ChinaRxiv

AI translation · View original & related papers at
chinarxiv.org/items/chinaxiv-201810.00052

Construction and Updating of User Interest Models for News Recommendation (Postprint)

Authors: Yuan Renjin, Gang Chen, Li Feng

Date: 2018-10-11T00:00:00+00:00

Abstract

To address the challenges of user interest model construction and user interest drift in news recommendation systems, this paper proposes a method for constructing and updating user interest models tailored for news recommendation. Initially, the vector space model and Bisecting K-means clustering algorithm are utilized to construct the original user interest model; subsequently, a forgetting function is formulated based on the Ebbinghaus forgetting curve and employed to apply temporal weighting to the user interest model, thereby achieving its updating. The experiments adopt user-based collaborative filtering recommendation and item-based collaborative filtering recommendation as baselines. Experimental results demonstrate that the constructed original user interest model achieves superior recommendation performance, with a 4% improvement in F-measure, while the updated model yields a 1.3% increase in F-measure compared to the original model.

Full Text

Preamble

User Interest Model Construction and Update for News Recommendation

Yuan Renjin, Chen Gang, Li Feng
(Institute of Geospatial Information, Information Engineering University, Zhengzhou 450001, China)

Abstract: To address the challenges of user interest model construction and user interest drift in news recommendation systems, this paper proposes a method for constructing and updating user interest models tailored for news recommendation. First, the vector space model and Bisecting K-means clustering algorithm are employed to construct the initial user interest model. Then, a forgetting function is constructed based on the Ebbinghaus forgetting curve and

applied to temporally weight the user interest model, thereby achieving model updates. Using user-based and item-based collaborative filtering recommendations as baselines, experimental results demonstrate that the proposed initial user interest model achieves superior recommendation performance, with a 4% improvement in F-value. Furthermore, the updated model shows a 1.3% F-value improvement over the original model.

Keywords: personalized recommendation; vector space model; user interest model; user interest drift; forgetting function

1 Introduction

In the era of information and big data, personalized recommendation technology has become the preferred solution for effectively utilizing massive information resources to provide customized services across various domains, including e-commerce, music, news, and film. User interest modeling represents one of the key technologies in recommendation systems. Collaborative filtering algorithms, as classical approaches in recommendation systems, have been widely applied to news recommendation by numerous scholars who have built user interest models based on these methods. However, collaborative filtering-based algorithms fail to consider news content and suffer from poor interpretability and data sparsity issues. To address these limitations, Okura et al. incorporated news content and user preferences, using Recurrent Neural Networks (RNN) to construct user interest models with promising results. Zhang et al. proposed a collaborative model combining matrix factorization, topic analysis, and knowledge graph representation to alleviate text scarcity. Additionally, news content and categorization significantly impact recommendation effectiveness. In terms of news classification, Gu Wanrong et al. and Li Jiashan summarized and analyzed news clustering algorithms but did not investigate user interest drift.

In practice, news exhibits strong timeliness, and user interests evolve over time. Current algorithms addressing user interest drift primarily fall into three categories: time window methods, forgetting function methods, and hybrid algorithms. Time window methods filter users' latest interests through sliding windows, as studied in the literature. Forgetting function methods adjust the weights of items users were interested in at different times using forgetting functions. Cheng Weidan applied the Ebbinghaus forgetting curve to describe user interest drift in collaborative filtering algorithms, while Sun et al. constructed a dynamic model for user interest drift based on clustering and nearest neighbors. Hybrid algorithms combine different approaches organically. Xing Chunxiao et al. proposed a hybrid algorithm to explore user interest changes based on collaborative filtering. However, these methods either focus solely on user interest drift or investigate it within the collaborative filtering framework, leaving room for further research on solving problems specific to news recommendation.

To address these issues, this paper proposes a method for constructing and

updating user interest models for news recommendation. First, user interest models are built based on the Vector Space Model (VSM) and Bisecting K-means clustering. Since the Ebbinghaus forgetting curve aligns better with human memory patterns, we then construct a forgetting function based on this curve to study user interest model updating. This approach retains the interpretability advantages of content-based recommendation algorithms while avoiding issues associated with editorial classification through clustering. By investigating user interest model updates, the method better reflects changes in user interests and improves news recommendation accuracy.

The framework for user interest model construction and updating is illustrated in [Figure 1: see original paper]. We first describe several key concepts:

a) **News keywords**: The most representative words in news that characterize its uniqueness and distinctiveness, typically extracted using text processing algorithms.

b) **News feature vector**: Since news content is textual, a multi-dimensional vector ($d = (w_1, w_2, \dots, w_m)$) represents the news content. The result of vectorizing news text is called a news feature vector.

---

2 User Interest Model Construction

2.1 News Text Vectorization

For a news collection $D = \{d_1, d_2, \dots, d_n\}$, the VSM representation is:

$$M = \begin{bmatrix} w_{11} & \cdots & w_{1m} \\ \vdots & \ddots & \vdots \\ w_{n1} & \cdots & w_{nm} \end{bmatrix}$$

where $[w_{i1}, w_{i2}, \dots, w_{im}]$ represents the feature vector of news item i, and w_{ij} denotes the weight of keyword j in news i. Building VSM involves two key aspects: determining the dimension m of the keyword set and calculating the weights w_{ij}.

a) **Keyword set dimension** m: First, keywords are extracted from each news article, then the TF-IDF algorithm is applied to determine the dimension m of the keyword set for the news collection.

b) **Weight calculation** w_{ij}: The most common and effective weight calculation method is the TF-IDF representation, a mature technique in information retrieval that we will not elaborate on here.

To ensure weight values fall within the $[0, 1]$ interval and that news can be represented by equal-length vectors, cosine normalization is applied. The weight calculation formula is:

$$w(i,j) = \frac{\text{TF-IDF}(i,j)}{\sqrt{\sum_{j=1}^{m} \text{TF-IDF}(i,j)^2}}$$

2.2 Constructing Hierarchical User Interest Models

In this paper, user interest models are constructed from users' browsing history and represented using a three-layer hierarchical structure: user—news category —news article. As shown in [Figure 2: see original paper], the first layer is the user node, the second layer represents news categories browsed by the user, and the third layer contains individual news articles the user has browsed.

If a user has browsed m different news categories, the user interest model can be represented as:

$$user = \{(T_1, w_1, n_1), (T_2, w_2, n_2), \ldots, (T_m, w_m, n_m)\}$$

where T_i represents the feature vector of the i-th news category, w_i denotes the weight of the i-th news category, and n_i indicates the number of news articles browsed by the user in the i-th category.

The feature vector of a news category is calculated as the weighted average of all browsed news feature vectors within that category based on interest degree. The formula for calculating the feature vector T_i of the i-th news category is:

$$T_i = \frac{\sum_{e_j \in E_j} e_j \cdot I_j}{\sum_{e_j \in E_j} I_j}$$

where E_j represents the set of news articles browsed by the user in category i, e_j denotes the news feature vector, and I_j represents the user' s interest degree in the j-th news article in that category. Since browsing a news article indicates user interest, we set $I_j = 1$, simplifying equation (4) to:

$$T_i = \frac{\sum_{e_j \in E_j} e_j}{\sum_{e_j \in E_j} 1}$$

The value of w_i is calculated based on the proportion of news articles browsed in category i relative to the total number of news articles browsed by the user, as shown in equation (6).

During calculation, the user interest model is represented as:

$$V_{user} = (w_1 \cdot T_1, w_2 \cdot T_2, \ldots, w_m \cdot T_m)^T$$

Finally, cosine similarity is used to calculate the similarity between candidate news d_i and the user, with the calculation formula shown in (8):

$$sim(user, V_{di}) = cos(w_i \cdot T_i^T, V_{di})$$

where $w_i \cdot T_i^T$ is the feature vector of the news category to which candidate news d_i belongs, and V_{di} is the feature vector of d_i.

---

3 User Interest Model Update Based on Forgetting Function

3.1 Forgetting Function

Research by the renowned German psychologist Ebbinghaus indicates that human forgetting follows a non-linear pattern over time rather than a linear one. In recommendation systems, user interests also change over time and should conform to this forgetting pattern. News articles browsed earlier leave weaker impressions in users' minds and thus should carry less weight in the user interest model.

Based on this principle, we construct a time-based forgetting function similar to the Ebbinghaus forgetting curve, defined as:

$$F(u, i) = a + (1 - a)e^{-\frac{T_{ui}}{T_u}}$$

where T_{ui} represents the time interval between when user u browsed news i and their most recent browsing activity; T_u denotes the time interval between the user' s earliest and most recent news browsing activities; and a is the weight regulation factor, with $a \in [0, 1]$.

[Figure 3: see original paper] illustrates the Ebbinghaus forgetting curve, showing the relationship between memory retention ratio and time. [Figure 4: see original paper] displays the trend of our constructed forgetting function, showing how the memory retention ratio for news changes with browsing time intervals, where the value of a affects the function' s trend. The a value can be dynamically adjusted according to different recommendation systems to achieve optimal performance.

3.2 User Interest Model Update Using Forgetting Function

The impression of browsed news fades over time. We quantify the weight of browsed news in users' minds using the forgetting function from equation (9), which shows that news weights change over time. For user u, the weight t_{ui} of news i is:

$$t_{ui} = F(u, i)$$

We introduce the forgetting function to update the user interest model proposed above. For a browsed news article j, its content representation changes from $\langle e_j \rangle$ to $\langle e_j, t_{uj} \rangle$. By incorporating news weights into equation (5), we obtain the updated feature vector T_i' for the i-th news category:

$$T_i' = \frac{\sum_{e_j \in E_j} e_j \cdot t_{uj}}{\sum_{e_j \in E_j} t_{uj}}$$

The updated user interest model transforms from equation (7) to equation (12):

$$V_{user}' = (w_1 \cdot T_1', w_2 \cdot T_2', \ldots, w_m \cdot T_m')^T$$

---

4 Experiments and Comparative Analysis

4.1 Data Preparation

This study uses news browsing records provided by DataCastle as the experimental dataset. Randomly collected from Caixin Net in China, the dataset includes 116,225 browsing records from 10,000 users during March 2014. Each record contains user ID, news ID, browsing timestamp, and news text content. For preprocessing, we use Python' s jieba tokenizer for word segmentation and apply an improved stopword list from the Harbin Institute of Technology' s Information Retrieval Center to remove stopwords.

The raw data is preprocessed by filtering out users with more than 30 browsing records, resulting in 32,770 records from 417 users. These data are randomly divided into five groups, each containing 200 users. For each user, the last 10 browsing records are used as the test set. Since sampling results overlap across groups, we average the experimental results from all five groups to ensure objectivity.

4.2 Evaluation Metrics

To balance accuracy and recall, we adopt the F-value as our evaluation metric, which combines precision and recall. Precision and recall are derived from the confusion matrix shown in .

Table 1. Confusion Matrix

	Predicted Positive	Predicted Negative
Actual Positive	true positive (TP)	false negative (FN)

	Predicted Positive	Predicted Negative
Actual Negative	false positive (FP)	true negative (TN)

Precision P and recall R are calculated as:

$$P = \frac{TP}{TP + FP}, \quad R = \frac{TP}{TP + FN}$$

The F-value is computed using both precision P and recall R:

$$F = \frac{2PR}{P + R}$$

4.3 Experimental Results and Analysis

To validate the feasibility and effectiveness of our proposed method, we conduct comparative experiments using user-based collaborative filtering and item-based collaborative filtering as baselines. For these collaborative filtering algorithms, we consider neighbor counts of (5, 10, 15, 20) and five different recommendation list sizes (10, 15, 20, 25, 30).

Impact of Cluster Number M on Recommendation Performance

To investigate how the number of clusters M in the news set affects recommendation performance, the Bisecting K-means clustering algorithm considers M values of (10, 15, 20, 25, 30). The corresponding evaluation results are shown in .

Table 2. Impact of Cluster Number M on F-value

M	$N = 10$	$N = 15$	$N = 20$	$N = 25$	$N = 30$
10	F=0.164	F=0.175	F=0.180	F=0.173	F=0.162
15	F=0.182	F=0.196	F=0.205	F=0.192	F=0.187
20	F=0.193	F=0.211	F=0.215	F=0.187	F=0.179
25	F=0.175	F=0.197	F=0.201	F=0.198	F=0.183
30	F=0.171	F=0.179	F=0.185	F=0.178	F=0.169

The results indicate a non-linear relationship between the number of clusters M and the F-value: as M increases, the F-value first rises and then falls, reaching its maximum when $M = 20$, which yields optimal recommendation performance.

Comparison of Proposed Model with Collaborative Filtering Algorithms

We compare the optimal results from each method across different conditions, focusing on two experiments: (a) comparing our constructed model with collaborative filtering algorithms, and (b) comparing the updated model with the original model.

The average F-values across five experimental groups for different algorithms are presented in .

Table 3. F-value Comparison Between Proposed Model and Collaborative Filtering

Algorithm	$N = 10$	$N = 15$	$N = 20$	$N = 25$	$N = 30$
User-based CF	F=0.166	F=0.173	F=0.168	F=0.157	F=0.148
Item-based CF	F=0.162	F=0.174	F=0.164	F=0.160	F=0.146
Our Model	F=0.193	F=0.211	F=0.215	F=0.187	F=0.179

[Figure 5: see original paper] illustrates the F-value trends across different algorithms. Our model demonstrates significantly improved recommendation performance compared to both collaborative filtering methods. When the recommendation list size falls within [15, 20], all algorithms perform relatively best, with our model' s F-value averaging 4% higher than the collaborative filtering algorithms.

Comparison Between Updated Model and Original Model

The recommendation performance of the updated model is influenced by the weight regulation factor a in equation (9). To understand the relationship between a and recommendation performance, we experiment with a values of (0, 0.2, 0.4, 0.6, 0.8). shows the performance of the updated model under different a values and recommendation list sizes.

Table 4. Impact of a Value on Recommendation Performance

Updated Model	$N = 10$	$N = 15$	$N = 20$	$N = 25$	$N = 30$
$a = 0$	F=0.185	F=0.191	F=0.193	F=0.183	F=0.172
$a = 0.2$	F=0.192	F=0.209	F=0.211	F=0.183	F=0.174
$a = 0.4$	F=0.203	F=0.221	F=0.223	F=0.210	F=0.194
$a = 0.6$	F=0.202	F=0.219	F=0.220	F=0.203	F=0.189
$a = 0.8$	F=0.173	F=0.185	F=0.186	F=0.169	F=0.157

[Figure 6: see original paper] visualizes how a values affect recommendation performance. The optimal performance occurs when $a = 0.4$. Under this forgetting function, we compare the updated model with the original model, as shown in [Figure 7: see original paper]. The updated model consistently outperforms the original model, with an average F-value improvement of 1.3%, demonstrating

that user interests do drift over time and that our forgetting function facilitates model updating, though the improvement is modest and warrants further investigation.

---

5 Conclusion

This paper addresses the impact of news content and categorization on news recommendation system performance. We first propose a user interest model construction method based on VSM and Bisecting K-means clustering. Recognizing that user interests drift over time, we construct a forgetting function inspired by the Ebbinghaus forgetting curve and apply temporal weighting to update the model. Experimental results show that:

a) Compared to user-based and item-based collaborative filtering algorithms, our user interest model constructed with VSM and Bisecting K-means clustering achieves better recommendation performance, with an average F-value improvement of 4%.

b) The updated user interest model using the forgetting function shows slight performance improvement over the original model, with an average F-value increase of 1.3%.

c) The F-value consistently exhibits a pattern of initially increasing and then decreasing across experimental data, a phenomenon that currently remains unexplained and requires further investigation.

Overall, the proposed method can serve as a viable approach in news recommendation, offering a reference for model construction and forgetting function-based updating. Future work will further investigate user interest model updating and explore news recommendation methods incorporating geographic location information.

References

[1] Leng Yajun, Lu Qing, Liang Changyong. Survey of recommendation based on collaborative filtering [J]. PR&AI, 2014, 27(8): 720-734.

[2] Kong Yanli. Research on personalized recommendation based on collaborative filtering [D]. Beijing: Beijing University of Technology, 2016.

[3] Wu Yanwen, Qi Min, Yang Rui. A new recommendation system based on an improved collaborative filtering algorithm [J]. Computer Engineering & Science, 2017, 39(6): 1179-1185.

[4] Garcin F, Zhou Kai, Faltings B, et al. Personalized news recommendation based on collaborative filtering [C]//Proc of IEEE/WIC/ACM International

Conferences on Web Intelligence and Intelligent Agent Technology. [S. l.]: IEEE Press, 2013: 437-441.

[5] Peng Feifei, Qian Xu. Personalized recommendation system for news based on user concern [J]. Application Research of Computers, 2012, 29(3): 1005-1007.

[6] Okura S, Tagami Y, Ono S, et al. Embedding-based news recommendation for millions of users [C]//Proc of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. [S. l.]: ACM Press, 2017: 1933-1942.

[7] Zhang Kuai, Xin Xin, Luo Pei, et al. Fine-grained news recommendation by fusing matrix factorization, topic analysis and knowledge graph representation [C]//Proc of IEEE International Conference on Systems, Man and Cybernetics. [S. l.]: IEEE Press, 2017: 918-923.

[8] Gu Wanrong, Dong Shoubin, He Jingchao, et al. News recommendation method based on secondary clustering [J]. Journal of South China University of Technology: Natural Science, 2014, 42(7): 15-20+32.

[9] Li Jiashan. Research and implementation of text clustering for personalized news recommendation system [D]. Beijing: Beijing University of Posts and Telecommunications, 2013: 20-29.

[10] Fei Hongxiao, Dai Yi, Mu Jun, et al. Method of drifting user' s interests based on time window optimization [J]. Computer Engineering, 2008, 34(16): 210-211.

[11] Cheng Weidan. Research on collaborative filtering algorithm based on forgetting function and user [D]. Hangzhou: Zhejiang University of Technology, 2016.

[12] Sun Baoshan, Dong Lingyu. Dynamic model adaptive to user interest drift based on cluster and nearest neighbors [J]. IEEE Access, 2017, PP(99): 1.

[13] Xing Chunxiao, Gao Fengrong, Zhan Sinan, et al. A collaborative filtering recommendation algorithm incorporated with user interest change [J]. Journal of Computer Research and Development, 2007, 44(2): 296-301.

[14] Yao Qingyun. Research of VSM-based Chinese text clustering algorithms [D]. Shanghai: Shanghai Jiao Tong University, 2008: 27.

[15] Abuaiadah D. Using bisect K-means clustering technique in the analysis of Arabic documents [J]. ACM Trans on Asian and Low-Resource Language Information Processing, 2016, 15(3): 1-13.

[16] Yu Zhuang. Symmetric repositioning of bisecting K-means centers for increased reduction of distance calculations for big data clustering [C]//Proc of IEEE International Conference on Big Data. [S. l.]: IEEE Press, 2017: 2709-2715.

[17] Ebbinghaus H. Memory: a contribution to experimental psychology [J]. Annals of Neurosciences, 2013, 20(4): 155.

[18] Xiang Liang. Recommended system practice [M]. Beijing: Post & Telecom Press, 2012: 44-59.

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

Source: ChinaXiv —Machine translation. Verify with original.