

# A Study on Comparison Analysis of Collaborative Filtering in Java and R

Aziz Nasridinov, Young-Ho Park

Dept. of Multimedia Science, Sookmyung Women's University, Seoul, South Korea

{aziz, yhpark}sm.ac.kr

The mobile application market has been growing extensively in recent years. Currently, Apple's App Store has more than 400,000 applications and Google's Android Market has above 150,000 applications. Such growth in volumes of mobile applications has created a need to develop a recommender system that assists the users to take the right choice, when searching for a mobile application. In this paper, we study the recommendation system building tools based on collaborative filtering. Specifically, we present a study on comparison analysis of collaborative filtering in Java and R statistical software. We implement the collaborative filtering using Java's Apache Mahout and R's recommenderlab package. We evaluate both methods and describe the advantages and disadvantages of using them in order to implement collaborative filtering.

## 1. Introduction

The mobile application market has been growing extensively in recent years. Currently, Apple's App Store has more than 400,000 applications and Google's Android Market has above 150,000 applications. According to a recent study, the mobile application market will reach \$17.5 billion by 2012. By then, the number of mobile application downloads will have also grown to nearly 50 billion from just over 7 billion in 2009 [1]. Such growth in volumes of mobile applications has created a need to develop a recommender system that assists the users to take the right choice, when searching for a mobile application.

A lot of mobile applications have emerged that provide app recommendation and aggregation services which filter, rank and recommend the best apps to end users. For example, Appazaar [2] is a recommender system for Android apps. Users can look into how people use all the apps that are available on the market. The app provides stats on the daily and location-based usage. Appaware [3] is a mobile application that captures and shares installations, updates, and removals of Android programs in real time. Users can participate to this social market anonymously or share their history via Twitter.

In this paper, we study the recommendation system building tools based on collaborative filtering. Specifically, we present a study on comparison analysis of collaborative filtering in Java and R statistical software. We implement the collaborative filtering using Java's Apache Mahout and R recommenderlab package. We evaluate both of these methods, describe the advantages and disadvantages of using them in order to implement collaborative filtering. The rest of this paper proceeds in as follows. Section 2 discusses the related work. Section 3 explains the collaborative filtering. Section 4 presents the results of comparison of collaborative filtering approach in Java and R. Section 5 highlights conclusions.

## 2. Related Work

The Android Market proposes basic features of recommender system, such as "Featured" and "Top" applications list. "Featured" applications are considered by a recent release and a rapidly increasing download rate. "Top" applications are highest ranked applications in terms of

download number. Android Market also provides the reviews (i.e. stars score and the comments) of each application, which enables users think and evaluate when searching for a mobile application.

Appazaar [2] is a recommender system that suggests mobile applications to its users that is available on the Android Market Store. It learns which applications you find interesting by tracking your application usage and comparing you to other people with similar interests and by taking your real-time location into account. Appaware [3] is a mobile application that captures and shares installations, updates, and removals of Android programs in real time. In other words, AppAware is a social network for Apps and Games. It shows which Android Apps are currently popular among friends and the community. By using AppAware, any user can immediately see which applications are popular by observing how many people install or update them. Moreover, users can also see which programs are being downloaded around their location and find the top applications for their specific device.

## 3. Collaborative Filtering

Collaborative filtering is a technique used to build recommender systems. Overall, collaborative filtering is the process of filtering for information or patterns using techniques involving collaboration among multiple agents, viewpoints, data sources, etc [6]. To predict the item's rating for the target user, collaborative filtering techniques typically proceed as follows [7]:

- Identify a set of like-minded users (in terms of the target user)
- Use the ratings given by these users on the target product to predict the rating of the target user.

Like-minded users are those users that rated similarly to the same application the target user have rated, i.e. users that tend to like and/or dislike the same items as the target user. The set of like-minded users is typically calculated by using a similarity function like Cosine or the Pearson Correlation on the user ratings. For collaborative filtering to work there have to be an enough number of co-rated applications between users. These are the steps usually followed by a collaborative filtering algorithm [7]:

- Y Identification of the ratings of the target user
- Y Identification of the users with ratings more similar to the ratings of the target user
- Y Identification of the applications evaluated by the similar users
- Y Generation of a rating for each identified application
- Y Recommendation of the top-*N* applications to the target user

#### 4. Comparative Analysis of Collaborative Filtering

In this section, we present the results of comparison of collaborative filtering in Java and R statistical software.

In this section, we describe the implementation of app recommendation system. We implement the proposed system in Java language. We used the computer with an Intel i5-760 quad core processor running at 2.80GHz in Windows Operating System with 16GB of main memory. The java implementation of collaborative filtering is Apache Mahout. Apache Mahout [4] is a new open source project by the Apache Software Foundation (ASF) with the primary goal of creating scalable machine-learning algorithms that are free to use under the Apache license. Mahout's primary features are [4]:

- Y Taste collaborative filtering. Taste is an open source project for collaborative filtering started by Sean Owen on SourceForge and donated to Mahout in 2008.
- Y Several Map-Reduce enabled clustering implementations, including k-Means, fuzzy k-Means, Canopy, Dirichlet, and Mean-Shift.
- Y Distributed Naive Bayes and Complementary Naive Bayes classification implementations.
- Y Distributed fitness function capabilities for evolutionary programming.
- Y Matrix and vector libraries.

According to [7], there are following advantages and disadvantages of Apache Mahout. Advantage of Mahout is that it gives a good novelty of recommendation, since the prediction is issued by other user likings and not on applications that are similar to what the user has liked. There is also no need for additional information such as a description of the user or product features. However, there are too less data to compute the similarity between the users. One possibility is to fill in a questionnaire when the user starts to use the recommender system.

In recent years, the R statistical software has emerged as one of the most used tools of data scientists. According to Revolution Analytics [4], R statistical software is used by over 2 million analysts worldwide both in industry and academic world. R statistical software can be used to solve challenging problems in fields ranging from computational biology to quantitative finance. This resulted in adoption of R statistical software by large organizations such as Facebook, Google and LinkedIn.

The R statistical software proposed a package called recommendlab [5] that implements the collaborative filtering technique. The package uses the abstract ratingMatrix to provide a common interface for rating data. ratingMatrix implements many methods typically available for matrix-like objects. For example, dim(), dimnames(), colCounts(), rowCounts(), colMeans(), rowMeans(), colSums() and

rowSums(). Additionally sample() can be used to sample from users (rows) and image() produces an image plot.

There are several reasons that make R statistical software a good choice for analyzing the Big Data. First, R statistical software includes all data manipulation, statistical model, and chart that you need when analyzing the Big Data. You can also easily find, download and use methods in statistics and predictive modeling from popular researchers in data science. Second, R statistical software provides a powerful graphical tool to analyze the Big Data. Specifically, it offers not only basic charts and graphs, but also multidimensional data with multi-panel charts, 3-D surfaces and more. Third, R is a programming language designed for data analysis, who works with Big Data. Experienced R programmers produce data analyses faster than users of legacy statistical software, with the flexibility to mix-and-match models for the best results. However, with R the user interface of such systems can be very complex.

#### 5. Conclusion

In this paper, we have studied the recommendation system based on collaborative filtering. We presented a study on comparison analysis of collaborative filtering in Java and R statistical software. Specifically, we implemented the collaborative filtering using Java's Apache Mahout and R recommendlab package. We evaluated both methods and described the advantages and disadvantages of using these methods in order to implement collaborative filtering. The results of comparison showed that using R statistical software is more beneficial in making collaborative filtering. Because, R statistical software includes all data manipulation, statistical model, and chart that you need when analyzing the Big Data. You can also easily find, download and use methods in statistics and predictive modeling from popular researchers in data science.

#### Acknowledgement

This work was supported by the IT R&D program of MKE/KEIT. [10041854, Development of a smart home service platform with real-time danger prediction and prevention for safety residential environments].

#### Reference

- [1] S. Perez. Mobile app martetplace. Available at: <http://www.readwriteweb.com>
- [2] Appazaar. Available at: <http://appazaar.net/>
- [3] Appaware. Available at: <http://appaware.com/>
- [4] Apache Mahout. Available at: <http://mahout.apache.org/>
- [5] Recommenderlab. Available at: <http://cran.r-project.org/web/packages/recommenderlab/index.html>
- [6] Wikipedia. Collaborative Filtering. Available at: [http://en.wikipedia.org/wiki/Collaborative\\_filtering](http://en.wikipedia.org/wiki/Collaborative_filtering)
- [7] Pergher M., Steingruber T. Android Apps Recommender System. Free University of Bolzanom Bozen, 2011.