**COMP9417**

**Machine Learning and Data Mining**

**Assignment2**

**Topic: No.9 Recommender system using collaborative filtering**

**Group member: Yichen Zhu z5098663**

**Difan Xu z5106246**

**Zheyu Li z5099346**

**Chi Xu z5096261**

**Introduction**

A **recommender system** or a **recommendation system** (sometimes replacing "system" with a synonym such as platform or engine) is a subclass of information filtering system that seeks to predict the "rating" or "preference" that a user would give to an item.

Recommender systems have become increasingly popular in recent years, and are utilized in a variety of areas including movies, music, news, books, research articles, search queries, social tags, and products in general. There are also recommender systems for experts,collaborators, jokes, restaurants, garments, financial services, life insurance, romantic partners (online dating), and Twitter pages.

**Collaborative filtering**

One approach to the design of recommender systems that has wide use is collaborative filtering. Collaborative filtering methods are based on collecting and analyzing a large amount of information on users’ behaviors, activities or preferences and predicting what users will like based on their similarity to other users. A key advantage of the collaborative filtering approach is that it does not rely on machine analyzable content and therefore it is capable of accurately recommending complex items such as movies without requiring an "understanding" of the item itself. Many algorithms have been used in measuring user similarity or item similarity in recommender systems. For example, the k-nearest neighbor (k-NN) approach and the Pearson Correlation as first implemented by Allen.

Collaborative filtering is based on the assumption that people who agreed in the past will agree in the future, and that they will like similar kinds of items as they liked in the past.

When building a model from a user's behavior, a distinction is often made between explicit and implicit forms of data collection.

Examples of explicit data collection include the following:

* Asking a user to rate an item on a sliding scale.
* Asking a user to search.
* Asking a user to rank a collection of items from favorite to least favorite.
* Presenting two items to a user and asking him/her to choose the better one of them.
* Asking a user to create a list of items that he/she likes.

Examples of implicit data collection include the following:

* Observing the items that a user views in an online store.
* Analyzing item/user viewing times.
* Keeping a record of the items that a user purchases online.
* Obtaining a list of items that a user has listened to or watched on his/her computer.
* Analyzing the user's social network and discovering similar likes and dislikes.

The recommender system compares the collected data to similar and dissimilar data collected from others and calculates a list of recommended items for the user. Several commercial and non-commercial examples are listed in the article on collaborative filtering systems.

One of the most famous examples of collaborative filtering is item-to-item collaborative filtering (people who buy x also buy y), an algorithm popularized by Amazon.com's recommender system.[[24]](https://en.wikipedia.org/wiki/Recommender_system#cite_note-patft.uspto.gov-24) Other examples include:

* As previously detailed, Last.fm recommends music based on a comparison of the listening habits of similar users, while Readgeek compares books ratings for recommendations.
* Facebook, MySpace, LinkedIn, and other social networks use collaborative filtering to recommend new friends, groups, and other social connections (by examining the network of connections between a user and their friends). Twitter uses many signals and in-memory computations for recommending who to follow to its users.

Collaborative filtering approaches often suffer from three problems: cold start, scalability, and sparsity.

* Cold start: These systems often require a large amount of existing data on a user in order to make accurate recommendations.
* Scalability: In many of the environments in which these systems make recommendations, there are millions of users and products. Thus, a large amount of computation power is often necessary to calculate recommendations.
* Sparsity: The number of items sold on major e-commerce sites is extremely large. The most active users will only have rated a small subset of the overall database. Thus, even the most popular items have very few ratings.

A particular type of collaborative filtering algorithm uses matrix factorization, a low-rank matrix approximation technique.

Collaborative filtering methods are classified as memory-based and model based collaborative filtering. A well-known example of memory-based approaches is user-based algorithm and that of model-based approaches is Kernel-Mapping Recommender.

**Implementation**

1. **Class UserBasedRecommender**

**To**

1. **Class ItemBasedRecommender**
2. **Class Similarity**

**Find the similarity between new user and certain exist user and similarity between new item and certain exit item.**

1. Class Evaluation

**Experimentation**

TO BE CONTINUED

**Results**

TO BE CONTINUED

**References**

TO BE CONTINUED