

An new method to collaborative filtering recommendation based on DBN and HMM

Yong Mei Zhao¹, Hai Rong Wan², Zheng Xin Guo

¹ Department of Electric and Science, College of Science, Air Force Engineering University, China

² Department of Urban and Regional Planning, College of Urban and Environmental Sciences, Peking University, China

³ Department of Civil Engineering, Hunan University, China

Abstract: The main problems of collaborative filtering are initial rating, data sparsity and recommendation in time. A recommendation approach based on HMM model, which creates nearest neighbour set by simulating the user behaviours of web browsing, is a good way to solve the above problems. However, the HMM or model parameters constantly vary with customer's changing preference. When there is a new type of data to join, the HMM can only be discovered by relearn, which will affect real time of recommendation. Therefore a recommendation approach based on DBN and HMM is proposed. The approach will improve real time recommendation, and experiments shows that it has high recommendation quality.

Key words: hidden markov model(HMM), dynamic bayes network(DBN), collaborative filtering recommendation

1 Introduction

The main problems of collaborative filtering ^[1] are data sparsity, initial rating, recommendation in time and the expansion of data space. In order to solve these problems, an approach to collaborative filtering recommendation based on HMM is one of the good solution ^[2]. User behaviour changes constantly with the diversity of online consumer products, by which model or model parameters should be modified. Previous recommendation model once formed, model parameters cannot be changed arbitrarily, thus when there is a new

type of data to join, the HMM can only be discovered by relearn, which will affect real time of user behaviour recommendation. It is necessary to study how to improve the adaptive capability of recommend model on the basis of using previous model by updating its structure.

Short for dynamic Bayesian networks model (DBN), because of its flexibility of modelling, is widely used in many fusion algorithms. We can create recommendation model based on DBN to implement network structure learning--adding new features on the basis of previous collaborative filtering recommendation

technology, and combining all previous training set and new sample for leaning, which will both save time and optimize the network structure, then makes recommendation model meets user's needs well.

2 A collaborative filtering prediction model based on HMM^[3]

The HMM collaborative filtering model with users preference makes great improvements both in efficiency and accuracy recommendation results. The HMM collaborative filtering model is defined as following

$$P_{aj} = \frac{1}{n} \sum_{i=1}^n P_i(O | \lambda) \times A_{ij} \quad (1)$$

Where , A_{ij} is the preferences of usr i for commodity j ; Commodity j is what the path j represents; P_{aj} is the references probability of target user a for commodity j ; $P_i(O | \lambda)$ is the similarity of the nearest user i .

3 A collaborative filtering recommendation model based on DBN

3.1 Definition of DBN model^[5,6]

The probability equation of observation variable:

$$p(X_t, A_t | Y_t) = \sum_{m=1}^M p(M_t = m | Y_t) p(X_t, A_t | Y_t, M_t = m) \quad (2)$$

Where, $p(A_t | Y_t, M_t = m)$ is expressed in a simple distribution like passion distribution;

$$p(X_t | A_t, Y_t, M_t = m)$$

is expressed in conditional Gaussian model as following

$$p(X_t | A_t, Y_t, M_t = m) = N(X_t; u_m^t + BA_t, \sum_m^u) \quad (3)$$

Equation of state:

$$\theta_t = G_t \theta_{t-1} + w_t, w_t \sim N[0, W_t] \quad (4)$$

$$\text{The initial prior: } (\theta_0 | D_0) \sim N[m_0, C_0] \quad (5)$$

Where θ_t is the state parameter at time t , which is a $n \times 1$ matrix; G_t is $n \times n$ state transition matrix; v_t is observation error and w_t is state

variables, they are independent.

W_t can hardly be determined in practice, in order to solve the faults, the discount factor δ $0 < \delta < 1$ is incorporated into the model. A first order polynomial model with the discount factor is called a first order polynomial discount model, the specific implementation is as follows:

$$\mu_t = \alpha + \beta_t = \alpha + \beta_{t-1} + \beta_{r_t} = \mu_{t-1} + \beta_{r_t}$$

$\beta_t = \beta_{r_t} = (r_t / r_{t-1}) \beta_{t-1}$. Where, μ_t is the level of the sequence at time t , β_t is the growth of sequence level from time t_{i-1} to t_i .

The state equation:

$$\begin{cases} \mu_t = \mu_{t-1} + (r_t / r_{t-1}) \beta_{t-1} + w_{t,1} \\ \beta_t = (r_t / r_{t-1}) \beta_{t-1} + w_{t,2} \end{cases} \quad (6)$$

Where

$$w_t = \begin{bmatrix} w_{t,1} \\ w_{t,2} \end{bmatrix}$$

The initial information:

$$\left(\begin{bmatrix} \mu_{t_0} \\ \beta_{t_0} \end{bmatrix} \middle| D_{t_0} \right) \sim N[m_0, C_{t_0}] \quad (7)$$

Where

$$m_{t_0} = \begin{bmatrix} m_{t_0} \\ b_{t_0} \end{bmatrix}, C_{t_0} = \begin{bmatrix} C_{t_{0,1}} & C_{t_{0,3}} \\ C_{t_{0,3}} & C_{t_{0,2}} \end{bmatrix}$$

v_t is a term of observation error, which follows

normal distribution with zero mean and variances of V_{t_i} . D_{t_i} is the information

collection about system before time t_i .

$$W_{t_i} = (\delta^{-r_i} - 1) G_{t_i} G_{t_{i-1}} G_{t_i}^T$$

$$F = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, G_{t_i} = \begin{bmatrix} 1 & r_i / r_{i-1} \\ 0 & r_i / r_{i-1} \end{bmatrix}, \theta_{t_i} = \begin{pmatrix} \mu_{t_i} \\ \beta_{t_i} \end{pmatrix}$$

3.2 The model of reasoning and learning algorithm

1) Model reasoning algorithm^[7]

In the process of model reasoning, we mainly calculate two parameters: λ and π , which are served as β and in HMM.

Parameter λ and π can be calculated by the following formula:(assume the observation vector is e)

$$P(e) = \sum_j \lambda_j^i * \pi_j^i, \forall i \quad (8)$$

$$P(X_i = j | e) = \frac{\lambda_j^i * \pi_j^i}{\sum_j \lambda_j^i * \pi_j^i}, \forall i \quad (9)$$

The specific reasoning algorithm is as follows(algorithm 1):

For each variable X_i (poster order traversal)

If X_i is a leaf node,

If $\forall j$ are the values of observation vectors,
 $\lambda_j^i = 1$,

Else $\lambda_j^i = 0$,

Else

If $\forall j$ are the values of observation vectors,

$$\lambda_j^i = \prod_{c \in \text{children}(X_i)} \sum_f \lambda_f^c * P(X_c = f | X_i = j)$$

Else $\lambda_j^i = 0$

For each variable X_i (preordering traverse)

If X_i is a root node,

$$\pi_j^i = P(X_i = j)$$

Else

$$\pi_j^i = \sum_{v \in \text{CON}(P)} P(X_i = j | X_p = v) * \pi_v^p * \prod_{s \in \text{silbings}(X_i)} \sum_f \lambda_f^s * P(X_s = f | X_p = v)$$

(where X_p is the father of X_i)

2) Model learning algorithm^[8,9]

When the observation vector is incomplete with any given network structure, the classical EM algorithm is suitable. The training process of the algorithm is:(Algorithm 2)

Step E:

Assume the observation vector is e , the conditional probability of node C_i will be shown in equation (10):

$$P(C_i = w | e) = \frac{\lambda_w^i * \pi_w^i}{\sum_w \lambda_w^i * \pi_w^i} \quad (10)$$

The probability of every solidification node can be calculated according to equation(10), using the derived algorithm based on connected-tree algorithm.

Then calculate N_{ijk} --the number of times each variable X_i appeared, assuming $X_i = k$, parents(X_i)= j . First choose one solidification node, which contains both X_i and its father node.

Let V_{jk}^i be the collection of solidification nodes which satisfy $X_i = k$ and parents (X_i)= j , then

$$N_{ijk} = \sum_{w \in V_{jk}^i} P(C_i = w | e) \quad (11)$$

N_{ijk} increases with the addition of the training sample.

Step M:

After calculating N_{ijk} , the conditional probability of each variable can be reevaluate as shown in formula (12):

$$P(X_i = k | \text{parents}(X_i) = j) = \frac{N_{ijk}}{\sum_k N_{ijk}} = \frac{N_{ijk}}{N_{ij}} \quad (12)$$

3.3 Collaborative filtering recommendation algorithm based on DBN (algorithm 3)

Step 1: Initialize the X_0 ;

Step 2: Perform the collaborative filtering prediction model based on HMM (Model 1), training model based on the standard characteristics;

Step 3: Perform algorithm 1, algorithm 2 and establish DBN;

Step 4: Calculate $P(X_T | Y_{1:T})$, chose X_T maximizing $P(X_T | Y_{1:T})$.

3.4 The update model based on DBN

Combining the previous model with the nearest neighbour similarity recommended by HMM, the original forecast model of collaborative filtering updating for formula:

$$P_{aj} = \frac{1}{n} \sum_{i=1}^n S_i \times A_{ij} \tag{13}$$

$$S_i = \psi * P_i(O|\lambda) + \zeta * P_i(X_T^i|Y_{1:T}) \tag{14}$$

Where, A_{ij} is the preference of user i for commodity j ; Commodity j is what the path j represents. P_{aj} is the probability of which target user a like commodity j . S_i is the similarity of the nearest user i . ψ and ζ are balance index

which are used to define how HMM and DBN training results influence the model.

3.5 Nearest neighbour collaborative filtering method after updating

There are three parts of the updated collaborative filtering recommendation model: data pre-processing, HMM filtration, DBN updating model, collaborative filtering prediction model. Figure 1 depicts its structure:

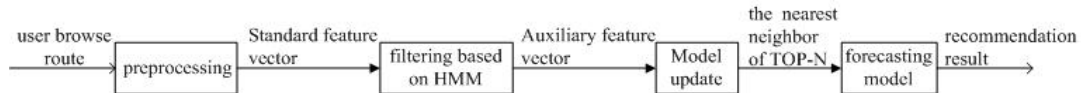


Figure 1. The updated collaborative filtering process

In nearest neighbour recommended phase, obviously, calculate $P_i(O|\lambda)$ and $P_i(X_T|Y_{1:T})$ separately, and choose X_T to maximize $P(X_T|Y_{1:T})$ and $P_i(O|\lambda)$. S_i can be considered as the similarity between user i and the target user, wherever λ is HMM parameter for a target user. N users of the most similarity will join the nearest neighbor set.

4 Evaluation of experiment

Experimental process is: (1) Calculate the similarity between the users by training set, and find the nearest neighbours of users in various scenarios.(2) Predict all of the items of the whole users, then figure out the collection of recommended results.(3) Using recommendation results and real record in test set, calculate recommended efficiency according to the evaluation standards.

Five parameters are used in the experiment, the access time of queries, IP address, URL, browse number and page load time. We take experiment with five users(for user1, user2, and user3, user5) data. User1 is a target user randomly generated the original 20000 feature vectors(the feature vector has been treated, so there was little

obvious mistake in the random data), 10000 of which are used for training network, the format is as follows:(2012-01-2222 : 10 : 15 , 192. 168. 1. 23, http://www.haoting.com/, 15 , 24)(2015-06-0212 : 02 : 15 , 192 . 168 . 1 . 21 , http://www.haoting.com/zhangjie/ , 15 , 122).....

The left 10000 of feature vectors are used for the experiment data of the nearest set recommendation. The same as user1, other users also randomly generate 20000 feature vectors for the recommendation experiment.

First of all, training the nearest set after model updating:

Perform algorithm 3, and calculate the similarity between users (Calculate $P_i(X_T|Y_{1:T})$, and choose X_T to maximize $P_i(X_T|Y_{1:T})$).When

$\psi = \zeta = 0.5$, new similarity S_i shows in table 1:

Table 1. The similarity of users after model updating

user	State number	Similarity based on HMM	Similarity after model updating

user1	153	0.993	0.996
user2	90	0.647	0.641
user3	145	0.975	0.985
user4	76	0.296	0.287
user5	69	0.188	0.218

This figure shows S_i --the similarity of different users in different states. Predict similarity between user1 and itself is 0.996. It is very intuitive from the figure that user3 is closest to target user1, which can be classified as nearest neighbour set.

Second, the recommendation process of user preference commodities after updating model:

We find the commodity 1001 meets the greatest degree of preferences of user3, and the recommend probability of target users is: $P_{user1, 1001} = 0.9894$.

Add user2 to the nearest neighbour set, calculate formula (14), we still find the commodity 1001 meets the greatest degree of preferences of user3.

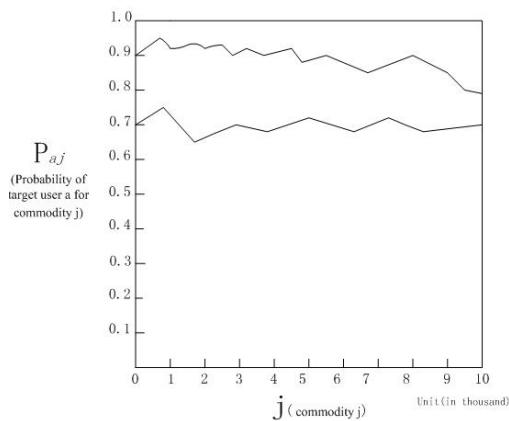


Figure 2. Results of experiment

Experiment result is shown in figure 2. The top curve providing a forecast of preferences probability of target user1 for commodity j with the nearest neighbour set (user3) while the bottom curve providing the results with the nearest neighbour (user3, user2).

5 Conclusion

We mainly introduces the method of updating the model of collaborative filtering

recommendation based on access path of users, which takes user's other characteristics into consideration, such as the user's static ratings, by using the high integration of dynamic Bayesian. In this way, we update the model, and achieve the dynamic behaviour data combined with static characteristics of the data, then acquire a more accurate recommendation.

References

1. You Wen, Ye Shui-sheng. A Survey of Collaborative Filtering Algorithm Applied in E-commerce Recommender System[J]. Computer Technology and Development, 2006, 16(9): 57-61.
2. Wang Xia, Liu Qin. Research on Application of Collaborative Filtering in Recommender Systems [J]. Applications of The Computer Systems, 2005, 4: 23-27.
3. Huang Guang-qiu, Zhao Yong-mei, Approach to collaborative filtering recommendation based on HMM [J]. Journal of Computer Applications, 2008.28(6): 1601-1604.
4. ZHANG Zi-li, LIU Wei-yi, State prediction based on the dynamic Bayesian network [J]. 2007, 29(1): 35-39.
5. LI Xiao-yi, XU Zhao-di, SUN Xiao-wei. Study on Parameter Learning of Bayesian Network [J]. Journal of Shenyang Agricultural University, 2007-02, 38(1): 125-128
6. Wei Fang, Discovering User Interests Based on Bayesian Network [D]. Xi'an: Xi'an University of Architecture and Technology, 2007: 25-29.
7. Liu Jie, Chinese Proper Names Recognition Based on Dynamic Bayesian Network [D], Shanxi: Shanxi University, 2006: 11-13.
8. Sun A-li, Research on Dynamic Bayesian Network Models for Audio-Visual Speech Recognition [D]. Xi'an: Northwestern Polytechnical University, 2007: 28-40.
9. Wang Xing-bin, Research on Application of Bayesian Net in Robust Speech Recognition [D]. Henan: The PLA Information Engineering University, 2006: 17.