

An efficient recommender system approach for collaborative filtering

1. Chinnapaga Ravi 2. M Bal Raju 3. N Subhash Chandra

1. Research scholar 2. Professor 3. professor

Computer science and engineering, JNTUH, Hyderabad, India

Abstract:

Many companies are used recommender system to give basic knowledge and feedback information about product and service to the customers. Generate the recommendations based on the likes usage history and preferences. Collaborative filtering generates recommendation by using the previous recommendations of other users. The collaborative filtering is the basic idea for analysis and customer behaviour from the past history that gives the future prediction that may like similar items. When process the large amount of data that old and previous recommender techniques are not efficient for sparsity, efficiency and scalability. Single node machines are the time consuming on large data sets. Parallel distributed file filtering model (PDFM), The PDFM recommender system is working on parallel files for MapReduce framework and uses correlation as similarity metric. Parallel distributed file filtering model (PDFM) allows distributed processing of big data across multiple clusters of nodes.

Introduction

Recently, researchers have developed new methods for collaborative filtering (Goldberg et al. 1992; Koren 2008; Rendle 2012). A new direction is to apply transfer learning to collaborative filtering (Li, Yang, and Xue 2009b; Pan et al. 2011b), so that one can make use of auxiliary data to help improve the rating prediction performance. However, in many industrial applications, precise point-wise user feedbacks may be rare, because many users are unwilling or unlikely to express their preferences accurately. Instead, we may obtain estimates of a user's tastes on an item based on the user's additional behaviour or social connections. For example, suppose that a person Peter is watching a 10-minute video. Suppose that Peter stops watching the video after the first 3 minutes. In this case, we may estimate that Peter's preference on the movie is in the range of 1 to 2 stars with a uniform distribution. As another example in social media, suppose that Peter reads his followers' posts in a microblog about a certain movie¹. Suppose that his followee John posts a comment on the movie with 3 stars. In addition, Peter's other followees Bob gives 4 stars, and Alice gives 5 stars. Then, with this social impression data, we should be able to obtain a potential rating distribution for Peter's preference on the movie. We call such a rating distribution as an uncertain rating, since it represents a rating spectrum involving uncertainty instead of an accurate point-wise score.

To leverage such uncertain ratings as described above, we plan to exploit techniques in transfer learning (Pan and Yang 2010). To do this, we have to answer two fundamental questions: "what to transfer" and "how to transfer" in transfer learning (Pan and Yang 2010). In particular, we have to decide (1) what knowledge to extract and transfer from the auxiliary uncertain ratings, and (2) how to model the knowledge transfer from the auxiliary uncertain rating data to the target numerical ratings in a principled way. As far as we know, there has not been existing research work on this problem.

Several existing works are relevant to ours. Transfer learning approaches are proposed to transfer knowledge in latent feature space (Singh and Gordon 2008; Yoo and Choi 2009; Pan et al. 2010; Cao, Liu, and Yang 2010; Pan et al. 2011b; Vasuki et al. 2011), exploiting feature covariance (Adams, Dahl, and Murray 2010) or compressed rating patterns (Li, Yang, and Xue 2009a; 2009b). In collaborative filtering, transfer learning methods can be adaptive (Li, Yang, and Xue 2009a; Pan et al. 2010) or collective (Singh and Gordon 2008; Li, Yang, and Xue 2009b; Yoo and Choi 2009; Cao, Liu, and Yang 2010; Pan et al. 2011b; Vasuki et al. 2011). Other works, such as that by Ma et al. (Ma, King, and Lyu 2011), tend to use auxiliary social relations and extend the rating generation function in a model-based collaborative filtering method (Salakhutdinov and

Mnih 2008). Zhang et al. (Zhang et al. 2010) generate point-wise virtual ratings from sentimental polarities of users' reviews on items, which are then used in memory-based collaborative filtering methods for video recommendation. However, these works do not address the uncertain rating problem.

RELATED WORK:

Priority and ranking system of product in e-commerce, that more helpful in marketing areas for consumers decision making. Based on the collaborative method recommender system widely addressed the ranking system of the items. There are many previous techniques like

Collaborative Filtering and Prediction uncertainty: collaborative filtering main goal is to help consumers make necessary move by providing relevant items information what they may be interested. Sometimes previous feedback is not sufficient and inappropriate to choose good ranking product, then customer preferences may change, that is called sparsity problem. To avoid that Item neighbourhood based collaborative filtering technique was proposed for scalability by calculating item- item similarities.

Recommender Systems: Based on the previous history of the customer purchase behaviour and what items are preferred, recommender system gives ranked items list and relevant items history for the customer knowledge.

3 types of recommended systems:

1). Content based recommended systems: these recommend items based on a comparison between user data and content of items. It is also called as *cognitive filtering*. It has some limitations like content description, subject domain specialization, over specialization.

2). Collaborative filtering based recommender systems: it generates recommendations by using the previous recommendations of other users. It is also called social filtering. The basic idea behind these is, if a user likes a certain type of items in past then it may like similar items in future.

a) User- user collaborative filtering.

b) Item- item collaborative filtering.

3). Hybrid recommender systems: these recommender systems overcome the above limitations. Most of the recommender systems have not consider user preferences. We addressed this problem and proposed new recommender system. We used novel approach for CF. the preferences of user are taken in the form of keywords. The implementation is done on hadoop MapReduce parallel computing framework.

Collaborative Filtering and MapReduce: Collaborative Filtering calculation is a great customized proposal calculation; it's broadly utilized as a part of numerous business recommender frameworks. Community filtering calculation is a calculation in view of the accompanying three suppositions thought.

The Pearson relationship coefficient is the most broadly utilized and served as a benchmark for CF. Generally, we use the Cosine similarity measure method, its calculating equation as follows:

$$\text{Sim}(x,y) = \frac{\sum_{s \in S_{xy}} r_{x,s} r_{y,s}}{\sqrt{\sum_{s \in S_{xy}} r_{x,s}^2} \sqrt{\sum_{s \in S_{xy}} r_{y,s}^2}} \quad (1)$$

Where r_x and r_y is rating on item That x and y are coe evaluated for Item rating calculation. Neighbours weights are included for Item rating. The equation for rating as fallow

$$r_{x,s} - \bar{r}_x + \frac{\sum_{y \in S_{xy}} (r_{y,s} - \bar{r}_x) \text{sim}(x, y)}{\sum_{y \in S_{xy}} \text{sim}(x, y)} \quad (2)$$

\bar{r}_x is the average rating of user x.

We propose new algorithm for the Collaborative Filtering calculation on parallel distributed data sets.

Map-Reduce: The Map-Reduce Model has two steps, first one is mapping and second one is Reduce step. In Map data is summarized and aggregated make action on the consolidated data. Customer has limitation for applying key for reorganization of and for course of action. It aggregates characteristics for smaller game plan.

4. Proposed System: Parallel distributed file filtering model (PDFM) recommendation system has three components, as shown in Fig. The components are Hadoop nodes, distributed recommendation engine and Hbase Storage.

Data Extraction: The movie datasets consists of comments and ratings which will be stored in clusters with the customer Id. Customer ratings are compared with other customer preferences which are similar, those will be formed to show the customer. MapReduce method for store & extract the data with the technique of Parallel distributed file filtering model (PDFM).

Data Analysis: Big data analysis system with Hadoop for recommender system, previous techniques such as python and Java for MapReduce on large datasets. We use Parallel distributed file filtering model (PDFM) to analyse movie review system.

Collaborative Filtering performs:

- (1) Collecting the user most preferred items,
- (2) Identified similar items
- (3) Recommendations evaluation.

User preferences extraction is : <Customer ID Movie Title, Rating> (1)

Then, we use Pearson correlation coefficient (PCC) measure to calculate the similarity. Compared to cosine similarity and Euclidean distance the PCC is better. It first finds the items rated by both users. Then calculates the sums and the sum of the squares of the product ratings for both the users and calculates the sum of the products of their ratings. Finally, it uses these results to calculate the Pearson correlation coefficient.

$$S_{xx} = \sum X^2 - (\sum X)^2 / n$$

$$S_{yy} = \sum Y^2 - (\sum Y)^2 / n$$

$$S_{xy} = \sum XY - (\sum X)(\sum Y) / n$$

$$r = S_{xy} / \sqrt{S_{xx} S_{yy}} \text{ square root}$$

RESULTS:

MapReduce for Recommender: it consists of 3 phases as follows:

A. Map Phase: This is the first phase in this phase. The Hadoop mapper setup function to build Recipe List based on Preparation time and ingredients used.

B. Reduce Phase: In this stage the hadoop stage would produces a few reducers naturally. The reducer gathers the Recipe subtle elements and sorts them as indicated by the given arrangement.

C. Web application: In this stage we created web application using J2EE architecture where the application takes hadoop map reduced output files as input to database and present the recipes on the web front end to the user whenever user gives rating the ratings will be stored in the database.

6. Conclusion

This paper gives a scalable product recommendations collaborative filtering for big data on a Parallel distributed file filtering model (PDFM), An optimized HBase gives better performance. For low latency applications HBase is highly preferred because of distributed architecture and leverage the power of Apache Hadoop. As the size of data increases the Hadoop performs well by adding more data nodes into the processing. Collaborative Filtering is one of the best algorithm for the product recommendations.

References

- [1] Apache Hadoop, <https://hadoop.apache.org/>
- [2] Zhiyang Jia, Wei Gao, Yuting Yang, Xu Chen, User-based Collaborative Filtering for Tourist Attraction Recommendations, 2015 IEEE International Conference on Computational Intelligence & Communication Technology, 978-1-4799-6023-1/15 © 2015 IEEE DOI 10.1109/CICT.2015.20
- [3] Jyoti Gupta, Jayant Gadge, Performance Analysis of Recommendation System Based On Collaborative Filtering and Demographics, IEEE International Conference on Communication, Information & Computing Technology (ICCICT), 978-1-4799-5521-3, 2015 DOI: 10.1109/ICCICT.2015.7045675
- [4] Shunmei Meng, Wanchun Dou, Xuyun Zhang, and Jinjun Chen, KASR: A Keyword Aware Service Recommendation Method on MapReduce for Big Data Applications, IEEE 2014
- [5] Riyaz P.A., Surekha Mariam Varghese, Leveraging MapReduce with Hadoop for Weather Data Analytics, IOSR - Journal for Computer Science, 2015, volume 17, issue 3, pp 6-12.
- [6] Wu Yueping and Zheng Jianguo, "A research of recommendation algorithm based on cloud model", IEEE 2010.
- [7] Yanhong Guo et al., "An improved collaborative filtering algorithm based on trust in e-commerce recommendation system", IEEE 2010. [8] B.M. Sarwar et al., "Item-item Collaborative Filtering Recommendation Algorithms," 10th Int'l World Wide Web Conference, ACM Press, 2001, pp. 285-295.
- [9] B.M. Sarwar et al., "Analysis of Recommendation Algorithms for E-Commerce," ACM Conf. Electronic Commerce, ACM Press, 2000, pp.158-167.
- [10] Hao Ma, Irwin King and Michael R. Lyu, Effective Missing Data Prediction for Collaborative Filtering, ACM July 2327, 2007.
- [11] Xingyuan Li, A Review on Hadoop HDFS Infrastructure Extensions, Proceedings of 2013 IEEE Conference on Information and Communication Technologies (ICT 2013).
- [12] Mario Muoz-Organero, Gustavo A. Ramirez-Gonzalez, Pedro J. Muoz-Merino and Carlos Delgado Kloos, A Collaborative Recommender System Based on SpaceTime Similarities, IEEE CS 2010.
- [13] Bankim Patel, Atul Patel, Atul Patel, Big Data Analysis: Recommendation System with Hadoop Framework, 2015 IEEE International Conference on Computational Intelligence Communication Technology.
- [14] Martin Wiesner and Daniel Pfeifer, Health Recommender Systems: Concepts, Requirements, Technical Basics and Challenges, Int. J. Environ. Res. Public Health, 2014.
- [15] Ratnamala Mantri, Rajesh Ingle and Prachi Patil, SCDP: Scalable, Cost-Effective, Distributed and Parallel Computing Model for Academics, ICNCS, Volume: 5, 77-80, ISBN:978-1-4244-8677-9, IEEE 2011.

[16]Ratnamala Mantri, Ashwini Jawalikar, \textit{Implementation and Performance Analysis of Academic\$\$_\$MapReduce Algorithm (AcdMR) , International Journal of Computer Applications, Volume:121, Issue:19, July 2015.

[17]Sonali Ghodake, Ratnamala Mantri, Survey on Recommender System using Distributed Framework, International Journal of Science and Research (IJSR), Volume: 5, Issue: 1, January 2016.

