ISSN No: 2454-9614

Movie Recommendation System Based On Agglomerative Hierarchical Clustering

P. Rengashree, K. Soniya*, ZeenathJasmin Abbas Ali , K. Kalaiselvi

Department Of Computer Science and Engineering, S.N.S College of Engineering,
Coimbatore, Tamilnadu, India.

*Corresponding Author: ZeenathJasmin Abbas Ali,

E-mail: rengashree1995@gmail.com,

Received:10/11/2015, Revised: 15/12/2015 and Accepted: 08/03/2016

Abstract

An increasing number of services are emerging on the Internet. As a result, service-relevant data become too big to be effectively processed by traditional approaches. In view of this challenge, a Clustering-based Collaborative Filtering approach (ClubCF) is proposed in this paper, which aims at recruiting similar services in the same clusters to recommend services collaboratively. Technically, this approach is enacted around two stages. In the first stage, the available services are divided into small-scale clusters, in logic, for further processing. At the second stage, a collaborative filtering algorithm is imposed on one of the clusters. Since the number of the services in a cluster is much less than the total number of the services available on the web, it is expected to reduce the online execution time of collaborative filtering. Application use existing data partitioning and clustering algorithms to partition the set of items based on user rating data. Predictions are then computed independently within each partition. Ideally, partitioning will improve the quality of collaborative filtering predictions and increase the scalability of collaborative filtering systems. **Reviewed by* **ICETSET'16** *organizing committee*

1. Objective

Spurred by service computing and cloud computing, an increasing number of services are emerging on the Internet. As a result, service-relevant data become too big to be effectively processed by traditional approaches. In view of this challenge, a Clustering-based Collaborative Filtering approach (ClubCF) is proposed in this paper, which aims at recruiting similar services in the same clusters to recommend services collaboratively**.**

2. Introduction

2.1 Outline of the Project

BIG data has emerged as a widely recognized trend, attracting attentions from government, industry and academia. Generally speaking, Big Data concerns large-volume, complex, growing data sets with multiple, autonomous sources. Big Data applications where data collection has grown tremendously and is beyond the ability of commonly used software tools to capture, manage, and process within a “tolerable elapsed time” is on the rise. The most fundamental challenge for the Big Data applications is to explore the large volumes of data and extract useful information or knowledge for future actions.

3. Working Environment

3.1 Problem Definition and Description System Analysis

3.1.1 Problem Definition

The most fundamental challenge for the Big Data applications is to explore the large volumes of data and extract useful information or knowledge for future actions. The basic assumption of user-based CF is that people who agree in the past tend to agree again in the future. Different with user-based CF, the item-based CF algorithm recommends a user the items that are similar to what he/she has preferred before.

In traditional CF algorithms, to compute similarity between every pair of users or services may take too much time, even exceed the processing capability of current RSs. Consequently, service recommendation based on the similar users or similar services would either lose its timeliness or could not be done at all.

3.1.2 Features

- Creating a data set with the Kite SDK
- Developing custom Flume components for data ingestion
- Managing a multi-stage workflow with Oozie
- Analyzing data with Crunch
- Writing user-defined functions for Hive and Impala
- Transforming data with Morphlines Indexing data with Cloud era Search.

3.2 Existing System

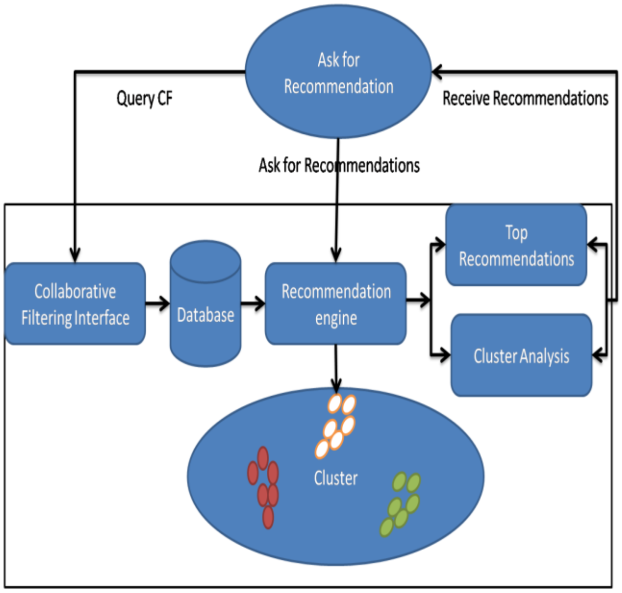
- The most fundamental challenge for the Big Data applications is to explore the large volumes of data and extract useful information or knowledge for future actions.
- The basic assumption of user-based CF is that people who agree in the past tend to agree again in the future. Different with user-based CF, the item-based CF algorithm recommends a user the items that are similar to what he/she has preferred before.
- In traditional CF algorithms, to compute similarity between every pair of users or services may take too much time, even exceed the processing capability of current RSs.
- Consequently, service recommendation based on the similar users or similar services would either lose its timeliness or couldn’t be done at all.

3.3 Proposed System

- We proposed a Agglomerative Hierarchal Clustering or Hierarchal Agglomerative Clustering

- Clustering are such techniques that can reduce the data size by a large factor by grouping similar services together.
- A cluster contains some similar services just like a club contains some like-minded users. This is another reason besides abbreviation that we call this approach ClubCF.
- This approach is enacted around two stages. In the first stage, the available services are divided into small-scale clusters, in logic, for further processing. At the second stage, a collaborative filtering algorithm is imposed on one of the clusters.
- This similarity metric computes the Euclidean distance *d between two such user points This value* alone doesn't constitute a valid similarity metric, because larger values would mean more-distant, and therefore less similar, users. The value should be smaller when users are more similar.

4. System Architecture



4.1 Description of Modules

Modules:

- Login Module
- Add movie Details
- Data Pre Processing
- Collaborative Filtering approach to build the recommendation Engine.

4.2 Modules Description

4.2.1 Data Pre Processing

- The training data, we are given a list of vectors (u; m; r; t), where u is a user ID, m is a movie ID; r is the rating u gave to m, and t is the date.
- After training, we output predictions for a list of user-movie pairs. We measure error by using the root mean squared error.
- After pre-processing we output the movie ids with the corresponding users and their ratings with; separated files.

4.2.2 Data Clustering

- We cluster the people based on the movies they watched and then cluster the movies based on the people that watched them.
- The people can then be re-clustered based on the number of movies in each movie cluster they watched.
- Movies can similarly be re-clustered based on the number of people in each person cluster that watched them.
- On the first pass, people are clustered based on movies and movies based on people.
- On the second and subsequent passes, people are clustered based on movie clusters, and movies based on people clusters.
- A cluster contains some similar services just like a club contains some like-minded users.

4.2.3 Collaborative Filtering approach to build the recommendation

4.2.3.1 Engine

- This similarity metric computes the Euclidean distance d between two such user points this value alone doesn't constitute a valid similarity metric, because larger values would mean more-distant, and therefore less similar, users. The value should be smaller when users are more similar.
- Therefore, the implementation actually returns 1 / (1+*d).*
- The upside of this approach is that recommendation is fast at runtime because almost everything is pre-computed.
- One could argue that the recommendations are less personal this way, because recommendations are computed for a group rather than an individual.
- This approach may be more effective at producing recommendations for new users, who have little preference data available.

4.2.4 Agglomerative Hierarchal Clustering or Hierarchal Agglomerative Clustering

- Clustering are such techniques that can reduce the data size by a large factor by grouping similar services together.

- A cluster contains some similar services just like a club contains some like-minded users. This is another reason besides abbreviation that we call this approach ClubCF.

Similarity by Euclidean distance:

- This similarity metric computes the Euclidean distance *d between two such user points This value* alone doesn't constitute a valid similarity metric, because larger values would mean more-distant, and therefore less similar, users. The value should be smaller when users are more similar.
- Therefore, the implementation actually returns 1 / (1+*d).*

4. Conclusion

In this paper, we present a ClubCF approach for big data applications relevant to service recommendation. Before applying CF technique, services are merged into some clusters via an AHC algorithm. Then the rating similarities between services within the same cluster are computed. As the number of services in a cluster is much less than that of in the whole system, ClubCF costs less online computation time.

Moreover, as the ratings of services in the same cluster are more relevant with each other than with the ones in other clusters, prediction based on the ratings of the services in the same cluster will be more accurate than based on the ratings of all similar or dissimilar services in all clusters. These two advantageous of ClubCF have been verified by experiments on real-world data set.

References

[1] M. A. Beyer and D. Laney, "The importance of "big data": A definition," Gartner, Tech. Rep., 2012.

[2] X. Wu, X. Zhu, G. Q. Wu, et al., "Data mining with big data," IEEE Trans. on Knowledge and Data Engineering.

[3] R. S. Sandeep, C. Vinay, S. M. Hemant, "Strength and Accuracy Analysis of Affix Removal Stemming Algorithms," International Journal of Computer Science and Information Technologies.

[4] T. Niknam, E. TaherianFard, N. Pourjafarian, et al., "An efficient algorithm based on modified imperialist competitive algorithm and K-means for data clustering," Engineering Applications of Artificial Intelligence.

[5] X. Liu, Y. Hui, W. Sun, et al., "Towards service composition based on mashup," in Proc. of IEEE Congress on Services

[6] X. Wu, X. Zhu, G. Q. Wu, et al., "Data mining with big data," IEEE Trans. on Knowledge and Data Engineering, vol. 26, no. 1, pp. 97-107, January 2014.

[7] A. Rajaraman and J. D. Ullman, "Mining of massive datasets," Cambridge University Press, 2012.

[8] Z. Zheng, J. Zhu, M. R. Lyu. "Service-generated Big Data and Big Data-as-a-Service: An Overview," in Proc. IEEE BigData, pp. 403-410, October 2013.

[9] A. Bellogín, I. Cantador, F. Díez, et al., "An empirical comparison of social, collaborative filtering, and hybrid recommenders," ACM Trans. on Intelligent Systems and Technology, vol. 4, no. 1, pp. 1-37, January 2013.

[10] W. Zeng, M. S. Shang, Q. M. Zhang, et al., "Can Dissimilar Users Contribute to Accuracy and Diversity of Personalized Recommendation?," International Journal of Modern Physics C, vol. 21, no. 10, pp. 1217-1227, June 2010.

[11] T. C. Havens, J. C. Bezdek, C. Leckie, L. O. Hall, and M. Palaniswami, "Fuzzy c-Means Algorithms for Very Large Data," IEEE Trans. on Fuzzy Systems, vol. 20, no. 6, pp. 1130-1146, December 2012.