CS771: Group 1
Project Write Up

Akshay Kumar
kakshay@iitk.ac.in
Dept. of CSE

Arpit Jain
arpitj@iitk.ac.in
Dept. of CSE

Dhananjay Sharma
djsharma@iitk.ac.in
Dept. of CSE

Indian Institute of Technology, Kanpur

Project Report
September 26, 2013

Abstract

Abstract of the project. We aim to build a recommendation system for the *movielens* database mentioned in Problem 2. As mentioned in the problem statement, it will perform the following two tasks: predict whether a user is likely to watch a given movie or not and secondly if yes, predic user's rating on a scale of 1 to 5.

1 Introduction

Online recommendation system is almost ubiquitous in present world. From flipkart to netflix, almost every consumer serving website we visit provides us some kind of recommendation or the other. It may vary from friend suggestions on facebook to books on flipkart or to articles liked on a newpaper website.

When it comes to salesperson, the immediate and tangible benefits of a successful recommendation system lies in increasing sales and creating revenue. When it comes to user (potential buyers), they are often overwlhelmed with a multitude of choices and options in online business experiences while at the same time they have limited resources and time to invest in the selection process.

The problem of building a prediction system for the movie database is infact very intimately related to making a recommendation system. Here, the recommended products will be the movies a user is likely to watch.

1.1 Problem Statement

Formally speaking, we aim to develop a prediction system that predicts the movie preferences for an unknown user based on some factors. As described in the problem statement, we essentially aim to target the following two grey areas:

1. Predict whether a user is likely to watch a movie or not
2. If yes, the predict the rating a user is likely to accord to that movie on a scale of 1 to 5

1.2 Current State of Art

A lot is already happening in the field of recommendation system. It all started with k-nearest problem however it has expanded its horizon immensly since then. The algorithms in this field are heavily centered around cluster analysis.

Traditionally used recommendation techniques include content based filtering, collaborative filtering and a nybrid of these two. Content based filtering build up a user profile based on his past preferences, likes etc. In contrast, collaborative filtering tries to form communities of users in a social network that share appreciations.

More recent recommendation system approach tend to be influenced by a myriad of other factors say for exmaple context awareness or semantics/ontology.

As far as movie recommendation system is concerned, the story in incomplete with the mention of Netflix. Netflix is an on-demand internet streaming

media provider. Succintly speaking, you can watch latest movies right from your couch. They say that most of the movie recommendation algorithms need to know the following tw things : **1)** What do you like? **2)** What movies are siimilar to te ones you like?

2 Algorithms

As described in the previous section their are traditionally two major approaches:

1. **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.

2. **Content-based Filtering** Content-based filtering methods are based on information about and characteristics of the items that are going to be recommended. In other words, these algorithms try to recommend items that are similar to those that a user liked in the past (or is examining in the present). In particular, various candidate items are compared with items previously rated by the user and the best-matching items are recommended.

Collaborative filtering approaches include $k-NN$ and the Pearson Correlation. It is characterized by : "People who buy x also buy y". It is used primarily by amazon, last.fm, facebook, etc. Content-based filtering approzhes include Bayesian Classifiers, cluster analysis, decision tress, artificial neural networks, etc.

3 The curious case of "The Netflix Prize"

At this point, it worthwhile mentioning about "The Netflix Prize" initiative. It was an event designed by Netflix where the competitors were provided with a database of over 100 million movie ratings and the objective was to return recommendations that are 10% more accurate than the one offered by the company. The most accurate algorithm used an ensemble of 107 different algorithms. This event spurred huge research in the field of recommender systems.

4 Issues in Recommendation System

We tend to target the following problems in a movie recommendation system:

- Scalability
- Shilling attack
- Sparsity

We would like to shed some light upon the problem of Shilling Attack. The greatest example is of The Godfather v/s Pulp Fiction fight and how The Shawshank Redemption eventually benefitted the most out of this. Their were two polarities, one dedicated to The Godfather and the other two Pulp Fiction and there always used to be stiff competition between these two. However, Pulp Fiction supporters started giving poor ratings to The Godfather and vice versa. As a result the rating of both the movie fell and the one which benefitted the most was The Shawshank Redemption. This is what shilling effect is all about.

5 Platform to be used

We plan to use either R or Matlab for implementing the recommendation systems. Both the languages have excellent statistical tools.