Collaborative Filtering for Information Recommendation Systems

Anne Yun-An Chen and Dennis McLeod
Department of Computer Science and Integrated Media System Center
University of Southern California, Los Angeles, California, USA

INTRODUCTION

In order to draw users' attention and to increase their satisfaction towards online information search results, search engine developers and vendors try to predict user preference based on the user behavior. Recommendations are provided by the search engines or online vendors to the users. Recommendation systems are implemented in commercial and non-profit web sites to predict the user preferences. For commercial web sites, accurate predictions may result in higher selling rates. The main functions of recommendation systems include analyzing user data and extracting useful information for further predictions. Recommendation systems are designed to allow users to locate the preferable items quickly and to avoid the possible information overloads. Recommendation systems apply data mining techniques to determine the similarity among thousands or even millions of data.

Collaborative filtering techniques have been successful in enabling the prediction of user preferences in the recommendation systems (Hill et al., 1995, Shardanand & Maes, 1995). There are three major processes in the recommendation systems: object data collections and representations, similarity decisions, and recommendation computations. Collaborative filtering aims at finding the relationships among the new individual and the existing data in order to further determine the similarity and provide recommendations. How to define the similarity is an important issue. How similar should two objects be in order to finalize the preference prediction? Similarity decisions are concluded differently by collaborative filtering techniques. For example, people that like and dislike movies in the same categories would be considered as the ones with similar behavior (Chee et al., 2001). The concept of the nearest-neighbor algorithm has been included in the implementation of the recommendation systems (Resnick et al., 1994). The designs of pioneer recommendation systems focus on entertainment fields (Resnick et al., 1994, Hill et al., 1995, Shardanand & Maes, 1995, Dahlen et al., 1998). The challenge of conventional collaborative filtering algorithms is the scalability issue (Sarwar et al., 2000a). Conventional algorithms explore the relationships among system users in large datasets. User data are dynamic, which means the data vary within a short time period. Current users may change their behavior patterns, and new users may enter the system at any moment. Millions of

user data, which are called neighbors, are to be examined in real time in order to provide recommendations (Herlocker et al., 1999). Searching among millions of neighbors is a time-consuming process. To solve this, item-based collaborative filtering algorithms are proposed to enable reductions of computations because properties of items are relatively static (Sarwar et al., 2001). Suggest is a Top-N recommendation engine implemented with item-based recommendation algorithms (Karypis. 2000, Deshpande & Karypis, 2004). Meanwhile, the amount of items is usually less than the number of users. In early 2004, Amzn Investor Relations (2004) states that Amazon.com Apparel & Accessories Store provides about one hundred and fifty thousands of items but has more than one million customer accounts that have ordered from this store. Amazon.com employs item-based algorithm for collaborative-filtering-based recommendations (Linden et al., 2003) to avoid the disadvantages of conventional collaborative filtering algorithms.

BACKGROUND

Collaborative filtering techniques collect and establish profiles, and determine the relationships among the data according to similarity models. The possible categories of the data in the profiles include user preferences, user behavior patterns, or item properties. Collaborative filtering solves several limitations in content-based filtering techniques (Balabanovic & Shoham, 1997), which decides user preference only based on the individual profile. Collaborative filtering has been expressed in different terminologies in literatures. Social Filtering and Automated Collaborative Filtering (ACF) are two frequently referred terminologies. Collaborative-filtering-based recommendation systems are also referred as Collaborative Filtering Recommender systems and Automated Collaborative Filtering systems.

Several existing collaborative-filtering-based recommendation systems have been designed and implemented since early 90’s. Collaborative filtering techniques have been proven to provide satisfying recommendations to users (Hill et al., 1995, Shardanand & Maes, 1995). Grouplens project, a recommendation system for netnews, has investigated the issues on automated collaborative filtering since 1992 (Resnick et al., 1994, Konstan et al., 1997). In the system design, the Better Bit Bureaus (BBBs) has been developed to predict user preferences based on computing the correlation coefficients between users and on averaging ratings for one news article from all. MovieLens is a movie recommendation system based on GroupLens technology (Miller et al., 2003). RECommendation Tree (RecTree) is one method using divide-and-conquer approach to improve correlation-based collaborative filtering and performing clustering on movie ratings from users (Chee et al., 2001). The ratings are

extracted from MovieLens Dataset. Ringo (Shardanand & Maes, 1995) provides music recommendations using a word of mouth recommendation mechanism. The terminology “social information filtering” was used instead of collaborative filtering in the paper. Ringo determines the similarity of users based on user rating profiles. Firefly and Gustos are two recommendation systems which employed the word-of-mouth recommendation mechanism to recommend products. WebWatcher has been designed for assisting information searches on the World Wide Web (Armstrong et al., 1995). WebWatcher suggests users which hyperlinks would lead to the information that users want. The general function serving as the similarity model is generated by learning from a sample of training data logged from users. Yenta is a multi-agent matchmaking system implemented with the clustering algorithm and the referral mechanism (Foner, 1997). Jester is an online joke recommendation system based on Eigentaste algorithm, which was proposed to reduce dimensionality of offline clustering and to perform online computations in constant time (Goldberg et al., 2000). The clustering is based on continuous user ratings of jokes.

One of the most famous recommendation systems nowadays is the Amazon.com Recommendation (Linden et al., 2003). This recommendation system incorporates a matrix of the item similarity. The formulation of the matrix is performed offline. Launch, music on Yahoo!, Cinemax.com, Moviecritic, TV Recommender, Video Guide and the suggestion box, and CDnow.com are other successful examples of collaborative-filtering-based recommendation systems in the entertainment domain.

Many methods, algorithms, and models have been proposed to resolve the similarity decisions in collaborative-filtering-based recommendation systems. One of the most common methods to determine the similarity is the cosine angle computation. Amazon.com Recommendation system (Linden et al., 2003) uses this cosine measure to decide the similarity between every two items bought by each customer and to establish the item matrix, which contains item-to-item relationships. Several algorithms that combine the knowledge from Artificial Intelligence (AI) (Mobasher et al. 2004), Network (Chien et al., 1999), and other fields have also been implemented in the recommendation systems. Genetic algorithm along with Naïve Bayes Classifier is to define the relationships among users and items (Ko et al., 2001). Genetic algorithm first completes clustering for discovering relationships among system users in order to find the global optimum. On the other hand, Naïve Bayes classifier defines the association rules of items. Then, similarity decisions would be performed to match the clusters of users or clusters of items, and the system can decide the final user profiles. The user profiles only consist of associated rules. Expectation Maximization

(EM) algorithm (Charalambous & Logothetis, 2000) provides a standard procedure to estimate the maximum likelihood of latent variable models, and this algorithm has been applied to estimate different variants of the aspect model for the collaborative filtering (Hoffman, 1999). Heuristic of EM algorithm can be applied on latent class models to perform aspect extracting or clustering.

Meanwhile, hierarchical structures are employed to describe the relationships among users (Jung et al., 2001). The preferences of each user can be described in a hierarchical structure. The structure represents the index of categories, which are the labels of the nodes. Matching one structure to another with all category labels results in that each node contains a group of users with similar preferences. Hierarchical structures can also be applied on similarity computations for items (Ganesan et al., 2003). Edges in the structure clearly define how items are related for the item-to-item relationships. A hierarchical structure, a tree, specified the relative weights for the edges provide information on how much two items are related. A method of the order-based similarity measurement has been proposed for building a personal computer recommendation system (PCFinder) (Xiao et al., 2003). Instead of using 0/1 for the search, this method uses the concept in Fuzzy Logic to estimate the similarity.

Two popular approaches, the coefficient correlation computation and the nearest-neighbor algorithm, have their limitations on scalability and sparsity. Clustering (Breese et al., 1998), Eigentaste algorithm (Goldberg et al., 2000), and Singular Value Decomposition (SVD) (Sarwar et al., 2000) are introduced to collaborative-filtering-based recommendation systems to break these barriers. Eigentaste and genetic algorithms enable the constant time computations for online processes. Item-based collaborative filtering algorithms are proposed to further decrease the computation time (Linden et al., 2003).

MAIN THRUST OF THE MANUSCRIPT

Privacy Issues and User Identification

Do users always agree on being monitored by the systems? Not every user is comfortable if each page the user visited is recorded. Some users even disable the cookies in their browsers. Recommendation systems usually require user registrations in order to utilize user data for future recommendations. There exist users that prefer not to login systems every time they visit. Can the behavior patterns of random users be included in the data mining processes? It depends on the properties of the similarity models. Unregistered users only provide few continuously behavior patterns.

These data may be hardly useful if the similarity models require the quantity of the behavior patterns to reach a certain level. At the same time, these data may be treated as neighbors and included in the clustering processes for the recommendation computations. The computational time will be increased when more neighbors are included. The necessity to include the data of segmental user behavior patterns depends. If enlarging the data coverage enables the increase on the prediction accuracy, there is a trade-off between the computation time length and the coverage scale.

Drawbacks

There are still several drawbacks of the collaborative filtering. First, the lack of the information would affect the recommendation results. For the relationship mining, new items not-yet-rated or not-yet-labeled can be abandoned in the recommendation processes. The second problem is that the collaborative filtering may not cover the extreme case. If the scales of the user profiles are small or the users have unique tastes, similarity decisions are unable to be established. The third problem is the update frequency. If any new information of users has to be included in the recommendation processes in real time, data latency will increase the waiting time for the query result. The complexity of the computation for the recommendation affects the waiting time of the user directly. Synchronization is another issue of the profile updates in the system. When hundreds of users query the system within a very short time period, two new problems occur: who should be considered in one certain clustering process and how to pipeline the computational power of the system server.

Hybrid Methods

A new approach is designed to comprise both content-based and collaborative filtering techniques in order to provide the accurate prediction on user preferences. The decisions of how accurate the predictions are depend on the subjective opinions from the users. A recommendation system including both technologies is a hybrid recommendation system (Balabanovic & Shoham, 1997). Hybrid methods solve the problem of extreme case coverage that collaborative filtering techniques unable to handle.

The Next Evaluation Tool for Information Retrieval (IR)

Precision and Recall are two conventional measurements of data accuracy. User satisfaction has become an important issue in the IR area since a decade ago. Recommendation system developers need to focus on what the users prefer and avoid what the users dislike. Evaluating user satisfaction is not an easy job. There are two

ways to perform the evaluation of user satisfaction. The first one is to survey the users. The problem of this approach is that frequent surveys would cause a lot of disturbance in the online searching experience. The second approach is to decide the criteria to quantify the degrees of user satisfaction. One criterion is the involvement time length for the search result. However, the starting and ending time points are hard to be determined in the Web environment (Shahabi et al., 1997).

FUTURE TRENDS

An ideal recommendation system should be dynamic, which indicates that the updates on profiles can be performed approximately in real time. Although the innovations of hardware designs advance the computational speed, algorithms and techniques with low time computational complexity are expected in the recommendation system developments. User data flow in every second. In order to keep the profiles up-to-date, online computations require many resources, such as memories and computational power. Therefore, it is important to maximize the offline computations. The computation time depends on two factors: the number of items and the number of users in the database. The impact of the first factor, the number of items, may be reduced. The items usually are not added into the databases continuously (the opposite is the data stream, e.g. video stream). However, the decision on the frequency of updates on user profiles is more complex. How often should the updates be performed in order to keep track of the user preference trends? If the updates are required to be performed approximately in real time, an algorithm or a technique with low memory computational complexity is essential to reduce the system loading and eliminate the potential effects on the system synchronization.

CONCLUSION

Different collaborative filtering techniques have been proposed to decrease the processing time and the data latency. The results from different recommendation systems indicate that collaborative filtering techniques afford the systems enough ability to provide recommendations to users. Consequently, the recommendation systems can predict user behavior patterns without any knowledge of the user in advance, and to evaluate the accuracy by the comparing the prediction and the reality. If clustering is performed by Genetic, Nearest-Neighbor algorithm, or the algorithm developed based on any of these two, the gaps among data affect the accuracy of the prediction a lot. This also means that missing data would lower the accuracy of the prediction. The situation is the same for EM approaches. This is because EM approaches perform better when the probability space is more complete. The accuracy of the prediction performed by hierarchical approaches may also be affected since the

recommended items would be too general due to the lack of detailed categorizations. These statements indicate that the reduction of the missing or insufficient data is not simple, and that some approximations are required to be performed in order to provide better predictions of user preferences.

An approach to decrease online computational time is to allow recommendation systems perform the clustering offline (Chee et al, 2001). Several algorithms and techniques proposed perform computations in a constant time (Goldberg et al. 2001, Lemire 2003). These algorithms and techniques provide the possibility of real time updates on profiles in the recommendation systems.

REFERENCES

Aggarwal, C. C., Gates, S. C., & Yu, P. S. (2004). On using partial supervision for text categorization. *IEEE Transactions on Knowledge and Data Engineering, 16 (2),* 245-255.

Amzn Investor Relations. (2004). Amazon.com Apparel & Accessories Store Reaches One Millionth Customer Account. *Amazon.com 2004 Press Releases*. Retrieved 17 August 2004, from http://www.amazon.com.

Armstrong, R., Freitag, D., Joachims, T., & Mitchell, T. (1995). Webwatcher: A learning apprentice for the world wide web. *AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments,* 6-12.

Balabanovic, M., & Shoham, Y. (1997). Fab: Content-based collaborative recommendation. *Communications of the ACM , 40(3),* 88-89.

Breese, J., Heckerman, D., & Kadie, C. (1998). Empirical analysis of predictive algorithm for collaborative filtering. *Microsoft Technical Report MSR-TR-98-12*.

Charalambous, C. D., & Logothetis, A. (2000). Maximum likelihood parameter estimation from incomplete data via the sensitivity equations: the continuous-time case. *IEEE Transactions on Automatic Control (IEEE) 45, no. 5,* 928-34.

Chee, S. H. S., Han, J., & Wang, K. (2001). RecTree: An Effcient Collaborative Filtering Method. *Lecture Notes in Computer Science, 2114,* 141-151.

Chen, A. Y., & McLeod D. (2004). Semantic-based customization of information retrieval on the World Wide Web. manuscript in preparation.

Chien, Y. H., & George E. I. (1999). A Bayesian Model for Collaborative Filtering. *Proceedings of the Seventh International Workshop on Artificial Intelligence and Statistics*, San Francisco, CA.

Dahlen, B. J., Konstan, J., Herlocker, J., Good, N., Borchers, A., & Riedl, J. T. (1998). Jump-starting movielens: User benefits of starting a collaborative filtering system with "dead data". *University of Minnesota TR 98-017*.

Deshpande, M., & Karypis, G. (2004). Item-based top-N recommendation algorithms. *ACM Transactions on Information Systems (TOIS), 22(1)*, 143-177.

Foner, L. (1997). Yenta: A multi-agent, referral-based matchmaking system. *Proceedings of The First International Conference on Autonomous Agents*. ACM. xvi+549, 301-7.

Ganesan, P., Garcia-Molina, H., & Widom, J. (2003). Exploiting hierarchical domain structure to compute similarity. *ACM Transactions on Information Systems*. ACM. 21, no. 1, 64-93.

Goldberg, K., Roeder, T., Gupta, D., & Perkins, C. (2001). Eigentaste: A constant time collaborative filtering algorithm. *Information Retrieval 4(2)*, 133–151.

Herlocker, J., Konstan, J., Borchers, A., & Riedl, J. T. (1999). An algorithmic framework for performing collaborative filtering. *Proceedings of ACM SIGIR'99*. ACM. xvi+339, 230-7.

Hill, W., Stead, L., Rosenstein, M., & Furnas, G. (1995). Recommending and evaluating choices in a virtual community of use. *Proceedings of CHI '95*. ACM. xx+598, 194-201.

Hofmann, T., & Puzicha, J. (1999). Latent class models for collaborative filtering. *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence*. Morgan Kaufmann Publishers. xxii+1452, vol.2, 688-93.

Huberman, B. A., & Kaminsky, M. (1996). Beehive: A system for cooperative

filtering and sharing of information. *Computer Human Interaction*. 210-217.

Lee, W. S. (2000). Online clustering for collaborative filtering. *School of Computing Technical Report TRA8/00*.

Lemire D. (2003). Scale and Translation Invariant Collaborative Filtering Systems. *Journal of Information Retrieval, 8(1),* 129-150.

Lin, C. H., & McLeod, D. (2002). Exploiting and learning human temperaments for customized information recommendation. *Proceedings of the 6th IASTED International Conference on Internet and Multimedia Systems and Applications*. ACTA. iv+430, 218-23.

Jung, J. J., Yoon, J. S., & Jo, G. S. (2001). Collaborative information filtering by using categorized bookmarks on the web. Web Knowledge Management and Decision Support. *14th International Conference on Applications of Prolog*. Revised papers (Lecture Notes in Artificial Intelligence Vol.2543). Berlin, Germany : Springer-Verlag. x+305, 237-50.

Karypis, G. (2000). Evaluation of item-based top-N recommendation algorithms. *Technical Report 00-046, University of Minnesota, Department of Computer Science*.

Ko, S. J., & Lee, J. H. (2001). Discovery of user preference through genetic algorithm and Bayesian categorization for recommendation. *Conceptual Modeling for New Information Systems Technologies, ER 2001 Workshops, HUMACS, DASWIS, ECOMO, and DAMA*. Revised Papers (Lecture Notes in Computer Science Vol.2465). Berlin, Germany : Springer-Verlag,. xvii+500, 471-84.

Konstan, J. A., Miller, B. N., Maltz, D., Herlocker, J., Gordon, L., & Riedl, J. T. (1997). GroupLens: Applying collaborative filtering to Usenet news. *Communications of the ACM, 40(3),* 77-87.

Linden, G., Smith, B., & York, J. (2003). Amazon.com recommendations. *IEEE Internet Computing 7. no. 1, (Jan.-Feb. 2003)* , 76-80.

Miller, B. N., Albert, I., Lam, S. K., Konstan, J. A., & Riedl, J. T. (2003, January). MovieLens unplugged: Experiences with an occasionally connected

recommender system. *Proceedings of ACM 2003 International Conference on Intelligent User Interfaces (Accepted Poster)*.

Mobasher, B., Jin, X., & Zhou, Y. (2004). Semantically Enhanced Collaborative Filtering on the Web. *Proceedings of the European Web Mining Forum, LNAI (Volume TBD)*. Berlin, Germany : Springer-Verlag. To appear.

Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., & Riedl, J. T. (1994). GroupLens: An open architecture for collaborative filtering of Netnews. *Proceedings of the Conference on Computer Supported Cooperative Work*. ACM. xi+464, 175-86.

Sarwar, B. M., Karypis, G., Konstan, J. A., & Riedl, J. T. (2000a). Analysis of recommendation algorithms for E-commerce. *Proceedings of the 2nd ACM Conference on Electronic Commerce*. ACM. vii+271, 158-67.

Sarwar, B. M., Karypis, G., Konstan, J. A., & Riedl, J. T. (2000b). Application of dimensionality reduction in recommender system -- A case study. *ACM WebKDD 2000 Web Mining for E-Commerce Workshop*, Boston, MA.

Sarwar, B. M., Karypis, G., Konstan, J. A., & Riedl, J. T. (2001). Item-based collaborative filtering recommendation algorithms. *Proceedings of the 10th International World Wide Web Conference*, 285-295.

Shahabi, C., Zarkesh A., Adibi, J., & Shah, V. (1997). Knowledge discovery from users web-page navigation. *Proceedings of Workshop on Research Issues in Data Engineering*. IEEE Computer Society. 20-31.

Shardanand, U., & Maes, P. (1995). Social information filtering: Algorithms for automating 'Word of Mouth'. *Proceedings of CHI '95*. ACM. xx+598, 210-17.

Xiao, B., Aimeur, E., & Fernandez, J. M. (2003). PCFinder: an intelligent product recommendation agent for e-commerce. *Proceedings IEEE International Conference on E-Commerce*. IEEE Computer Society. xiv+414, 181-8.

TERMS AND THEIR DEFINITIONS

Recommendation System: A system that retrieves information based on users'

preference**.**

Collaborative Filtering: An approach to provide recommendations based on the preference of similar users.

Content-based Filtering: An approach to provide recommendation based on the individual preference.

Profile: An organized dataset of information on users or items.

Similarity Model: A set of schematic descriptions that specify the measurement standard for the degrees of being similar.

Nearest-neighbor Algorithm: An algorithm that determines and ranks the distance between a target object and any other available object.

Clustering: A task that segments objects into groups according to the object similarity.

Data Latency: An experienced time delay when a system or an agent sends data to a receiver.