPS accumulation and SPS population for the incoming sequences of the watched TV program contents in Table 4. As the final results in Step 1, we can obtain the accumulated occurrence and net occurrence values for a TV program content P_i of DPS and SPS in the *P–S* relations. The accumulated occurrence and net occurrence values are then normalized by the total number of broadcast times for P_i within $\Phi^{\text{on}}(t)$, which are denoted as $O_{P_i}^{\Phi^{\text{on}}(t)}$ and $Q_{P_i}^{\Phi^{\text{on}}(t)}$, respectively.

Step 2: Update of occurrence and net occurrence values of watched TV program contents

In Step 1, we expressed the TV program contents in the *P–S* relation where their normalized occurrence $O_{\text{P}_i}^{\Phi^{\text{on}}(t)}$ and net occurrence $Q_{\text{P}_i}^{\Phi^{\text{on}}(t)}$ values are represented in the DPS's and SPS's. Notice that $O_{\text{P}_i}^{\Phi^{\text{on}}(t)}$ and $Q_{\text{P}_i}^{\Phi^{\text{on}}(t)}$ are calculated only within the current observation window $\Phi^{\text{on}}(t)$. It should also be noted that $O_{\text{P}_i}^{\Phi^{\text{on}}(t)}$ and $Q_{\text{P}_i}^{\Phi^{\text{on}}(t)}$ must be distinguished from the instantaneously accumulated occurrence and net occurrence, $\hat{o}_{i_m}$ and $\hat{q}_{i_m}$ of P_{i_m}, in (11). The updated occurrence and net occurrence values $O_{\text{P}_i}^{\text{on}}(t)$and $Q_{\text{P}_i}^{\text{on}}(t)$ are given by

$$O_{\text{P}_i}^{\text{on}}(t) = O_{\text{P}_i}^{\text{on}}(t-1) + \Delta O_{\text{P}_i}^{\text{on}}(t) \Big/ O_{\text{P}_i}^{\text{avg,on}}(t) \tag{12}$$

$$Q_{\text{P}_i}^{\text{on}}(t) = Q_{\text{P}_i}^{\text{on}}(t-1) + \Delta Q_{\text{P}_i}^{\text{on}}(t) \Big/ Q_{\text{P}_i}^{\text{avg,on}}(t) \tag{13}$$

where $\Delta O_{\text{P}_i}^{\text{on}}(t)$ and $\Delta Q_{\text{P}_i}^{\text{on}}(t)$ indicate the change in the occurrence between $O_{\text{P}_i}^{\Phi^{\text{on}}(t)}$and $O_{\text{P}_i}^{\text{on}}(t-1)$, and the change in the net occurrence between $Q_{\text{P}_i}^{\Phi^{\text{on}}(t)}$ and $Q_{\text{P}_i}^{\text{on}}(t-1)$, respectively, which are given by

Table 5 Example of the result of total *P–S* relations for given database

DPS	*P–S* relations for the sequences of watched TV program content in an observation window
P_1	$P_1^{2,0.5}\vert P_2^{1,0.2}P_3^{2,0.5}P_6^{1,0.6}P_4^{2,0.5}$
P_2	$P_2^{2,0.8}\vert P_3^{1,0.1}P_4^{1,0.3}$
P_3	$P_3^{4,1.6}\vert P_6^{2,1.1}P_4^{3,0.9}$
P_4	$P_4^{3,0.9}$
P_6	$P_6^{2,1.1}\vert P_4^{1,0.2}$

$$\Delta O_{\mathrm{P}_i}^{\mathrm{on}}(t) = O_{\mathrm{P}_i}^{\Phi^{\mathrm{on}}(t)} - O_{\mathrm{P}_i}^{\mathrm{on}}(t-1) \quad \text{and}$$
$$\Delta Q_{\mathrm{P}_i}^{\mathrm{on}}(t) = Q_{\mathrm{P}_i}^{\Phi^{\mathrm{on}}(t)} - Q_{\mathrm{P}_i}^{\mathrm{on}}(t-1). \tag{14}$$

In (12) and (13), $O_{\mathrm{P}_i}^{\mathrm{avg,on}}(t)$ and $Q_{\mathrm{P}_i}^{\mathrm{avg,on}}(t)$ indicate the averages of the occurrence and net occurrence values from the beginning to the current time t, respectively, which can be recursively calculated with the previous average values. The updates in (12) and (13) reflect the user's past preferences on TV program contents in computing the current preferences.

Step 3: Extraction of the sequential patterns

In Step 3, we first find as the length-1 sequential pattern the DPS P_i for which its $O_{\mathrm{P}_i}^{\mathrm{on}}(t)$ and $Q_{\mathrm{P}_i}^{\mathrm{on}}(t)$ are greater than the predefined minimum support thresholds $O_{\mathrm{Min\_sup}}$ and $Q_{\mathrm{Min\_sup}}$, respectively. For DPS P_i, we extract the TV program content P_j for which its $O_{\mathrm{P}_j}^{\mathrm{on}}(t)$ and $Q_{\mathrm{P}_j}^{\mathrm{on}}(t)$values exceed $O_{\mathrm{Min\_sup}}$ and $Q_{\mathrm{Min\_sup}}$ from its SPS, respectively. Similarly, this can be done for the other DPSs $\mathrm{DPS}_1, \ldots, \mathrm{DPS}_{K^{\mathrm{on}}}$ for which DPS_k, $1 \le k \le K^{\mathrm{on}}$, has the selected SPS's as $\mathrm{SPS}_1^k, \ldots, \mathrm{SPS}_{M^{\mathrm{on}}}^k$ where M^{on} is the total number of selected TV program contents in the SPS of DPS_k, respectively. Then, we generate a sequential pattern of length-2 by concatenating P_j selected from the P_i's SPS to P_i. To further extract a sequential pattern of length-3 based on that of length-2, we construct the *P–S* relation for the selected TV program content P_j from the P_i's SPS, and compute the occurrence and net occurrence values by the updating scheme in (12) and (13). Then, we extract the TV program content P_k for which its $O_{\mathrm{P}_k}^{\mathrm{on}}(t)$ and $Q_{\mathrm{P}_k}^{\mathrm{on}}(t)$ values exceed the predefined minimum support thresholds from the generated SPS of P_j, and concatenate the selected TV program content P_k to the sequential pattern $P_i - P_j$ of length-2, which become a sequential pattern $P_i - P_j - P_k$ of length-3.

3.4 Hybrid SPM method

Until now, we have discussed the offline and online SPM methods. The offline SPM method reflects the entire usage history of user's watched TV program contents in finding sequential patterns, which reflects users' watching tendency of a long-term period. On the other hands, the online SPM put more emphasis on the recently watched TV program contents in a non-overlapped sliding observation window, which considers users' recent watching tendency. To take both advantages, a hybrid SPM method is proposed by combining both the offline and online SPM methods. Figure 4 illustrates the computation of occurrence and net occurrence values in the proposed hybrid SPM method. $O_{\mathrm{P}_i}^{\Phi^{\mathrm{off}}(t)}$ and $Q_{\mathrm{P}_i}^{\Phi^{\mathrm{off}}(t)}$ values are computed in the same way as the offline SPM method in the offline observation window $\Phi^{\mathrm{off}}(t)$ of a relative longer length using (9) and (10), which are then used for the updates of $O_{\mathrm{P}_i}^{\Phi^{\mathrm{on}}(t)}$ and $Q_{\mathrm{P}_i}^{\Phi^{\mathrm{on}}(t)}$ in the online observation window.

Based on these occurrence and net occurrence values computed in $\Phi^{\mathrm{off}}(t)$ and $\Phi^{\mathrm{on}}(t)$, the occurrence and net occurrence values in the hybrid SPM method are calculated by

$$O_{\mathrm{P}_i}^{\mathrm{hybrid}}(t) = O_{\mathrm{P}_i}^{\Phi^{\mathrm{off}}(t)} + \Delta O_{\mathrm{P}_i}^{\Phi^{\mathrm{on}}(t)} \Big/ O_{\mathrm{P}_i}^{\mathrm{avg,off}}(t) \tag{15}$$

$$Q_{\mathrm{P}_i}^{\mathrm{hybrid}}(t) = Q_{\mathrm{P}_i}^{\Phi^{\mathrm{off}}(t)} + \Delta Q_{\mathrm{P}_i}^{\Phi^{\mathrm{on}}(t)} \Big/ Q_{\mathrm{P}_i}^{\mathrm{avg,off}}(t) \tag{16}$$

where $\Delta O_{\mathrm{P}_i}^{\Phi^{\mathrm{on}}(t)}$ and $\Delta Q_{\mathrm{P}_i}^{\Phi^{\mathrm{on}}(t)}$ indicate the change in the occurrence between $O_{\mathrm{P}_i}^{\Phi^{\mathrm{off}}(t)}$ and $O_{\mathrm{P}_i}^{\Phi^{\mathrm{on}}(t)}$, and the change in the net occurrence between $Q_{\mathrm{P}_i}^{\Phi^{\mathrm{off}}(t)}$ and $Q_{\mathrm{P}_i}^{\Phi^{\mathrm{on}}(t)}$, respectively, which are given by

$$\Delta O_{\mathrm{P}_i}^{\Phi^{\mathrm{on}}(t)} = O_{\mathrm{P}_i}^{\Phi^{\mathrm{on}}(t)} - O_{\mathrm{P}_i}^{\Phi^{\mathrm{off}}(t)} \quad \text{and}$$
$$\Delta Q_{\mathrm{P}_i}^{\Phi^{\mathrm{on}}(t)} = Q_{\mathrm{P}_i}^{\Phi^{\mathrm{on}}(t)} - Q_{\mathrm{P}_i}^{\Phi^{\mathrm{off}}(t)} \tag{17}$$

The updates of $O_{\mathrm{P}_i}^{\Phi^{\mathrm{on}}(t)}$ and $Q_{\mathrm{P}_i}^{\Phi^{\mathrm{on}}(t)}$ in $\Phi^{\mathrm{on}}(t)$ are made with $O_{\mathrm{P}_i}^{\Phi^{\mathrm{off}}(t)}$ and $Q_{\mathrm{P}_i}^{\Phi^{\mathrm{off}}(t)}$ values computed in $\Phi^{\mathrm{off}}(t)$ until the current observation window escapes the offline observation window $\Phi^{\mathrm{off}}(t)$. At the time that $\Phi^{\mathrm{on}}(t)$ gets out of $\Phi^{\mathrm{off}}(t)$, $\Phi^{\mathrm{off}}(t)$ is shifted to the front point of $\Phi^{\mathrm{on}}(t)$ as shown in Fig. 4. Then $O_{\mathrm{P}_i}^{\Phi^{\mathrm{off}}(t)}$ and $Q_{\mathrm{P}_i}^{\Phi^{\mathrm{off}}(t)}$ values are recalculated accordingly. Also, $O_{\mathrm{P}_i}^{\mathrm{avg,off}}(t)$ and $Q_{\mathrm{P}_i}^{\mathrm{avg,off}}(t)$ indicate the averages of the occurrence and net occurrence values from the beginning to the current time t for the offline observation window $\Phi^{\mathrm{off}}(t)$, respectively, which can be recursively calculated with the previous average values.

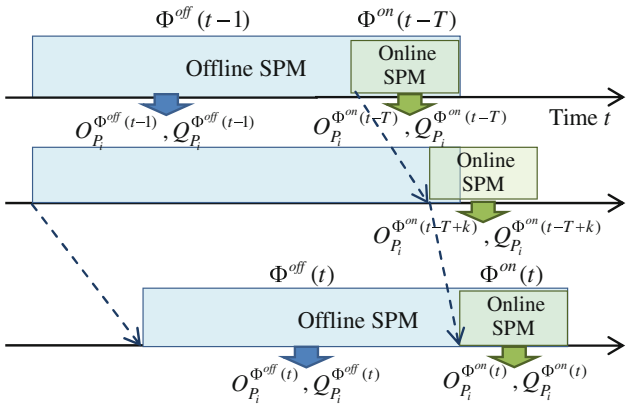

Fig. 4 An example of computing the net occurrence and occurrence values in the hybrid SPM method

4 Experimental results

For the experiments to demonstrate the effectiveness of the proposed three SPM methods for recommendation of sequential TV program contents, we used a TV usage history dataset, collected by AC Neilson Korea. The TV usage history dataset consists of the TV watching history of 86 users on six terrestrial TV channels during 6 months from 1 December 2002 to 1 May 2003. We used TV program watching history data of 86 users from the total usage history data for effective similar user grouping and sequential pattern extraction. The 86 users are the female users of ages between 61 and 65, who have watched TV programs more frequently than other users. In this way, we used the demographic information for initial user filtering among the total TV watching history data. For the experiments, the data set of the first 3 months is used for training and the remaining data set is for testing purpose. Also, the TV programs that exist both in the training and test periods are considered for recommendation.

Under TV watching environments in general, TV's can sometimes be left turned-on without real watching, which may cause a data fidelity issue for data-driven model-based recommendation systems. To eliminate the history data of low fidelity, we removed the usage history data of watched TV programs that had been watched less than 20 % of their total lengths. Nevertheless, it is very difficult to further discriminate the unintentionally watching history data of TV programs more than 20 % of their total lengths from the whole data set.

4.1 Similar user grouping results

Before we compare the performance of PW-NMRR and NMRR-based similar user grouping, we show the necessity of similar user grouping when we extract sequential patterns by comparing the precisions of offline SPM method with similar user grouping and without similar user grouping. For the experiment of similar user grouping, we set 30 target users and cluster similar user groups for each target user according to the grouping threshold G_{th}. Based on the 30 similar user groups for 30 target users, the average precisions for the extracted sequential patterns are computed in 7 days of the week. Also, for the experiment of precision performance based on offline SPM without similar user grouping, we just extract the sequential patterns from the watching history data of whole users based on offline SPM and compute the precision for the same 30 target users.

Figure 5 shows the average precision of the length-1 and -2 patterns using offline SPM method with and without similar user grouping. As shown in Fig. 5, the similar user grouping enhances the precision performance for the extracted sequential patterns. Therefore, this result verifies that the similar user grouping is necessary for effective SPM. Also, we compare the performances of two similar user grouping metrics, NMRR and PW-NMRR. For this, we firstly compare the extracted numbers of sequential patterns between the similar user groups clustered by NMRR and those by PW-NMRR.

Figure 6 shows the average numbers of extracted sequential patterns of length-1, -2 and -3 based on NMRR and PW-NMRR using the offline, online and hybrid SPM methods. The grouping threshold G_{th} is empirically set to 0.7 in similar user grouping for NMRR and PW-NMRR. As shown in Fig. 6, the PW-NMRR allows larger numbers of the extracted sequential patterns than the NMRR for all minimum support threshold values for the three SPM methods. The more the similar users are found, the larger the number of the extracted sequential patterns is. When we extract the sequential patterns using the offline, online and hybrid SPM methods, we compute the occurrence and

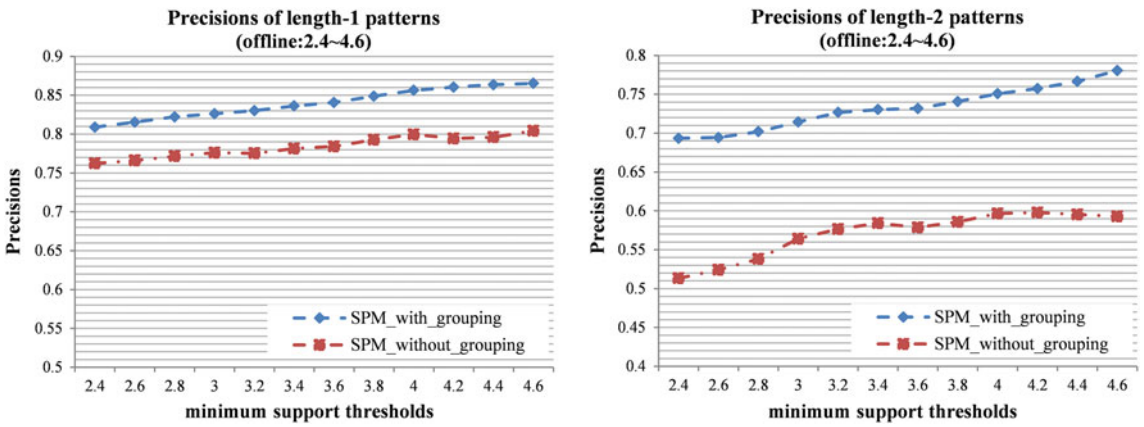

Fig. 5 Average precision of the length-1 and -2 patterns using offline SPM method with similar user grouping ($G_{\text{th}} = 0.7$) and without similar user grouping

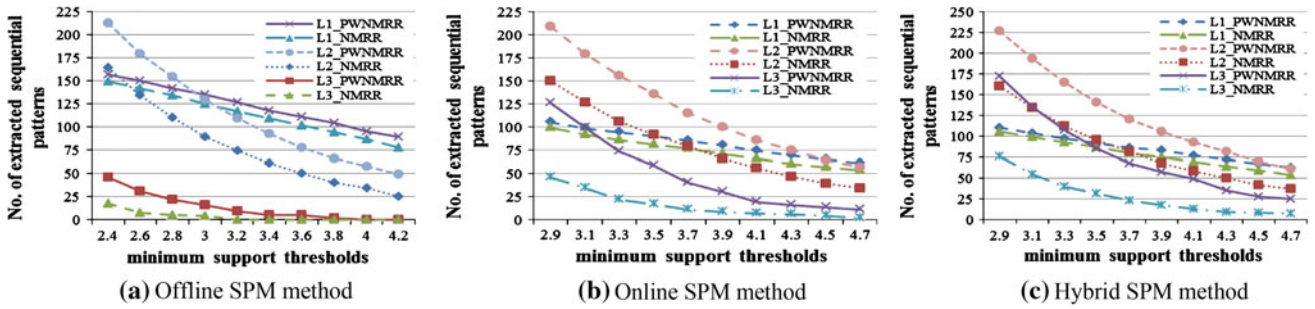

Fig. 6 Number of extracted sequential patterns of length-1, -2 and -3 for different minimum support thresholds using NMRR and PW-NMRR

net occurrence values for each watched TV program content, and then extract the sequential patterns with the occurrence and net occurrence values that exceed predefined minimum support thresholds $O_{\text{Min\_sup}}$ and $Q_{\text{Min\_sup}}$. Since the net occurrence reflects the preference of the group users for TV program contents, the PW-NMRR for similar user grouping can allow extracting more meaningful sequential patterns than the NMRR which does not take into account the preference values. The number of similar users for the target user can vary, depending on the grouping threshold G_{th}. We select the similar users who have smaller PW-NMRR values than a predefined grouping threshold value G_{th}. If G_{th} is small, the number of users in a similar user group becomes smaller. Table 6 shows the average, minimum and maximum numbers of users in the similar user groups for the 30 target users using PW-NMRR and NMRR, respectively.

In Table 6, the numbers of similar users clustered by PW-NMRR are generally smaller than those by NMRR. For the similar user groups, we compare the precisions for different grouping threshold values G_{th}. The effectiveness of the proposed similar user grouping method is examined in terms of the precision performance for extracted sequential patterns based on the results of similar user grouping. Table 7 shows the precisions of extracted sequential patterns of length-1, -2 and -3 by varying grouping threshold values

Table 6 Average, minimum and maximum numbers of users in the similar user groups for 30 target users

# of users	G_{th}				
	0.66	0.68	0.7	0.72	0.74
PW-NMRR					
Average	16.8	19.7	22.4	25.4	28.2
Min	4	5	8	10	13
Max	31	33	37	37	38
NMRR					
Average	22.4	25.5	29.6	32.6	35.6
Min	7	9	15	19	21
Max	34	37	40	43	46

G_{th} for PW-NMRR and NMRR. The precision for the recommended sequential patterns of length-n is measured by taking into account how many recommended sequential patterns of length- n are actually watched in the test data set by a target user. The average precision is then calculated by averaging the precisions for all target users and 7 days of the week. The precisions of the sequential patterns of length-1 generated by the offline, online and hybrid SPM methods are almost similar for four different grouping threshold values. Also, the precisions of the sequential patterns of length-2 and -3 are somewhat affected by the threshold G_{th} values for the offline, online and hybrid SPM methods. However, the parameters that affect the precision include not only the grouping threshold G_{th} for similar user grouping, but also the type of a SPM method used, the minimum support threshold, and the length of extracted sequential patterns. Nevertheless, the precisions of extracted length-1 sequential patterns based on PW-NMRR are generally a little higher than those based on NMRR. This is also observed for the precisions of the extracted sequential patterns of length-2, -3 for PW-NMRR- and NMRR-based similar user groups. It means that the similar user groups that are clustered based on PW-NMRR result in better recommendation precisions than those based on NMRR. (In Table 7, there are three cases that NMRR-based precisions are larger than PW-NMRR: (1) Offline SPM for length-3 with $G_{\text{th}} = 0.74$; (2) Hybrid SPM for length-1 with $G_{\text{th}} = 0.74$; and (3) Hybrid SPM for length-3 with $G_{\text{th}} = 0.7$. As G_{th} becomes larger, the number of users being grouped into a user group also becomes larger. So, the grouping results based on PW-NMRR and NMRR becomes indistinguishable because the preference based on PW-NMRR gets less effective. So, for $G_{\text{th}} = 0.74$, we obtained the reversed precision performances with cases (1) and (2). For the case (3), PW-NMRR and NMRR yielded 57 and 40 extracted sequential patterns, respectively. In this case, the smaller extracted pattern number (40) for NMRR resulted in higher precision performance than the one (57) based on PW-NMRR.

Table 7 Precisions based on PW-NMRR and NMRR

Precision	G_{th}			
	0.68	0.7	0.72	0.74
Offline SPM				
Length-1				
PW-NMRR	0.857	0.860	0.859	0.852
NMRR	0.853	0.854	0.852	0.846
Length-2				
PW-NMRR	0.730	0.751	0.743	0.747
NMRR	0.725	0.718	0.739	0.715
Length-3				
PW-NMRR	0.622	0.595	0.575	0.598
NMRR	0.575	0.551	0.537	0.614
Online SPM				
Length-1				
PW-NMRR	0.885	0.877	0.873	0.871
NMRR	0.874	0.874	0.870	0.871
Length-2				
PW-NMRR	0.791	0.772	0.774	0.773
NMRR	0.765	0.767	0.771	0.765
Length-3				
PW-NMRR	0.457	0.437	0.426	0.374
NMRR	0.421	0.404	0.404	0.354
Hybrid SPM				
Length-1				
PW-NMRR	0.872	0.877	0.870	0.865
NMRR	0.870	0.872	0.864	0.869
Length-2				
PW-NMRR	0.776	0.779	0.771	0.776
NMRR	0.760	0.765	0.770	0.773
Length-3				
PW-NMRR	0.498	0.470	0.490	0.455
NMRR	0.469	0.496	0.481	0.427

4.2 Performance evaluation of proposed SPM-based TV program recommendation schemes

We extract the sequential patterns for the training data set by the offline, online and hybrid SPM methods for the PW-NMRR-based similar user groups. The extraction of sequential patterns is performed for each day of the week. For each day of week, we extract the sequential patterns of length-1, -2 and -3 that all exceed the predefined minimum support thresholds. The minimum support threshold is an important factor that directly influences the resulting number of extracted sequential patterns. To fairly compare the performances of the three methods in terms of precision, we set the minimum support values for the three methods to generate similar numbers of extracted sequential patterns of length-1, -2 and -3. Figure 7 shows the numbers of extracted sequential patterns of length-1, -2 and -3 versus their minimum support thresholds $Q_{\text{Min\_sup}}$ for the PW-NMRR-based similar user groups with $G_{th} = 0.7$. The minimum support thresholds $O_{\text{Min\_sup}}$ are all set to 20 %.

Table 8 shows the ranges of minimum support thresholds $Q_{\text{Min\_sup}}$ that generate similar numbers of extracted sequential patterns of length-1, -2 and -3 for the three SPM methods. The left most, middle and right most plots correspond to the appropriate ranges of minimum support thresholds for the number of extracted sequential patterns of length-1, -2 and -3, respectively. In this experiment, the total number of target users is 30. The observation window size for $\Phi^{on}(t)$ is set to 3 weeks while that of the offline SPM $\Phi^{off}(t)$ is set to 12 weeks. For the off-line SPM, the length of the observation window is defined as the same as the total training period. For the on-line SPM, the length of the observation window is empirically set to 3 weeks by considering the amount of usage history data. The same $\Phi^{on}(t)$ and $\Phi^{off}(t)$ are also used for the hybrid SPM method. We found ten minimum support thresholds by which we obtain similar numbers of extracted sequential patterns of length-1 and length-2 for the offline, online and hybrid SPM methods. On the other hand, six minimum support thresholds were found to have similar numbers of the extracted sequential patterns of length-3 for the offline, online and hybrid SPM methods.

Figure 8 shows the precision performances of the offline, online and hybrid SPM methods for the recommended sequential patterns of length-1, -2 and -3.

As shown in Fig. 8a, the online SPM method outperforms the other two in precision performance for the recommended sequential patterns of length-1. This is because the online SPM method can best reflect the user's trend for the recently watched TV program contents of a relatively short length. In Fig. 8b, the hybrid SPM method outperforms the other two for the recommended sequential patterns of length-2. In Fig. 8c, the offline SPM method outperforms the other two for the recommended sequential patterns of length-3. It is interesting to see that the offline SPM method becomes more advantageous as the length of the recommended sequential patterns increases, and the online SPM method gets more advantage for the recommended sequential patterns of a shorter length. The hybrid SPM method takes its advantages between the short length and long length of the recommended sequential patterns. The highest precision for the recommended sequential patterns of length-1 is 0.877 for $Q_{\text{Min\_sup}} = 4.7$ by the online SPM method. For the recommended sequential patterns of length-2, the hybrid SPM method yields the precision performance of 0.793 for $Q_{\text{Min\_sup}} = 4.1$. The offline SPM method best performs for the recommended

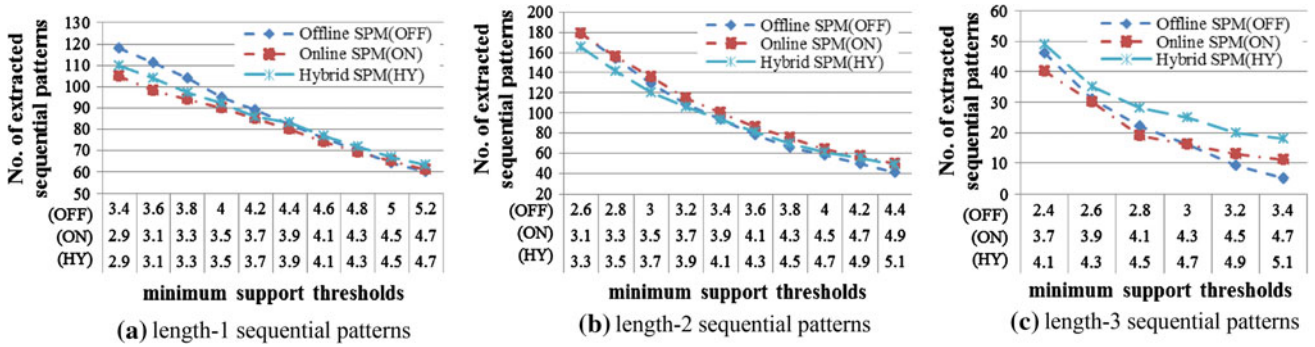

Fig. 7 Extracted sequential patterns of length-1, -2 and -3 for different minimum support thresholds

Table 8 Appropriate range of minimum support thresholds of length-1, -2, and -3 for offline, online and hybrid SPM methods

Minimum support threshold	Length-1	Length-2	Length-3
Offline SPM	3.4–5.2	2.6–4.4	2.4–3.4
Online SPM	2.9–4.7	3.1–4.9	3.7–4.7
Hybrid SPM	2.9–4.7	3.3–5.1	4.1–5.1

sequential patterns of length-3, producing the precision of 0.619 for $Q_{\text{Min\_sup}} = 3.4$.

4.3 Comparison between the offline SPM and PrefixSpan

Since the proposed SPM schemes are first proposed with the recommendation nature of TV program contents in sequential time orders, it is difficult to compare it with the other recommendation methods which do not deal with the sequential recommendation. Instead, we compare the precision performances of our proposed offline SPM method with those of the PrefixSpan algorithm for given minimum support thresholds. The minimum support threshold values of occurrence $O_{\text{Min\_sup}}$ for both methods are set to 10, 15, 20, 25 and 30 % of the total number of TV program content sequences watched by the users in each similar user group. To have similar numbers of extracted sequential patterns for length-1, -2 and -3, the offline SPM method uses the minimum support threshold $Q_{\text{Min\_sup}}$ of 2.4 for selecting the sequential patterns of length-1 and length-2, and $Q_{\text{Min\_sup}}$ of 1.6 for selecting the sequential patterns of length-3. Figure 9 shows the precisions of length-1, -2 and -3 sequential patterns extracted by the offline SPM method and the PrefixSpan algorithm when $O_{\text{Min\_sup}}$ is equal to 20 %.

As shown in Fig. 9, the precision performances of the offline SPM method are all higher than those of the PrefixSpan algorithm for the extracted length-1, -2 and -3 sequential patterns. Table 9 summarizes the performance comparisons between the offline SPM method and the PrefixSpan algorithm for various minimum support thresholds of the occurrence in terms of the average precisions. As shown in Table 9, the offline SPM method outperforms the PrefixSpan algorithm in the average precisions for the extracted sequential patterns of length-1, -2 and -3 for various $O_{\text{Min\_sup}}$ thresholds except for the length-3 sequential patterns when $O_{\text{Min\_sup}} = 30\,\%$. This is because the PrefixSpan algorithm takes more advantage of finding relevant sequential patterns for more selected sequential patterns under this particular condition while the offline SPM method extracts relatively a smaller number

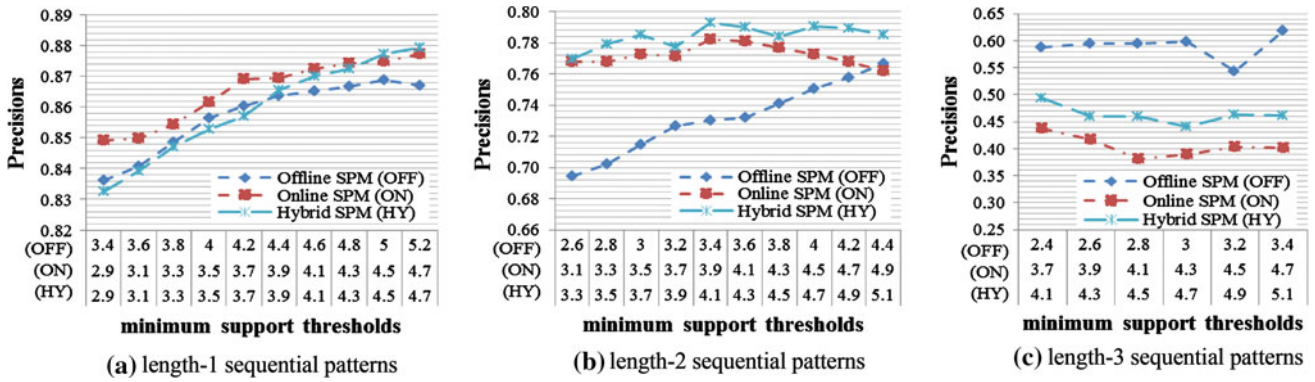

Fig. 8 Average precision of online, offline and hybrid SPM methods

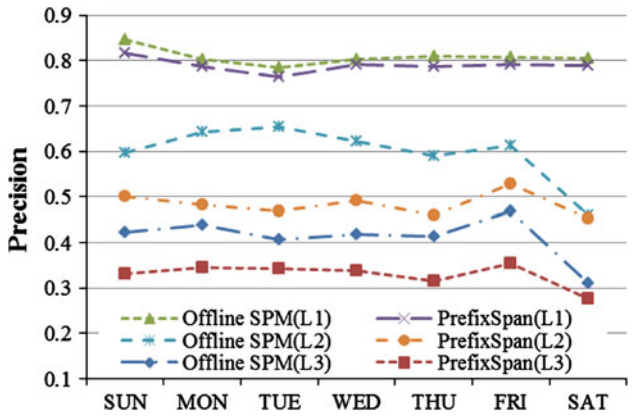

Fig. 9 Precision comparisons between the offline SPM method and the PrefixSpan algorithm

Table 9 Average precisions of the offline SPM method and the PrefixSpan algorithm

$O_{\text{Min\_sup}}$	Precisions				
	10 %	15 %	20 %	25 %	30 %
Length-1					
Offline SPM	0.789	0.794	0.806	0.831	0.861
PrefixSpan	0.708	0.753	0.790	0.829	0.861
Length-2					
Offline SPM	0.595	0.595	0.597	0.612	0.639
PrefixSpan	0.355	0.422	0.484	0.554	0.625
Length-3					
Offline SPM	0.410	0.412	0.411	0.434	0.436
PrefixSpan	0.178	0.253	0.329	0.421	0.512

(close to 0) of sequential patterns due to its more conservative criterion with an additional minimum support threshold of net occurrence.

It is interesting to see that the offline SPM method exhibits more superior performance in average precisions for the extracted sequential patterns of longer lengths when the minimum support threshold of occurrence is smaller. That is, the offline SPM method finds more meaningful sequential patterns among larger sets of the extracted sequential patterns of longer lengths for smaller minimum support threshold values of occurrence.

4.4 Computational complexity

In this section, we will discuss about the processing time of our proposed SPM methods. Figure 10 shows three kinds of processing time (second); 1) the processing time of offline SPM for total training data, 2) online SPM for total training data and 3) hybrid SPM for total training data with various minimum support values. As shown in Fig. 10, the processing times of offline SPM radically change when the minimum support value is changing. On the other hand, the processing times of online approach are homogeneous against the minimum support values and they are much smaller than offline approach. Offline SPM extracts patterns using projection, whereas online SPM extract patterns using Item-Sequence relationship representation and update the pattern values. So, as the minimum support value becomes smaller, the number of extracted patterns becomes larger and the number of execution of projection increases.

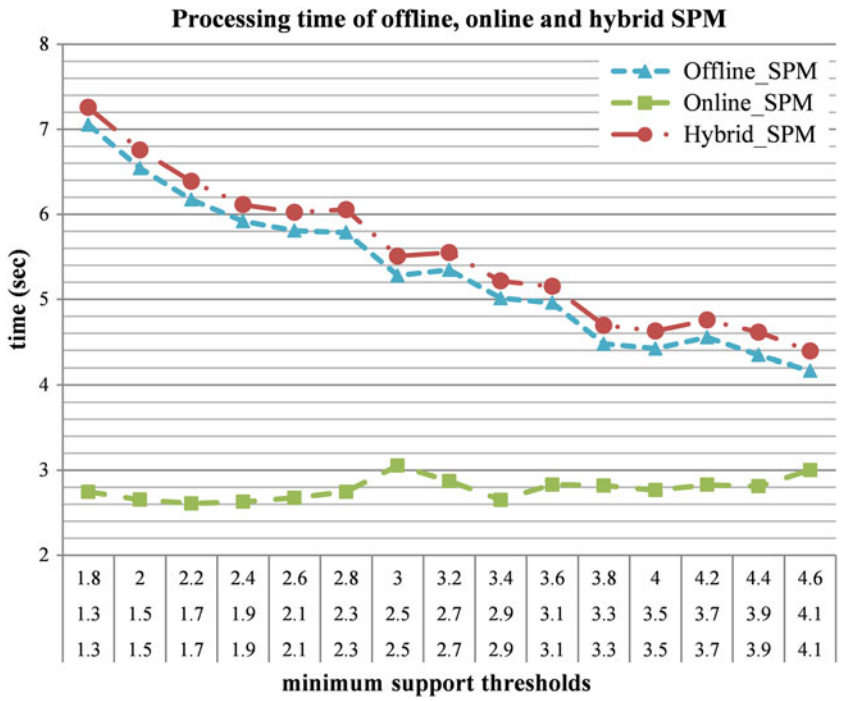

Fig. 10 Processing time of offline, online and hybrid SPM

However, in online SPM case, the minimum support value did not affect the execution time of the algorithm. The reason is that the number of execution for Item-Sequence relationship is independent with minimum support value. Also, online SPM is much faster than offline SPM for the same amount of data. When we find the sequential patterns based on online SPM, we divide the data set into 4 subsets and we re-organize each subset of database into Item-Sequence relationship. These transformation procedures for the 4 subset of database just scan the given dataset once and update the pattern values. So, computational complexity is lower than offline SPM which executes projection process for each selected frequently watched program. Also, the processing time for the hybrid SPM is slightly larger than that for the offline SPM because the hybrid SPM takes the total processing time of the offline SPM and the online SPM within an observation window for training.

4.5 Discussion

For reliable recommendation, the data sparsity and cold start problems must be discussed: First, for an active user who has sparsely recorded TV usage history data, it is difficult to directly recommend based only on the active user's TV usage history data, which might yield in imprecise recommendation results. One way of alleviating this is to utilize the common set of TV usage history data from the similar user group to which the active user belongs, as the proposed SPM scheme; second, for the cold start problem that a new user comes into a the recommendation system, any recommendation scheme cannot effectively work without some initial information. However, the quality and efficiency of the recommendation system to solve the cold start problem depend on the ability of predicting good recommendations with the minimum amount of initial information about new users or new items [22]. For the recommender systems of TV program contents user profile information such as gender and age can be used as initial information for the cold start recommendation. That is, a user group is generated with the existing users of the same gender and age of a new user. The cold start recommendation for the new user can then be made based on the TV usage history data of the group.

5 Conclusions and future work

In this paper, we proposed an automatic recommendation scheme of a sequence of TV program contents in a time-ordered manner using the offline, online and hybrid SPM methods. Before we extract sequential patterns based on the three SPM methods, similar user grouping is performed based on the proposed PW-NMRR metric. The PW-NMRR metric is used to measure the similarity between a non-target user and a target user by taking into account not only the watching order but also the preference values of preferred TV program contents. So, the PW-NMRR metric is more effective to similar user grouping for SPM, compared to the NMRR metric which only considers the watching order. Based on the similar users for each target user, we extract more meaningful sequential patterns to target users by considering the occurrence and net occurrence of watched TV program contents.

The Offline SPM method extracts the sequential patterns from the whole TV watching history database with projection process, and online SPM methods extracts the sequential patterns from a sliding observation window based on the updated occurrence and net occurrence values of watched TV program contents, and represents them in the *P–S* relation to effectively find longer sequential patterns in the subsequent DPS's and SPS's. The hybrid SPM method extracts the sequential patterns by combining the offline and online SPM methods with overlapping observation window. The offline SPM method is superior in relative long-sequence recommendation by utilizing accumulated long-term history of watched TV program contents. On the other hand, the online SPM method is effective for short-sequence recommendation by instantaneously reflecting more recently watched TV program contents in the *P–S* relation with the occurrence and net occurrence updating schemes. The hybrid SPM method compromises its performance between the offline and online SPM methods. The maximum precisions of 0.877, 0.793 and 0.619 for the recommendation of sequential patterns of length-1, -2 and -3 were obtained from the online, hybrid and offline SPM methods, respectively.

As our future work, we plan to reflect the target TV user's opinions for the automatically recommended TV programs into recommender systems as a relevance feedback. Also, for a composite sequence for TV program recommendation will be studied where the sequence has multi-paths so that a target TV user can traverse the paths (sequences) of recommended TV programs for their choices, instead of a one single rigid sequence path.

Acknowledgments This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2012-01120197). This work was supported by the IT R&D program of MKE/KEIT. [10039161, Core UI technologies for improving Smart TV UX].

References

1. Kim, E., Pyo, S., Park, E., Kim, M.: An automatic recommendation scheme of TV program contents for (IP)TV personalization. IEEE Trans. Broadcast. **57**(3), 674–684 (2011)

2. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. IEEE Trans. Knowl. Data Eng. **17**(6), 734–749 (2005)
3. Burke, R.: Hybrid recommender systems: survey and experiments. User Model. User-Adapt. Interact. **12**(4), 331–370 (2002)
4. Agrawal, R., Srikant, R.: Mining sequential patterns. 11th International Conference on Data Engineering, Taipei, Taiwan, pp 3–14 (1995)
5. Pyo, S., Kim, E., Kim, M.: Automatic recommendation of (IP) TV program schedules using sequential pattern mining. Adjunct Proceedings of EuroITV 2009, Leuven, Belgium, pp 50–53 (2009)
6. Cooley, R., Mobasher, B., Srivastava, J.: Web mining: Information and pattern discovery on the World Wide Web. In: Proceedings of the 9th IEEE International Conference On Tools With Artificial Intelligence, Newport Beach, CA, pp 558–567 (1997)
7. Wu, H.-Y., Zhu, J.-Y., Zhang, X.-Y.: The explore of the web-based learning environment base on web sequential pattern mining. In: Proceedings of the International Conference on CiSE, Wuhan, pp 1–6 (2009)
8. Tseng, S.-M., Tsui, C.-F.: Mining multilevel and location-aware service patterns in mobile web environment. IEEE Trans. Syst. Man Cybernet. B **34**(6), 2480–2485 (2004)
9. Pei, J., Han, B., Mortazavi-Asl, B., Pinto, H.: PrefixSpan: mining sequential patterns efficiently by prefix-projected pattern growth. Proceedings of the 17th International Conference on Data Engineering, Heidelberg, Germany, pp 215–226 (2001)
10. Pei, J., Han, J., Mortazavi-Asl, B., Wang, J., Pinto, H., Chen, Q., Dayal, U.: Mining sequential patterns by pattern-growth: the PrefixSpan approach. IEEE Trans. Knowl. Data Eng. **16**(11), 1424–1440 (2004)
11. Zhou, B., Hui, S.C., Chang, K.: An intelligent recommender system using sequential web access patterns. In: Proceedings of the 2004 IEEE Conference on Cybernetics and Intelligent Systems, vol 1. Singapore, pp 393–398 (2004)
12. Huang, J.-W., Tseng, C.-Y., Ou, J.-C., Chen, M.-S.: A general model for sequential pattern mining with a progressive database. IEEE Trans. Knowl. Data Eng. **20**(9), 1153–1167 (2008)
13. Ayres, J., Gehrke, J., Yiu, T., Flannick, J.: Sequential pattern mining using a bitmap representation. In: Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Edmonton, Alberta, Canada, pp 429–435 (2002)
14. Agrawal, R., Srikant, R.: Mining sequential patterns: generalization and performance improvements. In: Proceedings of the 5th International Conference on Extending Database Technology, vol. 1057. Avignon, France, pp. 3–17 (1996)
15. Han, J., Pei, J., Yan, X.: Sequential pattern mining by pattern-growth: principles and extensions. In: Chu, W., Lin, T. (eds.) Foundations and Advances in Data Mining, Studies in Fuzziness and Soft Computing 180, pp. 183–220. Springer, Berlin (2005)
16. Zhao, Q., Bhowmick, S.S.: Sequential Pattern Mining: A Survey. Technical Report, CAIS, Nanyang Technological University, Singapore, No. 2003118 (2003)
17. Mabroukeh, N.R., Ezeife, C.I.: A taxonomy of sequential pattern mining algorithms. ACM Comput. Surv. **43**(1), 3:1–3:41 (2010)
18. Manjunath, B.S., Ohm, J.-R., Vasudevan, V.V., Yamada, A.: Color and texture descriptors. IEEE Trans. Circuits Syst. Video Technol. **11**(6), 703–715 (2001)
19. Ndjiki-Nya, P., Restat, J., Meiers, T., Ohm, J.-R., Seyferth, A., Sniehotta, R.: Subjective Evaluation of the MPEG-7 Retrieval Accuracy Measure (ANMRR). ISO/IEC JTC1 SC29 WG11, Geneva, Switzerland, Doc. M6029 (2000)
20. Manjunath, B.S., Ohm, J.-R., Vasudevan, V.V., Yamada, A.: Color and texture descriptors. IEEE Trans. Circuits Syst. Video Technol. **11**(6), 703–715 (2001)
21. Wong, K.-M., Po, L.-M.: MEPG-7 dominant color descriptor based relevance feedback using merged palette histogram. IEEE Int. Conf. Acoust. Speech Signal Process. **3**, 433–436 (2004)
22. Rodríguez, R.M., Espinilla, M., Sánchez, P.J., Martínez, L.: Using linguistic incomplete preference relations to cold start recommendations. Internet Res. **20**(3), 296–315 (2010)