CNS: A Task-based Hybrid Collaborative Filtering Recommender Service

Sampath Jayarathna and Frank Shipman
Computer Science & Engineering
Texas A&M University
College Station, TX 77843-3112 USA
{sampath, shipman}@cse.tamu.edu

Abstract—In this paper, we introduce a task-based hybrid collaborative filtering (CF) recommender service to support users’ information gathering tasks. Even with the best web search engines (WSEs), and the most effective query formulations, information gathering tasks require people to work through long lists of documents to determine potentially relevant documents. We propose and implement Singular Value Decomposition (SVD) based Latent Semantic Indexing (LSI) to find similarity in information gathering tasks to find task clusters and use these clusters for recommendations. Our method alleviates major challenges in traditional CF such as data scalability, sparsity, synonymy and privacy protection (using interest profiles via a web service).

Keywords – Latent Semantic Indexing, User Interest Modeling, Recommender Systems

I. INTRODUCTION

A Recommender System can support its users by recommending documents that best match user’s interests during an open-ended information gathering task, thereby ensuring that user’s time is spent efficiently on the most relevant documents. These recommendations may derive from the interests, activities, and outcomes of similar information tasks performed by a community of users. In this work, we explore recommendations of web documents based on a task-based hybrid CF framework. We assume that users interact with WSEs with a certain information task in mind, and hence finding similar task preferences from many users (collaborating) may provide informative recommendations of interest. In doing so, we intend to support the searching of web-based documents “by task to be executed”, instead of “by documents to be retrieved”[1].

In the traditional CF, the fundamental assumption is that if users X and Y rate N items similarly, then they will rate or act on other items similarly [2]. In other words, CF uses the known preferences of a group of users to make recommendations or predictions of the unknown preferences of the other users. This work aims to investigate users’ behavior in everyday information gathering tasks by taking into consideration similar tasks identified in a community of recommenders.

The traditional CF uses only the user-item ratings data to make predictions and recommendations whereas content-based (CB) systems rely on the features of user-items. While CF systems do not explicitly incorporate feature information, CB systems do not incorporate preference and similarity of individual users [2]. Furthermore, current task-based recommender systems [1] rely only on long-term query log submitted to the WSEs. The query logs may constitute a good starting point to build a user behavior models, but these may lack task specific semantic labels to extract tasks from the logs.

II. USER INTEREST MODEL

A. Interest Profiles

This paper describes our efforts to use interest profiles of a community of users (Figure 1) as a basis to create task clusters that draw a new user’s attention to similar documents and document parts during an information gathering task. During information tasks, useful documents may be long and cover multiple subtopics; users may read some segments and ignore others. In order to record which portion(s) of the document pique the user’s interests, an explicit annotation capturing tool is used [3]. The derived User Interest Profile consists of a unique task ID, documents IDs (URLs) and the respective term vectors (user annotations). In addition, the Interest Profile Manager (IPM) calculates an interest rating per each document, but this interest rating is not used as an attribute in the current work.

![Figure 1. Community Navigation Service Overview](image)

B. Interest Profile Manager (IPM)

The IPM (Figure 1) acts as a local interest profile server which collects information from multiple applications, including a Reading Application, and aggregates and stores this information in the user’s interest profile and broadcasts it to the participating applications [4]. Any application can be modified to include the IPM local server. Currently Visual Knowledge...
Builder (VKB organizing application) [3], WebAnnotate (Reading Application) and CNS Recommender Client support two-way communication with IPM. The IPM communicates individual tasks to the CNS at a time maintaining user anonymity and reduces the risk that the information at the CNS is subject to de-anonymization attacks.

Figure 2. CNS Web Service Architecture

III. COMMUNITY NAVIGATION SERVICE

Figure 2 shows simplified schematic web service architecture of the components involved in running the prototype wrapper application of the CNS. The center elements are the two java classes that we need to instantiate the client and the actual service. Also note that we can implement the client version of the code in any SOAP aware language and therefore are not constrained to use Java/Axis2 [5]. The following sections explain the functionality provided by the CNS Service class to calculate a list of document recommendations for a new user task.

A. Task Similarity and Clustering

Each atomic task in the user interest profile includes web documents related to the information gathering task and their respective term vectors. Our goal is to create a \( T \times T \) task similarity matrix given a set of \( T \) tasks to be clustered. Given two tasks \( \alpha \) and \( \beta \), we define the task similarity objective function as:

\[
tSim(\alpha, \beta) = \frac{\sum_{i=1}^{m} \sum_{j=1}^{n} dSim(i,j)}{m \times n}
\]

(1)

Similarity computation between two documents \( dSim(i,j) \) is a critical step in memory-based CF approaches. There are many different methods to compute the similarity between documents but dimensionality reduction techniques such as SVD can alleviate the data sparsity and scalability problem and quickly produce quality recommendations. The SVD based Latent Semantic Indexing (LSI) [6] can take a large matrix of term document association data and construct a semantic space where terms and documents that are closely associated can be detected with cosine similarity. The resultant average aggregated similarity \( tSim(\alpha, \beta) \) is calculated for all the given \( T \) tasks and a \( T \times T \) task similarity matrix is used in the clustering process. In this work, hierarchical agglomerative clustering [7] with complete linkage and auto-clustering is used.

IV. TASK-BASED RECOMMENDATIONS

During the recommendation phase, a user’s current information gathering task \( t \) consisting of document IDs and respective term vectors can be compared in the semantic space with the inferred task clusters \( N \) from the community. The new task must first be translated into the concept space via the same transformation used on documents in the task-cluster semantic concept space. In order to find the task cluster closely associated with the new task, the task similarity objective function is used by calculating the average task similarity per each cluster and finding cluster with the highest average task similarity given by,

\[
\text{max} \left\{ cSim_x = \frac{\sum_{i=1}^{n} tSim(i,t)}{n} \right\} \quad x = 1, 2, ..., N
\]

(2)

V. CONCLUSION AND FUTURE WORK

In this work, we used CF, one of most successful approaches to build recommender systems based on task preferences of a community of users to make recommendations for other users. By using SVD based LSI over individual tasks, our method alleviates main challenges of CF such as data scalability, synonymy and privacy protection (using interest profiles).

In our future works, we will incorporate CNS with the IPM’s functionality necessary to characterize user interests, including the ability to collect, parse, and determine similarity among common forms of Web documents. We are also working on a large-scale user study to evaluate the performance of the CNS in a digital library environment.

REFERENCES