Visvesvaraya Technological University

A Project Report
on

A Weight based Personalized Recommendation using Idiocentric and Collaborative Recommendation

submitted in partial fulfillment of the requirements for the award of the degree of
BACHELOR OF ENGINEERING
in
COMPUTER SCIENCE & ENGINEERING

by

Vijesh M- 1PI09CS119
Sanjay G Huilgol- 1PI09CS083
Vijay Mahantesh SM- 1PI09CS118

under the guidance of
Dr. Kavi Mahesh

January 2013 - May 2013

Department of Computer Science & Engineering
PES Institute of Technology
(An autonomous institute under VTU, Belgaum and UGC, New Delhi)
100 Feet Ring Road, BSK-III Stage, Bangalore-560085

PES Institute of Technology

(An autonomous institute under VTU, Belgaum and UGC, New Delhi)

100 Feet Ring Road, BSK-III Stage, Bangalore-560085

Department of Computer Science & Engineering

CERTIFICATE

Certified that the project work entitled **”A Weight based Personalized Recommendation using Idiocentric and Collaborative Recommendation”** is bonafide work carried out by

Vijesh M, 1PI09CS119
Sanjay G Huilgol, 1PI09CS083
Vijay Mahantesh SM, 1PI09CS118

in partial fulfillment for the award of degree of Bachelor of Engineering in Computer Science and Engineering of the Visvesvaraya Technological University, Belgaum during the academic semester January 2013 - May 2013. It is certified that all corrections and suggestions indicated for internal assessment have been incorporated in the report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of the project work prescribed for the said Bachelor of Engineering degree.

Signature of the Guide	Signature of the HOD	Signature of the Principal
Dr. Kavi Mahesh	Prof. Nitin V Pujari	Dr. K N B Murthy

External Viva

Name of the examiners   Signature with date

1. ....

2. ....

Acknowledgement:

The thrill and excitement of finishing a project can be felt in the last stage when the report is being written. When we look back from where it all started we recollect all the people who brought this project from infancy to its completion.

At the outset, we would like to express our heartfelt thanks towards our Principal **Dr. K N Balasubramanya Murthy** for providing us with a congenial environment to carry out our academic projects.

We express our deep sense of gratitude to our HOD, **Prof. Nitin V Pujari** for his invaluable support and guidance and providing all the necessary resources, which always encourages us to excel.

We consider it a privilege to acknowledge and express gratitude to our guide, **Dr. Kavi Mahesh** for his continuous support and motivational guidance throughout.

We also express our sincere thanks to all the teaching and non-teaching staff of the Department of Computer Science, PESIT for their kind support and ever helping nature. We would also like to thank our project coordinator, **Mr. Badri Prasad**, for all his cordial support during the project timeline.

A friend in need is a friend indeed is rightly said. We take this opportunity to thank all our friends for their constant and invaluable feedback which encouraged us to a great extent. In particular, we would like to thank **Sandeep Raju P** and **Rakesh Kumar R** for their invaluable effort in helping us with designing the approach and building the user interface.

Lastly, we are very grateful to several web resources like Google, Wikipedia and other useful technical articles without which this project would not have seen the light of day.

- The Team

Abstract:

Recommending items to users is one of the most frequently stumbled upon requirements in the e-commerce, or the internet in general. Many techniques have been developed for recommending items. The techniques can be broadly classified into three categories: Collaborative Filtering, Content Based Filtering and Hybrid Recommendations. In this project, we have developed a Graph-based recommendation algorithm that uses both the idiocentric and collaborative behavior of the user to recommend the items back to the users. We have developed methods to carry out unsupervised dimensionality reduction on the dataset. We have deduced some of the characteristic qualities of the users and have constructed a custom built profile for each of them. We have used the movielens_1m dataset to test the accuracy of our algorithm. A survey paper on movielens dataset by Bao and Xia, from autumn 2012 Machine Learning Course at Stanford University, compares four parameterized approaches for rating prediction. The least RMS error from all the approaches was found to be 0.917. Using the non-parameterized approach that we have developed, we were able to achieve an RMS error of 0.903.

Contents

Chapter 1

Introduction

In this age of information overload, people use a variety of strategies to make choices about what to buy, how to spend their leisure time, what movies to watch, etc. Recommender systems automate some of these strategies with the goal of providing affordable, personal, and high-quality recommendations. Recommender Systems are primarily directed towards individuals who lack sufficient personal experience or competence to evaluate the potentially overwhelming number of alternative items that a Web site, for example, may offer. Since recommendations are usually personalized, different users or user groups receive diverse suggestions.

In their simplest form, personalized recommendations are offered as ranked lists of items. In performing this ranking, recommender systems try to predict what the most suitable products or services are, based on the user's preferences and constraints. In order to complete such a computational task, they collect from users their preferences, which are either explicitly expressed, e.g., as ratings for products, or are inferred by interpreting user actions. For instance, navigation to a particular product page can be considered as an implicit sign of preference for the items shown on that page.

The major problems faced in recommender systems are as follows:

- In-homogeneity in properties of items An item is associated with a set of properties. Given a set of items, there is no constraint that all the items should possess the same set of properties. Generally, recommendation systems assume homogeneity in properties and are not applicable to inhomogeneous item datasets.

- Dimensionality Reduction and Feature Selection on the item dataset In machine learning and statistics, dimensionality reduction is the process of reducing the number of features under consideration, and can be divided into feature selection and feature extraction. Feature selection approaches try to find a subset of the original features. Feature extraction transforms the data in the high-dimensional space.

- User Profiling A user profile indicates the information needs or preferences on items that users are interested in. There are a lot of recommendation techniques that utilize the user profiles to personalize the recommendations.

- Combining Idiocentric and Collaborative Recommendation Idiocentric recommendation involves recommending items to users based only on the user's history. Collaborative recommendation involves recommending items to users based on user-user similarity or item-item similarity. Each of the methods give varied results and combining them is a challenging task.

In our approach to the solution, we are addressing the problems mentioned above. We verify the accuracy of our approach on the movielens dataset. We also compare the results that we've obtained against the results presented in one of the survey papers from the Machine Learning Course, Stanford University, Autumn 2012.

The solution that we propose is robust in the perspective of in-homogeneity in properties of items. That is, it is not mandatory that all the items have the same set of properties. We also perform unsupervised dimensionality reduction on the item dataset, relative to the user dataset. Considering the pattern in which the users have consumed the items, we also perform user profiling and determine the weight that the users inherently have towards attribute of the items. From the usage pattern, we deduce the extent of idiosyncrasy and use it to combine the results from idiocentric and collaborative recommendation.

Chapter 2

Problem Definition

We frequently stumble upon applications in e-commerce and the Internet that requires recommending items to users. From ages, many techniques have been proposed for recommending items to users, but as it involves a large dataset most of the techniques are costly in terms of space and time. Often, the dataset available are inhomogeneous in terms of item properties. If the item contains large number of properties i.e. a very high dimensionality, it becomes tedious for the system user to select features that have a significant impact factor. Good personalized recommendations demand an efficient and intuitive user profiling. So an effective way of solving this recommendation problem is of great importance.

Chapter 3

Literature Survey

Recommendation systems have gained popularity in web based systems since the appearance of papers on collaborative filtering in the 1990s [7, 11, 5].

- [7]explains the concept of collaborative filters. They introduce GroupLens as a system for collaborative filtering of netnews, to help people find articles they will like in the huge stream of available articles. They hypothesize that the users who have agreed on a certain aspect in the past will probably agree again.
- [11]describes a technique for making personalized recommendations to a user based on the similarities between the interest profile of that user and those of other users. They also test and compare four different algorithms for making recommendations using social information filtering.
- [5]presents an approach for collaborative recommendation where in the history of other users is used in the automation of a social method for informing choices to the user. Their results show that the communal history-of-use data can serve as a powerful resource for use in interfaces.
- [6]determines the potential predictive utility for Usenet news. They develop a specially modified news browser that accepts ratings and displays predictions on a 1-5 scale. They compute the predictions using collaborative filtering techniques and compare the results with non collaborative approaches.
- [8]describes several algorithms designed for collaborative filtering, including techniques based on correlation coefficients, vector-based similarity calculations, and statistical Bayesian methods. They compare the predictive accuracy of the various methods in a set of representative problem domains. Over the past decade, web systems have moved towards personalized recommendations for better user experience.
- [3]describes a tag-based system for personalized recommendation. They propose an approach which extends the basic similarity calculus with external factors such as tag popularity, tag representativeness and the affinity between user and tag.
- [1]investigates user modeling strategies for inferring personal interest profiles from social web interactions. They analyze individual micro-blogging activities on twitter and compare different strategies for creating user profiles based on the twitter messages.
- [10]presents a personalization algorithm for recommendation in folksonomies, which relies on hierarchical tag clusters.
- [9]explores and analyzes different item-based collaborative techniques. They look into different techniques for computing item-item similarities and various techniques to obtain recommendations from them. Significant developments in learning using graph data has led to recent advances in recommendation techniques.
- [2]presents a recommendation algorithm that includes different types of contextual information. They model the browsing process of a user on a movie database by taking random walks over the contextual graph.

- [4]models personalized tag recommendation as a ”query and ranking” problem. They also propose a novel graph-based ranking algorithm for interrelated multitype objects.

Chapter 4

Project Requirement Definition

4.1 Target Users

The primary target users are academic researchers and firms that use some form of recommendation in their product. The algorithm is open sourced and the users are free to contribute towards the development of a more efficient algorithm.

4.2 Software Requirements

4.2.1 Python

Python is a general-purpose, high-level programming language whose design philosophy emphasizes code readability. It is an expressive language which provides language constructs intended to enable clear programs on both a small and large scale with its own built-in memory management and good facilities for calling and cooperating with other programs. Python supports multiple programming paradigms, including object-oriented, imperative and functional programming styles. Python packages that we use in our implementation are:

- networkx - Used for the creation, manipulation, and study of the structure, dynamics, and functions of complex networks.
- pickle - pickle is the standard mechanism for the object serialization and de-serialization. It was developed as a pure python pickle module in its beta version. We use pickle to store graph objects.
- numpy - numpy is an extension to the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large library of high-level mathematical functions to operate on these arrays.

4.2.2 Gephi

Gephi is an interactive visualization and exploration platform for all kinds of networks and complex systems, dynamic and hierarchical graphs. We use it to visualize the Item graph while displaying the output.

4.2.3 Pencil

Pencil is an open-source GUI prototyping tool that we used to create UI mockup.

4.2.4 Git Versioning System

In software development, Git is a distributed revision control and source code management (SCM) system with an emphasis on speed. Every Git working directory is a full-fledged repository with complete history and full revision tracking capabilities, not dependent on network access or a central server. We use it to effectively control our software development.

4.2.5 Qt

Qt is a cross-platform application framework that is widely used for developing application software with a graphical user interface (GUI) (in which cases Qt is classified as a widget toolkit), and also used for developing non-GUI programs such as command-line tools and consoles for servers. Qt uses standard C++ but makes extensive use of a special code generator (called the Meta Object Compiler, or moc) together with several macros to enrich the language. Qt can also be used in several other programming languages via language bindings. It runs on the major desktop platforms and some of the mobile platforms. It has extensive internationalization support. Non-GUI features include SQL database access, XML parsing, thread management, network support, and a unified cross-platform application programming interface (API) for file handling.

4.3 Hardware Requirements

This being partially a research project, we are coming up with an algorithmic solution to the recommendation problem. So the hardware requirements are minimal and basic. The execution demands a linux-based operating system. The exact configuration of the system is highly application dependent. Larger the dataset, more powerful the system should be.

4.4 Assumptions and Dependencies

As a user of our recommendation system, we assume that he/she has the knowledge about the following aspects:

- The user should possess the working knowledge of linux-based system.
- The user should know how to execute projects in Qt.
- The user is expected to know the abstract flow of the algorithm.
- The user is expected to know what a dataset item and a dataset user is.
- The user is expected to know what the properties of the items are.
- The user should have knowledge about the format of our input dataset, as specified in the input requirements.
- The user should know what the terminologies idiocentric and collaborative recommendations represent.
- The user should be able to interpret what the relative attributive importance and relative value importance are.
- The user should know what dimensionality reduction is.
- The user should have a basic knowledge in the concepts involved in similarity and clustering.
- If the user is keen on understanding how the recommendations were given, he/she should have knowledge about graphs, data-mining and machine learning concepts.

As a researcher or a developer who wishes to extend our system, we assume that he/she has the knowledge about the following aspects:

- All the assumptions specified for a normal user in the above sub-section hold good for a researcher/developer.
- The researcher is expected to have deep understanding about graphs, data-mining and machine learning concepts.
- The researcher should be aware of the previous research works carried out in this domain.
- The researcher should have a deep understanding about our algorithm.

- The researcher should have deep understanding about the concepts of similarity and clustering (overlapping clustering in specific).
- The researcher should understand the difficulties in unsupervised dimensionality reduction.
- The researcher should be able to identify the improvements in various sections of the algorithm.
- The developer is expected to be knowledgeable about the following technologies:
  - Python
  - Qt
  - Git
  - Linux Based Operating System
- The developer should be able to recognize and use the existing interfaces to plug-in custom algorithms.

4.5 Languages Used

- Python
- Qt for GUI Programming