New Attributes for Neighborhood-based Collaborative Filtering in News Recommendation

Asghar Darvishy, Hamidah Ibrahim, Aida Mustapha, and Fatimah Sidi
Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, Malaysia
Email: a_darvishy@azad.ac.ir, {hamidah.ibrahim, aida_m, fatimah }@upm.edu.my

***Abstract*—How can recommendation accuracy be improved in a personalized news recommender system? In this paper, we utilize new attributes in addition to standard recency and popularity such as *Reading Rate* and *Hotness*. These attributes are defined in the user profile and news metadata and are used in neighborhood-based collaborative filtering in news recommendation. We analyze the proposed attributes in the user profile construction and the news metadata enrichment by exploring similar users' interests in news reading. This is carried out via experiments using *k*-means. We then compare the precision, recall and F1-score in a series of experiments to evaluate the news recommendation with these attributes. The experimental results show that the proposed attributes improved the accuracy in news recommendation with higher precision and F1-score. We conclude that *Reading Rate* and *Hotness* in news have a significant impact on personalized news recommendation systems.**

***Index Terms*—news recommendation, collaborative filtering, news hotness, user profile, news metadata**

I. Introduction

A huge and growing volume of news is provided on the Web. In light of this, news recommendation has become one of the most significant subjects in research. Major web providers such as *Yahoo! News* and *Google News* deal with getting the news from infinite sources to the users based on their preferences. Recommendation services can widely help users to acquire the correct information on the items that they are interested in. Creating a fine user profile and rich news metadata can help in the preparation of an accurate and efficient recommendation. A high quality personalized news recommender system should provide the news set to each user based on his/her interests and behavior.

The news recommendation system is different in several aspects compared to other types of recommendation systems. News recency has greater importance than other items like music or movies [1]. Secondly, despite the fact that news articles have many words in common between them, it is not necessarily indicative of a relationship between them [2]. Thirdly, news analysis is not easy as news articles have an unstructured format. Also, it is common for users to prefer reading news in various topics and areas. The recommended news set should be diversified enough to cover user's interests [1, 3]. Finally, breaking news can be interesting even if it is not in line with user's preferences [1, 4]. This provides opportunities for improvement in news recommendation systems such as news selection, news representation, news processing, and user profiling.

In an effort to improve news selection and user profiling, this paper focuses on one aspect in the news recommendation system, which is the news attributes. In this research, we focus entirely on empirical experiments in collaborative filtering to evaluate recommendations with new attributes in news metadata and user profile. We developed a news recommender system in neighborhood-based collaborative filtering and enhanced it by adding attributes such as *ReadingRate*. Based on *ReadingRate*, we predict the number of news items that a user prefers to read and rank the news by their *Hotness,* which can help to recommend a better news set to a given user. Our results imply that enhancement with new attributes increases accuracy in news recommendation.

The remainder of this paper is organized as follows: in Section 2, we present a brief summary of previous works in neighborhood-based CF as well as works in news recommendation. In Section 3, the proposed new attributes in news recommendation systems are discussed, while Section 4 details the experimental setup, Section 5 discusses the results and finally, Section 6 presents the conclusion to the article and proposes some recommendations for related future research.

II. Related Works

Recommendation systems are not only ubiquitous in the generation and consumption of online information [5] but are also robust in facilitating users to access huge amounts of data such as articles, news, videos, and even jobs [6]. One of the big issues in recommendation systems is finding out how to accurately recommend information to the users [1]. Recommendation systems are based on two major approaches, namely: collaborative filtering (CF) and content-based filtering (CB).

The term collaborative filtering (CF) was coined in [7, 8] to refer to personalized recommendations that are generated based on user preferences. CF-based

Manuscript received May 2, 2015; revised August 11, 2015.

©2015 Journal of Emerging Technologies in Web Intelligence
doi: 10.12720/jetwi.7.1.13-19

recommendation systems retain user preferences data and utilize them to identify groups of similar users. Then, user-liked items can be recommended to similar users in the group. These methods recommend news based on previous readings of other users by identifying similar users to the active user [7-10]. For this reason, CF-based systems are also referred to as social-filtering recommendation systems.

On the contrary, content-based (CB) recommendation systems generate recommendations using comparative representations of content relating an item to representations of content that are of interest to the user. In this method, news recommendation is done based on content similarity between the news articles that a user has read, with the newly-published news by considering news recency. As such, many methods have classified this issue as a matter of information retrieval (IR), where the content related to the user's preferred choices is regarded as an equiry, and the unrated documents are evaluated on the basis of relevance/similarity to this enquiry [11-13].

In order to exploit the fortes of content-based and collaborative recommenders, many hybrid methods have been recommended based on a combination of the two. One simple method is to accommodate both collaborative filtering and content-based methods to obtain two separately-rated lists of recommendations, and then arrive at a final list, which is a merger of the two results. The two predictions employing an adaptive weighted average are combined, where the weight of the collaborative component increases in line with the increase in the number of users accessing an item [14, 15].

There are many hybrid methods developed on the basis of traditional collaborative filtering, but which also retain the content-based profile for individual users. Such profiles, instead of co-rated items, are utilized to identify similar users. In [1, 16], each user profile is characterized by a vector of weighted words obtained from positive training examples employing the Winnow algorithm. Predictions are made through the application of CF directly to the matrix of user-profiles (in contrast to the user-ratings matrix). Alternatively, an approach known as FAB [11, 17] makes use of relevance feedback to simultaneously mold a personal filter together with a communal "topic" filter. This method initially ranks a document according to the topic filter and then transmits it to a user's personal filter. The relevant feedback of the user is then used to alter both the personal filter and the originating topic filter.

A few news recommender systems utilize recency and popularity to modify ranking in the news recommended list. One example of a news recommendation system is the SCENE [1], which uses both of them as the exclusive attributes of the news articles. In this system, recency and popularity are the dynamic characteristics of the news metadata. The normalized values of both are combined and based on that the recommended news articles are ordered. In PENETRATE [12] and LOGO [13], in the final step, the recommended news set is adjusted based on recency and popularity. LOGO is a content-based news recommender system and uses recency and popularity in the last step of recommendation. Some news recommender systems employ just recency to generate the final recommended list. In NewsWeeder [18], the news articles in each rating category are converted into Term Frequency-Inverse Document Frequency (TF-IDF) word vectors, and then averaged to obtain a prototype vector of each category for a user. To categorize recent news, a comparison is made with each prototype vector and a predicted rating is provided on the basis of the cosine similarity to each category. Other examples include Tapestry [7] a commercial recommender system, Netflix [19] , and GroupLens [9].

The Tapestry system is proposed based on collaborative filtering. A basic idea of the Tapestry work is that more effective filtering can be done by involving humans' behavior in the filtering process [7]. Many other recommendation systems have used collaborative filtering in hybrid methods such as: *Google News* [20], *Yahoo! news* [13, 21], and SERUM [3].

III. Collaborative Filtering in News Recommendations

It is interesting to note that many users prefer to know what is interesting in news rather than reading news based on their reading behavior [22] and the news can be recommended based on neighbor user's interest and rating. A CF-based recommendation system works based on user feedback on their rating of items, a click on news is rated 1, indicating the user is interested in the news article, whereas a non-click is rated 0, which means the user is not interested in the news [1]. CF-based methods are categorized into neighborhood-based (commonly referred to as memory-based) approach and model-based approach [8].

A. Neighborhood-based CF

Neighborhood-based or memory-based CF depends on the similarities between an active user and a subset of selected users. News predictions for the active user are produced by a weighted combination of their readings [8]. Generally, all of these algorithms include the following steps:

- Compute the similarity weight across all users with respect to the active user.
- Select k users with the highest similarity weight compared to the active user (their neighbors).
- Predict from the selected neighbors' readings and their combined weights.

In these steps, it is necessary to measure similarity pair-to-pair of all users using Pearson Correlation Coefficient [9], cosine similarity [23] or Jaccard similarity [24].

B. Model-based CF

In comparison with the neighborhood-based or memory-based CF, model-based CF attempts to model users according to their previous readings and predict the readings on non-clicked news. This approach employs the probabilistic models, cluster models, and Bayesian models [25]. Nonetheless, Model-based CF is lacking in user categorization, where each user belongs to a single group while in reality a user has diverse interests in news reading.

IV. The Proposed New Attributes in User Profile and News Metadata

Existing news recommendation systems mainly rely on two attributes, which are *Recency* that indicates how recent the news has been published and *Popularity* that indicates how popular the news is in terms of number of users who have read the news. In addition to these two attributes, we propose two new attributes of user profile and news metadata, which are *ReadingRate* and *Hotness*, in an effort to recommend a better set of news to the users, based on the *ReadingRate*, the number of news topics that the user prefers to read are predicted and the news items are arranged by their *Hotness*. The user profile and news metadata are constructed as follows.

A. User Profile Construction

A recommendation system starts to capture the user's reading behavior to construct the user profile. In traditional collaborative filtering, user profile includes user's click behavior. Based on the user's news reading time and news crawled time new attributes are defined that improve recommendation. We propose a user profile with a new structure in CF, including similar user access pattern, *ReadingRate*. Each profile is parameterized with the tuple, *U = <Nu, R, Hr>*
where:

- *Nu* represents a set of neighbor users $\{nu_1, nu_2, \ldots nu_n\}$ with similar reading behavior.
- *R* represents *ReadingRate* which determines the daily reading behavior of an active user.
- *Hr* represents *HotnessRate* which denotes the average of news *Hotness* that the user has read.

These three dimensions make a rich user profile and their correlation helps to achieve higher accuracy in recommendation. *ReadingRate* is a new term and attribute in the user profile and is defined as follows:

DEFINITION 1. *ReadingRate*: User's daily news reading average, which is the average number of news items a user reads per day.

As each user has different news reading behavior, user interests differ in terms of the number of news topics read. For example, user *a* reads *i* news topics on average per day and user *b* reads *j* news topics per day. It is better to recommend a different number of news topics to each user based on their reading interests and behavior. In previous works, the experiments were evaluated with top 20, top 30 or top 50 recommended news topics [1, 12, 21]. However, in reality, different users have different reading behaviors; hence we argue that the number of recommended news items should differ according to the individual user. The attribute *ReadingRate* is proposed to achieve this objective.

DEFINITION 2. *HotnessRate*: The average of news *Hotness* that the user has already read.

Based on news *Hotness* that a user prefers to read, *HotnessRate* is computed. *HotnessRate* is an average value of news hotness that the user has read. For example, some users can follow sports news on weekends and they need to access all news articles related to sport. It means these users like to read news topics with lower degree of *Hotness*. In contrast, some users prefer to read news when it is *hot*. *HotnessRate* shows user's interest in reading the hottest news topics whether he/she prefers to read or not.

B. News Metadata Creation

Regardless of the news content, summarization in content-based filtering, by considering the pure collaborative filtering, the rich news metadata can be created. News metadata include published-time and crawled-time. Crawled-time refers to the time that a news article becomes readable in the web environment. By comparing the news crawled time with the user reading time new advantageous attributes are defined to increase accuracy in recommendation. Dynamic news metadata are defined to capture the values of these attributes. When a news *N* is read by user *U*, a new multi-dimensional tuple of metadata is created :{ *U, N, reading-time, crawled-time, Recency, Popularity, Hotness*}. This tuple is constructed as news metadata. The new attributes in this tuple are defined below.

DEFINITION 3. *Recency*: The time difference, in seconds, between the time the news is crawled on the web and the time the user reads the news. To each news article, the score for *Recency* is calculated as follows:

$$Recency = CurrentTime - CrawledTime \quad (1)$$

where *PublishedTime* determines when a news item is published in the web environment and *CurrentTime* is the user's news reading time. This means newly-published news has a lower *Recency* score.

DEFINITION 4. *Popularity*: The number of users accessing a news article. *Popularity* increases every time a user reads a news item. This attribute determines whether a news article is interesting and how many users have already read the news. To each news article, *Popularity* is calculated as follows:

$$Popularity = Popularity + 1 \quad (2)$$

DEFINITION 5. *Hotness*: is defined in news metadata and represents the count of the news reading within a certain duration of time. *Hotness* is calculated as follows:

$$Hotness = Popularity \,/\, Recency \quad (3)$$

These attributes: *ReadingRate*, *HotnessRate*, *Recency*, *Popularity*, and *Hotness* are placed in the user profile and news metadata for further processing in news recommendation system.

C. Ranking Adjustment

In news recommendation, recency is an important attribute of news metadata. Normally, users prefer to read more recent news rather than similar news to their interests and preferences. As mentioned, news is a specific item with different properties in recommendation systems. Some news (breaking news) is read thousands of times per minute. If we add popularity to news metadata, news prediction can be improved. Now, the news items with less recency and high popularity can be in the prediction list. Combination of these values creates hotness and by ordering news based on hotness, the newly published and more popular news can be selected to recommend. In other words, hotness determines how many times a news item is read in a particular duration of time?

Generally hotness is less than 1.0 (a float number between <0, 1.0>) but, if reading of a news item increases more than one time per second, it can increase to 1.0. We normalize hotness to ensure all values are between <0, 1.0>. The normalization is done as

$$Hotness_{norm} = \frac{Hotness - Hotness_{min}}{Hotness_{max} - Hotness_{min}} \quad (4)$$

where $Hotness_{max}$ is maximum value and $Hotness_{min}$ is minimum value of hotness in news metadata table.

V. EXPERIMENTAL SETUP

Typically, a recommendation system is composed of three phases: data preparation and preprocessing, pattern discovery, and recommendation [22]. In this paper, we focus on empirical experiments in neighborhood-based collaborative filtering to evaluate the proposed new attributes. Fig. 1 shows the sections of a news recommendation system using neighborhood-based Collaborative Filtering (CF).

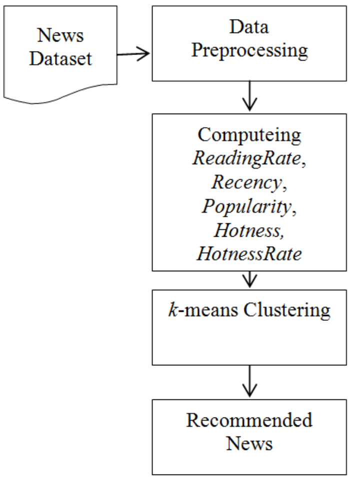


Figure 1. The proposed neighborhood-based CF with new attributes

A. News Dataset

The dataset was retrieved from crawled Twitter information streams over a period of more than two months, from October 2010 to January 2011. In this information stream, more than 20,000 users were crawled with total news reads more than 10 million times. To link the tweets with the news articles, more than 60 RSS feeds of prominent news media such as *CNN*, *BBC*, and *New York Times* were monitored. The tweets made a total of 77, 544 news items [26].

As we are interested in analyzing the user profiles and the attributes in news metadata, a sample was gathered of 1,009 users, who read at least four news items. This sample dataset contained 1,161,798 news reading records. In our sample, 38,737 news items were derived. Table I represents the dataset characteristics and statistical information.

TABLE I. DATASET CHARACTERISTICS

No. of News items	38,737
No. of Users	1,009
News Readings	1,161,798
Daily News Average	355

B. Data Preprocessing

The tweet data consist of many items not related to news recommendation, such as the tweet contents, username, and news URL. We performed three steps of data preprocessing, namely: Data cleaning, data transformation, and data reduction. Data cleaning is required to solve some data problems such as missing data. Data filtering and session identification are performed to complete the data preparation. Data transformation is required because data are distributed across a number of tables such as the user table, news table, and relation table. By joining these tables, it is possible to identify outliers and smooth out noisy data. Next, the pattern discovery phase commenced, which included measuring user similarity and clustering the users in order to locate the nearest users to an active user.

C. Computation of News Attributes

In our experiment, we evaluated the proposed news attributes: *ReadingRate* and *Hotness* in addition to the standard attributes, *Recency* and *Popularity*. The candidate news list is prioritized and adjusted based on their *Hotness* comparison with user's *HotnessRate* and *ReadingRate*.

D. User Clustering Using k-means

In this step, we performed user clustering by running *k*-means algorithm with Jaccard similarity, where *k*-means is a clustering method of vector quantization that is popular in data mining. This method is able to partition two-way, two-mode data into k classes (c_1, c_2, …, c_k), where c_i is a set of n_i objects in cluster i, and k is given. *k*-means constructs clusters with the distance between the row vector for any object and the centroid vector of its respective cluster is at least as small as the distance to the centroids of the remaining clusters [27]. Distance and similarity between the two items A and B are convertible to each other as follows:

$$Dis(A,B) = 1 - Sim(A,B) \quad (5)$$

In our experiment, we used Jaccard similarity coefficient [24], which compares similarity and distance of the items. Jaccard similarity coefficient measures similarity between finite sets as presented, given A and B are two sets in U:

$$Sim(A,B) = \frac{|A \cap B|}{|A \cup B|} \text{ where } 0 \leq Sim(A,B) \leq 1 \quad (6)$$

©2015 Journal of Emerging Technologies in Web Intelligence

However, in the news item, the item vector is binary. The binary version of Jaccard Similarity formula is placed to compute user similarity matrix and then cluster the users, by k-means. The binary version of Jaccard similarity is defined as follows:

$$B_{sij} = \frac{\sum_l (h_{i,l} \wedge h_{j,l})}{\sum_l (h_{i,l} \vee h_{j,l})} \quad (7)$$

Each $h \in H$ represents a click history on a news item which the user has clicked on in click history *H*. The similarity between two users u_i and u_j is defined as the overlap between their click histories: h_i and h_j, given based on the above formula. This means the users are classified into k groups based on the highest similarity ratio. By considering the neighbors of an active user *U*, we can weigh the candidate news to select a news set and recommend it to the active user *U*.

We evaluate the performance of the recommendation method based on quality and accuracy metrics [28]. The dataset is divided into a training set (70% of dataset) and a test set (30% of dataset) in the offline manner. We report the precision, recall and F1-score as follows:

- Precision is one of the most common metrics in assessing prediction accuracy in recommendation systems. Precision is the probability that a recommended item fits the user's interests and preferences. Precision considers the number of relevant items ranked in the top-N places [29]. In our study, the relevant items are the news items that the user clicked. If all the recommended news items fit the user interests, e.g., the precision scores the 1.0. Fig. 2 determines how precision is computed based on equation (8).

$$Precision = \frac{\{read\ news\} \cap \{recommended\ news\}}{\{recommended\ news\}} \quad (8)$$

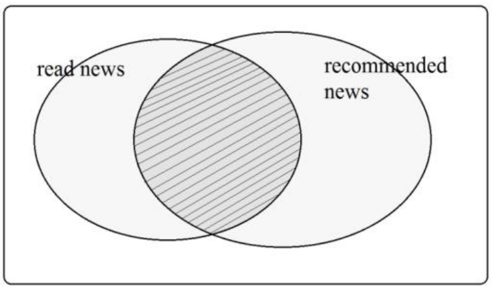


Figure 2. Subscription of {read news} and {recommended news} to compute precision and recall

- Recall is another metric to measure prediction quantity in recommendation systems. It shows the proportion of the selected items to all relevant items. Precision and recall can be computed for each user in each recommendation. In recommendation systems, a perfect recall scores 1.0 that determines excellent items were recommended to the user. A higher precision value shows more accuracy in prediction rather than recall. Equation (9) determines how to compute recall.

$$Recall = \frac{\{read\ news\} \cap \{recommended\ news\}}{\{read\ news\}} \quad (9)$$

- Precision and recall are related inversely. In most cases, increasing the size of the recommendation set will decrease precision and increase recall [30].
- F1-score or F1-measure is a harmonic mean of precision and recall and measures the accuracy of predictions in recommender systems. In computing the F1-score, both values of precision and recall are considered as follows:

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (10)$$

E. *News Recommendations*

Finally, the recommendation system will generate recommended news set to an active user *U* based on the reading behavior of neighbore users and comparing his/her *HotnessRate* with news items' *Hotness*. The selected news set is prioritized and adjusted by this comparison. Then, the top of this weighted set will be limited to the coefficient of *ReadingRate*.

VI. Results and Discussions

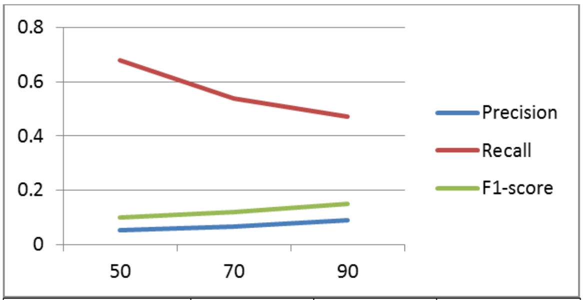
In this research, k-means clustering algorithm is used to cluster similar users into the same groups. The active user’s interest is predicted by exploring similar users' behaviors. To test the effect of our proposed news attributes, three experiments were done: Typical *k*-means clustering, *k*-means clustering with *ReadingRate*, and *k*-means clustering with user's *ReadingRate*, news *Hotness* and user *HotnessRate*. We then compared the precision, recall and F1-score for all three experiments under the same experimental environment.

- Experiment 1: Typical *k*-means in Neighborhood-based CF [1, 8, 27].
- Experiment 2: *k*-means with *ReadingRate*.
- Experiment 3: *k*-means with *ReadingRate, Hotness* and *HotnessRate*.

Each experiment was run separately twelve times, by considering different numbers of clusters in *k*-means to find the higher accuracy in recommendation. The result of 50, 70 and 90 of clusters are reported as below. In the experiments precision, recall and F1-score were computed as the accuracy metrics.

Experiment 1: Typical *k*-means in Neighborhood-based CF

In this experiment, the typical *k*-means is performed. A member of each cluster is selected randomly, and then the other members are compared based on their similarities. In each cluster a user is selected randomly. Recommended set to this user could be built based on the other members in the same cluster, by integrating the neighbors’ read news set. Comparison between the user’s real reading behavior and the recommended set determines the recommendation's accuracy. Fig. 3 shows the accuracy of this experiment:



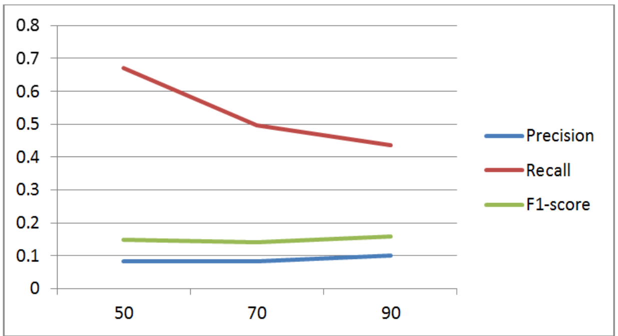
No. of Clusters	50	70	90
Precision	0.053623	0.066164	0.088624
Recall	0.680353	0.537772	0.470085
F1-score	0.099412	0.117831	0.149132

Figure 3. Precision, recall and F1-score in experiment1

The accuracy metrics in Experiment1 include precision, recall and F1-score. In this experiment, all the selected news articles are recommended. Because of the high number of recommended news items, recall is high while precision and F1-scores are very low.

Experiment 2: k-means with *ReadingRate*

In Experiment 2, the *k*-means clustering is run similar to Experiment 1. However, the recommendation of the news set the limited to the *ReadingRate*. The recommendation works like a typical *k*-means with the difference that the number of the recommended news items is limited to the coefficient of the *ReadingRate*. This means the recommendations are more tailored to the user's historical reading behavior, whereby the system recommends a news set to the active user based on his/her historical behavior in news reading. Fig. 4 indicates the precision, recall and F1-score:



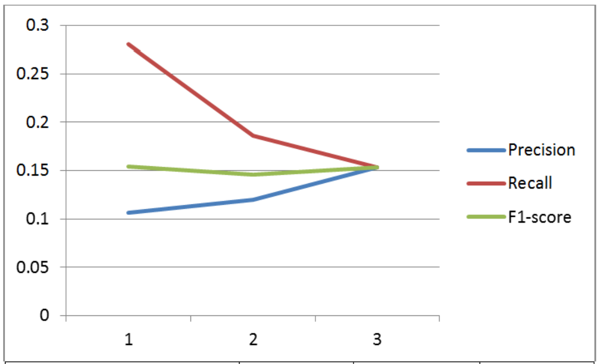
No. of Clusters	50	70	90
Precision	0.084118	0.082929	0.10205
Recall	0.67182	0.496114	0.43565
F1-score	0.149513	0.14204	0.15856
Improvement in F1-score	50.39%	20.54%	6.32%

Figure 4. Precision, recall and F1-score in Experiment 2

In Experiment 2, improvement in recall and F1-score is impressive. In comparison with Experiment1, there was little reduction in recall because of the limited number of recommended news items.

Experiment 3: k-means with *ReadingRate*, *Hotness* and *HotnessRate*

Experiment 3 is similar to Experiment 2 with a little change: adding other proposed news attributes, which are *Hotness* and *HotnessRate*. In this experiment, the primitive recommended news set, which is gathered based on typical *k*-means, prioritized and adjusted based on news *Hotness* and user *HotnessRate*, while the high rated news set is limited by the coefficient of *ReadingRate*. The finalized news set is recommended to the active user. In Fig. 5, precision, recall, F1-score and the improvement of these results are compared with Experiment 1 as follows:



No. of Clusters	50	70	90
Precision	0.1063933	0.1197079	0.15353
Recall	0.280474	0.186232	0.153005
F1-score	0.154268	0.145737	0.153267
Improvement in F1-score	55.18%	23.68%	2.77%

Figure 5. Precision, recall and F1-score in experiment3

In addition to *ReadingRate*, news *Hotness* and user *HotnessRate* are considered, in Experiment 3. The prioritizing and adjusting of the news set is on the *Hotness* and *HotnessRate*, which provide a more accurate prediction of the news items to recommend.

A. *Accuracy Evaluation*

In order to verify the effectiveness of our proposed attributes in news selection, we report a detailed comparison between traditional k-means and ours. Also, we implemented a recommender system in the CF-based model. For each approach, we randomly selected 50 users to provide recommendation. We plot the recall and precision pair to each user. Fig. 6 shows the comparison results. From Fig. 6, it can be observed that besides the higher precision, the performance distribution in the proposed approach is more than the others. It demonstrates the sagacity of the proposed approach in news selection.

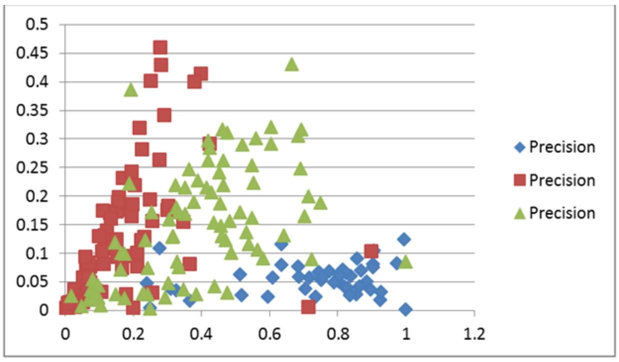


Figure 6. Recall-precision plots to different experiments. Remark: "◆" represents typical k-means results, "■" denotes k-means and ReadingRate results and "▲" represents the results of k-means and all proposed attributes.

In this experiment, all the users are equally treated as the experimental subject. Here, the comparison base includes three approaches. In Fig. 6, "♦" represents typical k-means results, "■" denotes *k*-means and *ReadingRate* results and "▲" represents the results of k-means of all proposed attributes. From the comparison, we observe that our approach achieves a reasonable recommendation result. The reason is that besides considering similar news access behavior and the users' historical behavior, we also measure news hotness and user's interest in reading hot news as hotnessrate.

VII. Conclusions and Future Works

Researches in recommendation systems continuously seek to enhance the accuracy of recommendations. To achieve this objective, this paper proposed new attributes in user profile and news metadata of news recommendations. The proposed attributes are tested in user clustering experiments using k-means to measure whether it would increase accuracy in news recommendation systems. The accuracy measurement metrics in these experiments include: precision, recall, and F1-score. By considering our new attributes including *ReadingRate*, *Hotness* and *HotnessRate*, recommendation becomes more accurate. Precision increases impressively with an 82.31% improvement overall and F1-score has a reasonable improvement with 23.72% increment.

In future, we hope to propose new algorithms in clustering including fuzzy methods in clustering and user profile construction. Proposing a hybrid personalized news recommendation system is our primary plan. Another direction for subsequent research is to evaluate these algorithms in an online environment. However, while offline studies consider accurate recommendation, online recommendation systems are often limited by time.

References

[1] L. Li, D. Wang, T. Li, D. Knox, and B. Padmanabhan, "Scene: A scalable two-stage personalized news recommendation system," in *Proc. the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2011, pp. 125-134.

[2] L. Li, D. D. Wang, S.-Z. Zhu, and T. Li, "Personalized news recommendation: A review and an experimental investigation," *Journal of Computer Science and Technology*, vol. 26, pp. 754-766, 2011.

[3] S. A. Munson, D. X. Zhou, and P. Resnick, "Sidelines: An algorithm for increasing diversity in news and opinion aggregators," in *Proc. ICWSM*, 2009.

[4] M. Tavakolifard, J. A. Gulla, K. C. Almeroth, J. E. Ingvaldesn, G. Nygreen, and E. Berg, "Tailored news in the palm of your hand: a multi-perspective transparent approach to news recommendation," in *Proc. the 22nd International Conference on World Wide Web Companion*, 2013, pp. 305-308.

[5] A. Said, J. Lin, A. Bellogín, and A. D. Vries, "A month in the life of a production news recommender system," in *Proc. the 2013 Workshop on Living Labs for Information Retrieval Evaluation*, 2013, pp. 7-10.

[6] J. Rao, A. Jia, Y. Feng, and D. Zhao, "Taxonomy based personalized news recommendation: Novelty and diversity," in *Proc. Web Information Systems Engineering–WISE 2013*, Springer, 2013, pp. 209-218.

[7] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry, "Using collaborative filtering to weave an information tapestry," *Communications of the ACM*, vol. 35, pp. 61-70, 1992.

[8] P. Melville and V. Sindhwani, "Recommender systems," in *Encyclopedia of Machine Learning*, Springer, 2010, pp. 829-838.

[9] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "GroupLens: An open architecture for collaborative filtering of netnews," in *Proc. the 1994 ACM Conference on Computer Supported Cooperative Work*, 1994, pp. 175-186.

[10] D. Maltz and K. Ehrlich, "Pointing the way: Active collaborative filtering," in *Proc. the SIGCHI Conference on Human Factors in Computing Systems*, 1995, pp. 202-209.

[11] M. Balabanović and Y. Shoham, "Fab: Content-based, collaborative recommendation," *Communications of the ACM*, vol. 40, pp. 66-72, 1997.

[12] L. Zheng, L. Li, W. Hong, and T. Li, "Penetrate: Personalized news recommendation using ensemble hierarchical clustering," *Expert Systems with Applications*, 2012.

[13] L. Li, L. Zheng, and T. Li, "Logo: A long-short user interest integration in personalized news recommendation," in *Proc. the fifth ACM Conference on Recommender Systems*, 2011, pp. 317-320.

[14] P. Cotter and B. Smyth, "Ptv: Intelligent personalised tv guides," in *AAAI/IAAI*, 2000, pp. 957-964.

[15] M. Claypool, A. Gokhale, T. Miranda, P. Murnikov, D. Netes, and M. Sartin, "Combining content-based and collaborative filters in an online newspaper," in *Proc. ACM SIGIR Workshop on Recommender Systems*, 1999.

[16] M. J. Pazzani, "A framework for collaborative, content-based and demographic filtering," *Artificial Intelligence Review*, vol. 13, pp. 393-408, 1999.

[17] Y. Xue, C. Zhang, C. Zhou, X. Lin, and Q. Li, "An effective news recommendation in social media based on users' preference," in *Proc. Education Technology and Training, International Workshop on Geoscience and Remote Sensing. ETT and GRS*, 2008, pp. 627-631.

[18] K. Lang, "Newsweeder: Learning to filter netnews," in *Proc. the Twelfth International Conference on Machine Learning*, 1995.

[19] J. Bennett and S. Lanning, "The netflix prize," in *Proc. KDD Cup and Workshop*, 2007, p. 35.

[20] A. S. Das, M. Datar, A. Garg, and S. Rajaram, "Google news personalization: Scalable online collaborative filtering," in *Proc. the 16th International Conference on World Wide Web*, 2007, pp. 271-280.

[21] S. Cleger-Tamayo, J. M. Fernández-Luna, and J. F. Huete, "Top-N news recommendations in digital newspapers," *Knowledge-Based Systems*, vol. 27, pp. 180-189, 2012.

[22] D. M. Desyaputri, A. Erwin, M. Galinium, and D. Nugrahadi, "News recommendation in Indonesian language based on user click behavior," in *Proc. 2013 International Conference on Information Technology and Electrical Engineering*, 2013, pp. 164-169.

[23] A. Singhal, "Modern information retrieval: A brief overview," *IEEE Data Eng. Bull.*, vol. 24, pp. 35-43, 2001.

[24] P. N. Tan, M. Steinbach, and V. Kumar, "Introduction to data mining: Pearson addison wesley," 2005.

[25] J. S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," in *Proc. the Fourteenth Conference on Uncertainty in Artificial Intelligence*, 1998, pp. 43-52.

[26] F. Abel, Q. Gao, G. J. Houben, and K. Tao, "Analyzing user modeling on twitter for personalized news recommendations," *User Modeling, Adaption and Personalization*, Springer, 2011, pp. 1-12.

[27] D. Steinley, "K−means clustering: A half−century synthesis," *British Journal of Mathematical and Statistical Psychology*, vol. 59, pp. 1-34, 2006.

[28] D. M. Powers, "Evaluation: From precision, recall and F-measure to ROC, informedness, markedness & correlation," *Journal of Machine Learning Technologies*, vol. 2, pp. 37-63, 2011.

[29] L. Lü, M. Medo, C. H. Yeung, Y. C. Zhang, Z. K. Zhang, and T. Zhou, "Recommender systems," *Physics Reports*, vol. 519, pp. 1-49, 2012.

[30] V. A. Rohani, Z. M. Kasirun, S. Kumar, and S. Shamshirband, "An effective recommender algorithm for cold-start problem in academic social networks," *Mathematical Problems in Engineering*, vol. 2014, 2014.