A Collaborative Filtering Recommendation Methodology for Peer-to-Peer Systems

Hyea Kyeong Kim[1], Jae Kyeong Kim[1,⋆], and Yoon Ho Cho[2]

[1] School of Business Administration, KyungHee University,
1, Hoeki-dong, Dongdaemoon-gu, Seoul, 130-701, Korea,
Tel: +82-2-961-9355, Fax: +82-2-967-0788
{kimhk, jaek}@khu.ac.kr
[2] School of E-Business, KookMin University,
861-1 Jungnung, Sungbuk, Seoul, 136-702, Korea,
Tel: +82-2-910-4950, Fax: +82-2-910-4519
www4u@kookmin.ac.kr

Abstract. To deal with the image recommending problems in P2P systems, this paper proposes a PeerCF-CB (Peer oriented Collaborative Filtering recommendation methodology using Contents-Based filtering). PeerCF-CB uses recent ratings of peers to adopt a change in peer preferences, and searches for nearest peers with similar preference through peer-based local information only. The performance of PeerCF-CB is evaluated with real transaction data in S content provider. Our experimental result shows that PeerCF-CB offers not only remarkably higher quality of recommendations but also dramatically faster performance than the centralized collaborative filtering recommendation systems.

1 Introduction

According to a recent report, 93% of information produced worldwide is in digital form and the unique data added each year exceeds one exabyte, and more than 513 million people around the world are now connected to the global information resource [8]. However, many of those people have problems to search for digital contents they are most interested in. This trend calls for recommender systems with scalable searching capability. A recommender system is defined as a system that assists users in finding the items they would like to use. It has been used to help users search for products or multimedia contents in Web environment. One of the most successful recommendation techniques is Collaborative Filtering (CF), which has been widely used in a number of different applications [1], [2], [4], [5], [6], [7]. Collaborative filtering is an information filtering technique that depends on human beings' evaluations of items. It identifies users whose tastes are similar to those of a given user and it recommends items those users have liked in the past.

The peer-to-peer (P2P) systems are developed to facilitate direct communication or collaboration between two or more agents, such as personal computers or

⋆ Corresponding author.

K. Bauknecht et al. (Eds.): EC-Web 2005, LNCS 3590, pp. 98–107, 2005.

devices. P2P applications such as Napster and Gnutella are increasingly popular for file sharing through direct exchange. These applications offer the advantages of decentralization by distributing the storage capacity and load across a network of peers and scalability by enabling direct and real-time communication [10]. For example, to search for contents, the agent of peer broadcasts a search request to peers connected, and propagates the requests to their own peers and so on. An increasing number of P2P users and shared contents also raise a serious complexity for the users selecting their desired contents. Accordingly recommender systems in P2P systems are emerging as a successful solution to overcome these difficulties [1], [6], [7]. However, existing research and practice in recommender systems are mostly based on centralized client-server architecture.

In this paper, we propose an adaptive CF recommendation methodology in P2P systems, PeerCF-CB (Peer oriented Collaborative Filtering recommendation methodology using Contents-Based filtering), to deal with the problems we face in recommending images. Although CF has been used successfully as a recommendation technique for client-server architecture, it is necessary to adapt the CF methodology for recommending images in P2P systems. For such a purpose, Content-based image retrieval (CBIR) is employed, which performs similarity-based image retrieval using its visual features such as color, texture and shape [4], [9]. In CBIR, the peer describes visual characteristics of desired images using a query that is a set of example images. To learn about the peer's true intention, the peer's current preference on the presented images needs to be fed back so that CBIR can learn from this preference to retrieve images more similar to the one the peer really wants. This learning process is an essential mechanism for a faster search of desired images in PeerCF-CB.

PeerCF-CB essentially follows the ground principle of CF and CBIR techniques, while we suggest the following modification to be applied in the P2P systems; ***an event-driven recommendation*** - whenever a peer finds relevant contents, the contents are forwarded to other peers in real time, ***a recent rating-based filtering*** - recent observations can better represent the current peer's interests than the past observations, and ***a dynamic neighbor re-formation*** - to reflect the change in recent interests, neighbor peer set is frequently re-formed using peer-based local information only, which results in the performance improvement with much less computation time.

Several experiments are performed to compare the performance of PeerCF-CB with that of a centralized CF system using real transaction data in S content provider, and their results are discussed.

2 Peer Model

Peer network in P2P system consists of interconnected peers and they collaborate each other by exchanging preference information. It is assumed that peers have distinctive preference and they are willing to share what images they prefer. Each peer, named as a *host peer* participates in the peer network, and has an individual *peer model*. The peer model is composed of three parts, *host peer profile, neighbor peer set,* and *target peer set*. A host peer profile includes

information about what images a host peer prefers. Such information is used to find similar peers as neighbor peer set to receive recommendations. Target peer set is composed by requests of other peers, and a host peer forwards images to them as recommendations.

The success of recommendation depends to a large extent on the ability to represent the host peer's actual preference. Images saved on a host peer's computer include information about peer's preference on images. Therefore, saved images, called as a *preferred image set* are used to create host peer profile. Whenever a host peer h saves or deletes an image, the preferred image set is updated. Preferred image set, P^h consists of multiple images, and is defined as $\{q_1^h, q_2^h, ..., q_i^h, ...q_L^h\}$, which denotes that host peer h has L saved images on his/her personal computer. Each image is represented as collection of all possible visual features that describe its perceptual properties such as HSV (i.e. hue, saturation, and value of color) based color moment, shape and texture. $q_1^h$is composed of S-dimensional visual feature values, and defined as $\{q_{i1}^h, q_{i2}^h, ..., q_{is}^h, ...q_{iS}^h\}$ where S denotes the number of visual features, and q_{is}^h denotes sth feature value on image i of the host peer h. Each image in a preferred image set is represented as a point in the multidimensional space of those features.

Host peer h receives recommendations from its neighbor peers. Each peer estimates *neighbor similarity, NS(h,n)*, between host peer h and other peer n, to select neighbor peers, which have higher *NS(h,n)* than others. A neighbor peer set of h, $N^h$is defined as $\{n_1^h, n_2^h, ..., n_j^h, ...n_M^h\}$ where M is the predefined number of neighbor peers. Once a peer is selected as a neighbor peer, it is dynamically exchanged with a more similar peer in candidate neighbor set. In PeerCF-CB, the neighbor peer set of host peer h's most similar neighbor peer set is defined as a candidate neighbor set, CN^h, to limit the exploration boundary. When one of the candidate neighbor peers who has higher neighbor similarity than a neighbor peer is detected, the peer becomes a new neighbor of h.

Target peer set, T^h is a peer set which is recommended by host peer. T^h is defined as $\{t_1^h, t_2^h, ..., t_e^h, ...t_N^h\}$ where N is the predefined number of target peers. The target peer set is organized by the request of other peers with similar tastes.

3 Recommendation Procedure

PeerCF-CB consists of the following three cooperating distinct procedures, *an Event-driven recommendation procedure, a CBIR procedure, and a Neighbor reformation procedure.*

3.1 Event-Driven Recommendation Procedure

Event-driven recommendation is generated with a push way, which is that whenever a peer saves an image, the newly saved image is added to the preferred image set of the peer and forwarded to other peers in real time. The push way can

particularly emphasize the most recent ratings of neighbor peers, which leads to faster spread of newly obtained images. Host peer is allowed to push the newly saved image to only limited number of target peers, however recommendations of host peer can reach away beyond target peers.

3.2 CBIR Procedure

The pushed images from neighbor peers are accumulated in a queue of host peer. And the CBIR procedure selects top-k recommendation list among the images in the queue.

In CBIR procedure, a distance between each preferred image of a host peer and each image in a queue is calculated based on visual features and k images having the shortest distance are selected as top-k recommendation list. The pushed image set, X^h in the queue of h is defined as $\{x_1, x_2, ..., x_C\}$, where C is the maximum number of images in a queue. A queue keeps on maintaining recently received C images. $P^h$is used as a query for searching similar images. The query, which is internally represented as multiple query points, is continuously updated by adding the newly saved images to the query points in P^h. Since a query is allowed to have multiple query points, the distance function between an image x_i, and a query P^h aggregates multiple distance components from the image to related query points. We use the following aggregate distance function;

$$Dist(x, P^h) = \sqrt{\frac{L}{\sum_{i=1}^{g} 1/dist^2(x, q_i)}}, \tag{1}$$

where L is the number of query points in a query P^h, q_i is the ith query point of P^h, and $dist(x, q_i)$ is a distance function between an image x and a query point q_i. We derived the equation (1) from the FALCON's formula [11]. It treats an image with the shortest distance component to any one of query points as the image with the shortest aggregate distance. The $dist(x, q_i)$ in Equation (1) is defined as;

$$dist(x, q_i) = \sqrt{\sum_{s=1}^{S} w_s (x_s - q_{is})^2}, \tag{2}$$

where S is the number of dimensions of feature space, w_s is a weight of the sth dimension in the feature space, and x_s and q_{is} are coordinates of an image x and a query point q_i on the sth dimension, respectively. w_s is defined as $1/\sigma_s$ where σ_s is a standard deviation of coordinates of sth dimension of images. Note that σ_s is calculated using all images in P^h.

CBIR procedure generates top-k recommendation list for the host peer. The retrieved k images are presented to the peer and the peer skims through the list to see if there are any images of interest. Then, the peer may save desired images on the peer's computer.

3.3 Initial Neighbor Formation Procedure

Neighbor re-formation procedure decides whom to keep as neighbors in accordance with distance-based neighbor similarity. Before explaining neighbor reformation, this section explains about building an initial neighbor peer set to participate in the peer network. As an initial neighbor peer set, M nearest neighbor peers are generated based on the similarity between host peer and other peers using an average inter-cluster distance function[3]. To take an initial neighbor set for h, the agent calculates *Neighbor similarity, NS(h,n)*. Given P^h and P^n , *NS(h,n)* is defined as;

$$NS(h,n) = 1 - \left(\frac{1}{|P^h||P^n|} \sum_{q^h \in P^h} \sum_{p^n \in P^n} sim(q^h, q^n) \right), \tag{3}$$

$$where\ sim(q^h, q^n) = \sum_{s=1}^{S} |q_s^h - q_s^n|, \tag{4}$$

$|P^h|$ and $|P^n|$ are the size of the P^h and the P^n respectively, q^h and q^n are images in the P^h and the P^n respectively, and $sim(q^h, q^n)$ is a feature-based distance function between P^h and P^n. In equation (4), S is the number of dimensions of the feature space and q^h and q^n are coordinates of q_s^h and q_s^n on the sth dimension respectively.

Using the *NS(h,n)*, initial neighbor peer set is determined by comparison of the degree of similarity between saved image sets. Note that for a new peer without any saved image, an initial neighbor set is composed of peers having the most frequently saved images. As preferred image set is often updated by newly saved images, the similarity is also changed with the passage of time. Neighbor reformation procedure attempts to adapt the change in real time, i.e., dynamic neighbor reformation is occurred.

3.4 Neighbor Re-formation Procedure

The neighbor re-formation procedure is implemented with a learning algorithm to constitute better relevant neighbor peer set. In the procedure, each host peer decides whom to disconnect from neighbor peer set and whom to add to the neighbor peer set. The neighbor peers with consistently similar preference to a host peer are kept as neighbors. But when the preference of a neighbor peer becomes different from the host peer, the neighbor peer is disconnected from the neighbor peer set. For the replacement of a disconnected neighbor, PeerCF-CB makes the host peer explore the candidate neighbor set, CN^h . If a more similar peer is discovered among the CN^h than any n_i, the cn^h is included to the N^h and the n_i is discarded. This always leads N^h to be composed of peers with more similar preferences.

Figure 1 illustrates the neighbor re-formation procedure, where neighbor peer A is the most similar neighbor peer, and neighbor set of peer A are candidate neighbor peer set. If peer C has higher neighbor similarity than current neighbor

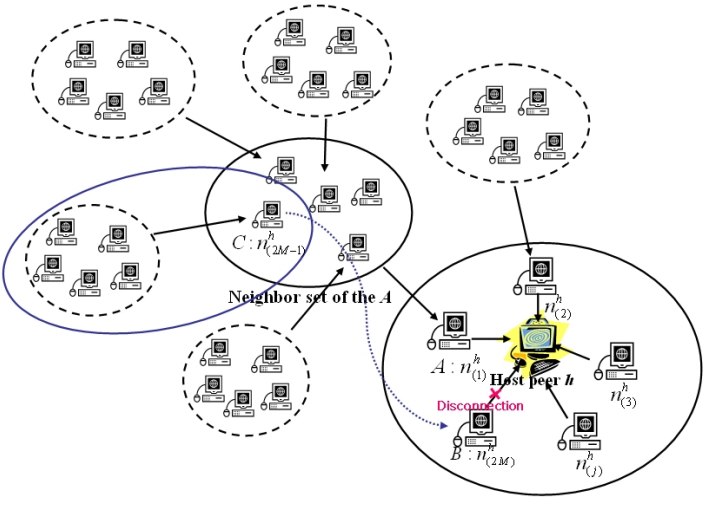

Fig. 1. Neighbor re-formation

peer B, the peer C becomes new neighbor peer of host peer h, while peer B is excluded from h's neighbor set. According to this mechanism, any neighbor peer of peer C may be also included in neighbor peer set of host peer h later time.

4 Experimental Evaluation

4.1 Experiment

The performance of PeerCF-CB is compared with that of two centralized benchmark recommender systems, *CentralizedCF-CB* and *CentralizedCF*. The *Centralized CF-CB* is similar to PeerCF-CB, but neighbor re-formation procedure of *CentralizedCF-CB* uses all peers' ratings. And CBIR procedure of *CentralizedCF-CB* is performed based on all past ratings of neighbor set. *CentralizedCF* is similar to *CentralizedCF-CB*, but it follows a pure CF principle. The *Centralized CF* adapts PLS(Purchase Likeliness Score) [4] to select top-k recommendation list instead of a CBIR procedure of *CentralizedCF-CB*.

The systems to perform our experiments were implemented using Visual Basic 6.0 and ADO components. MS-Access is used to store and process all the data necessary for our experiments. We run our experiments on Windows 2003 based PC with Intel Pentium IV processor having a speed 2.80 GHz and 1GB of RAM.

The comparative experiment is performed with real transaction data offering character images from S content provider, a leading Korean company, in mobile commerce. The data contain 8,776 images, 1,921 customers, and 55,321 transactions during the period between June 1, 2004 and August 31, 2004. The transaction data during the three months are divided into two sets, a training

set and a test set. Host peers are determined as the users who have purchased at least one image during the training period, and initial preferred image set of each host peer is generated from transaction records of the training period. Initial neighbor peer set is then formed based on the initial preferred image set. Each host peer receives recommendations from his/her neighbor peers at each connection date for the test period, and then we observe whether the recommended images match the real purchased images of each host peer or not.

HSV (Hue, Saturation, and Value of color) based color moment was selected as visual features characterizing images [4], [9]. For all pixels in images, we translated the values of three-color channels (i.e. RGB; red, green, and blue) into HSV values. Then, the mean, standard deviation and skewness for HSV values were calculated to represent images as vectors in nine dimensional feature spaces.

This research employs two metrics, ***hit ratio*** and ***response time*** for the evaluation of accuracy and performance of suggested recommendation methodology respectively. The hit ratio is defined as the ratio of hit set size to the test set size, where hit set size means the success number of recommendations, in our experiment, and test set size means the number of connections. The response time is defined as the amount of time required to generate recommendations for the test set.

4.2 Results and Discussion

This section presents experimental results performed by different parameter set, and the performance of PeerCF-CB is compared with those of CentralizedCF-CB and CentralizedCF.

Among different parameter set, the queue size of each peer and neighbor peer size are determined to be most important parameters impacting on the recom-

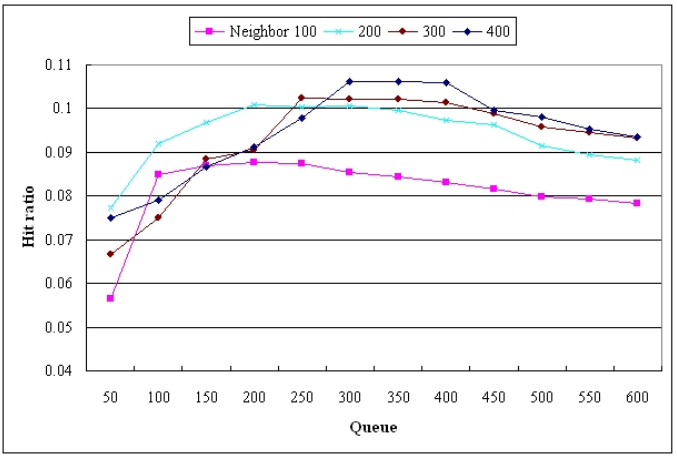

Fig. 2. Neighbor re-formation

mendation quality. Experiments are performed as we varied the queue size from 50 to 600 with an increment 50 at each neighbor peer size from 100 to 400. Figure 2 shows the results. From the results, we make an important observation over all neighbor peer sizes that the quality of recommendation improves as the queue size is increased. As the queue of each peer stores images recommended from its neighbors, recommendation based on large queue size will have a higher hit ratio especially in the domains of newly released images. But after a certain level the improvement slows down and eventually the recommendation quality becomes worse. This indicates that the excessive queue size may cause violation of reflecting the current preference, which leads to lower quality of recommendations. It confirmed that our recent rating-based filtering using queue is a reasonable suggestion to enhance the quality of recommendations.

To compare with the centralized benchmark systems, the experiments were carried out with varied number of neighbors at top-20 recommendation list, and computed the corresponding hit ratio and response time. Note that, to make the comparisons fair with the centralized benchmark systems, we also experimentally determined the optimal queue size of PeerCF-CB for each number of neighbors and tune the system to perform to its ideal level.

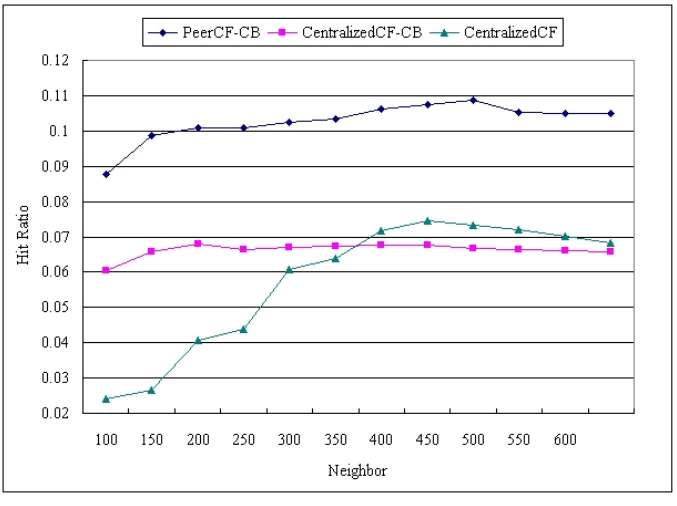

Fig. 3. Recommendation quality comparison

Figure 3 shows the sensitivity of the neighbor size from 100 to 650 with an increment 50. The recommendation quality of CentralizedCF improves as the neighbor size is increased, and after a certain point the improvement slows down and eventually the recommendation quality becomes flat. This result is similar to those of other CF recommender systems [2], [5]. On the other hand, the results of PeerCF-CB and CentralizedCF-CB do not have much variance over all neighbor size, similarly to other content-based filtering systems. Figure 3 also shows that

PeerCF-CB gains improvement of over 40% on the average from the results of centralized benchmark systems. From these results, we make an observation that PeerCF-CB works better than the centralized benchmark systems at all neighbor sizes and the recommendation quality of PeerCF-CB is robust.

Table 1. Performance comparison

	PeerCF-CB	CentralizedCF-CB	CentralizedCF
Response Time (sec)	0.0497	2.3620	2.0484

Table 1 shows the performance comparison represented by average response time. The PeerCF-CB is about 47 times and 41 times faster than CentralizedCF-CB and CentralizedCF respectively. When top-k list is generated, the benchmark centralized procedure uses all past ratings of its neighbors. But PeerCF-CB uses only recent ratings in the queue, which makes PeerCF-CB reflect the up-to-date preference of peers. This makes the PeerCF-CB offer not only higher accuracy but also dramatically faster performance improvement. Moreover, the benchmark centralized procedures perform neighbor re-formation using the preferred image sets of all peers, while PeerCF-CB uses the preferred image sets of neighbor peers and neighbor peer's neighbor peers only, which leads to the dramatically improvement of response time.

5 Conclusion

With the pervasive deployment of personal computers, P2P systems are receiving increasing attention in research and practice. In this paper, we suggest an adaptive CF-based recommendation methodology in P2P systems, PeerCF-CB, to deal with the problems we face in recommending multimedia contents. The characteristics of PeerCF-CB is as follows. First, using the queue of each peer, PeerCF-CB reflects the most current preference of peers, which results in significant quality improvement. Second, each Peer's event, such as saving an image, triggers recommendations with push way which leads to faster spread of new contents without centralized control. Finally, similar neighbor peers are dynamically determined based on peer-based local information only, which results in dramatically faster performance. Our experiment shows that PeerCF-CB offers not only remarkably higher quality of recommendations but also dramatically higher performance than the centralized benchmark procedures. These results give much implication to developing recommender systems in P2P systems, because the number of contents and that of peers grow very fast and personal computers have inherently a limited computing power only. PeerCF-CB is expected to be a realistic solution to the problems currently encountered in multimedia content recommendations in P2P systems.

PeerCF-CB has flexibility to share any multimedia contents, therefore we plan to extend PeerCF-CB to varying contents, such as music and text as a

further research area. Furthermore, it will be also a promising research area to develop a robuster recommender system with high degree of tolerance against errors and attack.

References

1. Canny, J.: Collaborative filtering with privacy, In Proc. of the IEEE Symposium on Re-search in Security and Privacy. (2002) 45-57
2. Cho, Y.H., Kim, J.K.: Application of Web usage mining and product taxonomy to collaborative recommendations in e-commerce. Expert Systems with Applications. 26 (2004) 233-246
3. Jiawei H., Micheline K.: Data Mining Concepts and Techniques. Morgan Kaufmann Pub-lishers. (2001)
4. Kim, C.Y., Lee, J.K., Cho, Y.H., and Kim, D.H.: VISCORS: a Visual Contents Recom-mender System on the Mobile Web, IEEE Intelligent Systems, Special issue on Mining the Web for Actionable Knowledge, Vol. 19, pp.32-39.
5. Kim, J.K., Cho, Y.H.: Using Web Usage Mining and SVD to Improve E-commerce Recom-mendation Quality. Lecture Notes in Computer Science. 2891 (2003) 86-97
6. Olsson, T.: Bootstrapping and Decentralizing Recommender Systems. Ph.D. Thesis, Dept. of Information Technology, Uppsala Univ. (2003)
7. Peng, H., Bo, X., Fan, Y., Ruimin, S.: A scalable P2P recommender system based on dis-tributed collaborative filtering. Expert Systems with Applications. 27 (2004) 203-210
8. Prete, C.D., McArthur, J.T., Villars, R.L., Nathan, R. I., Reinsel, L. D.: Industry develop-ments and models, Disruptive Innovation in Enterprise Computing: storage. IDC. February. (2003)
9. Porkaew, K., Chakrabarti, K. Mehrotra, S.: Query Refinement for Multimedia Similarity Re-trieval in MARS, In Proc. Of the 7th ACM Multimedia Conference. (1999) 235-238.
10. Ramanathan, M. K., Kalogeraki, V., Pruyne, J.: Finding Good Peers in Peer-to-Peer Net-works. HP Labs. Technical Report HPL-2001-271. (2001)
11. Wu, L., Faloutsos, C., Sycara, K., Payne, T.: FALCON: Feedback Adaptive Loop for Con-tent-Based Retrieval. In Proc. 26th VLDB Conference. (2002). 297-306