International Journal of Grid and Distributed Computing
Vol. 9, No. 12 (2016), pp.23-34
http://dx.doi.org/10.14257/ijgdc.2016.9.12.03

A New Prediction Model Based on Web Access Behavior

Haiqin Ye [1, a,*], Huan Li [1, b] and Ailing Zhang [2, c]

[1]*School of Computer Science and Technology, Zhoukou Normal University, Zhoukou 466001, China*
[2]*Automation station, Anyang 455000, China*
[a]*onlyyhq@126.com,*[b]*15055594@qq.com,*[c]*zal715517@126.com*
** Corresponding Author: Haiqin Ye*

Abstract

In order to predict network users' access behavior accurately, this paper proposes a new prediction model based on Web access behavior. To improve prediction performance and reduce the state-space complexity, the model uses hybrid-order Markov chain structure and stores the sequences which share the same prefix. The problems that lower-order models have poor prediction performance and higher-order models result in high state-space complexity are solved in this paper. Simulation results have shown that the prediction model based on Web access behavior can improve the precision and recall to some extent.

Keywords*: Markov chain, prediction model, Web access behavior, state-space complexity, prediction performance.*

1. Introduction

The development of computer technology and communication technology, as well as the continuous growing of the demands for people connecting to the Internet, prompt the Internet to develop rapidly. In China, for example, the number of people using Internet is 668 million, and Internet penetration rate is 48.8 percent and 18.94 million people become new Internet users in half a year up to June 2015, said the thirty-sixth "statistics report of China Internet development"[1]released by CNNIC. Moreover, up to November 15, 2015, the number of people using Internet all of world is 3345832772 and Internet penetration rate is 46.4 percent according to the latest statistics [2]. It is obvious that the number of computers connecting to Internet and people using Internet presents a faster growth trend. From a user's perspective, the current Internet has become an information ocean in which we can develop and use data resources. The Internet prompts the transmission of network information extremely. People pay more attention to developing and making full use of information resources and these requirements play an immeasurable role in promoting the development of science, culture, economy and society.

With the massive increase in the number of Internet users, more and more users access the Internet resulting in the heavy internet traffic and the Web's popularity has significantly increased user-perceived latency. The obvious solution—to increase the bandwidth—is not viable, because we cannot easily change the Web infrastructure (the Internet) without significant economic cost. However, if we could predict future user requests, we could put those pages into the client-side cache when the browser is free. When a user requests one of the pages, the browser could retrieve it directly from cache. Thus, user-perceived latency can be reduced. It makes use of the combination of caching and prefetching. Caching and prefetching techniques both reduce the user-perceived latency through predicting users' access behavior. But the cache mechanism only uses the time locality of WWW access

ISSN: 2005-4262 IJGDC

pattern, it can't cache the documents which are not accessed before, so the quality of response can't be improved much yet. As a complement way, Web prefetching is the most effective method to break the upper bound of caching performance. And prefetching is one kind of the active caches that can cache the pages which are still not requested by the user, which is an expansion from the time locality to the space locality.

Markov model is a statistical model proposed by Andrei A. Markov. Markov models are widely used to model sequential processes, and have achieved many practical successes in areas such as web log mining, computational biology, speech recognition, natural language processing, robotics, and fault diagnosis. In Web domain, Markov models are used to predict users' access behavior. In general, they use the sequence of Web pages a user has accessed as input, with the goal of building Markov models with which they can predict the page the user will most possibly access next.

Presently Markov models have a lot of variations. The PPM (Prediction by Partial Match) prediction models[3-14] based on tree-like structure belong to multiple high-order Markov models; the DG(Dependency Graph) models [15-20]based on graph-like structure belong to 1-order Markov models. In many applications, first-order Markov models are not very accurate in predicting the user's browsing behaviors, since these models do not look far into the past to correctly discriminate the different observed patterns. As a result, higher-order models are often used. Unfortunately, these higher-order models have a number of limitations associated with high space complexity, reduced coverage, and sometimes even the worse prediction precision. Higher-order model states are different combinations of the actions observed in the data set, so the number of states tends to rise exponentially as the model order increases. This dramatic increase can significantly limit the applicability of Markov models for applications in which fast predictions are critical for real-time performance or for applications with tight memory constraints. Furthermore, many examples in the test set might not have corresponding states in higher-order Markov models, thus reduce their coverage.

A new hybrid-order Markov prediction model (NPM) is proposed in the paper. NPM stores the sequences which share the same prefix so as to reduce the state-space complexity. And its performance is assessed from many aspects.

2. NPM Model

Definition 1.Sequence Prefix: Given sequences Sequ1=< $a_1\ a_2\ \cdots\ a_n$ >, Sequ2=<$b_1\ b_2\ \cdots\ b_m$>, ($m<n$), the sequence Sequ2 is the prefix of the sequence Sequ1; iff $i<m$, $b_i = a_i$, all items of $b_m \subseteq a_m$and ($a_m - b_m$) rank after the item bm. The sequence Sequ2 can also be called prefixal sequence.

The first step of the model is to turn the event sequence into the tree-like model to store in order to the following mining. First of all, scan and access the sequence database, then construct the head table. The following is the algorithm of constructing the tree-like model.

Input: historical access sequence database DB
Output: k-order Tree
MSS_Generate(DB)
{ for each sequence s do
current=root
for each event e of sequence s do
if URL of current node's child nodes=URL of event e then

count of this child node++
current=this child node
else
current=Generate_New_Node (current event e)
endif
endfor
endfor
}Return Tree

In general, given the three sequences of three users: x1, x2; y1, y2, y3, y4, y5; z1, z2, z3, they are stored with PPM model and tree-like model respectively as shown in figure 1.

As figure 1 shows, they only need the space expenses of 11 nodes when they are stored with tree-like model; however, they need the space expenses of 56 nodes when they are stored with PPM model. Assuming the sessions which are formed with n different nodes, they only need the space expenses of (n+1) nodes when they are stored with tree-like model and the space complexity is O(n); however, they need the space expenses of $(0.5n^2+0.5n+1)$ nodes when they are stored with PPM model and the space complexity is $O(n^2)$ as shown as figure 2. For the sequences which have the same prefix, PPM models far exceed tree-like models on space complexity. The tree-like models have reduced the space complexity of the model greatly and saved the space expenses.

Definition 2 The header table is a hash table. Each item stores the name and the pointer of the header node of header-table queue.

Definition 3 The header table queue consists of a queue and another queue links all nodes of the same event.

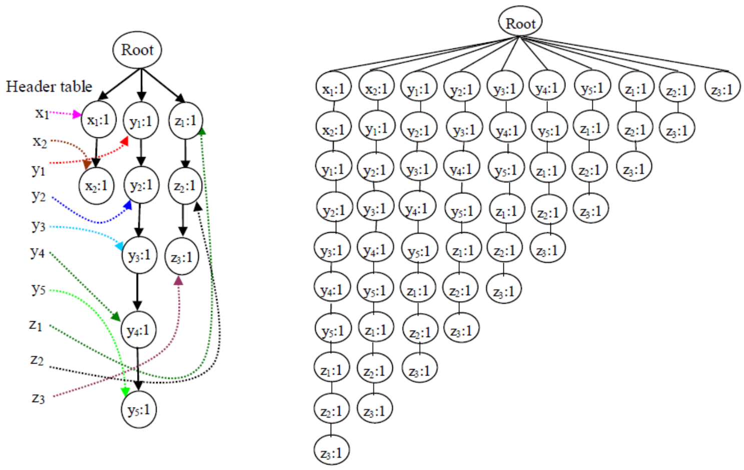


Figure 1. The Tree-like Structure and PPM Tree

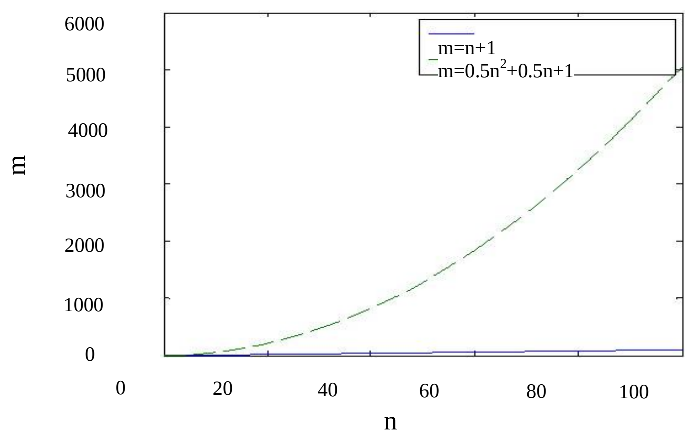


Figure 2. The Number of Note

Definition 4 Given a tree-like model G= (V, E), V is the aggregate of nodes and E is the aggregate of borders thereinto. Node contains the event name, counts and two quotations. The quotation of Next Header Queue Item points to the next node of header-table queue of the event; the quotation of Next Session Item points to next item of event sequence. The branch from the root node to the leaf node along with the quotation Next Session Item is a session.

After building the tree-like model, prediction algorithm is used to predict the result of hybrid-order model. The following is the prediction algorithm.

Input：The n-order tree-like structure Tree; a set of the last k requests, Requ[i], $0 \le i \le k \le n$; and the confidence threshold for each order, confth [j], $0 \le j \le n$

```
Output: prediction object ObjSetj
Method:
for(j=1;j<=maxOrder;j++){
for every header in Tree.HeaderTable{
for every node en in the header-queue{
Queue qu =Tree.JUnion(en,j-1);
if (qu.count>0)
AddToLink(en,qu.Dequeue(),Sqlist Lj);
Insert (child of qu.Dequeue(),Lj);}
}
  }
current_context[j]←node of depth j, representing the access sequence{Requ[k-j+1],…,Requ[k]};
ObjSetj←NULL;
for (length j=k;j>=1;j--){
  for every child_node chd of Current_context[j]
    if(occurrence_count of chd)/(occurrent_count of parent)>=conf th [j]
          ObjSetj←ObjSetj+chd;}
Return ObjSetj;
```

3. Simulations and Results Analysis

The NPM offers three methods for combining models: the accuracy voting method, selecting top value method and the top order method. To test performance, we compare the prediction results of our NPM with the results of HTMM [21].

NPM improves the accuracy voting method of HTMM. From Formula 1 to formula 3has shown how the prefetching value can be got from each order model. Firstly, choose the prediction page with the maximum prefetching value of j-order model as the prediction result, then get the prediction set PS={ P1, P2, …, PmaxOrder }, maxOrder∈N. Finally, compute the recommendation value of each page in PS using formula 3, where *Rec_j(PageX)* is j-order recommendation value of *PageX* in the current session.

Different from HTMM, the weight of each order isn't static and is computed using formula 2. The weight varies with the order. Order accuracy determines prediction weight. The precision of j-order model is computed using formula 1, where *rec(row)* is the prediction value and *row* is the row mark. The process of computing the precision of each order is dynamic. All the above operations are processed off-line.

Forluma 1: prediction accuracy.

$$Accuracy(j) = \frac{\sum_{row} rec^2(row)}{the number of row} \tag{1}$$

Forluma 2: prediction weight.

$$Weight(j) = \frac{Accuracy(j)}{\sum_k Accuracy(j)} \tag{2}$$

Forluma 3: recommendation value.

$$R(PageX) = \sum_{j=1}^{|PS|} rec_j(PageX) * Weight(j) \tag{3}$$

We have tested the precision, recall, prediction time and PRS of the two models. We coded all algorithms in Microsoft Visual C# and performed experiments on Pentium PC with 256 Mbytes RAM, 2.40GHZ CPU and Winxp operation system. We examined 1000000 records of Berkeley Trace and 98 World Cup trace respectively. After cleaning the original data, the URL of every requested page is coded according to the frequency, then forms the sessions.

3.1 Performance Parameters

We have evaluate the model in terms of precision, recall and PRS. Precision and recall which were proposed by Cleverdon in 1968[22] and have been used so far are the most commonly used metrics in the field of information retrieval field. Precision and recall were used for recommendation system by Billsus and Pazzani[23] in 1998 at the earliest , then are widely used in the field of recommendation gradually[24-33].

Definition 5: precision.

$$precision = P^+ / (P^+ + P^-) \tag{4}$$

where P^+ is the number of rightly predicted pages and P^- is the number of wrongly predicted pages.

Definition 6: recall.

$$recall = R^+ / |R| \tag{5}$$

where R^+ is the number of requests which is predicted by the model and $|R|$ is the number of all requests.

Precision and recall is often used as the valid metrics of Web prefetching. With the increase of prefetching data, it brings the high recall and low precision. It is necessary to use the metric PRS to weight the validity of prefetching synthetically.

Definition 7: PRS.

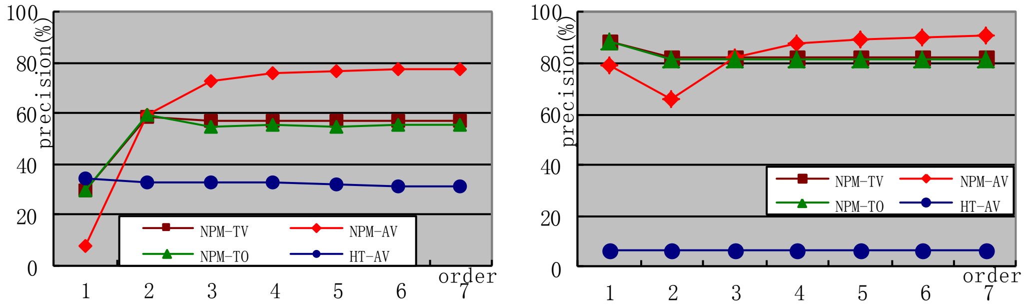
$$PRS = \frac{1}{\frac{1}{recall} + \frac{1}{precision}} = \frac{precision * recall}{precision + recall} \tag{6}$$

3.2 Results and Analysis

Table 1. English Name-English ab. Comparison table

English Name	Ab. English
top value method of NPM	NPM-TV
accuracy voting method of NPM	NPM-AV
top order method of NPM	NPM-TO
accuracy voting method of HTMM	HT-AV

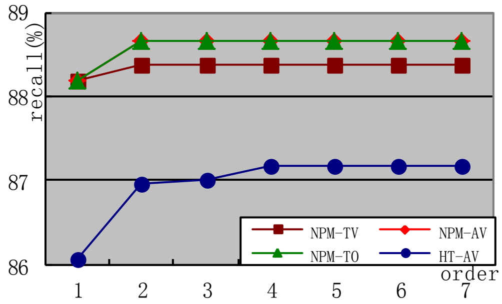
Figure 3 shows the relationship between precision and order. As figure 3 shows, of the three NPM combination methods, accuracy voting is more accurate as the order rises, the other two methods have similar precision regardless of order. From the second order, the three NPM combination methods are more accurate than HTMM combination method, while the HTMM combination method remains unchangeable nearly as the order increases. To all the combination methods, the precision remains unchangeable when the order reaches a certain value.



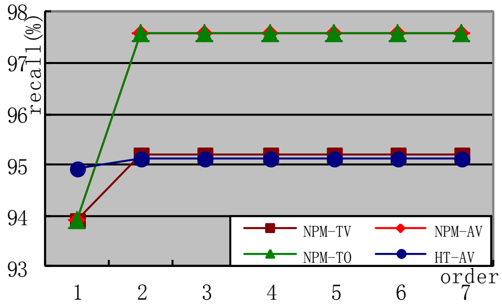
(a) Berkeley trace (b) 98 WC trace

Figure 3. Precision Comparison Curves

Figure 4 shows the relationship between recall and order. The performance of the three NPM combinations improves appreciably than that of the HTMM combination. Of the three NPM combination methods, accuracy voting and top order are better than top value, while the difference between accuracy voting and top order is very small as order increases. To all the combination methods, the recall remains unchangeable when the order reaches a certain value.



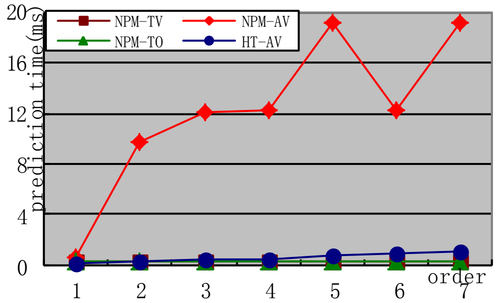
(a) Berkeley trace



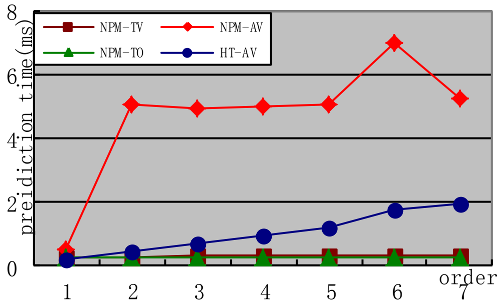
(b) 98 WC trace

Figure 4. Recall Comparison Curves

Figure 5 shows the relationship between the prediction time and order. As figure 5 shows, the variable set of the accuracy voting method of NPM is larger, while other three combinations change gently. Of the three NPM combination methods, top value and top order have similar prediction time, and, as the order rises, the time increases linearly and is shorter than the time for the HTMM combination method. We can see the prediction time for the accuracy voting method of NPM is the longest in all orders for all combination methods.



(a) Berkeley trace



(b) 98 WC trace

Figure 5. Prediction Time Comparison Curves

Figure 6 shows the relationship between PRS and order. PRS weights the validity of prefetching synthetically. As figure 6 shows, of the three NPM combination methods, accuracy voting is higher as the order rises, and the other two methods have similar PRS regardless of order. From the second order, the performance of the three NPM combination methods all improves compared with HTMM combination method, while the HTMM combination method remains unchangeable nearly as the order increases. To all the combination methods, PRS remains unchangeable when the order reaches a certain value.

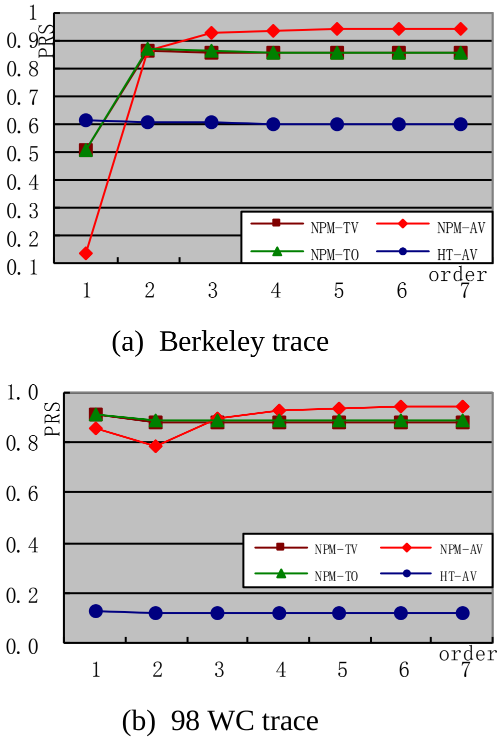


Figure 6. PRS Comparison Curves

4. Summary

Predicting users' access behavior precisely plays an important role in website design, E-commerce and personalized recommendation and so on. A new hybrid-order Markov prefetching model(NPM) is proposed in the paper. NPM stores the sequences which share the same prefix so as to reduce the space complexity. The NPM model integrated with diversified recommendation methods improves the quality of predicting users' access behavior, and enhances the quality of Intelligent Service such as Web prefetching and so on effectively and reduces the user-perceived latency.

Acknowledgments

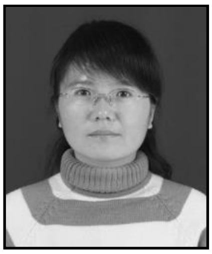
This work is supported by National Natural Science Foundation of China (project number: U1504613); the Soft Science Research Project of Henan Province (project number: 142400411220), China; Basis by the Frontier Projects in Henan Province (project number: 142300410432), China; Key Scientific Research Project of Colleges and Universities in Henan Province(project number: 15B520031).

References

[1] Information on: www.cnnic.cn/gywm/xwzx/rdxw/2015/201507/t20150723_52626.htm, **(2015)**.
[2] Information on: www.internetworldstats.com/stats.htm.
[3] C.-W. Lin and T.-P. Hong, "Maintenance of Prelarge Trees for Data Mining with Modified Records", Information Sciences, vol. 278, no. 10, **(2014)**, pp. 88-103.
[4] S. Su, S.-Z. Xu, X. Cheng, Z.-Y. Li and F.-C. Yang, "Differentially Private Frequent Itemset Mining via Transaction Splitting", IEEE Transactions on Knowledge & Data Engineering, vol. 27, no. 7, **(2015)**, pp. 1875-1891.
[5] Q. Wu and J.-X. Luo, "Improved Search Algorithm Based on Compressed FP-Tree", Computer Engineering and Design, vol. 36, no. 7, **(2015)**, pp. 1771-1777.
[6] J. Huang, M.-Q. Li and W.-Q. Guo, "Parallel FP-Growth Algorithm in Search Engines", Computer Science, vol. 42, no. 6A, **(2015)**, pp. 459-461.
[7] L. Shi, X.-G. Ding and W. Lin, "An Adaptive PPM Prediction Model", Journal of Computational Information Systems, vol. 2, no. 2, **(2006)**, pp. 633-638.
[8] Z.-G. Zhang and G.-L. Ji, "Parallel Algorithm for Mining Frequent Item Sets Based on FP-Growth", Computer Engineering and Applications, vol. 50, no. 2, **(2014)**, pp. 103-106.
[9] X.-S. Chen, S. Zhang, H. Tong and X.-J. Cui, "FP-Growth Algorithm Based on Boolean Matrix and Mapreduce", Journal of South China University of Technology(Natural Science Edition), vol. 42, no. 1, **(2014)**, pp. 135-141.
[10] D.-M. Fu and Z.-Q. Wang, "Mining Algorithm of Association Rule Based on FP-Tree and Constrained Concept Lattice and Application Research", Application Research of Computers. vol. 31, no. 4, **(2014)**, pp. 1013-1019.
[11] X. Chen and X.-D. Zhang, "A Popularity-Based Prediction Model for Web Prefetching", Computer, vol. 36, no. 3, **(2003)**, pp. 63-70.
[12] Y. Feng, J.-N. Yin and H.-Y. Xu, "Distributed Rules Mining Algorithm with Load Balance Based on Vertical FP-Tree", Journal of Computer Applications, vol. 34, no. 2, **(2014)**, pp. 396-400.
[13] Z.-B. Rong, D.-W. Xia and Z.-L. Zhang, "Complex Statistical Analysis of Big Data: Implementation and Application of Apriori and FP-Growth Algorithm Based on MapReduce", Proceedings of 4th IEEE International Conference on Software Engineering and Service science, **(2013)**; Beijing, China.
[14] L. Shi, Y. Zhang, Y.-X. Pei and Z.-M. Gu, "A PPM Prediction Model Based on Web Objects' Popularity", Journal of Chinese Computer Systems, vol. 27, no. 7, **(2006)**, pp. 1378-1382.
[15] G.-J. Han and H. Zhu, "Human Pose Estimation Algorithm Based on Pictorial Structure Model", Computer Engineering and Applications, vol. 49, no. 14, **(2013)**, pp. 30-33.
[16] H.-Y. Ye, Z.-W. Ni and L.-P. Ni, "Novel Graph-Based Recommender System with Interest Drift Detection", Journal of Chinese Computer Systems, vol. 33, no. 4, **(2012)**, pp. 700-706.
[17] S.-R. Yan, X.-L. Zheng, Y. Wang, W.-W. Song and W.-Y. Zhang, "A Graph-Based Comprehensive Reputation Model: Exploiting the Social Context of Opinions to Enhance Trust in Social Commerce", Information Sciences, vol. 318, **(2014)**, pp. 51–72.
[18] B. Xu, J.-J. Bu, C. Chen, C. Wang, D. Cai and X.-F. He, "EMR: A Scalable Graph-Based Ranking Model for Content-Based Image Retrieval", Knowledge & Data Engineering IEEE Transactions, vol. 27, no. 1, **(2015)**, pp. 102-114.
[19] Y. Xu, Z.-W. Zhao, Q. Chen, H.-L. Liu, J. Du, J.-Q. Hu and Z.-H. Li, "Graph-Based Web Entity Ranking Method", Computer Science, vol. 41, no. 5, **(2014)**, pp. 219-222.
[20] L.-M. Wang and K. Li, "An Algorithms for Learning the Structure of Utility Graph Based on Nearest-Biclusters Collaborative Filtering", Chinese Journal of Computers, vol. 33, no. 12, **(2010)**, pp. 2291-2299.
[21] D.-S. Xing and J.-Y. Shen, "A New Markov Model for Web Access Prediction", Computing in Science and Engineering, vol. 4, no. 6, **(2002)**, pp. 34-39.
[22] J. L. Herlocker, J. A. Konstan, L. G. Terveen and J. T. Riedl, "Evaluating Collaborative Filtering Recommender Systems", ACM Transactions on Information Systems (TOIS), vol. 22, no. 1, **(2004)**, pp. 5-53.
[23] D. Billsus and M. J. Pazzani, "Learning Collaborative Information Filters", Proceedings of the Fifteenth International Conference on Machine Learning, **(1998)**; Madison, USA.
[24] J.-D. Zhang and C.-Y. Chow, "CoRe: Exploiting the Personalized Influence of Two-Dimensional Geographic Coordinates for Location Recommendations", Information Sciences, vol. 293, **(2015)**, pp. 163-181.
[25] P.-Y. Zhang, E.-H. Chen and B. Huang, "Personalized Requirements Oriented Trustworthy Services Recommendation Based on Social Network", Journal on Communications, vol. 34, no. 12, **(2013)**, pp. 49-59.
[26] R.-M. Li, H.-F. Lin and J. Yan, "Mining Latent Semantic on User-Tag-Item for Personalized Music Recommendation", Journal of Computer Research and Development, vol. 51, no. 10, **(2014)**, pp. 2270-2276.
[27] C. Liu, G.-L. Liang and G.-P. Tan, "Fusion of Personalized Recommendation Model Based on User's Interest Drifting", Computer Engineering and Design, vol. 34, no. 8, **(2013)**, pp. 2944-2950.

[28] Y. Zhang, B. Zhang, K. Gao, P.-W. Guo and D.-M. Sun, “Autonomy Oriented Personalized Tag Recommendation”, Acta Electronica Sinica, vol. 40, no. 12, **(2012)**, pp. 2353-2359.
[29] S.-Z. Liu, X.-P. Fan, Z.-F. Liao and J. Hu, “Hidden Markov Model Fused with Staying Time for Personalized Recommendation”, Journal on Communications, vol. 35, no. 9, **(2014)**, pp. 112-121.
[30] W.-J. Song, Q. Guo and J.-G. Liu, “Effect of the Time Window on the Personalized Recommendation Algorithm”, Complex Systems and Complexity Science, vol. 12, no. 1, **(2015)**, pp. 28-32.
[31] H. Yu and J.-H. Li, “Algorithm to Solve the Cold-Start Problem in New Item Recommendations”, Journal of Software, vol. 26, no. 6, **(2015)**, pp. 1395-1408.
[32] P.-H. Chen, J. He and Q. Liang, “A Personalized Recommendation Method Integrating LBS and Social Network Tag”, Computer Applications and Software, vol. 32, no. 6, **(2015)**, pp. 83-86.
[33] B. Liu, H. Xiong, S. Papadimitriou, Y.-J. Fu and Z.-J. Yao, “A General Geographical Probabilistic Factor Model for Point of Interest Recommendation”, IEEE Transactions on Knowledge & Data Engineering, vol. 27, no. 5, **(2015)**, pp. 1167-1179.
[34] B. L. Mark and Y. Ephraim, “An EM Algorithm for Continuous-Time Bivariate Markov Chains”, Computational Statistics & Data Analysis, vol. 57, no. 1, **(2013)**, pp. 504-517.
[35] T. Wang, H. Krim and Y. Viniotis, “A Generalized Markov Graph Model: Application to Social Network Analysis”, IEEE Journal of Selected Topics in Signal Processing, vol. 7, no. 2, **(2013)**, pp. 318-332.
[36] Y. Chen and J. F. Canny, “Recommending ephemeral items at web scale”, Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, **(2011)**; New York, USA.
[37] Q. Liu, E.-H. Chen, H. Xiong, H. Chris, Q. Ding and J. Chen, “Enhancing Collaborative Filtering by User Interest Expansion via Personalized Ranking”, IEEE Transactions on Systems Man & Cybernetics Part B Cybernetics A Publication of the IEEE Systems Man & Cybernetics Society, vol. 42, no. 1, **(2012)**, pp. 218-233.
[38] H. Feng and X.-M. Qian, “Recommend Social Network Users Favorite Brands”, Advances in Multimedia Information Processing – PCM, **(2013)**; Nanjing, China.
[39] L. Li, D.-D. Wang, S.-Z. Zhu and T. Li, “Personalized News Recommendation: a Review and an Experimental Investigation”, Journal of Computer Science &Technology, vol. 26, no. 5, **(2011)**, pp. 754-766.
[40] J. Li, X.-M. Qian, Y.-Y. Tang, L.-J. Yang and T. Mei, “GPS Estimation for Places of Interest from Social Users’ Uploaded Photos”, Multimedia IEEE Transactions, vol. 15, no. 8, **(2014)**, pp. 2058–2071.
[41] F.-K. Liu and J. L. Hong, “Use of Social Network Information to Enhance Collaborative Filtering Performance”, Expert Systems with Applications, vol. 37, no. 7, **(2010)**, pp. 4772–4778.
[42] V. Agarwal and K. K. Bharadwaj, “A Collaborative Filtering Framework for Friends Recommendation in Social Networks Based on Interaction intensity and Adaptive User similarity”, Social Network Analysis & Mining, vol. 3, no. 3, **(2013)**, pp. 359-379.
[43] N. Manouselis, R. Vuorikari and A. F. Van, “Collaborative Recommendation of E-Learning Resources: An Experimental Investigation”, Journal of Computer Assisted Learning, vol. 26, no. 4, **(2010)**, pp. 227-242.
[44] Y. Zheng and X. Xie, “Learning Travel Recommendations from User-Generated GPS Traces”, Acm Transactions on Intelligent Systems & Technology, vol. 2, no. 1, **(2011)**, pp. 389-396.
[45] L. Yu, L. Liu, X.-F. Li, “A Hybrid Collaborative Filtering Method for Multiple-Interests and Multiple-Content Recommendation in E-Commerce”, Expert Systems with Applications, vol. 28, no. 1, **(2005)**, pp. 67-77.
[46] C. Cechinel, M.-Á. Sicilia, S. Sánchez-Alonso and E. García-Barriocanal, “Evaluating collaborative filtering recommendations inside large learning object repositories”, Information Processing & Management, vol. 49, no. 1, **(2013)**, pp. 34-50.
[47] G. Adomavicius and J.-J. Zhang, “Stability of Recommendation Algorithms”, Acm Transactions on Information Systems, vol. 30, no. 4, **(2010)**, pp. 47-54.
[48] A. B. Barragáns-Martínez, E. Costa-Montenegro, J. C. Burguillo, M. Rey-López, F. A. Mikic-Fonte and A. Peleteiro, “A Hybrid Content-Based and Item-Based Collaborative Filtering Approach to Recommend TV Programs Enhanced with Singular Value Decomposition”, Information Sciences, vol. 180, no. 22, **(2010)**, pp. 4290-4311.

Authors



Haiqin Ye, She received the B.E and M.E degrees from ZhengZhou University, Zhengzhou, China, in 2004 and 2007, respectively. Now she is a teacher of School of Computer Science and Technonogy, Zhoukou Normal University, Zhoukou city of

Henan province, China. Her current research interests are personalized recommendation and big data.

Huan Li, She received the Engineering Master degree from HuaZhong University of Science and Technology, Wuhan, China, in 2011. Now she is a teacher of School of Computer Science and Technology, Zhoukou Normal University, Zhoukou city of Henan province, China. Her current research interests are computer application.

Ailing Zhang, She received the B.E and M.E degrees from ZhengZhou University, Zhengzhou, China, in 2004 and 2007, respectively. Her current research interests are cloud computing and big data.