Category: Concepts, Trends, and Challenges

Collaborative Filtering for Information Recommendation Systems

Anne Yun-An Chen
University of Southern California, Los Angeles, USA

Dennis McLeod
University of Southern California, Los Angeles, USA

INTRODUCTION

In order to draw users' attention and to increase their satisfaction toward online information search results, search-engine developers and vendors try to predict user preferences based on users' behavior. Recommendations are provided by the search engines or online vendors to the users. Recommendation systems are implemented on commercial and nonprofit Web sites to predict 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 preferable items quickly and to avoid possible information overload. 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 recommendation systems (Hill, Stead, Rosenstein, & Furnas, 1995, Shardanand & Maes, 1995). There are three major processes in recommendation systems: object data collections and representations, similarity decisions, and recommendation computations. Collaborative filtering aims at finding the relationships among new individual data and existing data in order to further determine their 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, Han, & Wang, 2001). The concept of the nearest-neighbor algorithm has been included in the implementation of recommendation systems (Resnick, Iacovou, Suchak, Bergstrom, & Riedl, 1994). The designs of pioneer recommendation systems focus on entertainment fields (Dahlen, Konstan, Herlocker, Good, Borchers, & Riedl, 1998; Resnick et al.; Shardanand & Maes; Hill et al.). The challenge of conventional collaborative-filtering algorithms is the scalability issue (Sarwar, Karypis, Konstan, & Riedl, 2000a). Conventional algorithms explore the relationships among system users in large data sets. 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, Konstan, Borchers, & Riedl, 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, Karypis, Konstan, & Riedl, 2001). Suggest is a top-*N* recommendation engine implemented with item-based recommendation algorithms (Deshpande & Karypis, 2004; Karypis, 2000). Meanwhile, the amount of items is usually less than the number of users. In early 2004, Amazon Investor Relations (2004) stated that the Amazon.com apparel and accessories store provided about 150,000 items but had more than 1 million customer accounts that had ordered from this store. Amazon.com employs an item-based algorithm for collaborative-filtering-based recommendations (Linden, Smith, & York, 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, and item properties. Collaborative filtering solves several limitations of content-based filtering techniques (Balabanovic & Shoham, 1997), which decide user preferences only based on the individual profiles. Collaborative filtering has been expressed with different terminologies in the literature. Social filtering and automated collaborative filtering (ACF) are two frequently referred-to terminologies. Collaborative-filtering-based recommendation systems are also referred as to as collaborative-filtering recommender systems and automated collaborative-filtering systems.

Several collaborative-filtering-based recommendation systems have been designed and implemented since the early '90s. Collaborative-filtering techniques have been proven to provide satisfying recommendations to users (Hill et al., 1995, Shardanand & Maes, 1995). The GroupLens project, a recommendation system for Netnews, has investigated the issues on automated collaborative filtering since 1992 (Konstan, Miller, Maltz, Herlocker, Gordon, & Riedl, 1997; Resnick et al., 1994). In the system design, better bit bureaus (BBBs) have been developed to predict user preferences based on computing the correlation coefficients between users and on averaging the ratings for one news article from all. MovieLens is a movie recommendation system based on the GroupLens technology (Miller, Albert, Lam, Konstan, & Riedl, 2003). Recommendation Tree (RecTree) is one method using the divide-and-conquer approach to improve correlation-based collaborative filtering and perform clustering on movie ratings from users (Chee et al., 2001). The ratings are extracted from the MovieLens data set. Ringo (Shardanand & Maes) provides music recommendations using a word-of-mouth recommendation mechanism. The terminology social information filtering was used instead of collaborative filtering in the literature. Ringo determines the similarity of users based on user rating profiles. Firefly and Gustos are two recommendation systems that employ the word-of-mouth recommendation mechanism to recommend products. WebWatcher has been designed for assisting information searches on the World Wide Web (Armstrong, Freitag, Joachims, & Mitchell, 1995). WebWatcher suggests to users hyperlinks that may lead to the information the 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 multiagent matchmaking system implemented with a clustering algorithm and referral mechanism (Foner, 1997). Jester is an online joke recommendation system based on the eigentaste algorithm, which was proposed to reduce the dimensionality of off-line clustering and to perform online computations in real time (Goldberg et al., 2001). 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 items' similarities. The formulation of the matrix is performed off line. 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 similarity is the cosine angle computation. The 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, Jin, & Zhou, in press), networks (Chien & George, 1999), and other fields have also been implemented in recommendation systems. The genetic algorithm along with the naïve Bayes classifier is used to define the relationships among users and items (Ko & Lee, 2001). The genetic algorithm first completes the clustering for discovering relationships among system users in order to find the global optimum. On the other hand, the naïve Bayes classifier defines the association rules of the items. Then, similarity decisions can 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. The 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 collaborative filtering (Hoffman & Puzicha, 1999). The heuristics of the EM algorithm can be applied to latent class models to perform aspect extracting or clustering.

Meanwhile, hierarchical structures are employed to describe the relationships among users (Jung, Yoon, & Jo, 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 each node containing a group of users with similar preferences. Hierarchical structures can also be applied to similarity computations for items (Ganesan, Garcia-Molina, & Widom, 2003). Edges in the structure clearly define how items are related for the item-to-item relationships. A hierarchical structure, a tree specifying the relative weights for the edges, provides 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, Aimeur, & Fernandez, 2003). Instead of using 0/1 for the search, this method uses the concept of fuzzy logic to estimate the similarity.

Two popular approaches, the coefficient correlation computation and the nearest-neighbor algorithm, have

Related Content