Postprint: Simulated Implementation and Evaluation of Recommender Systems in Knowledge Retrieval Scenarios

Authors: Cheng Xiufeng, Zhang Ziming, Meng Yaqi, Fan Xiaoying, Yang Jinqing

Date: 2023-07-26T00:00:00+00:00

Abstract

[Purpose/Significance] Information overload has long been a primary challenge faced by knowledge workers in the processes of collecting, processing, and creating knowledge. One consequence of this challenge is the difficulty in recalling the content details and specific locations of previously used documents, a problem that recommender systems can help alleviate. By investigating and comparing the strengths and weaknesses of different recommender systems for this task, knowledge workers can be better assisted in completing recall tasks.

[Methods/Process] Based on relevant theories, we simulated, implemented, and tested four different types of recommendation processes in the same scenario (knowledge retrieval), including Content-Based Recommendation (CBR), Collaborative Filtering Recommendation (CFR), Inference Network-Based Recommendation (INR), and Context-Aware Recommendation (CAS). The recommendation effectiveness was compared according to several established metrics (accuracy, context relevance, predictability, diversity).

[Results/Conclusions] The results demonstrate that the aforementioned recommender systems each possess distinct advantages in helping users recall and retrieve documents, while the context-aware recommender system exhibits superior performance in terms of context relevance and predicting user behavior.

Full Text

Simulation and Evaluation of Recommendation Systems for Knowledge Re-finding Scenarios

Cheng Xiufeng, Zhang Ziming, Meng Yaqi, Fan Xiaoying, Yang Jinqing

School of Information Management, Central China Normal University, Wuhan 430079

Abstract: Information overload has long been considered a major barrier for knowledge workers in the processes of gathering, processing, and producing information. One consequence is the difficulty in recalling previously used documents, while recommendation systems can alleviate this difficulty. Comparing recommendation efficiencies through representative recommendation mechanisms may assist knowledge workers in accomplishing knowledge re-finding tasks. Based on associated recommendation system theories, this paper presents a simulation of four different recommendation procedures (CBR, CFR, INR, and CAS) in a unified experimental scenario (knowledge re-finding). Four evaluation criteria (precision, context-relevance, action-prediction, diversity) are used to evaluate and compare the efficiency of corresponding recommendation systems. Results show that each recommendation procedure has its own advantages in knowledge re-finding from different perspectives, and CAS has advantages in both context-relevance and action-prediction.

Keywords: information overload; knowledge re-finding; recommendation system; context-awareness

1 Introduction

With the dramatic expansion of the Internet and exponential growth in information volume, the information service domain has consistently faced the dilemma of "information overload." This dilemma manifests primarily when individuals or systems receive information exceeding their processing capacity, leading to system failures [1]. General information retrieval systems developed from retrieval theory can "pull" information from massive datasets but cannot mine effective information based on users' current or historical behaviors, nor can they perceive users' current context for reasonable recommendations. In contrast, information recommendation schemes built upon user demand prediction can better achieve "on-demand customized services" and partially solve the information overload problem [2].

Currently, recommendation systems combining the following categories of models (algorithm sets) have achieved relatively mature development: First, filtering-based models, which primarily target "user preferences for resource items" or the "user-item" binary relationship for recommendations. These models have been commercially applied in B2C websites, advertising, news, and entertainment with obvious effects [3-5]. Second, annotation-based models, which perform semantic optimization and tag description on both ends of the "user-resource" relationship, enabling machines to recognize semantic information embedded in user preferences and resources, thereby forming more precise personalized recommendation solutions [6,7]. Third, inference-based models, which use inference rules or machine learning methods for reasoning and decision-making to find satisfactory results in resource networks [8,9]. Fourth, perception-based models, which incorporate contextual information into original models to improve recommendation effectiveness in ubiquitous scenarios [10-11]. However, such context-aware recommendation systems remain in development, with few mature applications [13-14]. Notably, these four categories are not mutually exclusive; hybrid recommendation systems often combine two or more models for better performance.

Many scholars have studied information overload in the context of network information resource acceptance and utilization, but few have focused on knowledge re-finding scenarios in daily work [15]. Knowledge workers' tasks typically include document writing, programming, data analysis, and text reading, with frequent switching between these activities causing them to repeatedly consult previously browsed documents. This often leads to the problem of "knowing it's stored somewhere but forgetting its location." This paper terms this scenario "knowledge re-finding" or "knowledge-assisted memory," and the associated problem "knowledge re-finding difficulty."

Most experienced users struggle to remember content discovered during initial queries, frequently re-finding information [21]. A Yahoo study based on query logs showed that 40% of queries attempted to re-find previously retrieved results [22], while user behavior monitoring experiments revealed even higher proportions (44% [23], 58% [24], 81% [25]). Despite the frequency of such re-finding behavior, the overall success rate remains below 20% [26], making improvement of re-finding success a key research question.

Researchers initially termed this "information re-finding," attempting to solve it through retrieval and recommendation systems. In practice, user-based [27] and content-based [28] recommendations have shown good performance in information re-finding scenarios. With the development of personal knowledge management systems, re-finding has evolved from historical retrieval records to knowledge that has been personally processed and understood, actively contributing to ongoing creative work [29]. Thus, information re-finding has gradually evolved into knowledge re-finding, placing higher demands on existing recommendation systems.

2 Related Work

2.1 Knowledge Re-finding

Since 1992 when Goldberg introduced the first recommendation system Tapestry [30] and 1994 when P. Resnick et al. launched the first automated collaborative filtering system GroupLens [31], recommendation system research has developed for over 20 years, forming a relatively independent research direction intersecting with computer science, information retrieval, behavioral cognition, and management science. Although no universally precise definition exists, several explanations have gained wide acceptance: (1) recommendation systems are essentially information discovery tools [32]; (2) they effectively alleviate information overload [33]; (3) they mine "user-item" binary relationships to establish association algorithms [34].

Among the four categories mentioned, filtering-based systems are most theoretically and practically mature. Numerous scholars have proposed improved models to address cold-start, sparsity, heterogeneity, and scalability issues [35], achieving significant commercial success. Annotation-based systems emerged after 2000 [36], optimizing description mechanisms for either end of the binary relationship through project ontologies, user attribute tagging, or mass classification. Inference-based systems originated from decision support systems. After 2002, increased research in machine learning association rules, topic extraction, and intelligent decision-making [37] led to mature case-based reasoning and multi-Agent engine systems, supported by expert knowledge bases for case recommendation, system integration, and real-time recommendation. Perception-based systems integrate collected contextual information into recommendation results, beneficially affecting decision validity and precision. Current context-aware recommendation research focuses primarily on context fusion [38-40], such as incorporating location information for movie, advertising, and news recommendations.

However, both non-contextual and context-aware recommendation systems have traditionally focused on recommendation objects and algorithmic mechanisms (e.g., product, document, music recommendations) while neglecting specific scenarios like knowledge re-finding in document recommendation. As personalized services become intelligent, optimal recommendation effectiveness requires tailoring solutions to specific scenario elements: information platforms, resource types, user groups, evaluation criteria, and perception capabilities.

3 Experimental Design

To evaluate various recommendation systems' performance in knowledge re-finding scenarios and comprehensively analyze their advantages and disadvantages, this paper selected four representative recommendation systems for simulation. The simulation matched recommendation lists with user needs to obtain measurement data. The basic approach simulated knowledge re-finding by assigning users a specific task (e.g., browsing documents to write a paper) requiring extensive document consultation. After one time period, a similar task was assigned (e.g., creating a presentation) that might require previously consulted documents. Recommendation systems needed to recommend potentially useful documents from the first task during the second task. Effectiveness was measured by comparing the actual document sequence used in the second task (or user-selected important documents) with recommendation results from the first task.

This study employed offline simulation of online environments, isolating recommendation processes for evaluation, offering advantages of high standardization, data accessibility, and precise assessment.

The implementation proceeded as follows: The experiment consisted of two periods, A and B. In period A, users selected one of t topics (t=3 in this experiment) and were provided with offline search document sets D_t = <d_{t1}…d_{tn}>, pre-classified and placed in directories corresponding to search engines (360 , Soso, Sogou, Baidu, Wanfang, CNKI), with each folder containing documents found by that engine. k users (k=20) each selected a topic and used D(t) to write an 800+ word article, producing article collection CA = <c_{A1}…c_{Ak}>. System recorded each user's actual behavior history HA_i during period A as input for the recommendation system. After period A, each participant subjectively selected 15 important documents MA_i.

Four weeks later in period B, users were recalled and given only document set MA_i, requiring them to create a 5-slide formatted presentation using MA_i, producing result collection CB = <c_{B1}…c_{Bk}>. Each user's actual behavior history HB_i was recorded, and after completing period B, each subjectively selected and scored 10 important documents MB_i (1-5). Contextual information for periods A and B was recorded using IE-History-View and Keylogger tools, detailed in Section 4.4. The experimental process is shown in Figure 1 [Figure 1: see original paper].

After simulation, recommendation result set R was obtained based on MA and non-context-aware models, while R' was obtained based on MA, HA and context-aware models. Both were compared with MB for precision and other metrics.

4 Recommendation Models

4.1 Content-Based Recommendation (CBR)

Content-based recommendation systems recommend items similar to those a user prefers, based on content preference. In knowledge re-finding scenarios, documents are treated as items, segmented words as item elements, and browsing duration as user preference. The system analyzes previously browsed document content to form user preference descriptions, pre-filters current work projects to form content descriptions, and compares similarities between users and items for recommendation [41].

The CBR system for knowledge re-finding has two modules: preprocessing and recommendation. The preprocessing module handles item content through segmentation, feature extraction, indexing, and dimensionality reduction to establish mathematical or document descriptions content(i). The recommendation module first uses TF-IDF to calculate element weights, creates user preference descriptions based on history, then uses a utility function R(u,i) = sim(u, content(i)) to represent item importance to user u.

The specific process: First, obtain user browsing history, segmentation, and stop-word removal using IE-History-View and segmentation tools [42]. Then, using the classic TF-IDF model (to maximally distinguish documents), construct target user u_i's preference vector for feature word set T = {t_1, t_2,

…, t_k} as u_i = {w_1,i, w_2,i, … w_k,i, …, w_|T|,i}, where any document can be represented by a feature word weight vector. The normalized importance of feature word t_k for document d_j is:

$$W(t_k, d_j) = TF(t_k, d_j) \times \log\left(\frac{TF(t_k, d_j) \times \log}{\frac{N}{n_k}}\right), \quad (d_j \in D) \tag{1}$$

where TF(t_k, d_j) is term frequency, N is total documents, n_k is documents containing t_k, and T is feature word count. Then, cosine similarity calculates the re-recommendation process:

$$\operatorname{sim}(u_i, d_j) = \frac{\sum_k w_{k,j} \cdot w_{k,j}}{\sqrt{\sum_k w_{k,j}^2} \cdot \sqrt{\sum_k w_{k,j}^2}} \tag{2}$$

where w_{k,j} represents user preference vector weight and document vector weight. Matrixing this (user-document) yields the recommended document set based on browsing history; setting thresholds produces recommendation rankings.

4.2 Collaborative Filtering Recommendation (CFR)

Collaborative filtering differs from CBR by being based on target user preferences versus selection based on other users' preferences. It operates on Goldberg's 1992 hypothesis: "If a group of users has similar preferences for some items, their ratings for other related resources will also be similar" [43]. CFR essentially vectorizes preferences and uses similarity metrics to find k-nearest neighbors (KNN) for target users/items, predicting rankings to form Top-N recommendations.

In knowledge re-finding, this resembles a small team's knowledge sharing: discovering other members' preferences to re-find needed documents. However, two problems arise: (1) Using 0/1 rating matrices from MA selections creates sparsity issues when MA doesn't intersect with actual browsing history HA, or when recommendation set R has low overlap with reference sets HB and MB. (2) Document preferences depend not just on selection but also on actual browsing, average dwell time, clicks, document type, and user knowledge level.

To address these, this paper combines subjective selection with objective records to determine preferences, build KNN, and generate recommendations. The approach: (1) Incorporate browsing dwell time t_u from HA into MA, establish threshold T (>30sec), de-binary MA, and use extended collaborative methods [44] to gradually fill sparse matrices until sparsity factor > 0.05. (2) Construct rating matrices for document sequences MA_i, where user u's rating for document i is given by formula (3).

Document similarity is measured using Pearson correlation coefficient for users who rated both documents i and j:

$$\text{sim}(i,j) = \frac{\sum_{u \in U_{ij}} (R_{ui} - \bar{R}_i)(R_{uj} - \bar{R}_j)}{\sqrt{\sum_{u \in U_{ij}} (R_{ui} - \bar{R}_i)^2} \sqrt{\sum_{u \in U_{ij}} (R_{uj} - \bar{R}_j)^2}} \tag{4}$$

where $\bar{R}_i$ and $\bar{R}_j$ are average ratings for documents i and j.

Document KNN and comprehensive filtering: For each target document i, search document set I_1 = {i'_1, i'_2, ..., i'*n} in the entire space I where i' I_1 and similarity sim(i,i') is sorted descendingly. Filter I_1 to remove documents not rated by target user u, keeping only rated sequence I_2. Then predict recommendation degree P*{u,i''} for document i" using:

$$P_{i'',u} = R_{u,i''}, \quad P_{i'',u} = \frac{\sum \text{sim}(i,i'') \times (R_{u',i''} - \bar{R}_{i''})}{\sum |\text{sim}(i,i'')|} \tag{6}$$

where I_1 is the nearest neighbor set, R_{u',i''} are other users' ratings, and $\bar{R}_{i''}$ is the average rating. The filtering from I_1 to I_2 is crucial as it ensures recommended documents exist in user u's commented/browsed set.

4.3 Inference Network Recommendation (INR)

Bayesian networks, proven efficient approximate inference algorithms over nearly 30 years, have succeeded in information retrieval, AI, expert systems, and pattern recognition [45]. Inference network recommendation (INR) models [46] use Bayesian network structures and conditional probability tables to calculate node probabilities. For knowledge re-finding, the algorithm captures latent dependencies between users and documents through behavior records: build index term occurrence event sequence $\bar{k}_1$, given prior document observation probability and conditional term probability, derive posterior query satisfaction probability.

A Bayesian network BN = <X, A, > is constructed through the inference chain document observation → index term occurrence → query satisfaction, leveraging conditional independence. <X,A> represents a directed acyclic graph (DAG), X denotes node types, A represents potential probabilistic dependencies, and quantifies dependency degrees via conditional distributions. The open-source tool Indri combines inference networks with language modeling, supporting structured queries for effective node estimation [47], making it representative for probabilistic document recommendation.

Indri uses TF-IDF for document-to-term node inference. Figure 2 [Figure 2: see original paper] shows Indri's inference network model, where each node represents a random event with conditional probability tables describing result set probabilities given parent nodes. The system calculates document query probabilities using prior/conditional probabilities for ranking.

In the experiment, each user's MA document sequence was input to Indri to obtain Indri values I, sorted and compared with MA. Results are shown in Table 1 . Removing the bottom 5 documents (MA_i has 15, MB_i has 10) yields overlap between recommendation set R and MB_i for evaluation.

Table 1: INR Recommendation Results Sample

User ID	I (Indri)
1	0.325196252
2	0.041227371
3	0.01912699
4	0.018069445
5	0.011333865
6	0.011095502
7	0.008886901
8	0.007074106
9	0.006728622
10	0.005564768

4.4 Context-Aware Recommendation (CAS)

Context-aware recommendation (CAS) differs from CBR/CFR by using context information HB_i to determine current user context and applying historical information HA_i in recommendation, making results more contextually appropriate. Given knowledge associations, this paper adopts the Spreading Activation (SA) model [48] for context inference, originally from information retrieval for term expansion [49]. If two terms co-occur or have semantic association, and one indexes a document, the other can too. SA builds clear knowledge networks, extracts context-relevant information, and enhances memory functions to reduce information overload [50].

The experiment uses document name keywords as indexing. First, an SA-based hierarchical model is built (Figure 3 [Figure 3: see original paper]) comprising event, context, and document layers with distinct functions.

Table 2: Event Data Sample

Event Data Item	Example
User	Huanlei
Event_{ID}	201734355T2315454
Location	DELL@file:///C:/Users/DELL/Desktop/Experiment/…/I
Index (Topic)	XiaoHong Introduction

Event Data Item	Example
Position/Domain/Click	NULL

The event layer contains discrete user events (browsing, clicks) recorded by IE-History-View, including mouse activities, window titles, URLs. Each opening event forms an Event Block indexed by document topic, providing data for context extraction.

Context information is classified into four basic types: location, time, objects, and activity [20,28]. In knowledge re-finding, context includes time, topic, and entity (current open document). The context layer builds networks via SA, while the document layer uses SA to construct inter-document association networks, ranking candidates by association strength. WordNet is used to query index associations. For example, detecting "cancer" topic activates semantically related "tumor" topic for expanded recommendations.

To prevent over-expansion reducing accuracy, TF-IDF principles constrain SA's context-document association strength: recommendation importance is proportional to context frequency within a document but inversely proportional across documents. Weight calculations between layers are shown in Table 3 .

Table 3: Context Information Weight Calculation

Context Item	Weight Calculation
Data/Time	Timestamp is unique browsing identifier; only one document browsed at a time
Entity	Probability of object appearing in document
Location	Document storage path is generally unique
Topic	Topic expansion weight depends on occurrence probability
ClickCount	Ratio of document clicks to maximum clicks

Since document matching depends mainly on topic, which determines thematic scope, context-aware recommendation focuses on context relevance to predict next document preferences. LDA extracts document topics combined with browsing records to mark features.

5 Evaluation Metrics

Many scholars have set various evaluation standards for recommendation methods [52], but most focus only on precision, rarely addressing diversity, predictability, or real-time performance. Based on knowledge re-finding requirements and F. Ricci [2] and M. Sappelli's [20] four criteria—precision, context relevance, prediction, and diversity—this paper selects specific metrics (MAE, P/R, P@N, F-measure, ROGUE-N [53]) for evaluation.

5.1 Precision

Precision evaluates overlap between recommendation set R and user subjective needs MB, similar to information retrieval precision. Higher precision reduces irrelevant document interference. This paper calculates precision P for top-10 ranked documents:

$$\text{Precision} = \frac{\sum(R \cap MB_i)}{k} \tag{8}$$

where k is user count.

5.2 Context Relevance

Context relevance evaluates how well R matches users' current context, reflecting: (1) continuous matching across context sequences S_i <s_1, …, s_n>; (2) ability to minimize user attention dispersion [20]; (3) capacity to understand user context. This helps remind users of contextually relevant previously browsed documents.

The experiment uses a context triple: Context = <time, index, document (currently open)>, all extractable from HB_i browsing records. The matching degree between R and context-relevant document set Context is:

$$\text{Context} = \frac{(\text{Context.time}) \cdot \text{Num}_i(\text{Context.index})}{\text{Num}_i(\text{Context})} \cdot \text{sim}(\text{documents}, R_i) \tag{9}$$

where Num_i(Context.time) counts R documents opened within time intervals (3min); Num_i(Context.index) counts R documents from same directory; sim(documents,R_i) is topic similarity via cosine formula; Num_i(Context) counts context occurrences (document opens). A threshold of 0.04 was experimentally determined to ensure successful matching. P@N measures context relevance quality.

5.3 Prediction

As recommendations aim to remind users what they "will" use rather than what they "liked," prediction evaluates whether recommendations effectively forecast next documents, saving selection time. Period B is divided into three intervals (10min$×$3), comparing each recommendation set R with documents opened in subsequent intervals:

$$\text{Prediction} = \frac{\sum \text{Num}(R_i^{T2}) + \sum \text{Num}(R_i^{T3})}{\text{Total comparisons}} \tag{10}$$

where Num(R_i^{Tj}) counts overlaps between R and documents opened in interval Tj. Prediction@1 and Prediction@10 are used.

5.4 Diversity

Task completion involves multi-domain knowledge, making recommendation diversity crucial for stimulating creativity. Diversity measures coverage of different subdirectories under the same topic. Documents from the same minimal subdirectory are grouped; groups with \$ \$2 documents are divided by group size, then averaged across users.

6 Results

A 20-person volunteer team was organized. Each user worked on one theme per experiment, conducted after class time. Simulation experiment 1 was followed by experiment 2 four weeks later, recording behavior data and selected documents for evaluation across four recommendation systems.

6.1 Precision

Table 4 shows precision results. P@1 indicates overlap at first position; P@10 indicates top-10 overlap. CFR and CAS show lower randomness than CBR and INR because CAS classifies contexts during recommendation, creating relationships between R and context set Context that don't significantly contribute to precision. CBR and CFR show the most uneven P@1 to P@10 distribution—while performing well at top ranks, quality decays rapidly.

Table 4: Precision Comparison

System	P@1	P@10
CBR	0.325	0.125
CFR	0.289	0.098
INR	0.301	0.112
CAS	0.267	0.089

6.2 Context Relevance

Table 5 shows context relevance results. Context@1 measures first-item matching; Context@10 measures top-10 matching. CAS scores highest, proving superior context awareness in knowledge re-finding. The SA model's context item calculation strongly correlates with Context@N—removing "click count" significantly impacts context extraction and relevance.

Table 5: Context Relevance Comparison

System	Context@1	Context@10
CBR	0.690	0.567
CFR	0.369	0.397
INR	0.578	0.401
CAS	0.728	0.583

6.3 Prediction

Table 6 shows CAS performs better in predicting next-period needs. CBR's correct next-document prediction remains around 3%; CFR fluctuates more but is less effective; INR is weakest (<2%); CAS reaches nearly 4% in later periods. Note that next-period recommendations cannot include current-period documents, as users often switch between documents without closing them, making repeated switches unpredictable.

Table 6: Prediction Comparison

System	Prediction@1	Prediction@10
CBR	0.009	0.033
CFR	0.021	0.015
INR	0.012	0.033
CAS	0.027	0.039

6.4 Diversity

Table 7 shows INR and CAS have higher diversity, indicating better adaptation to context changes, followed by CBR and CFR.

Table 7: Diversity Comparison

System	Diversity
CBR	0.037
CFR	0.041
INR	0.052
CAS	0.048

6.5 Visualization

Figure 4 [Figure 4: see original paper] visualizes matching degree P@1 to P@10 across four systems. Results show context-aware systems have greater fluctuation in precision, demonstrating separation between document relevance and context relevance. CBR's results depend more on document content, while CAS depends more on current context. Increased recommendation list length means more contexts covered. While CAS has higher context adaptability, its satisfaction across all contexts decays with list length, unlike CBR and INR. Both CAS and CBR perform well in preventing system interference.

If the goal is predicting next documents, CAS is optimal because its recommendations directly associate with context-activated documents, an association stable over time. In this experiment, maximum recommendation list length was 10, providing limited diversity detection. Figure 7 [Figure 7: see original paper] shows CAS also performs well in diversity, though differences among systems are modest.

7 Discussion

The four evaluation criteria cover multiple user requirements in knowledge re-finding. Some standards relate to explicit needs (precision), others to implicit needs (diversity). For personal/team knowledge management systems, these results demonstrate effectiveness and provide suggestions for system selection. However, the highest-scoring system in experiments may not always be optimal. For example, with concentrated contexts (e.g., health documents all day), CAS prevents interference. But with dispersed contexts (frequent switching), CAS's anti-interference weakens, distracting users.

Thus, the optimal recommendation scheme depends not only on experimental results but also on current tasks in knowledge re-finding scenarios. Each system has pros and cons: CBR is simple and reliable but suffers cold-start with new contexts; INR determines context without external information but may fail queries (14% of context words in this experiment). In complex scenarios, the preferred scheme may change even within a day. Task-driven selection and consistent use are key.

Future work will integrate non-contextual and contextual models, incorporating emerging models like latent semantic and deep learning models, comparing performance differences with/without context information, and using activation spreading to mine context-relevant documents for better applicability analysis in knowledge re-finding scenarios.

References

[1] BAWDEN D, ROBINSON L. The dark side of information: overload, anxiety and other paradoxes and pathologies[J]. Journal of information science, 2009, 35(2): 180-191.

[2] RICCI F, ROKACH L, SHAPIRA B, et al. Recommender systems handbook[M]. Berlin: Springer, 2011.

[3] SCHAEFER J B, DAN F, HERLOCKER J, et al. Collaborative filtering recommender systems[M]. Berlin: Springer, 2007.

[4] BELL R M, KOREN Y. Scalable collaborative filtering with jointly derived neighborhood interpolation weights[C]//Proceedings of the seventh IEEE international conference on data mining. Washington: IEEE Computer Society, 2007: 43-52.

[5] MANOUSELIS N, VUORIKARI R, ASSCHE F V. Collaborative recommendation of e-learning resources: an experimental investigation[J]. Journal of computer assisted learning, 2010, 26(4): 227-242.

[6] XIONG Huixiang, YANG Xueping, GAO Lianhua, et al. Personalized recommendation based on user interest topic model[J]. Journal of the China Society for Scientific and Technical Information, 2017, 36(9): 916-928.

[7] WANG Hao, LIU Dongsu. Collaborative filtering recommendation algorithm based on co-clustering and user feature extraction[J]. Journal of the China Society for Scientific and Technical Information, 2017, 36(8): 852-858.

[8] DOAN A H. Learning to map between ontologies on the semantic web[C]//Proceedings of the eleventh international conference on World Wide Web. New York: ACM, 2002: 622-673.

[9] BRASETHVIK T, GULLA J A. Natural language analysis for semantic document modeling[J]. Data & knowledge engineering, 2001, 38(1): 45-62.

[10] KIM J H, SONG C W, LIM K W, et al. Design of music recommendation system using context information[C]//Proceedings of the 2006 Pacific Rim international conference on agent computing & multi-agent systems. Berlin: Springer, 2006: 708-713.

[11] WANG Hao, LIU Dongsu. Collaborative filtering recommendation algorithm based on co-clustering and user feature extraction[J]. Journal of the China Society for Scientific and Technical Information, 2017, 36(8): 852-858.

[12] MA H, ZHOU T C, LYU M R, et al. Improving recommender systems by incorporating social contextual information[J]. ACM transactions on information systems, 2011, 29(2): 1-23.

[13] LIU N N, HE L, ZHAO M. Social temporal collaborative ranking for context-aware movie recommendation[J]. ACM transactions on intelligent systems and technology, 2013, 4(1): 1-26.

[14] WANG X, ROSENBLUM D, WANG Y. Context-aware mobile music recommendation for daily activities[C]//Proceedings of the 20th ACM multimedia. New York: ACM, 2012: 99-108.

[15] ELSWEILER D, RUTHVEN I, JONES C. Towards memory supporting personal information management tools[C]//Proceedings of the 2nd BCS-IRSG conference on future directions in information access. London: British Computer Society, 2008: 14-21.

[16] REINHARDT W, SCHMIDT B, SLOEP P, et al. Knowledge worker roles

and actions: results of two empirical studies[J]. Knowledge & process management, 2011, 18(3): 150-174.

[17] WANG W, LI W, WU Z, et al. An ontology-based context model for building context-aware services[C]//Proceedings of the second international conference on intelligent systems, Modelling and Simulation. Washington: IEEE Computer Society, 2011: 296-299.

[18] WAKELING S, CLOUGH P, SEN B. Investigating the potential impact of non-personalized recommendations in the OPAC: Amazon vs WorldCat.org[C]//Proceedings of the fifth information interaction in context symposium. New York: ACM, 2014: 96-105.

[19] CHARI S, CHRISTODOULIDES G, PRESI C, et al. Consumer trust in user-generated brand recommendations on Facebook[J]. Psychology & marketing, 2016, 33(12): 1071-1081.

[20] SAPPELLI M, VERBERNE S, KRAAIJ W. Evaluation of context aware recommendation systems for information re-finding[J]. Journal of the Association for Information Science & Technology, 2017, 68(4): 895-910.

[21] AULA A. Information search and re-access strategies of experienced web users[C]//Proceedings of the 14th international conference on World Wide Web. New York: ACM, 2005: 583-592.

[22] TEEWAN J, ADAR E, JONES R, et al. Information re-retrieval: repeat queries in Yahoo's logs[C]//Proceedings of the 2007 International ACM Sigir conference on research & development in information retrieval. New York: ACM, 2007: 151-158.

[23] OBENDORF H, WEINREICH H, HERDER E, et al. Web page revisitation revisited: implications of a long-term click-stream study of browser usage[C]//Proceedings of the 2007 SIGCHI conference on human factors in computing systems. New York: ACM, 2007: 597-606.

[24] TAUSCHER L, GREENBERG S. Revisitation patterns in World Wide Web navigation[C]//Proceedings of the 1997 ACM SIGCHI conference on human factors in computing systems. New York: ACM, 1997: 399-406.

[25] COCKBURN A, MCKENZIE B. What do web users do? an empirical analysis of web use[J]. International journal of human-computer studies, 2001, 54(6): 903-922.

[26] WEN J. Post-valued recall web pages: user disorientation hits the big time[J]. IT & society, 2003, 1(3): 184-194.

[27] HARPER S. Integrating memory context into personal information refinding[C]//Proceedings of the 2nd BCS-IRSG conference on future directions in information access. London: British Computer Society, 2008: 14-21.

[28] DENG T, ZHAO L, WANG H, et al. ReFinder: a context-based knowledge refinding system[J]. IEEE transactions on knowledge & data engineering, 2013,

25(9): 2119-2132.

[29] VÖLKEL M, ABECKER A. Cost-benefit analysis for the design of personal knowledge management systems[J]. Proc of ICEIS, 2009, 6030(1): 95-105.

[30] GOLDBERG D. Using collaborative filtering to weave an information tapestry[J]. Communications of the ACM, 1992, 35(12): 61-70.

[31] RESNICK P, IACOVU N, SUCHAK M, et al. GroupLens: an open architecture for collaborative filtering of netnews[C]//Proceedings of the 1994 ACM conference on computer supported cooperative work. New York: ACM, 1994: 175-186.

[32] RUAN Guangce, XIA Lei. Development of recommender systems and discussion on personalized information services in public libraries[J]. Library, 2016, 2016(2): 94-99.

[33] BAEZA-YATES R, RIBEIRO-NETO B. Modern information retrieval[M]. New York: ACM Press, 1997.

[34] RESNICK P, VARIAN H R. Recommender systems[J]. Communications of the ACM, 1997, 40(3): 56-58.

[35] LI Y, LU L, LI X. A hybrid collaborative filtering method for multiple interests and multiple content recommendation in e-commerce[J]. Expert systems with applications, 2005, 28(1): 67-77.

[36] JIA T, YEON C, KIM H N, et al. Collaborative tagging in recommender systems[M]. Berlin: Springer, 2007.

[37] WOOLDRIDGE M J. An introduction to multiagent systems[M]. Chichester: John Wiley & Sons, 2002.

[38] ADOMAVICIUS G. Incorporating contextual information in recommender systems using a multidimensional approach[J]. ACM transactions on information systems, 2005, 23(1): 103-145.

[39] LI Y, ZHANG D, LAN Z, et al. Context-aware advertisement recommendation for high-speed social news feeding[C]//Proceedings of the 2016 international conference on data engineering. Washington: IEEE Computer Society, 2016: 505-516.

[40] LOMMATZSCH A. Real-time news recommendation using context-aware ensembles[C]//Proceedings of the 2014 European conference on information retrieval. Berlin: Springer, 2014: 51-62.

[41] GOLDBERG D. Using collaborative filtering to weave an information tapestry[J]. Communications of the ACM, 1992, 35(12): 61-70.

[42] IEHistoryView [EB/OL]. [2017-09-24] https://iehistory-view.en.softonic.com/.

[43] GOLDBERG D, NICHOLS D, OKI B, et al. Using collaborative filtering to weave an information tapestry[J]. Communications of the ACM, 1992, 35(12),

61-70.

[44] AHN H J. A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem[J]. Information sciences, 2008, 178(1): 37-51.

[45] NAGARAJA R, SCUTARI M, LÈBRES S. Bayesian network inference algorithms[M]. Berlin: Springer, 2013.

[46] TURTLE H, CROFT W B. Evaluation of an inference network-based retrieval model[J]. ACM transactions on information systems, 1991.

[47] STROHMAN T, METZLER D, TURTLE H, et al. Indri: a language model-based search engine for complex queries[C]//Proceedings of 2005 IEEE international conference on information and automation. New York: ACM, 2005: 1-34.

[48] MCCLELLAND J L, RUMELHART D E. An interactive activation model of context effects in letter perception, part i: an account of basic findings[J]. Psychological review, 2013, 88(5): 375-407.

[49] CRESTANI F. Application of spreading activation techniques in information retrieval[M]. Dordrecht: Kluwer Academic Publishers, 1997.

[50] TIAN Ye, YANG Mei, ZHU Zhongming, et al. Research on linked data-driven query expansion technology[J]. Library and information service, 2015, 59(4): 122-128.

[51] DEY A K, ABOWD G D. CybreMinder: a context-aware system for supporting reminders[C]//Proceedings of the 2nd international symposium on handheld and ubiquitous computing. Berlin: Springer, 2000: 172-186.

[52] LÜ L, MEDO M, CHI H Y, et al. Recommender systems[J]. Physics reports, 2012, 519(1): 1-49.

[53] FLICK C. ROUGE: a package for automatic evaluation of summaries[C]//Proceedings of the workshop on text summarization branches out. Stroudsburg: ACL, 2004: 1-105.

Author Contributions: Cheng Xiufeng: research framework design and paper writing; Zhang Ziming: experimental design scheme revision; Meng Yaqi: data collection; Fan Xiaoying: data collection; Yang Jinqing: data analysis.

Correction Notice: In the paper "Social Media User Switching Behavior Influence Factors Model and Empirical Study" published in Issue 18, 2018, the fourth author should be Wei Yanan, not Wei Yanan. This is hereby corrected.

Note: Figure translations are in progress. See original paper for figures.