Personalizing Web Page Recommendation via Collaborative Filtering and Topic-Aware Markov Model

Qingyan Yang [1], Ju Fan [2], Jianyong Wang [3], Lizhu Zhou [4]

Tsinghua National Laboratory for Information Science and Technology,
Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China
{ [1]yqy08, [2]fan-j07}@mails.tsinghua.edu.cn { [3]jianyong, [4]dcszlz}@tsinghua.edu.cn

***Abstract*—Web-page recommendation is to predict the next request of pages that Web users are potentially interested in when surfing the Web. This technique can guide Web users to find more useful pages without asking for them explicitly and has attracted much attention in the community of Web mining. However, few studies on Web page recommendation consider personalization, which is an indispensable feature to meet various preferences of users. In this paper, we propose a personalized Web page recommendation model called PIGEON (abbr. for PersonalIzed web paGe rEcommendatiON) via collaborative filtering and a topic-aware Markov model. We propose a graph-based iteration algorithm to discover users' interested topics, based on which user similarities are measured. To recommend topically coherent pages, we propose a topic-aware Markov model to learn users' navigation patterns which capture both temporal and topical relevance of pages. A thorough experimental evaluation conducted on a large real dataset demonstrates PIGEON's effectiveness and efficiency.**

***Keywords*-Web Page Clustering; Personalized Recommendation; Collaborative Filtering; Markov model**

I. Introduction

With the rapid growth of the Web, it becomes more and more difficult for Web users to find useful information. In particular, a Web user often wanders aimless on the Web without visiting pages of his/her interests, or spends a long time to find the expected information. Web page recommendation is thus proposed to address this problem. It aims to understand the users' behaviors, and guide users to visit pages of their interests at a specific time.

An essential task of Web page recommendation is to understand users' navigation behaviors from their Web usage data, and devise a model to predict what pages the users are more likely to visit at the next step. To accomplish this task, variants of the Markov model have been widely employed [1,2]. These methods firstly extract *sessions*. Then they partition a session into several sub-sequences of pages, each of which is considered as a *state* in the Markov model. In addition, they calculate the conditional probabilities of the pages given one identified state. When the sequence of pages the user is visiting matches a state, the pages with the highest conditional probabilities are recommended.

However, existing Web page recommendation methods have the following limitations. Firstly, they cannot recommend *personalized* pages to meet preferences of different users. All users receive the same results if they are visiting the same sequence of pages. However, users are sometimes so different from each other that pages attracting attentions of one user may be bothersome to another. Therefore, it could be very useful to offer a personalized recommendation to satisfy users' tastes. Secondly, existing approaches cannot recommend Web pages related to users' browsing *topics*, since they only concern the temporal relation in state determination and neglect the topic information. Since a user generally focuses on a single topic for a while when surfing the Web, recommending topically coherent Web pages would improve the user experience.

Based on these observation, we propose a novel personalized Web page recommendation model called PIGEON (abbr. for PersonalIzed web paGe rEcommendatiON) to offer topically coherent Web pages satisfying different users' tastes. We employ a *collaborative filtering* framework for personalization. In this framework, we devise a novel clustering algorithm only based on URLs to detect the interested *topics* of each user from browsing logs, and represent the user profile as a set of topics. Then we measure user similarities based on the detected topics. In addition, we propose a topic-aware Markov model to recommend Web pages which are coherent with users' present missions. Compared with existing works, PIGEON not only recommends personalized pages satisfying tastes of different users, but also benefits from the "wisdoms of crowds" for better recommendations by considering user similarities in the ranking mechanism. Moreover, PIGEON improves the effectiveness of traditional Markov-model based methods by incorporating topically relevant pages as states, and thus better captures users' behaviors.

Our main contributions are summarized as follows.

- To the best of our knowledge, we are the first to propose a personalized Web page recommendation model.
- We propose a topic-aware Markov model to recommend Web pages coherent with users' current missions. The experimental results show that our method achieves better recommendation effectiveness.
- We employ a collaborative filtering framework for personalization. We devise a bipartite graph based method to calculate user similarities, which are integrated into

1550-4786/10 $26.00 © 2010 IEEE
DOI 10.1109/ICDM.2010.28

Table I
A SAMPLE BROWSING LOG.

timestamp	user id	IP	target	source
(09:44:44)	(0e0c...)	(211.90.-.-)	(http://datamining.it.uts.edu.au/icdm10/)	()
(09:44:58)	(0e0c...)	(211.90.-.-)	(http://datamining.it.uts.edu.au/icdm10/index.php/call-for)	(http://datamining.it.uts.edu.au/icdm10/)
(10:14:29)	(0e0c...)	(211.90.-.-)	(http://datamining.it.uts.edu.au/icdm10/index.php/topic-of-interest)	(http://datamining.it.uts.edu.au/icdm10/)

our ranking algorithm to get better recommendations.

The remainder of the paper is organized as follows. The related works are reviewed in Section II. Section III presents the problem formulation as well as some preliminaries. In Section IV, we introduce our topic-aware Markov model for modeling users' navigation behaviors. We discuss the personalized Web page recommendation in Section V and conduct experiments in Section VI. Finally, we conclude the paper in Section VII.

II. RELATED WORK

Learning users' navigation behaviors for Web page recommendation has attracted much attention in the community of data mining recently. Markov model and its variants have been proposed to model users' behaviors in many existing studies [1,2,3,4]. An essential task is to determine the *states*. Most approaches consider states as continuous Web pages within a given time window, and different lengths of states are combined to improve both predictive effectiveness and time efficiency [1,2,3]. Meanwhile, discontinuous pages are also allowed in [4]. Compared with existing approaches, we propose a novel model to detect *topically coherent* states, which capture users' interested topics, and thus can help users find more expected information.

Topic detection in our method is based on Web page clustering, which has been studied by some previous work. In [5], all pages are clustered into missions based on their content. In addition, many graph-based algorithms are devised for Web page clustering [6] and session clustering [7]. To the best of our knowledge, our method is the first one to cluster Web pages solely based on URLs rather than page content and structures, thus resulting in a simple yet efficient topic representation by a novel feature expansion algorithm.

Recently, *personalization* has been proposed to improve performance of recommendation and has attracted a lot of research interests. Collaborative filtering is the key technique in personalization [8]. It aims to leverage user similarities to predict the preferences of items to users. [9] gives a survey on algorithms for collaborative filtering. However, this indispensable feature is rarely studied in Web page recommendation. In this paper, we introduce personalization into Web page recommendation by exploring users' profiles from browsing logs and measuring user similarities. Therefore, we can provide pages satisfying different users' tastes.

III. PROBLEM FORMULATION

We first introduce some preliminaries in our proposed model and then give a formal statement of the personalized Web page recommendation problem.

A. Preliminaries

Browsing log. The raw data is a browsing log L, recording information about the surfing histories of the Web users. The records for a specific user u are denoted as L_u. Table I shows a sample browsing log. Each record has a target page p_j visited currently by the user and a source page p_i, from which the user "jumps" to p_j. The *jump relation* between p_i and p_j is denoted by $p_i \rightarrow p_j$. The source page could be empty since the user may directly access the target one. The distinct pages in L is denoted by P.

Navigation graph. Based on the jump relations between the pages, we construct a navigation graph $G_u = (V, E, \omega)$ for the user u from his/her browsing log L_u. V $(\subseteq P)$ contains all pages he/she visited, and E $(\subseteq V \times V)$ is the set of directed edges. A directed edge points to p_j from p_i if there is a jump relation $p_i \rightarrow p_j$. ω represents the edge weight. $\omega(p_i, p_j) = n$ if $p_i \rightarrow p_j$ appears n times, or 0 if no such jump relation exists.

B. Problem Formulation

Given N users $U = \{u_1, u_2, \cdots, u_N\}$ and their browsing log L, and for an active user $a(\in U)$ with the most recently visited $\ell(\geq 1)$ pages $\Phi = \langle p_1, p_2, \cdots, p_\ell \rangle$ (we call it a *prefix*), Web page recommendation is to offer k pages that he/she might be interested in. Our goal is to recommend Web pages which not only satisfy different user tastes, but also are topically coherent with users' present missions.

To provide these features, we propose an effective model PIGEON, whose main components are shown in Figure 1. In the offline part, we learn the behaviors of Web users respectively. We discover his/her interested topics on top of the navigation graph. Each topic is represented by an ellipse in *Topic Discovery* component. We further measure the similarities between users and build a topic-aware Markov model. Personalized recommender is the online part. Given the prefix, PIGEON recommends top-k Web pages to the active user according to his/her behavior model and user similarities. In the remaining part of this paper, we will delve into each component in details.

IV. TOPIC-AWARE MARKOV MODEL

In this section, we present a topic-aware Markov model which captures both temporal and topical relevance of Web pages. The key challenge is to *discover* users' interested topics from browsing logs and exploit the discovered topics to model users' navigation behavior. To address this challenge, we propose a graph based iteration algorithm to discover topics for each user in Section IV-A, and discuss the topic-aware Markov model in Section IV-B.

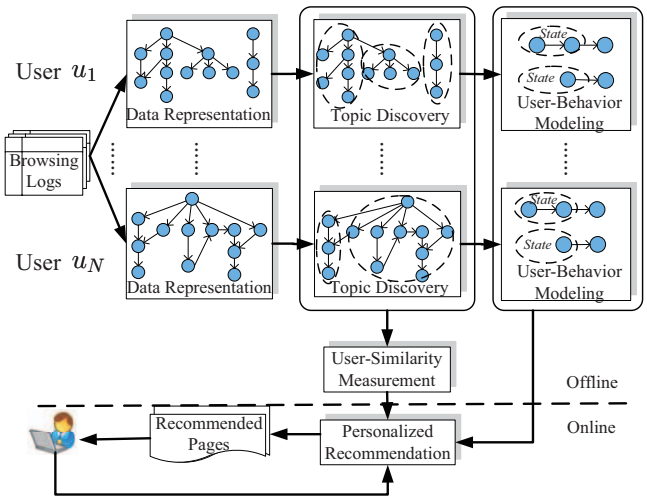


Figure 1. Overview of the PIGEON Model

A. Topic Discovery

The basic idea of topic discovery is to partition Web pages into several clusters, each of which contains topically relevant Web pages. As mentioned in Section III, we only have the URLs of Web pages in the browsing logs. Compared with page content or linking structure of pages, the URLs have limited topic clues to describe the pages and bring challenges to clustering.

Manifold-based keyword propagation [10] for semi-supervised learning is based on the consistency assumption that points in neighbor or on the same structure (typically a cluster or a manifold) likely have the same class label. Similarly, we assume that two pages are likely to be consistent with each other in terms of topics if they are involved in a *jump relation*. Under this assumption, the topic clues of pages in jump relations could be mutually complemented. To this end, we first extract features from URLs, and an iteration algorithm inspired by [10] is then exploited to expand the primitive features of the URLs.

We first develop some rules (e.g., stop-words elimination, GET-parameter parsing, etc.) to extract keywords from the URLs as their primitive features. Taken the URL *http://dblp.uni-trier.de/db/index.html* as an example, its features include *dblp*, *uni*, *trier*, *de* and *db*.

Then the URLs' features are expanded by keyword propagation iteratively on the navigation graph G_u as follows. All the keywords extracted from the pages V constitute a dictionary $\mathcal{D} = \{d_1, d_2, \cdots, d_{|\mathcal{D}|}\}$.

Firstly, a page-keyword matrix $F_{|V|\times|\mathcal{D}|}$ is initialized, where F_{ik} denotes the normalized term frequency tf of the keyword d_k in the page $p_i (\in V)$, i.e.,

$$F_{ik} = \frac{tf(p_i, d_k)}{\sum_{\ell=1}^{|\mathcal{D}|} tf(p_i, d_\ell)}$$

Secondly, we construct an affinity matrix $W_{|V|\times|V|}$. Intuitively, if two pages have many keywords in common and are often involved in jump relations, they are likely relevant to the same topic. Hence, the normalized matrix W is defined by

$$W_{ij} = \begin{cases} \frac{J(p_i,p_j)\cdot\omega(p_i,p_j)}{\sum_{\ell=1}^{|V|} J(p_i,p_\ell)\cdot\omega(p_i,p_\ell)} & : \quad \omega(p_i,p_j) \neq 0 \\ 0 & : \quad otherwise \end{cases}$$

where $J(p_i, p_j)$ is the Jaccard similarity between p_i and p_j.

Thirdly, we iterate $F(t+1) = \alpha \cdot W \cdot F(t) + (1-\alpha) \cdot F(0), \alpha \in (0,1)$ until convergence. The limit of the sequence $\{F(t)\}$ is denoted as F^*, each element of which reveals the keyword's relevance to the corresponding page.

With the expanded features, each Web page can be represented by a keyword vector. To accomplish Web page clustering, the similarity between every pair of pages is measured by the cosine similarity on top of their feature vectors. As the pages usually contain so many keywords after feature expansion, we sort the keywords of each page by their weights in descending order and select the top-k ones as its final features to reduce the complexity of Web page representation.

Finally, we exploit a relatively new clustering algorithm *Affinity Propagation* [11] with "preference" set to be -1. The result clusters make up the profile of the Web user. More specifically, each user could be described as a set of topics $T_u = \{t_u^1, t_u^2, \cdots, t_u^{|T_u|}\}$, where $t_u^i (1 \leq i \leq |T_u|)$ represents one topic, that is, the features of pages in the ith result cluster. We assign each feature by an average weight over all pages in the cluster to express its relevance to the corresponding topic, denoted as $rel(t_u^i, d_j), d_j \in \mathcal{D}$.

B. Topic-Aware Markov Model

We discuss the topic-aware Markov model based on the discovered topics in this section. As mentioned in Section II, Markov model has been widely applied to learn users' navigation behaviors for predicting the next step while surfing the Web. We first briefly introduce the traditional kth-order Markov model, and then discuss how to incorporate the topic information into it.

Markov-model based methods include the following steps.

(1) Session partition. A *session* is a sequence of pages ordered by access time and reflects user missions within a time interval t_θ (e.g., 30 minutes). Formally, a session with length m can be described as $\langle p_1, p_2, \cdots, p_m \rangle$, where the time difference of p_m and p_1 does not exceed t_θ. After being sorted first by user ids and then by timestamps, the records in the browsing log L are partitioned into sessions.

(2) State determination. Each k-gram of a session is called a *state*, and k denotes the model's order. Formally, given a session $\langle p_1, p_2, \cdots, p_m \rangle$, the jth state of a kth-order Markov model is $\mathcal{S}_j^k = \langle p_j, p_{j+1}, \cdots, p_{j+k-1} \rangle, 1 \leq j \leq m-k+1$.

(3) Conditional probability estimation. Given a state $\mathcal{S}_j^k$, the probability that the page p_i is requested next by the active user is estimated by the ratio of the frequency

of $\mathcal{S}_j^k$ followed by the page p_i to the frequency of $\mathcal{S}_j^k$, i.e., $P(p_i|\mathcal{S}_j^k) = frequency(\mathcal{S}_j^k, p_i)/frequency(\mathcal{S}_j^k)$.

Traditional approaches simply consider the temporal information of Web pages. As a result, the pages not immediately visited after the states will not be recommended, even though the user is likely interested in them due to their topical relevance to his/her present mission.

In this paper we propose a topic-aware Markov model to learn users' navigation behaviors. Two categories of states are to be determined, *temporal states* and *topical states*. Each k-gram of the sessions is called a *temporal state*. They are necessary, representing a not unusual surfing process where Web users have multiple missions concurrently. A *topical state* is a topically coherent sequence of Web pages, which may not be consecutively browsed. It is worthwhile to note that a state could be both temporal and topical, if the pages accessed consecutively are topical relevant at the same time. To obtain the topical states is tricky. Based on the result clusters, we partition a page sequence into subsequences. The pages in each subsequence ordered by access time belong to the same cluster. As a result, the topical states are the very k-grams in these subsequences. The model we build for each user u is denoted as $\mathcal{S}_u$.

V. Personalizing Web Page Recommendation

We exploit a collaborative filtering framework for personalized Web page recommendation. Collaborative filtering is a popular technique for personalization in many applications on the Web. We focus on two essential tasks in this section. We firstly introduce the methods of measuring user similarities in Section V-A. In Section V-B, we discuss the methods of recommending favorite pages of the similar users based on the learned topic-aware Markov model.

A. User Similarity Measure

We assume that two users are similar if they have visited many Web pages in common, or pages about relevant topics. Therefore, with each user represented by the topics discovered in Section IV-A, we take the similarity between two sets of topics as the similarity between the two corresponding users. Computing the overlap between two sets is straightforward but cannot capture the relationship within similar topics. In contrast, we measure the similarity between each pair of topics and adopt a maximum weight bipartite matching algorithm to derive the similarities between Web users.

In detail, a bipartite graph $\mathcal{BG}_{i,j} = \{\mathcal{V}, \mathcal{E}, \boldsymbol{b}\}$ is constructed for users u_i and u_j, where $\mathcal{V} = T_{u_i} \cup T_{u_j}$, and $T_{u_i} = \{t_{u_i}^1, t_{u_i}^2, \cdots, t_{u_i}^{n_i}\}$ is the set of topics attractive to the user u_i, similarly for T_{u_j} to the user u_j. The edge set is $\mathcal{E} = T_{u_i} \times T_{u_j}$, and $\boldsymbol{b} : E \mapsto \mathbb{R}$ represents the edge weights, which indicate the similarities among topics. Formally, for an edge $e(t_{u_i}^p, t_{u_j}^q), 1 \le p \le n_i, 1 \le q \le n_j$ connecting $t_{u_i}^p$ and $t_{u_j}^q$, the weight is $\boldsymbol{b}(e) = \sum_{d \in t_{u_i}^p \cap t_{u_j}^q} rel(t_{u_i}^p, d) \cdot rel(t_{u_j}^q, d)$. A matching $\mathcal{M}_{i,j}$ of $\mathcal{BG}_{i,j}$ is a subset of $\mathcal{E}$ where any two edges in $\mathcal{M}_{i,j}$ do not share a common topic. The weight of a matching is the sum of its edge weights, i.e., $weight(\mathcal{M}_{i,j}) = \sum_{e \in \mathcal{M}_{i,j}} \boldsymbol{b}(e)$. The number of edges in $\mathcal{M}_{i,j}$ is denoted by $|\mathcal{M}_{i,j}|$. The maximum matching $\mathcal{M}^*$ is the matching having the maximum weight, i.e., $\mathcal{M}^* = \text{argmax}_{\mathcal{M}_{i,j}} weight(\mathcal{M}_{i,j})$. The similarity between the users u_i and u_j is calculated by $sim(u_i, u_j) = 1/|\mathcal{M}^*| \times weight(\mathcal{M}^*)$. An example of measuring the similarity of two users u_1 and u_2 is shown in Figure 2. The edges in the maximum matching is represented by solid style. Hence, the similarity between u_1 and u_2 is $(0.9 + 0.8 + 1.0)/3 = 0.9$.

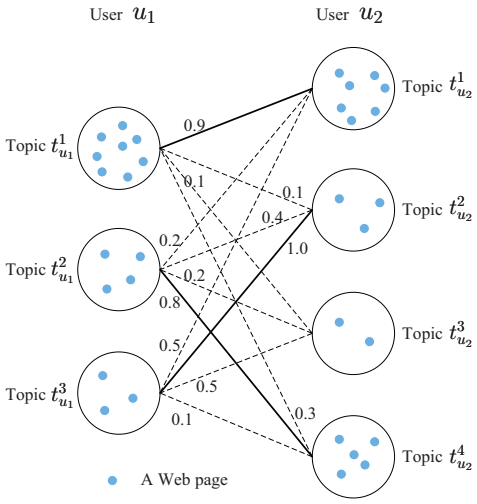


Figure 2. An Example of Measuring User Similarity

B. Personalized Recommendation

In this section, we discuss the problem of personalizing Web page recommendation, which can be formally described as follows. Given a prefix Φ for an active user a, the system is to recommend k pages probably attractive to the active user. Three steps are taken to solve the problem.

First, for each Web user $u_i, i = 1, 2, \cdots, N$, we try to match the prefix Φ with the states of his/her navigation behavior model. The conditional probabilities of each page p following Φ are estimated as stated before, and p is taken as a candidate. Second, the score of each candidate is calculated based on the idea of collaborative filtering. Specifically, for each candidate p, we combine its probabilities in each user's model with user similarities as follows.

$$\kappa(p|a, \Phi) = \sum_{u_i \in U} sim(u_i, a) \cdot \frac{frequency_{u_i}(\Phi, p)}{frequency_{u_i}(\Phi)} \quad (1)$$

Finally, we sort the candidates by their scores in descending order and recommend top-k pages to the active user a. Note that the candidate size may be smaller than k. If more recommendations are requested in this case, Φ will be augmented by one candidate page with the highest score to fetch more results. Details are described in Algorithm 1.

The set of candidates $\mathcal{R}$ is empty initially (Line 1). Based on each user's behavior model $\mathcal{S}_u$, k pages at most taken as candidates are stored in Λ (Lines 2-14), each element of which is a page-score pair (p, κ). For simplicity, we can ignore Lines 4-11 for a moment. Given the prefix Φ, the candidates are obtained using Formula 1 (Line 3). All candidates are added to $\mathcal{R}$, updating the scores if necessary (Lines 12-14). In case that the number of candidates are less than k, we would augment Φ with the candidate having the highest score (Lines 6-8). Further recommending based on the augmented prefix is performed (Line 9). Note that we multiply the scores of each new candidate with a penalty factor σ (Lines 10-11), since they are farther from the prefix.

Algorithm 1: `personalizedRec(`$\mathcal{S}$`, `a`, `Φ`, `k`)`

input : $\mathcal{S}$ The users' behavior models;
a The active user;
Φ The prefix visited by the user a
k The maximum number of recommendations.

1 $\mathcal{R} \leftarrow \emptyset$;
2 **foreach** $\mathcal{S}_u \in \mathcal{S}$ **do**
3 Get candidates $\Lambda : \{(p, \kappa)\}$ by Formula 1;
4 $\tilde{\Lambda} \leftarrow \Lambda$;
5 **while** $|\Lambda| < k$ *and* $\tilde{\Lambda} \neq \emptyset$ **do**
6 $\tilde{\Phi} \leftarrow \Phi$;
7 Poll the first page r in $\tilde{\Lambda}$;
8 $\tilde{\Phi} \leftarrow \Phi \cup \{r\}$;
9 **foreach** *candidate p, score* κ *w.r.t* $\tilde{\Phi}$ **do**
10 **if** $p \notin \Lambda$ **then** $\Lambda \leftarrow \Lambda \cup (p, \sigma \cdot \kappa)$;
11 **if** $p \notin \tilde{\Lambda}$ **then** Add $(p, \sigma \cdot \kappa)$ to $\tilde{\Lambda}$; ;
12 **foreach** $(p, \kappa) \in \Lambda$ **do**
13 **if** $p \in \mathcal{R}$ **then** Add the score of p by κ;
14 **else** $\mathcal{R} \leftarrow \mathcal{R} \cup (p, \kappa)$;
15 Recommend k pages with highest scores in $\mathcal{R}$;

VI. Experimental Evaluation

Data set. In this study we use a real data set originated from the browser of Sohu Company (www.sohu.com). It contains 1,402,371 records of 375 users from April 1, 2008 to May 4, 2008. We split the data set into a training set with the records in April, and a test set with the rest. Since the prefix's length could be varied, we prepared two prefix sets, *Test-1* and *Test-2*, from the test set. The target pages in the test set are extracted as an ordered sequence. Test-1 takes each page as a prefix, while Test-2 takes each bi-gram, if the two pages are within the pre-specified time window. Otherwise, the prefix only gets the first page. There are 26820 prefixes in Test-1, and 12210 ones in Test-2.

Experiment settings. The method for personalization based on 1st-order topic-aware Markov model is denoted by *1st-Order* PIGEON, while *All-2nd-Order* PIGEON represents that based on the combination of 1st-order and 2nd-order model. For comparison, we implemented the traditional Markov-based model as *Baseline* in 1st-order and all-2nd-order. It models users' behavior following the steps stated in Section IV-B with t_θ set as 10 minutes. In addition, we make clear other parameters in our model. The damp factor α in keyword propagation is 0.85. The detected topics are represented by top-10 keywords. The penalty factor σ is 0.9 in Algorithm 1, and we recommend 5 Web pages. All experiments were performed on a Pentium(R) D CPU, 2.8GHz computer with a 960MB main memory running Microsoft Windows XP Professional SP2.

Evaluation metrics. As the previous study in [8] do, we evaluate the effectiveness of our model by precision and recall. We denote the recommended result as $\mathcal{T}$ (i.e., Testing result), and $\mathcal{B}$ (i.e., Benchmark) means the Web pages actually visited by the active user within 30 minutes after the prefix. Therefore, the precision is calculated by $|\mathcal{T} \cap \mathcal{B}|/|\mathcal{T}|$ and the recall by $|\mathcal{T} \cap \mathcal{B}|/|\mathcal{B}|$.

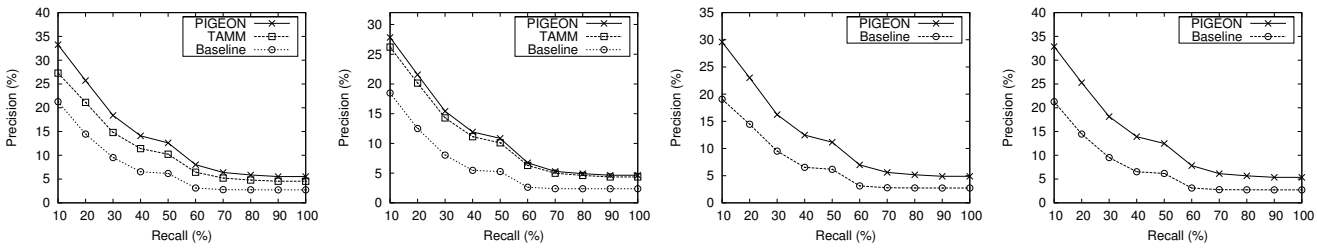
Another metric is model coverage. It is the the ratio of the prefixes matched by the states of the behavior model to all prefixes used to ask for recommendation.

A. Evaluation of Effectiveness

One main difference between PIGEON and Baseline is that PIGEON is able to derive topical states additionally. To illustrate the influence of the topical states on recommendation, we first consider the user similarities the same. Such a framework actually degenerate to the topic-aware Markov model (TAMM for short). We make recommendations based on 1st-Order and All-2nd-Order TAMM using the prefixes in Test-1 and Test-2 respectively, compared with 1st-Order and All-2nd-Order Baseline. The results of the precision and recall curve are shown in Figure 3(a) and Figure 3(b). They indicate that topical states could significantly improve the recommendation performance both in precision and recall. For example, at the point with recall 10%, the precision of 1st-Order Baseline is 21.2%, while that of 1st-Order TAMM is 27.2%, increased by 28.6% relatively.

To personalize recommendations, we further integrated user similarities into TAMM. The results are referred as PIGEON in Figures 3(a) and Figure 3(b). It is shown that the precision and recall are further improved. In addition, two other experiments were conducted to evaluate PIGEON using different length of prefixes. Figure 3(d) shows the results of All-2nd-Order PIGEON and Baseline tested on Test-1, and Figure 3(c) shows that of the 1st-Order models on Test-2. It is proved that our model performs much better than the Baseline. As TAMM performs similarly with the former, its results are omitted.

The coverage comparison is illustrated in Figure 4. Test-1 (Test-2 respectively) provides prefixes to ask 1st-Order (All-2nd-Order respectively) models for recommendations. It also shows the significant advantage of PIGEON over Baseline.



(a) Precision Recall Curves of 1st-Order Model on Test-1 (b) Precision Recall Curves of All-2nd-Order Model on Test-2 (c) Precision Recall Curves of 1st-Order Model on Test-2 (d) Precision Recall Curves of All-2nd-Order Model on Test-1

Figure 3. Effectiveness Evaluation of PIGEON Compared with the Baseline and the Topic-Aware Markov Model (TAMM for short).

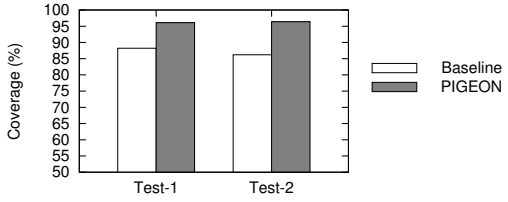


Figure 4. Coverage Comparison of PIGEON and the Baseline

Table II
NUMBER OF STATES IN DIFFERENT MODELS.

	PIGEON	Baseline	Increase
1st-Order	614,258	614,258	0
All-2nd-Order	1,851,930	1,538,327	20.4%

Table III
AVERAGE RESPONSE TIME TO ONE PREFIX.

	Test-1 (ms.)	Test-2 (ms.)
1st-Order PIGEON	35.9	43.8
All-2nd-Order PIGEON	38.9	56.0

B. Model Complexity and Recommendation Efficiency

Table II compares the number of states of our model and Baseline in different orders. Compared with Baseline, the number of states of All-2nd-Order PIGEON is increased by 20.4% relatively. Note that the state spaces of 1st-Order models are the same, since the states are actually the unique pages in the browsing log. Efficiency is also important for online service. We count the time spent on producing recommendations, averaged by the number of the prefixes. The average time summarized in Table III indicates that our model is efficient with a response time in milliseconds.

VII. CONCLUSION

In this paper, we have studied a new paradigm of personalized Web page recommendation to predict the pages that Web users are interested in without explicitly asking for them. Taking the user similarities into account, we personalized Web page recommendation to meet different preferences of Web users. We proposed an effective graph-based clustering algorithm to automatically discover the interested topics of Web users. Both the features of the URLs and the jump relations between Web pages are exploited for user profile learning. We devised a novel model for learning the navigation patterns, which contribute to the topically coherent recommendations. We have conducted extensive experimental study of our proposed model on a real data set. The results prove the effectiveness and efficiency of the new model to personalize Web page recommendation.

VIII. ACKNOWLEDGEMENT

This work was supported in part by National Natural Science Foundation of China under grant No. 60833003, the National Basic Research Program of China (973 Program) under Grant No. 2011CB302206, and the Program for New Century Excellent Talents in University under Grant No. NCET-07-0491, State Education Ministry of China.

REFERENCES

[1] M. Deshpande and G. Karypis, "Selective markov models for predicting web page accesses," *ACM Trans. Internet Techn.*, vol. 4, no. 2, pp. 163–184, 2004.

[2] J. Borges and M. Levene, "Evaluating variable-length markov chain models for analysis of user web navigation sessions," *IEEE Trans. Knowl. Data Eng.*, vol. 19, no. 4, pp. 441–452, 2007.

[3] M. Eirinaki, M. Vazirgiannis, and D. Kapogiannis, "Web path recommendations based on page ranking and markov models," in *WIDM*, 2005, pp. 2–9.

[4] G. Bonnin, A. Brun, and A. Boyer, "A low-order markov model integrating long-distance histories for collaborative recommender systems," in *IUI '09* New York, NY, USA: ACM, 2009, pp. 57–66.

[5] J. Li and O. R. Zaïane, "Combining usage, content, and structure data to improve web site recommendation," in *EC-Web*, 2004, pp. 305–315.

[6] R. Baraglia and F. Silvestri, "An online recommender system for large web sites," in *Web Intelligence*, 2004, pp. 199–205.

[7] S. Gündüz and M. T. Özsu, "A web page prediction model based on click-stream tree representation of user behavior," in *KDD*, 2003, pp. 535–540.

[8] A. Das, M. Datar, A. Garg, and S. Rajaram, "Google news personalization: scalable online collaborative filtering," in *WWW*, 2007, pp. 271–280.

[9] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 6, pp. 734–749, 2005.

[10] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," in *NIPS*, 2003.

[11] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, pp. 972–976, 2007.