Research Paper Engineering

Sentiment Analysis Based Hotel Recommendation System

Khushboo Ramesh Shrote | MTech student at Computer Science and Engineering Department Government College of Engineering Amravati Amravati, India

Prof. Anil V. Deorankar | Associate Professor at Computer Science and Engineering Department Government College of Engineering Amravati Amravati, India

ABSTRACT

Big data refers to datasets that aren't solely massive, however additionally high in variety and velocity that makes them troublesome to handle using tradition tools and techniques. Nowadays Web services are very widespread. Number of services, customers and information produced is increased now a days. Recommender systems represent user preferences for the aim of suggesting things to get or examine. A variety of techniques are projected for activity recommendation, including content-based, collaborative, knowledge-based and different techniques. In this paper, we are presenting "Sentiment Analysis Based Hotel Recommendation System", to deal with the above challenges. To improve the scalability and efficiency we proposed the system in Hadoop framework.

KEYWORDS Rating; Big data; Reviews; Hadoop; Map Reduce; Sentiment Analysis

Introduction

With the growing range of different services, effectively recommending services that users most well-liked has become a vital analysis issue. Service recommender systems are shown as valuable tools to assist users modify services over load and supply acceptable recommendations to them. Examples of such sensible applications include CDs, book, web content and numerous alternative product currently use recommender systems. Over the last decade, there has been a lot of analysis done each in business and world on developing new approaches for service recommender systems. Many major e-commerce Websites are used as recommendation systems to produce relevant suggestions to their customers.

problem definition

In traditional recommender systems we might face the problems of scalability and efficiency in case of large scale data. Capturing and analyzing this data is hard. Traditionally personalized requirements of end users are not considered. The rating list itself is provided as the recommendation. Passive users review about any product or system is ignored. In this paper a Review Based Service Recommendation method is proposed, to achieve an efficiency of a Recommender System. It aims at presenting a personalized service recommendation list and recommending the most appropriate services to the users effectively.

literature survey

Recommender systems are developing as a powerful tool in both industry and academia. In [1] authors propose a KASR method for personalized recommendation. In this paper user based collaborative filtering algorithm is used. To make the method more efficient and scalable it is implemented on Hadoop. Jaccard coefficient and Cosine similarity measure is used for evaluation. They show that the proposed recommendation method is better than the existing traditional methods. Merits are 1. Scalable 2. More efficient than traditional methods. Demerits are Jaccard Coefficient method is not so accurate. Users positive and negative reviews are not differentiated. Sentiments in the text is not considered for calculation. In [2] authors propose an active web service recommendation. Web usage history and QoS are the main criteria for recommendation. Using this approach top k services are generated for users. Usage history count is only used for ranking. Merits are 1.Higher recall ratio and accuracy; 2. Show the strength of the relationship between users. Demerits are Passive users reviews about the website is not considered. Usage history count is only used for ranking. In [3] authors propose a Bayesian inference based recommendation in online social networks. In this content ratings are shared with friends. Conditional probability is used for calculating rating similarity. Based on similarity score ranking is done. They show that the proposed Bayesian inference-based recommendation is better than the existing trust based recommendation. Merits are 1. Higher accuracy

via friends' recommendation; 2. Solve the problem of large size of particle in collaborative filtering recommendation Demerits are 1. There is a Cold start and rating sparseness problem. In [4] authors propose recommender system for sport videos, transmitted over the Internet and broadcast, in the context of large-scale events, which has been tested for Olympic Games. The recommendation is based on audiovisual consumption and not on the number of users, running only on the client side. Merits are 1. This avoids the concurrence, computation and privacy problems of central server approaches in scenarios with a large number of users, such as the Olympic Games. Whole video have to recommend. Demerits are 1. Specific video fragment can't be recommended using this approach. In [5] authors propose a probabilistic personalized travel recommendation model. For mining demographics for travel landmarks and paths people attributes and photos are used which are effective, and thus benefiting personalized travel recommendation services. In [6] authors propose quality of service ranking prediction for cloud services. Rating based approaches and ranking based approaches are studied in this paper. Merits are 1. the users can obtain QoS ranking prediction as well as detailed QoS value prediction. Demerits are 1. Applications in other field need further verification.

System architecture

System Architecture of proposed method is shown in Figure 2. A dataset of hotels is taken for evaluation. It consists of date followed with short comment and a long comment.

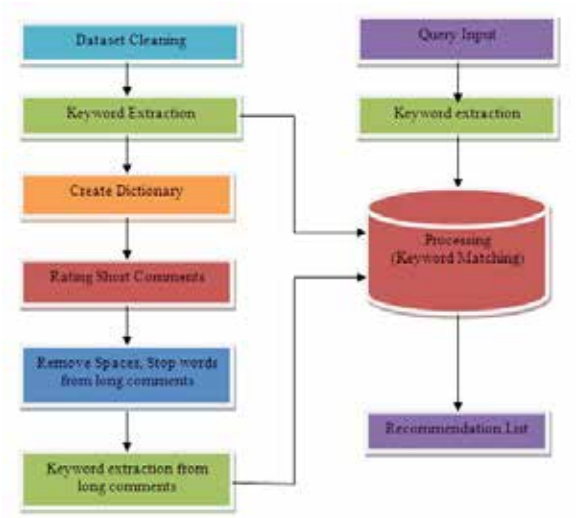

Figure 1. Proposed System Architecture

1. Preprocessing
Ignore data up to first tab as it is a date. Remove stop words in the reviews to avoid affecting the quality of the keyword extraction in the next stage. And the Porter Stemmer algorithm is used to remove the commoner morphological and in flexional endings from words in English. Its main use is as part of a term normalization process that is usually done when setting up Information Retrieval systems.

2. Keyword Extraction
Each review will be transformed into corresponding keyword set. For example if a user type word transport the corresponding similar meaning words such as transportation, ship, move, convey must be present in the domain thesaurus.

For keyword in case of long comments only those long comments are preferred whose short comment rating is above the threshold value. Map Reduce algorithm can be applied so that same keywords must not be repeated. Keywords obtained are stored in a database as [id, city, hotel, keywords]. Using queries in SQL we can retrieve any information such as to obtain whole information we fire a query select * from recommendationsystem.hotelinfo;

3. Create Dictionary
Dictionary is a repository where a collection of keywords and their corresponding rating value is stored. Keywords are extracted from short and long comments. Their rating value is calculated by manual testing and it is set spaced with a tab. When we extract any keyword we pass it to check its rating value. This value is then obtained to use in score calculation in case of Sentiment Analysis.

4. Rating Short Comments
All short comments of a particular hotel is captured and stored in a file. Name that file as file 1. Then keywords are extracted their corresponding rating values are checked. Sentiment Analysis is applied and a total rate value is calculated. If this value is above threshold value then that hotel is included otherwise ignored.

5. Recommendation List
Using the total score value a list of top-k services are recommended. Only positive values are considered here.

Implementation results
Each time you have to write following commands and run Hadoop on terminal :

1. su – hduser :This command switches normal user to hduser.
2. hdfs namenode –format: This command is used to format namenode. Namenode contains metadata, i.e. it contains address of data which is present on the datanode. Each time you have to format the namenode so as new data loaded will be a fresh data and metadata storage is efficient. Each time a new id is provided to the namenode.
3. start-all.sh : This command is used to start working of all the daemons .
4. jps : It is java virtual machine process status tool. It is used to see the number of daemons running on your local machine.
5. stop-all.sh :This command is used to stop the working of all hadoop daemons.

```
siva@siva-desktop:~$ jps
3168 SecondaryNameNode
2977 DataNode
3313 Jps
2853 NameNode
siva@siva-desktop:~$ start-yarn.sh
starting yarn daemons
starting resourcemanager, logging to /usr/lib/hadoop/hadoop-2.
localhost: starting nodemanager, logging to /usr/lib/hadoop/ha
siva@siva-desktop:~$ jps
3168 SecondaryNameNode
2977 DataNode
2853 NameNode
3831 ResourceManager
3963 NodeManager
4304 Jps
siva@siva-desktop:~$
```

Figure 2. Run Hadoop on Terminal

Once you select the city from the combo box and enter keywords of your interest. These keywords are matched with the words stored in the database. The corresponding hotel reviews are passed to sentiment analysis. A total score is calculated using rating dictionary.

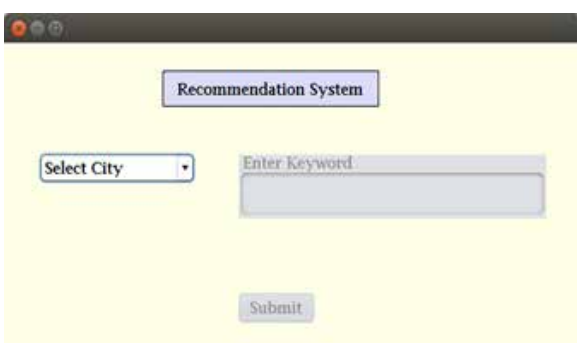

Figure 3. Recommendation System Home Page

We can add keywords of our own choice. These keywords entered if not in the lowercase they are converted to lowercase then compared with the keywords present in the database. Where a match is found that hotels sentiment score is calculated. If sentiment score is positive then it is recommended.

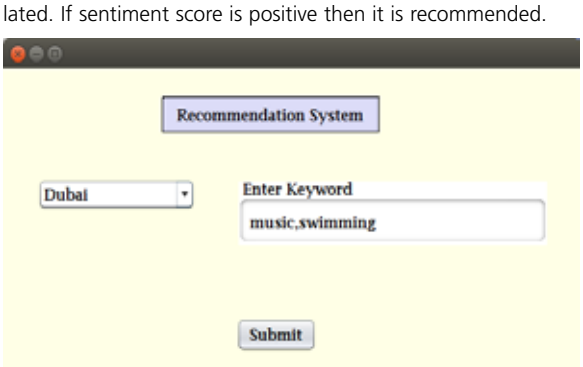

Figure 4. Run Application

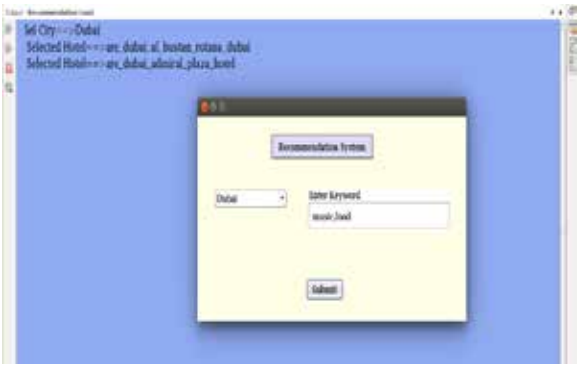

Figure 5. Top-k Recommendations

conclusions

In this paper sentiment analysis is used to recommend services to users. User based collaborative filtering algorithm is used to generate appropriate recommendations. Users can give more than one keyword as a preference. Hotel with highest rating value is ranked one and recommended first. This ranking is changeable. So we have to make updations in the rating dictionary as passive user's reviews changes. So, Recommendation is dynamic and more realistic. We are using

References

1. Shunmei Meng, Wanchun Dou, Xuyun Zhang, Jinjun Chen," KASR:A Keyword-Aware Service Recommendation Method on Map Reduce for Big Data Applications" IEEE Transactions On Parallel And Distributed Systems, TPDS-2013-12-1141.
2. X. Yang, Y. Guo, Y. Liu, "Bayesian-inference based recommendation in online social networks," IEEE Transactions on Parallel and Distributed Systems, Vol. 24, No. 4, pp. 642-651, 2013.
3. G. Kang, J. Liu, M. Tang, X. Liu and B. cao, "AWSR: Active Web Service Recommendation Based on Usage History," 2012 IEEE 19th International Conference on Web Services (ICWS), pp. 186-193, 2012.
4. Yan-Ying Chen, An-Jung Cheng, "Travel Recommendation by Mining People Attributes and Travel Group Types From Community-Contributed Photos" IEEE Transactions on Multimedia, Vol. 15, No. 6, October 2013.
5. M. Alduan, F. Alvarez, J. Menendez, and O. Baez, "Recommender System for Sport Videos Based on User Audiovisual Consumption," IEEE Transactions on Multimedia, Vol. 14, No.6, pp. 1546-1557, 2013.
6. Zibin Zheng, Xinmiao Wu, Yilei Zhang,Michael R. Lyu, Fellow,and Jianmin Wang," QoS Ranking Prediction for Cloud Services" IEEE Transactions On Parallel And Distributed Systems, Vol. 24, No. 6, June 2013.
7. G.Linden, B. Smith, and J. York, "Amazon.com Recommendations: Item to Item Collaborative Filtering," IEEE Internet Computing, Vol. 7, No.1, pp.76-80, 2003.
8. Fuzhi Zhang, Huilin Liu, Jinbo Chao, "A Two-stage Recommendation Algorithm Based on K-means Clustering In Mobile E-commerce", Journal of Computational Information Systems, Vol. 6, Issue 10, pp. 3327-3334, 2010.
9. Brian McFee, Luke Barrington and Gert Lanckriet, "Learning Content Similarity for Music Recommendation" IEEE Transactions on Audio, Speech, and Language Processing, Vol. 20, No. 8, 2012.
10. Z. D. Zhao, and M. S. Shang, "User-Based Collaborative-Filtering Recommendation Algorithms on Hadoop," In the third International Workshop on Knowledge Discovery and Data Mining, pp. 478-481, 2010.
11. D. Agrawal, S. Das, and A. El Abbadi, "Big Data and Cloud Computing: New Wine or Just New Bottles?" Proc. VLDB Endowment, vol. 3, no. 1, pp. 1647-1648, 2010.
12. J. Dean and S. Ghemawat, "MapReduce: Simplified Data Processing on Large Clusters," Comm. ACM , vol. 51, no. 1, pp. 107-113, 2005.
13. S. Ghemawat, H. Gobioff, and S. T. Leung, "The Google File System," Proc. 19th ACM Symp. Operating Systems Principles , pp. 29- 43, 2003
14. Z. Luo, Y. Li, and J. Yin, "Location: A Feature for Service Selection in the Era of Big Data," Proc. IEEE 20th Int'l Conf. Web Service, pp. 515-522, 2013.
15. B. Issac and W.J. Jap, "Implementing Spam Detection Using Bayesian and Porter Stemmer Keyword Stripping Approaches,"Proc. IEEE.