Chapter IX

COBRAS: Cooperative CBR Bibliographic Recommender System

Hager Karoui

Université Paris XIII, France

ABSTRACT

In this chapter, the authors propose a case-based reasoning recommender system called COBRAS: a Peer-to-Peer (P2P) bibliographical reference recommender system. COBRAS’s task is to find relevant documents and interesting people related to the interests and preferences of a single person belonging to a like-minded group in an implicit and an intelligent way. Each user manages their own bibliographical database in isolation from others. Target users use a common vocabulary for document indexing but may interpret the indexing vocabulary differently from others. Software agents are used to ensure indirect cooperation between users. A P2P architecture is used to allow users to control their data sharing scheme with others and to ensure their autonomy and privacy. The system associates a software assistant agent with each user. Agents are attributed three main skills: a) detecting the associated user’s hot topics, b) selecting a subset of peer agents that are likely to provide relevant recommendations, and c) recommending both documents and other agents in response to a recommendation request sent by a peer agent. The last two skills are handled by implementing two inter-related data-driven case-based reasoning systems. The basic idea underlying the document recommendation process is to map hot topics sent by an agent to local topics. Documents indexed by mapped topics are then recommended to the requesting agent. This agent will provide later, a relevance feedback computed after the user evaluation of the received recommendations. Provided feedbacks are used to learn to associate a community of peer agents to each local hot topic. An experimental study involving one hundred software agents using real bibliographical data is described. The Obtained results demonstrate the validity of the proposed approach.
INTRODUCTION

Recommender systems (Resnick & Varian, 1997) aim at learning users’ preferences over time in order to automatically suggest products that fit the learned user model. For example, recommender systems are used in e-commerce web sites to help customers in selecting products more suitable to their needs. Various techniques are used in order to compute recommendations such as:

- **Collaborative filtering**: is the most popular recommendation technique that aggregates data about customer’s preferences (products’ ratings) to recommend new products. A popular example is Amazon.com (Linden et al., 2003) which recommends items to customers. Another example is MovieLens (Miller et al., 2003) which recommends movies.

- **Content-based filtering** exploits the preferences of a specific customer to build new recommendations to the customer. Content-based filtering approaches recommend items for the user based on the description of previously evaluated items. User profiles are created using features extracted from these items and each user is assumed to operate independently. NewsDude (Billsus & Pazzani, 1999) recommends news stories that follow up on stories the user read previously. It observes what online news stories the user has read and not read and learns to present the user with articles he/she may be interested to read. Letizia (Lieberman, 1995) which recommends Web pages during browsing based on user profile.

- **Hybrid filtering** exploits features of content-based and collaborative filtering. For example, Fab (Balabanovic, 1997) for document recommendation where user profiles based on content analysis are maintained and closely compared to determine users with similar preferences for collaborative recommendation. COSYDOR (Jéribi et al., 2001) is a system for user’s assistance by helping the users to reformulate their queries based on the experience reuse and on the user’s profiles. WebWatcher (Armstrong et al., 1995) presents an agent recommending appropriate hyperlink given the current Web page viewed by the user.

In collaborative filtering the recommendation depends on customers information, and a large number of previous user/system interactions are required to build reliable recommendations. In content-based systems only the data of the current user are exploited in building a recommendation. It requires a description of user interests that is either matched in the items catalog or provided as input for the learned user model to output a recommendation. Both approaches, if not trained with lot of examples (product ratings or pattern of user preferences), deliver poor recommendations. This limitation mostly motivated a third approach, knowledge-based, that tries to better use preexisting knowledge specific of the application domain (e.g. travels vs. computers) to build a more accurate model requiring less training instances (Lorenzi & Ricci, 2003). The knowledge-based approach is considered complementary to the other approaches (Burke, 2000). In this approach, knowledge about customers and the application domain are used to reason about what products fit the customers preferences. The most important advantage is that this approach does not depend (exclusively) on customers rates, hence avoiding the mentioned difficulty in bootstrapping the system. Case-Based Reasoning (CBR) is one of the most successful machine learning methodologies that exploits knowledge-rich representation of the application domain (Vatson, 1997; Aha, 1998). It is a cyclic and integrated problem solving process that supports learning from experience (Aamodt & Plaza, 1994) and is based on human reasoning. In this chapter, we propose a general CBR approach for object recommendation such as document, musique, image, etc.. In the treated example, the