USER PREFERENCE BASED INFORMATION PERSONALIZATION FOR EASY ACCESS TO MULTIMEDIA CONTENTS

Jeongyeon Lim, Saggil Kang and Munchurl Kim

Laboratory for Multimedia Computing, Communications and Broadcasting
Information and Communications University (ICU)
103-6 Munji-dong Yusong-gu Daejeon 305-714 Korea
{jylim, sang, mkim}@icu.ac.kr

ABSTRACT

In this paper, we introduce a method of presenting user preference based TV program information by a user preference learning algorithm using a Bayesian network. And, we introduce a content recommendation system based on the proposed user preference learning method. Finally, we provide experimental results using a large set of usage history data for watching TV programs. The experimental results show that our proposed method is effective in learning user's timely changing preference so that appropriate recommendation for user's preferred contents can be made for easy access to alleviate the overloaded multimedia information presented to users. Our user preference learning is not limited for digital broadcasting but can also be applicable to Web surfing.

1. INTRODUCTION

For digital broadcasting environment with large amounts of TV programs being serviced, it is difficult for a user to surf the TV programs and select the user's preferred TV program contents in time, thus resulting in troublesome burden to users. By predicting or recommending the user's preferred contents (TV programs), it is possible to alleviate the user's burden in finding what she/he wants.

In this paper, we use as a user description model the *User Preference* description scheme in MPEG-7 MDS [3] and introduce a user preference learning method developed based on a Bayesian network (BN). Our user preference learning method devises a weighting scheme on input data to the BN in order to reflect recent usage history data more significantly in the computation of attribute values in the network. Then, a statistic BN can produce a recommended or predicted list for the user preferred TV programs in a more reasonable way for timely changing user preference (for example, on TV program genre) in a short period of time. In order to complete these tasks, we apply supervised learning (SL) on the data weighting scheme to trace a trend of the user preference depending on time. Finally, we apply this proposed algorithm to our recommendation system for TV programs.

The remainder of this paper is organized as follows: Section 2 describe the framework including BNs and the data weighting scheme. Section 3 illustrates our supervised learning method. Section 4 illustrates our recommendation system according to the user preference. In Section 5, we present experimental results for our proposed method in terms of the accuracy. We then conclude our paper in Section 6.

2. FRAMEWORKS

2.1 Bayesian networks

A BN, as shown in Figure 1, is an acyclic graphical expression of the causal relationship of variables, which can induce the conditional probability distribution of a random variable from a set of variables directly connected to the variables.

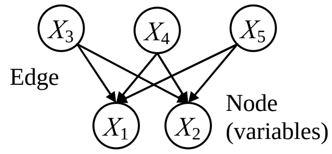


Figure 1 an example of causal networks

A BN can be denoted as $B = \langle G, X \rangle$, where G is the edges and X is a set of random variables in B. Let's consider a random vector $X = [X_1, X_2, ..., X_M]$, where a variable X_i can have r_i possible attributes. The conditional probabilities (CPs) of a state x_{ij} in a variable X_i can be calculated when evidence is embedded to the parent variables and can be denoted as

$$\theta_{x_{ij}|Pa_i} = p(X_i = x_{ij} \mid Pa_i, B) = n_{ij} / N_i \tag{1}$$

Here, N_i is the total number of samples in X_i and n_{ij} is the sample number of x_{ij}. The user preference models are usually utilized to statistically recommend or predict the user's near future preference (or intention) by analyzing the usage history data for contents (for example, TV programs that were viewed by the user). Usually, an attribute having the highest conditional probability is recommended when evidences are embedded in its parent variables Pa_i.

2.2 A data weighting scheme

Usage of contents (consumption of TV programs) is recorded over time. And, from (1), the CP at a state x_{ij} is calculated by the ratio of the total number (total number of content usage) of samples to the sample number (number of content usage) that corresponds to x_{ij}. Due to this, (1) is not appropriate to more significantly reflect the user's recent usage of contents to CP computation. So we partition the content usage data into two folds, old preference data and current preference data, by windowing as shown in Figure 2. The old preference data is represented in a old preference region $OPR_i(s)$ using M windows of a time length L, and the current preference data is located in the rightmost window indicated as $CPR_i(s)$. Here the subscript index i indicates that $OPR_i(s)$ and $CPR_i(s)$ are considered for the variable X_i with the same index i. The usage data is collected at a variable i with specific evidences that were embedded to its parent variables. The CPs (1) can be modified by considering the weights:

$$\hat{\theta}_{xj|Pa_i} = p(X_i = x_{ij} \mid Pa_i, B, \alpha s') = \sum_{m=1}^{M} \alpha_m n_{ij,m} \Big/ \sum_{m=1}^{M} \alpha_m N_{i,m} \quad (2)$$

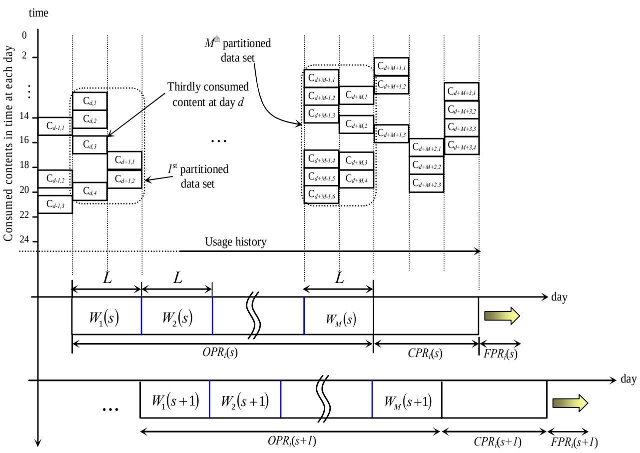


Figure 2 Window-weighting factors, *OPR_i*, *CPR_i*, and *FPR_i* in the training data as sliding one window

In order to impose different significances on partitioned data sets in different intervals in the windows $W_1(s), W_2(s), \cdots, W_M(s)$, each window $W_M(s)$ in the $OPR_i(s)$ has the corresponding weighting factor (or weight) $\alpha_m(s)$ where $m = 1, 2, \cdots, M$. The optimal weights are obtained in the sense that the mutual information (MI) $I(CPR_i(s); \hat{O}PR_i(s))$ is maximized [2].

3. USER PREFERENCE MODEL

3.1 derivation of the learning algorithm

Our supervised learning method is to determine the optimal weights in which $I(CPR_i(s); \hat{O}PR_i(s))$ is used for the external teacher to update the window weight s, as viewed in Figure 3 in which $\hat{O}PR_i(s)$ is the weighted $OPR_i(s)$.

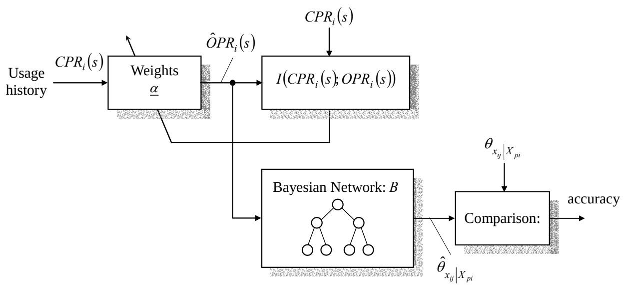


Figure 3 supervised learning a Bayesian network

At each sliding, the weights are adapted to the direction in which the mutual information (MI) is maximized. The *MI* can be defined as

$$\begin{aligned} I(\tilde{O}PR_i(s); CPR_i(s)) &= \log\left(\frac{p(CPR_i(s) \mid \hat{O}PR_i(s))}{p(CPR_i(s))}\right) \\ &= \log(p(CPR_i(s) \mid \hat{O}PR_i(s))) - \log(p(CPR_i(s))). \end{aligned} \quad (3)$$

In (3), the *MI* is a function of weights $\alpha_m(s)$ and we compute the partial derivative of the mutual information with respect to weights $\alpha_m(s)$ for obtaining a set of optimal weights. We assume that occurrence of each attribute is independent of that of others. In order to obtain $p(CPR_i(s) \mid \hat{O}PR_i(s))$, we first need to determine the conditional probability $p(x_{ij}(s) \mid \hat{O}PR_i(s))$ which means the probability of the attribute $x_{ij}(s)$ happening in the given $\hat{O}PR_i(s)$ domain.

$$p(x_{ij}(s) \mid \hat{O}PR_i(s)) = \sum_m \alpha_m(s) n_{ij,m}(s) \Big/ \sum_m \alpha_m(s) N_m(s) \quad (4)$$

Here, $N_m(s)$ and $n_{ij,m}(s)$ are the total number of samples and the number of j^{th} sample at the m^{th} window in $\hat{O}PR_i(s)$, respectively. The $p(CPR_i(s) \mid \hat{O}PR_i(s))$ can be obtained by just multiplying up the conditional probabilities of all elements in $CPR_i(s)$ because they are independent one another.

$$\begin{aligned} p(CPR_i(s) \mid \hat{O}PR_i(s)) &= \prod_j p(x_{ij}(s) \mid \hat{O}PR_i(s))^{n_{xij}(s)} \\ &= \prod_j \left(\sum_m \alpha_m(s) n_{ij,m}(s) \Big/ \sum_m \alpha_m(s) N_m(s)\right)^{n_{xij}(s)} \end{aligned} \quad (5)$$

Therefore, the weights are updated every epoch which is defined one sweep through implementing the GA. The amount of update $\Delta\alpha_m(s)$ can be obtained using the "delta rule" defined as

$$\Delta\alpha_m(e,s) = \eta \frac{\partial I(\hat{O}PR_i(s); CPR_i(s))}{\partial \alpha_m(s)}. \quad (6)$$

Here, notation e in $\alpha_m(e,s)$ is defined as the number of epoch and η is the learning-rate parameter of the GA, which determines the degree of searching step in the weight space during the gradient ascent process. We can express the weight update in every epoch e as

$$\alpha_m(e,s) \leftarrow \alpha_m(e-1,s) + \Delta\alpha_m(e,s). \quad (7)$$

We stop the training when the MI does not increase for some predetermined epoch [2].

4. USER PREFERENCE DESCRIPTION MODEL

Table 1 shows the content of MPEG-7 *User Preference* description scheme (DS) [3]. The *User Preference* DS contains *Usage Preference* DS and User Identifier which is used to identify users for user preference description data. The Usage Preference DS has two child elements: *Browsing Preference* DS and *Filtering and Searching Preference* DS. The *Filtering and Searching Preference* DS specifies creation preference, source preference and classification preference. The *Browsing Preference* DS has the *Summary Preference* DS that contains *SummaryTypePreference*, *PreferredSummaryTheme*, and other summary types.

Table 1 User Preference Types [3]

Browsing Preference DS	FilteringAndSearchPreference DS		
	Creation Preference	Classification Preference	Source Preference

· Summary Type Preference · Preferred Summary Theme	· Title · Creator · Keyword · Location · DatePeriod	· Country · DatePeriod · Language · Genre · Subject · Media Review · Parental Guidance	· Publication Type · Publication Source · Publication Place · Publication Date · Publisher · MediaFormat

Figure 4 shows the syntax of the User Preference DS.

```
<complexType name="UsagePreferencesType">
 <complexContents>
  <extension base="mpeg7:DSType">
   <sequence>
    <element name="FilteringAndSearchPreferences"
             type="mpeg7:FilteringAndSearchPreferencesType"
             minOccurs="0" maxOccurs="unbounded"/>
    <element name="BrowsingPreferences"
                  type="mpeg7:BrowsingPreferencesType"
                  minOccurs="0" maxOccurs="unbounded"/>
   </sequence>
    <attribute name="allowAutomaticUpdate" type="mpeg7:userChoiceType"
             use="default" value="false"/>
  </extension>
 </complexContents>
</complexType>
```

Figure 4 Syntax of User Preference DS [3]

5. EXPERIMENTAL RESULTS

5.1 user preference learning algorithm

We implemented our supervised learning process with the Digital TV genre recommendation problem. AC Nielsen Korea, one of the authorized market research company in Korea, provided two users' (Person A and B) watching history collected from December 1, 2002 to May 31, 2003. Person A is a teen-age girl and person B is a fifties-age man. The watching history was collected by a set-top box installed in the users' house, which can record login and logout time, broadcasting time and day, program genre, etc.

Due to limitation of information in the data, we can extract only TV watching time and day for genre variable. We considered the watching time from 6 pm. to 12 pm. The watching time (WT) variable has six states by slotting it by every one hour, so WT can be expressed as WT = {6~7pm, 7~8pm, …, 11~12pm}. Also, the watching day (WD) variable includes seven days as WD = {Monday, Tuesday, …, Sunday}. The genre (G) includes eight attributes as G = {Education, Drama&Movie, News, Sports, Children, Entertainment, Information, Etc.}. The states in WT and WD can be considered as evidences (E) for estimating the conditional probabilities of the attributes so there are 48 evidences obtained from the combination of the elements in WT and WD. If there is any missing week, the window of the week was replaced with the next week. We trained and plotted the weight curves and learning curves for $M=3$ ($\alpha_1(4), \alpha_2(4), \alpha_3(4)$) and then for two windows $M=2$ ($\alpha_1(5), \alpha_2(5)$), obtained by eliminating the first window from the three windows. The learning parameters are $\eta = 0.1$ and epoch = 300. From Figure 5, $\alpha_1(5)$ of $M=2$ and $\alpha_2(5)$ of M = 2 have a similar tracing behavior with $\alpha_2(4)$ of $M=3$ and $\alpha_3(4)$ of $M=3$, respectively, as we expected. In Figure 6, both windows converge to the same value of the MI so the redundant window does not affect the learning process. From Figure 5, 6, the ideal window number M is when no missing attributes as well as no redundant window exists in $OPR_i(s)$.

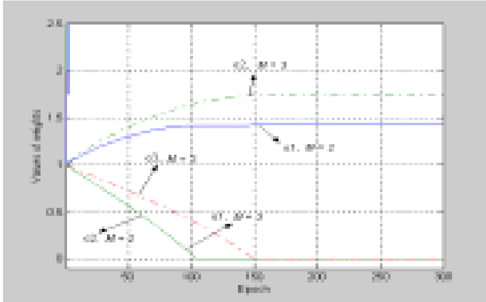


Figure 5 Weight adaptations

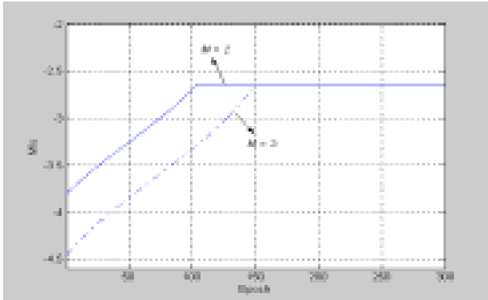


Figure 6 Learning curves for MI

5.2 Accuracy of our learning method

In this section, we show the numerical accuracy of our learning method by comparing with that of no training whose weights are all ones. The accuracy is the error of the recommended rank order compared with those of the true rank order in $FPR_i(s)$. As sliding one window at a time until arriving to the last window, we repeated training and calculating the accuracy, and then we tabulated the mean accuracies according to the different parameters such as M, η and δ as shown in Tables 2 and 3. The equation (8) is for the improvement of the accuracy by using our learning method, compared with no learning method.

$$\text{Improvement}(\%) = \frac{Accuracy\ of\ no\ train - Accuracy\ of\ train}{Accuracy\ of\ no\ train} \times 100 \tag{8}$$

From the tables, it is shown that the performances of our learning method are better than no learning method for the two persons, regardless of values of the parameters. For Person A, the biggest improvement we obtained is when $M=5$, for Person B, when $M=3$.

Table 2 Accuracy comparison: Person A

δ	η	The number of windows in *OPRi*(*s*) = *M*					
		M=3		*M*=4		*M*=5	
		Train	No train	**Train**	No train	**Train**	No train
0.01	0.01	**0.92**	1.05	**0.96**	1.12	**1.03**	1.22
	0.1	**0.91**	1.05	**0.89**	1.12	**0.91**	1.22
0.05	0.01	**0.87**	1.02	**0.89**	1.08	**0.93**	1.12
	0.1	**0.85**	1.02	**0.83**	1.08	**0.85**	1.12
Overall		**0.89**	1.04	**0.89**	1.10	**0.93**	1.17
Improvement		**14.4%**		**19.1%**		**20.5%**	

Table 3 Accuracy comparison: Person B

δ	η	The number of windows in *OPRi*(*s*) *M*		
		M=3	*M*=4	*M*=5

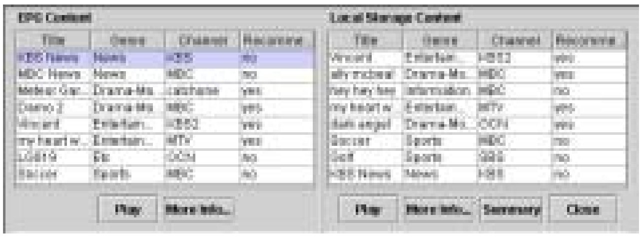
		Train	No train	**Train**	No train	**Train**	No train
0.01	0.01	**1.33**	1.44	**1.41**	1.49	**1.46**	1.55
	0.1	**1.26**	1.44	**1.32**	1.49	**1.33**	1.55
0.05	0.01	**1.18**	1.37	**1.22**	1.33	**1.26**	1.32
	0.1	**1.14**	1.37	**1.17**	1.33	**1.16**	1.32
Overall		**1.23**	1.41	**1.28**	1.41	**1.30**	1.44
Improvement		12.8%		9.2%		9.7%	

5.3 personalized EPG based on learned user preference

Figure 7 shows the GUI of our personalized EPG system. Our personalized EPG system presents to a user the EPG information based on his/her preference on TV program genres.

Figure 7 GUI of a personalized EPG system

Figure 8 shows personalized EPG's presented for Person A and Person B at the time of TV turn-on.

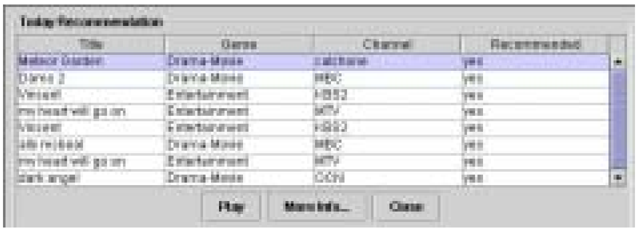


(a) Person A

(b) Person B

Figure 8 Personalized EPG's

Figure 9 shows the lists of recommended TV programs for Person A at 8 P.M. and for Person B at 10 P.M. on a Saturday. It can be easily noticed that the recommended TV program lists are different because Person A and Person B are in a teen-age and fifties-age with their own different user preferences.



(a) Recommended programs for Person A: 8 P.M. on Saturday

(b) Recommended programs for Person B: 10 P.M. on Saturday

Figure 9 Recommended TV program lists for a specific time and day

6. CONCLUSIONS

In this paper, we presented a user preference based information personalization method with a digital broadcasting application. We developed a new supervised learning of a Bayesian network which allows for dynamic learning for timely changing user's preference on content consumption by weighting the usage history data in time. The user preference model used in this paper is based on the MPEG-7 MDS *User Preference* description scheme which is also used as a basis in MPEG-21 DIA user characteristics description tools. From the experimental results with a large amount of realistic data, our user preference learning method is shown that the user preference on TV program genres can effectively be calculated in an automatic way. Prediction and recommendation on TV programs can be made for user's preferred genres for a given day and time.

REFERENCES

[1] M. Kim, G. Ryu, B. Bae, J. Nam, K. Kang and J. Kim, "Intelligent program guide for digital broadcasting," *Proceedings of the IWAIT2002*, pp. 257-263, Hualien, Taiwan, R.O.C., May 28, 2002.

[2] S. Kang, J. Lim, M. Kim, "Modeling the user preference on broadcasting contents using Bayesian belief network," *proceeding of VCIP,* San Jose, January, 2004.

[3] ISO/IEC JTC1/SC29/WG11, "ISO/IEC 15938-5 FDIS : Multimedia content Description Interface" MPEG2001/N4242, Sydney, Australia , July 2001.