A Customized Non-Exclusive Clustering Algorithm for News Recommendation Systems

Asghar Darvishy[1], Hamidah Ibrahim [2], Fatimah Sidi [3], Aida Mustapha [4]
[1]*Islamic Azad University, Tehran, South Branch, Iran*
[2,3]*Universiti Putra Malaysia, Malaysia,*
[4]*Universiti Tun Hussein Onn, Malaysia*

[1]a_darvishy@azad.ac.ir; [2]hamidah.ibrahim@upm.edu.my; [3]fatimah@upm.edu.my; [4]aidam@uth.edu.my

ARTICLE INFO
Submission date: 1/10/2018
Acceptance date: 4/11/2018
Publication date: 11/1/2019

Abstract

Clustering is one of the main tasks in machine learning and data mining, and is being utilized in many applications including news recommendation systems. In this paper, we propose a new non-exclusive clustering algorithm named Ordered Clustering (OC) with the aim is to increase the accuracy of news recommendation for online users. The basis of OC is a new initialization technique that groups news items into clusters based on the highest similarities between news items to accommodate news nature in which a news item can belong to different categories. Hence, in OC, multiple membership in clusters is allowed. An experiment is carried out using a real dataset which is collected from the news websites. The experimental results demonstrated that the OC outperforms the *k*-means algorithm with respect to Precision, Recall, and *F*1-Score.

Keywords: clustering algorithm, non-exclusive clustering, news recommendation, similarity weight

Introduction

Clustering is one of the main tasks in machine learning and data mining that has been widely applied in the field of time series prediction, recommendation, and parameter estimation [1,2]. The items with the highest similarities are grouped together in the same cluster and those with considerable dissimilarity are grouped into different clusters [3,4]. In news recommendation systems, scalability is one of the issues that requires delicate algorithms to effectively deal with huge amount of news articles [5]. To address the scalability issue, several strategies can be used such as MinHash [6] and clustering algorithms. The most commonly used clustering algorithms in recommendation systems are hierarchical clustering [5,7] and *k*-means [8,9]. Nevertheless, these clustering algorithms do not take into consideration the news nature in clustering the news items. Consequently, a news item will only belong to a single cluster while in reality, a news item can be categorized in more than one news category. Moreover, it is obvious that users' interests are not limited to one news category.

Hence, the clustering algorithm to be employed in news clustering should be able to cluster news items without limiting their membership to a single cluster. It is also important to ensure that the clustering algorithm will not add any additional complexity. It is impossible to exclusive clustering approaches to cluster news items into different clusters. Any other way, fuzzy clustering approaches are very complicated without any considerable improvement [10].

In this paper an efficient clustering algorithm named Ordered Clustering (OC) is proposed for news clustering based on the news nature in which a news item can belong to more than one category with the aim to achieve accurate and diverse recommendations. Our algorithm considered the highest peer-to-peer item similarities and grouped the items into multiple clusters.

The rest of this paper is organized as follows. The related works are presented in the following section. This is then followed by description of the proposed clustering algorithm. The experimental evaluations are then presented which is followed by a summary of this research work.

Related Works

Scalability is one of the issues in news recommendation that requires effective algorithms to deal with large news corpus. One of the common strategies used for solving scalability is clustering. In news recommendation systems, news retrieval is performed based on the user's access pattern in news reading and the news content is compared to the users' read news contents. Selecting a suitable clustering algorithm is essential for achieving reasonable results. Some extensively utilized clustering algorithms in the news recommendation systems are reviewed as follows.

Locality Sensitive Hashing (LSH)

The Locality Sensitive Hashing (LSH) technique [11] is introduced to answer the near-neighbor search problem. Since, many applications utilizing LSH have been found in numerous fields [12]. The main idea of the LSH technique is to use several hash functions to hash the data points. Thus, for each hash function, the probability of collision has to be higher for the items near to each other than for those that are far away from each other. Then, near neighbors could be determined by hashing the query point and stored by the elements retrieved from the buckets including that point. LSH schemes are identified to exist for the following similarity or dissimilarity (distance) measures: Jaccard's coefficient [13,14], Hamming norm [15], Earth Mover's Distance (EMD), and cosine distance [16].

Min-Hash (Min-wise Independent Permutations) is a LSH scheme first introduced by Cohen [14]. It is a probabilistic clustering method that places a couple of users in a cluster with a probability proportional to the overlap between the sets of the news items these users have accessed. A given user u_i is represented by a set of the news items that the u_i has read based on his/her click behavior. Click history c_{u_i} represents the u_i's click behavior. The similarity ratio $S(u_i, u_j)$ between the two users $u_i$and u_j is defined as the intersection between their news sets computed based on Jaccard's coefficient. Jaccard's coefficient is a value between 0 and 1. The distance function is defined as $D(u_i, u_j) = 1 - S(u_i, u_j)$ [16]. Min-Hash uses a simple pruning technique. The users, who have read at least one news without reducing the number of candidates to a manageable number owing to the presence of popular news stories, are realized via the hash table.

Probabilistic Latent Semantic Indexing (PLSI)

To conduct a collaborative filtering, Probabilistic Latent Semantic Indexing (PLSI) models were developed by Hofmann [17]. Accordingly, modeling of the news items ($n \in N$) and users ($u \in U$) as random variables is done by taking their values from the spaces of all possible news and users. The joint distribution of the news and users is modeled to learn the relationship between the news and users. To find this relationship, a hidden variable Z with the values derived from $z \in Z$ is presented while $\|Z\| = L$. L represents the news categories and user communities. This model can be formally written as a mixture model proposed in Equation (1):

$$p(n|u;\theta) = \sum p(z|u)p(n|z) \quad (1)$$

The Conditional Probability Distributions (CPDs) of $p(n|z)$ and $p(z|u)$ are displayed by parameter θ, through which the model can be completely specified. The model mainly introduces the latent variable Z leading to the conditional independence of users and items. In this generative model, state z of the latent variable Z is selected for u as a random user with regard to CPD $p(z|u)$. Then, the sampling of the item s is followed based on z selected from CPD $p(n|z)$.

Hierarchical Clustering

A hierarchical clustering algorithm [18] partitions data items into a tree of clusters. Hierarchical clustering methods are categorized as either *divisive* or *agglomerative*, it depends on whether the hierarchical decomposition is planned in a splitting (top-down) or merging (bottom-up). Hierarchical clustering algorithm suffers from its inability to accomplish adjustment once a split or merge decision has been performed. Because of it, if a particular split or merge decision later turns out to have been a poor option, the method is not able to back down and correct it. Latest research studies have accentuated the integration of hierarchical agglomeration with iterative relocation methods.

K-means

The *k*-means algorithm [18] takes *k*, as a input parameter, and clusters a set of *n* data items into *k* clusters so that the intra-cluster similarity result is high but the resulting inter-cluster similarity is low. Cluster similarity is computed by consdering the *mean* value of the items in a cluster, tht is viewed as the cluster's *center* or *centroid of gravity*. The *k*-means algorithm performs as follows. Firstly, it arbitrarily selects *k* of the data items, each of which primarily represents a cluster center or mean. To each remaining data item, an item is assigned to the cluster to which has higest similarity, based on the distance between the cluster center and the data item. It then measures the new centroid for each cluster. This process repeats until the criterion function converges. Commonly, the square-error criterion is utilized, defined as follows.

$$E = \sum_{i=2}^{k} \sum_{p \in C_i} |p - m_i|^2 \quad (2)$$

where E is the summation of the square error for all data items in the dataset; m_i is the mean of cluster C_i; and p is the point in space which represents a given data item (both m_i and p are multidimensional). Namely, for each data item in any cluster, the distance from the data item to its cluster mean is squared, and the distances are summed. This measure attempts to generate the resulting k clusters as separate and as compact as possible.

Application of Clustering Algorithm in News Recommendation System

PENERATE [7] used a group-based hierarchical clustering method. This approach firstly separates user items into different sets based on their historical behaviors and each user item might be allocated to a number of sets. In SCENE [5], LSH [15] and hierarchical clustering are integrated to address the scalability issue of news recommendation. Initially, the recently published news items are partitioned into small sets based on the news content using the LSH while a 2-layer hierarchical clustering is employed in the next step. The leaf nodes indicate the sets accompanied by their topic distributions and the inner nodes hold a pair of news sets representing more common news topics. Google News is a Collaborative Filtering (CF) based on the personalized news recommendation system [19]. News recommendation is generated by using 3 approaches, namely: Min-Hash clustering, PLSI, and co-visitation counts of the news items. CCNS is a vertical news recommendation system that focuses on helping users to find their preferred news in a specific field and utilizes adjusted *k*-means for clustering users [8]. Table 1 presents a brief comparison among the aforementioned clustering approaches.

Table 1. Comparison of Approaches Based on Common Methods (CF, Content-Based (CB) and Hybrid).

Approach Name		Methods			Clustering Method
		CF	CB	Hybrid	
1	PENETRATE	-	-	√	Group-based hierarchical clustering
2	SCENE	-	-	√	LSH and hierarchical clustering
3	Google News	√	-	-	Min-Hash clustering
4	CCNS	-	-	√	Adjusted *k*-means clustering

The Proposed Algorithm

In this paper, a new non-exclusive clustering algorithm is designed that is called Ordered Clustering (OC). To examine OC a three phases approach is designed as shown in Figure 1. These phases are historical data conversion into a User Click Behavior (UCB) Matrix, Similarity Matrix (SM) constructoin, and user clustering.

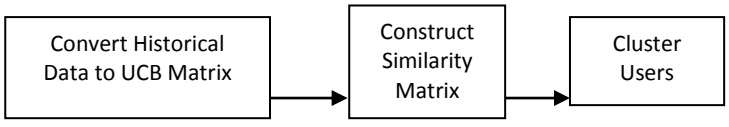

Figure 1. The Phases of the Proposed Approach.

Formation of User Click Behavior Matrix

This phase intends to convert the user click behavior (UCB) into a binary matrix. Users' historical behaviors are stored in a structured database which determines user u_i clicked on news item n_j at *read-time* time. This data can be shown as a triple $<$

$u_i, n_j, read - time >$, where u_i denotes the *ith* user, n_j represents the *jth* news item, and *read-time* stands for the time that the user accessed the news item. The entry of *UCB* is 1 if user u_i has accessed the news item n_j and 0 otherwise. Table 2 presents an instance of a UCB binary matrix.

Construct Similarity Matrix

The main goal of this phase is to calculate the similarities between users based on their historical reading behaviors. By calculating peer-to-peer similarities between the users, a Similarity Matrix (*SM*) is constructed. Binary Jaccard's similarity measure is used to calculate the similarities between the users [19]. Each user's reading behavior can be determined as a bit string. For example, based on Table 2, the bit string of the reading behavior of user u_1 (the first row of *UCB* matrix) is (10010110101001110011), where "1" and "0" denote read and unread, respectively.

Table 2. User Click Behavior Matrix on News Reading.

User / New	n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8	n_9	n_{10}	n_{11}	n_{12}	n_{13}	n_{14}	n_{15}	n_{16}	n_{17}	n_{18}	n_{19}	n_{20}
u_1	1	0	0	1	0	1	1	0	1	0	1	0	0	1	1	1	0	0	1	1
u_2	1	1	1	0	0	0	1	0	1	1	1	0	1	0	0	0	0	1	0	1
u_3	1	1	1	1	1	1	0	0	0	0	0	1	0	0	0	0	1	0	1	1
u_4	0	0	1	1	0	1	0	1	0	0	1	1	0	0	0	0	1	1	1	1
u_5	0	1	0	0	1	0	0	0	0	0	1	0	1	1	1	1	1	0	1	1
u_6	0	1	0	0	1	0	1	0	1	0	1	0	0	1	1	0	1	0	1	1
u_7	1	1	1	1	1	0	0	1	0	1	1	1	0	1	1	1	1	1	0	0
u_8	1	1	1	1	1	0	0	0	0	1	0	1	1	1	1	1	1	0	0	0
u_9	0	1	1	0	1	1	0	0	1	1	1	0	1	0	0	0	0	1	0	1
u_{10}	0	0	1	1	0	1	1	0	0	1	0	1	1	0	1	1	0	1	0	0
u_{11}	0	1	0	0	1	0	0	1	1	0	0	0	1	0	1	1	1	0	1	1
u_{12}	0	0	1	1	0	0	1	0	1	0	1	0	0	1	0	1	1	1	1	0
u_{13}	1	1	1	1	1	1	0	1	0	0	0	1	0	0	0	0	1	0	1	1
u_{14}	0	0	1	1	0	0	0	1	1	1	0	0	1	0	1	1	0	0	1	1
u_{15}	1	0	1	1	0	1	0	1	0	0	1	1	0	0	1	1	1	0	0	0
u_{16}	0	0	1	1	0	0	1	1	0	1	0	1	1	0	0	1	0	1	0	1
u_{17}	0	1	1	0	1	0	0	1	1	1	1	1	0	0	1	0	1	0	0	1
u_{18}	0	0	1	0	1	1	0	1	0	1	0	1	0	1	1	1	1	0	0	0
u_{19}	1	1	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1	1	0
u_{20}	1	0	1	0	0	1	0	0	1	1	0	0	1	1	0	0	1	0	1	1

Ordered Clustering Algorithm

The purpose of this phase is to cluster users based on the Ordered Clustering (OC) algorithm. OC algorithm includes new features and could not be classified into exclusive or fuzzy clustering classifications. It is generated based on the news nature and user reading behavior in news reading. Each user may be interested in a variety of news categories and a news article could be accessed by various users with different behaviors and preferences. In OC algorithm, multiple memberships are allowed with no membership weights or values and hence called a non-exclusive clustering. For instance, a user may be interested to read both sport news and economic news. The

sport news may be read by several users. In this way, a news item should be recommended to several users, and a user may be categorized into several groups of news.

The objective of clustering is to group the dataset D consisting of d items into k clusters. In OC, the number of clusters is determined during the execution of the clustering algorithm. A multiple binary cluster of D can be defined as a family of subsets $\{C_i | \ 1 \leq i \leq k\} \subset P(D)$ ($P(D)$ is the power set of D) with the below properties:

$$\bigcup_{i=1}^{k} C_i = D, \tag{3}$$

$$\exists \ C_i, C_j, \qquad C_i \cap C_j \neq \emptyset, 1 \leq i \neq j \leq k, \tag{4}$$

$$\emptyset \subset C_i \subset D, 1 \leq i \leq k \, . \tag{5}$$

Equation (3) means that the union of all the subsets C_i contains all the data in D. The subsets can be added, as expressed by Equation (4), and none of the subsets is empty or contains all the data in D as presented in Equation (5). In terms of membership functions, a cluster can be expediently represented by the cluster matrix $CM = [\mu_{ic}]_{k \times d}$. The *ith* row of the CM matrix includes values of the membership function μ_i of the *ith* subset C_i of D. It follows from equations 3, 4, and 5 that the elements of CM must satisfy the below conditions:

$$\mu_{ic} \in \{0, 1\}, \qquad 1 \leq i \leq k, \qquad 1 \leq c \leq d \tag{6}$$

$$\mu_{1c} \, OR \, \mu_{2c} \, OR \, \dots OR \, \mu_{kc} = 1, \qquad 1 \leq c \leq d \tag{7}$$

$$0 < \sum_{i=1}^{k} \mu_{ik} < d, \qquad 1 \leq i \leq k \tag{8}$$

Ordered Clustering algorithm selects a pair of users with the highest similarity ratio in the Similarity Matrix and groups these users into the same cluster. This process is repeated with the next highest similarity ratio. This means that the users in a cluster is ordered in descending order based on the similarity ratios between the users. Consequently, given a user u_j of cluster C_i, the left-hand side neighbors of u_j is said to be more similar than the right-hand side neighbors of u_j. For example, refer to Figure 2 which shows a cluster C_i with l members. To the given user u_j the left-hand side neighbors u_x, u_y, ..., u_{j-1} are more similar compared to the u_j's right-hand side neighbors u_{j+1}, ..., u_l as the similarity ratio values on the left-hand side of u_j are greater than those on the right-hand side.

u_x	u_y	u_z	...	u_{j-1}	u_j	u_{j+1}	...	u_l

Figure 2. An Array List of Cluster C_i Consisting of Users u_x to u_l

The similarities between the users are shown as follows:

$$Sim(u_x, u_y) > \ Sim(u_y, u_z) \ and \ \dots and \ Sim(u_{j-1}, u_j) > \ Sim(u_j, u_{j+1}) \ and \ \dots and \ Sim(u_{l-2}, u_{l-1}) > \ Sim(u_{l-1}, u_l)$$

Ordered Clustering Algorithm

Figure 3 illustrates the Ordered Clustering algorithm. Step 4 repeats and ensures that eventually all users of *U* are considered and belong to at least a cluster. In Step 6, the highest similarity ratio in *SM* is identified. This value indicated by SM_{ij} is assigned to a variable called Max (Step 7). In Step 8, the entry SM_{ij} of SM is set to 0 to avoid it from being chosen again in the next iteration. In Step 9, the existing clusters are checked and if user u_i is a member of an existing cluster say c_l then user u_j is inserted into the same cluster c_l of user u_i (Step 13). However, if u_i is not found in the cluster c_l but u_j is found to be a member of cluster c_l then u_i is inserted into the cluster c_l (Step 20). Yet, if both users u_i and u_j do not belong to any clusters, then a new cluster c_k is created and both u_i and u_j are inserted into the cluster c_k (Step 28). The algorithm is terminated when all users are members of at least one cluster, i.e. $U = \emptyset$.

OC Algorithm
Input: $U = \{u_1, u_2, \dots, u_n\}$ as Set of Users, Similarity Matrix *SM*
Output: $C = \{c_1, c_2, \dots, c_k\}$ as Set of Clusters

```
1. BEGIN
2.     k = 0
3.     Found = F
4.     WHILE U ≠ ∅ DO
5.       BEGIN
6.         Find the maximum value in SM
7.         Max = SM_ij
8.         SM_ij = 0
9.         FOR each cluster c_l in C AND Found ≠ T DO
10.          BEGIN
11.            IF u_i is a member of the cluster c_l THEN
12.              BEGIN
13.                Insert u_j into c_l
14.                U = U − u_j
15.                Found = T
16.              END
17.            ELSE
18.              IF u_j is a member of the cluster c_l THEN
19.                BEGIN
20.                  Insert u_i into c_l
21.                  U = U − u_i
22.                  Found = T
23.                END
24.                ELSE
25.                  BEGIN
26.                    k = k + 1
27.                    Create a new cluster c_k
28.                    Insert u_i and u_j into c_k
29.                    U = U − u_i
30.                    U = U − u_j
31.                    Found = T
```

```
32.                    END
33.              END
34.            Found = F
35.         END
36. END
```

Figure 3. The Ordered Clustering Algorithm.

By running the algorithm on the *CM* given in Figure 4, the clusters created are as shown in Table 4.

$$CM = \begin{bmatrix} 00110000000010100100 \\ 00010011000000101100 \\ 10001100001101010001 \\ 00000000010001010000 \\ 01000000100000000000 \\ 10000100000100000011 \end{bmatrix}$$

Figure 4. Cluster Matrix.

Table 4. The Clustering Results Based On Oc.

Cluster No	Members
c_1	$\{u_3, u_{13}, u_4, u_{15}, u_{18}\}$
c_2	$\{u_7, u_8, u_{15}, u_{18}, u_{17}, u_4\}$
c_3	$\{u_5, u_6, u_{11}, u_{14}, u_{16}, u_1, u_{12}, u_{20}\}$
c_4	$\{u_{10}, u_{16}, u_{14}\}$
c_5	$\{u_2, u_9\}$
c_6	$\{u_1, u_{19}, u_6, u_{12}, u_{20}\}$

Experiment Environment

In this section, an experimental evaluation is provided to show how the proposed clustering algorithm differed from *k*-means for news recommendation system. First, the real dataset used in the experiments are introduced. Then, the results of the *k*-means method and the proposed OC algorithm are presented.

News Dataset

The dataset was gathered from Twitter information streams that was crawled over a period of more than 60 days, from October 2010 to January 2011. In this dataset, the streamed news articles were accessed by more than 20,000 users with the more than 10 million times. To relate the tweets with the news articles, more than 60 well-known news agencies, such as *New York Times*, *CNN, and BBC* were monitored. The tweets create a total of 77,544 news articles [20].

We are interested in analyzing the clustering algorithms and their accuracy in prediction. Thus, we generated a sample of 1,009 users, who read at least 4 news items per day. This sample dataset contained 1,161,798 news-reading records. From our sample, 38,737 news items were derived. Table 5 presents the characteristics and the descriptive statistics of the dataset.

Table 5. Characteristics of The Dataset.

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

Evaluation of Clustering Effectiveness

The effectiveness of the proposed clustering algorithm, OC, is compared to the result of the *k*-means algorithm. The detailed experiments are illustrated as follows.

Both clustering algorithms were implemented using Java on a Pentium V PC with MS-Windows 8.0 and 4 GB of RAM. Each experiment was run 10 times. The recall, precision, and *F*1-scores have been measured for OC and *k*-means algorithms. Each user is assumed as an entry in the clustering.

Figure 5 depicts the results of the evaluation. It can be concluded from Figure 5 that the proposed OC algorithm significantly outperforms *k*-means in terms of accuracy with 20.6% improvement in Recall, 51.5% improvement in Precision, and 46% improvement in *F*1-score.

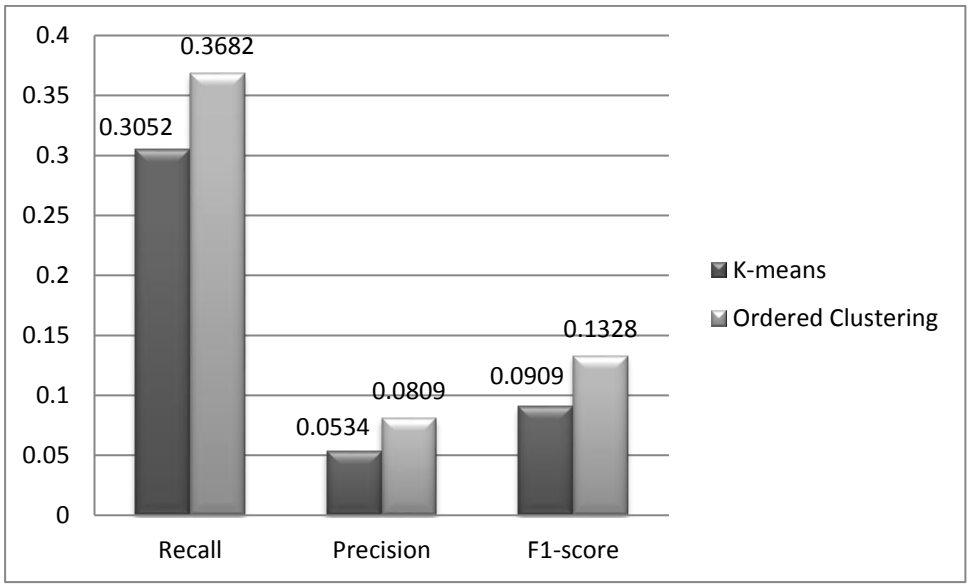

Figure 5. Accuracy Metrics in Different Clustering Algorithms.

To corroborate the effectiveness of our proposed clustering algorithm, a detailed comparison was also provided between our method and the general recommendation method that utilized *k*-means based on pairwise similarities. For each approach, we arbitrarily selected 100 users to provide recommendation for them. We then plotted the precision and recall of the news items recommended to each user. Figure 6 presents the results of recall and precision of this experiment. In Figure 6, the row denotes recall of the algorithms, "♦" shows the precision values for *k*-means, and "▲" demonstrates the Ordered Clustering precision.

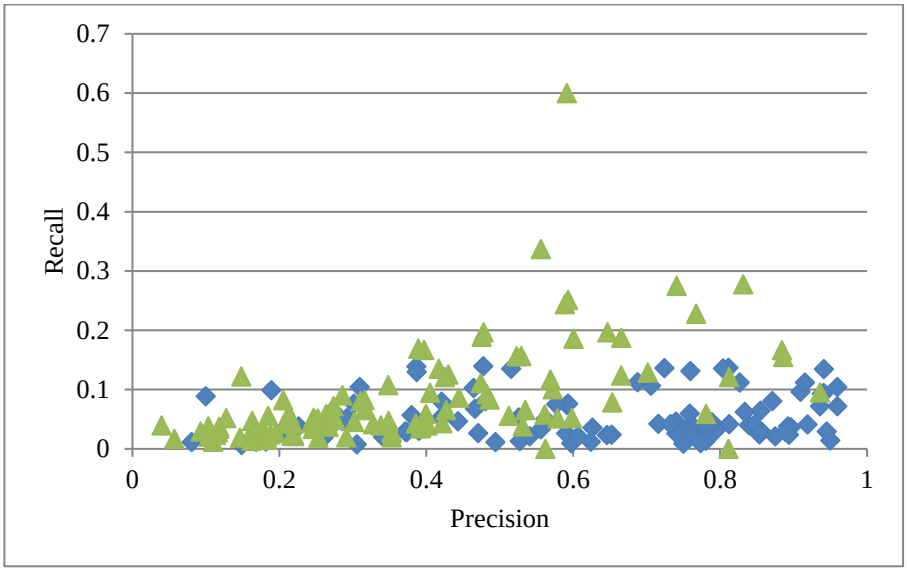

Figure 6. Recall-precision plot for different user clustering algorithms; remarks: "▲" represents news recommendation results using ordered clustering and "♦" denotes news recommendation results obtained from *k*-means clustering.

It can be concluded from Figure 6 that besides obtaining a higher recall and precision in the OC algorithm, the performance distribution of OC algorithm is more dense than that of the *k*-means algorithm. This assures the efficiency of OC algorithm for the news recommendation system. In the accomplished experiments of this study, all the users were equally treated as the experimental subjects. Actually, users with different news reading behaviors, such as various daily reading frequencies, might have different patterns of news topic preferences and then, the dynamic interests in the news items could vary very much.

Conclusion Remark and Future Research Direction

In this paper, we proposed a new non-exclusive clustering algorithm named Ordered Clustering (OC) that is dedicated to news recommendation. In this algorithm the highest peer-to-peer item similarities is considered and these items are grouped into multiple clusters. OC is a qualified and specified clustering algorithm in news recommendation based on the news nature. The results indicated that multiple memberships in the clusters contribute to the accuracy enhancement.

The experimental results and the higher *F*1-score demonstrated that OC is more efficient for clustering news items and generating accurate recommendations than the *k*-means. Our evaluation is done in an offline manner with real data. Nevertheless, a better evaluation can be performed in an online recommendation system. For future work, to achieve more precise results, assessment should be done in a real online environment.

Conflict of Interests.

There are non-conflicts of interest .

References

[1] P. Berkhin, "A survey of clustering data mining techniques," *Proceeding of the Grouping multidimensional data* pp. 25-71, 2006.

[2] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "From data mining to knowledge discovery in databases," *AI magazine*, vol. 17, no. 3, pp. 37-54, 1996.

[3] T. Hastie, R. Tibshirani, and J. Friedman, "Unsupervised learning," in *The elements of statistical learning*, Springer, 2009, pp. 485–585.

[4] W. M. Rand, "Objective criteria for the evaluation of clustering methods," *J. Am. Stat. Assoc.*, vol. 66, no. 336, pp. 846–850, 1971.

[5] L. Li, D. Wang, T. Li, D. Knox, and B. Padmanabhan, "SCENE: a scalable two-stage personalized news recommendation system," In Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2011, pp. 125-134. ACM.

[6] J. Liu, P. Dolan, and E. R. Pedersen, "Personalized news recommendation based on click behaviour," In Proceedings of the 15th International Conference on Intelligent User Interfaces, 2010, pp. 31-40. ACM.

[7] L. Zheng, L. Li, W. Hong and T. Li, "PENETRATE: Personalized news recommendation using ensemble hierarchical clustering," *Journal of Expert Systems with Applications*, vol. 40, no. 6, pp. 2127-2136, 2013.

[8] S. Jiang, and W. Hong, "A vertical news recommendation system: CCNS—An example from Chinese campus news reading system," In International Conference on Computer Science & Education (ICCSE), 2014, pp. 1105-1114. IEEE.

[9] L. Li, W. Chu, J. Langford, and R. E. Schapire, "A contextual-bandit approach to personalized news article recommendation," In Proceedings of the 19th International Conference on World Wide Web, 2010, pp. 661-670. ACM.

[10] Z Lassoued, and K Abderrahim. "PWARX Model Identification Based on Clustering Approach." In *Complex System Modelling and Control Through Intelligent Soft Computations*, pp. 165-193, 2015. Springer, Cham.

[11] A. Andoni, and P. Indyk, "Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions," In Proceeding of the 47th Anual IEEE Symposium on Foundations of Computer Science, 2006, pp. 459-468. IEEE.

[12] M. Muja, and D. G. Lowe, "Scalable nearest neighbor algorithms for high dimensional data," *Journal of IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 11, pp. 2227-2240, 2014.

[13] A. Z. Broder, "On the resemblance and containment of documents," In Proceeding of the compression and complexity of sequences, 1997, pp. 21-29. IEEE.

[14] E. Cohen, "Size-estimation framework with applications to transitive closure and reachability," *Journal of Computer and System Sciences*, vol. 55, no.3, pp. 441-453, 1997.

[15] A. Gionis, P. Indyk, and R. Motwani, "Similarity search in high dimensions via hashing," In Proceeding of the Very Large Data Base, vol. 99, no. 6, pp. 518-529, 1999.

[16] M. S. Charikar, "Similarity estimation techniques from rounding algoithms," In Proceedings of the Thiry-fourth annual ACM Symposium on Theory of Computing, 2002, pp. 380-388. ACM.

[17] T. Hofmann, "Latent semantic models for collaborative filtering," *Journal of ACM Transactions on Information Systems (TOIS)*, vol. 22, no. 1, pp. 89-115, 2004.

[18] J. Han, J. Pei, and M. Kamber, *Data mining: concepts and techniques*. Elsevier, 2011.

[19] A. S. Das, M. Datar, A. Garg, and S. Rajaram, "Google news personalization: scalable online collaborative filtering," In Proceedings of the 16th International Conference on World Wide Web, 2007, pp. 271-280. ACM.

[20] F. Abel, Q. Gao, G. J. Houben, and K. Tao, "Analyzing user modeling on twitter for personalized news recommendations," In Proceeding of the International Conference on User Modeling, Adaptation, and Personalization, 2011, pp. 1-12. Springer.