Volume 4, Issue 5, May-2017, pp. 161-168 ISSN (O): 2349-7084

International Journal of Computer Engineering In Research Trends

Available online at: www.ijcert.org

Product Rating using Opinion Mining

[1]Sunil B. Mane, [2]Kruti Assar, [3]Priyanka Sawant, [4]Monika Shinde

[1] *Assistant Professor, Department of Computer Engineering and Information Technology, College of Engineering Pune Pune - 411005, Maharashtra, India.*

[2,3,4] *B.Tech Student, Information Technology, College of Engineering Pune Pune - 411005, Maharashtra, India.*

Abstract :-Amazon.com is one of the largest electronic commerce website in the world which allows users to purchase different products and submit reviews on each one of them. The reviews allow the first-time buyers to understand the quality of the products and decide whether to make a purchase or not. The reviews result in unstructured big data which can be analyzed and used for recommendation of a product on the website. However, it is possible that some customers write fake reviews to promote or defame a particular brand. So it is important to detect and remove the fake reviews for providing the correct rating to the product. Also, it is necessary to create a fast and efficient system for analyzing big data. The present systems used for big data analysis are quite slow. So here, we use the Apache Spark framework for increasing the speed of processing the Amazon reviews. This paper provides a new implementation for analyzing Amazon reviews which involve detection of fake reviews, processing the genuine reviews using Apache Spark and finally rating the products.

Keywords: Opinion Mining, Apache Spark, Product Rating, Fake Review Detection, Natural Language Processing, Sentiment Analysis.

1. Introduction

Over the years, with advancement in technology, the business strategy has also changed. Now many e-commerce websites have emerged which are adopting new innovative ideas for publicity. One of the most important marketing schemes is providing a platform for online customer reviews. These online reviews help the customer to analyze different products and services and also provide a platform for comparing prices before taking a decision. Moreover, companies and vendors can frame new business strategies depending on the opinions provided by customers.

Amazon encourages its customers to give their feedback and write reviews on its website which are then analyzed and the one with most "helpful" hits is displayed on the front page. However, what if these reviews are fake? Few cases have been identified where people post incorrect reviews to defame a brand, or sometimes exquisite reviews are posted to increase the sales of a product. Hence it is important to detect certain opinions and eliminate the rest.

The real reviews can further be classified into positive and negative opinions, and sentiment analysis is performed on this data to finally rate different products to help the customer select one out of them.

2. Literature review

Much research has been carried out on Opinion Mining which is explained below.

2.1 Fake Review Detection

Presently, Amazon website uses some machine learning algorithms to select relevant features and decide the
Final rating of a product. However, it does not apply any algorithm to detect whether a review is fake or not. Few websites like Yelp.com and Fakespot.com can be used to detect fake reviews online, but there is no particular algorithm known to the world to filter reviews. Only a few relevant rules are designed for

this which we will be applying on our dataset in the form of database queries.

2.2 Opinion Mining

When opinion mining is performed on reviews, we try to find whether the reviewer is happy with the product or not. Hence the reviews written in natural language are processed to derive the sentiments from the words used. There are two main approaches available for sentiment analysis: supervised and unsupervised learning.

In supervised learning of sentiments, the text is annotated manually, and the algorithm classifies the sentiments based on this annotation. For such analysis, a large training dataset needs to be created which is labeled and is domain-specific. Thus the model generated by the supervised algorithm for one domain may not give accurate results for some other data belonging to a different domain.
On the other hand, unsupervised sentiment analysis creates a model that can be used on data from different domains. The lexicon-based approach allows us to use a dictionary of sentiments having a polarity assigned to each word. At the end, the total polarity of a sentence can be calculated, and the review can decide to be positive or negative by extracting the opinion.

Analysis of Twitter data has been the latest trend. There is much research carried out on designing better approaches for performing sentiment analysis of tweets. Sunil B. Mane, Y. Sawant, S. Kazi, and V. Shinde (2014) [9] have devised a new method for this which uses Naive-Bayes algorithm and a Hadoop cluster for distributed data processing. Another approach suggested by Eman M.G. Younis [5] uses the lexicon-based approach of sentiment analysis in R.

Most of the research for sentiment analysis is carried out for tweets, but there has been some study involving the Amazon reviews dataset too. Maria Soledad Elli, Yi-Fan Wang [20] mainly focus on finding the correlation between review's sentiment and customers and the correlation between a brand and its pricing design after performing sentiment analysis of reviews. They have used SVM algorithm along with Naive-Bayes for classification.

Since we want to analyze different categories of electronics products, we have decided to use the lexicon-based approach which works on various datasets of Amazon products.

2.3 Apache Spark

Apache Spark is a fast and efficient cluster computing and data processing framework. The architecture of Apache Spark is shown in the figure below:

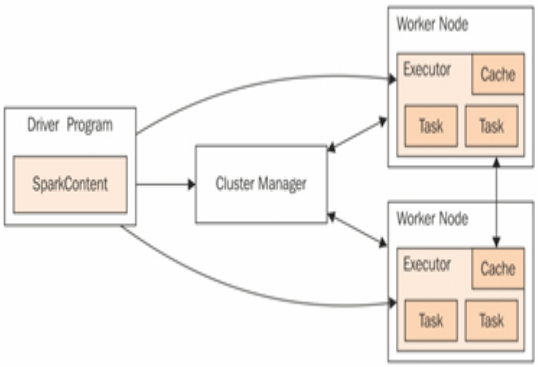


Figure 1. Architecture of Apache Spark framework

Spark has a driver (SparkContext) which is connected to the Cluster Manager or the Master node that is responsible for allocating resources on different Worker nodes following which the processes are distributed among the workers. Apache Spark performs in-memory cluster computing which helps in increasing the speed of the application. Thus Spark is considered about 10 times faster than Hadoop.

The most common platform used for processing big data at present is the Hadoop framework. The Hadoop Map-Reduce framework is used for analyzing sentiments, but it is said to be slow since it reads data from a cluster, performs the operations and writes back to each group separately unlike Apache Spark which processes entire data at once. The basic structure created by Spark is Resilient Distributed Dataset which means the dataset is divided into different logical partitions which allow parallel computation and faster processing of data. RDDs are also fault-tolerant because if one node fails, the other can take up the work of processing that data. Hence at many places, Hadoop is now being replaced with Spark.

K. Waddar and K. Shrinivas [18] in their paper of opinion mining on big data, have used Hadoop framework for data processing, Apache Mahout for machine learning and Hbase for creating databases. Here classification is performed using the Bayes classification algorithm. The model is first trained and then tested using sample data.

The majority of the methods of sentiment analysis stated in different papers use Hadoop Map Reduce. However, some researchers have also tried to use the latest Apache Spark framework for distributed processing of data. Nikolaos Nodarakis et al. (2016) [1] have used Apache Spark for sentiment analysis of Twitter data instead of Map Reduce. The paper highlights the advantage to Spark over Hadoop Map Reduce and for sentiment analysis, their algorithm focuses on hashtags and emoticons in the tweets. After extracting the features and constructing vectors, sentiment classification is performed using kNN

classification algorithm. Here it is stated that Spark is chosen because it works faster than Hadoop.

It is clear from the above analysis that Apache Spark provides a higher speed of computation as compared to Hadoop Map Reduce, so we have decided to use Spark as the underlying framework for processing our data.

3. Problem Definition

Detection of fake reviews, process only genuine reviews by performing opinion mining with Apache Spark as the underlying framework and use the results for rating products.

4. Dataset

The dataset used for analysis is the Amazon electronics product reviews. It is a large dataset which consists of a unique parameters about the product and the reviewers like in and reviewer ID respectively along with other attributes such as reviewer name, reviewer text message and time of review. The product details like title, brand, price, etc. can be extracted from the metadata provided which will be used for sentiment analysis.

5. Proposed Approach

The following diagram shows the architecture of our proposed system:

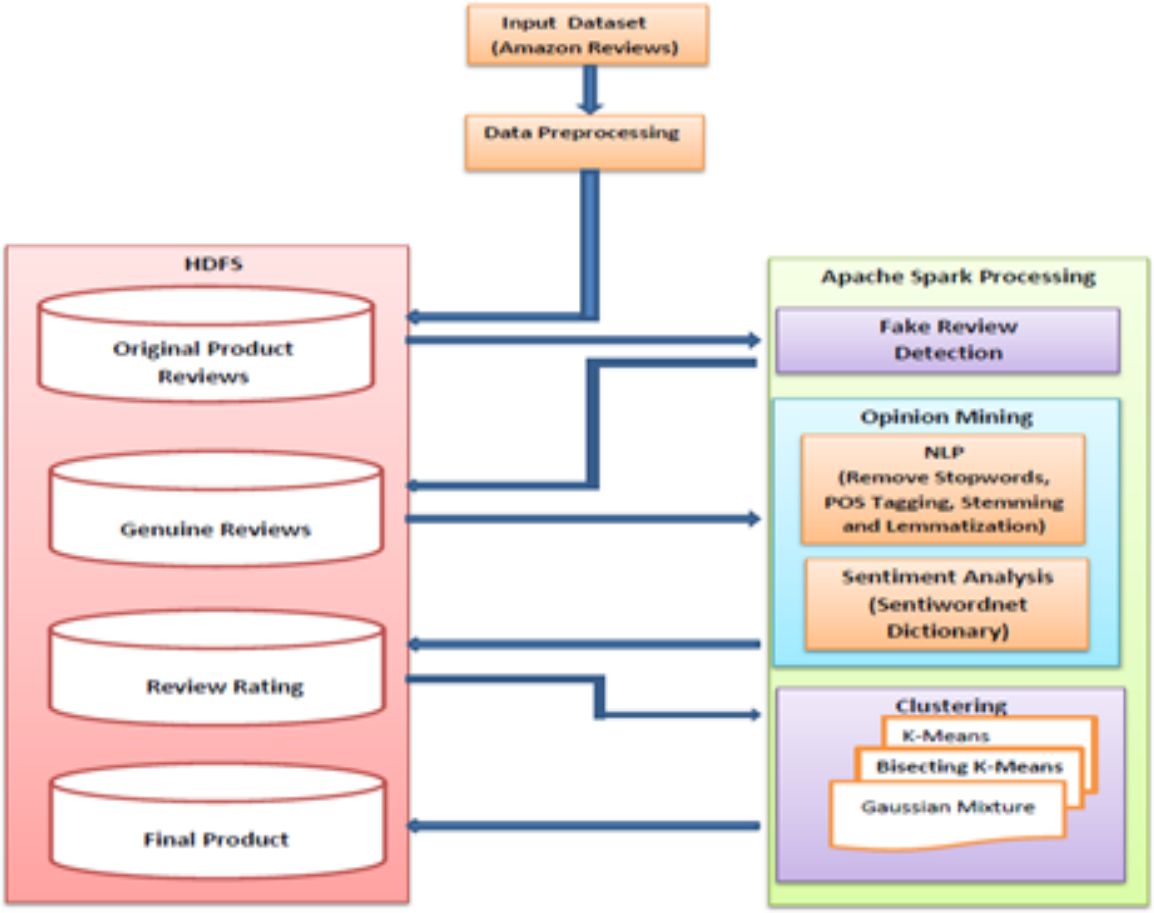


Figure 2. Proposed System Architecture

After obtaining the Amazon Electronics reviews dataset from Amazon and doing some pre-processing such as removing empty and duplicate reviews, the original reviews database is created. The Hadoop Distributed File System is used for storage. Hence, all the databases and files are saved on HDFS, and all the processing is done on Spark.

5.1 Fake Review Detection

The original reviews database is the input to the first process, the Fake review discovery. Some logical rules that are used to detect fake reviews are stated below:

- Same reviewer posting reviews frequently about the same product
- Many reviews being posted about the same product at the same time
- A particular user gives only the highest or the lowest rating to every product
- Multiple references to other people such as 'my family,' 'my sister,' etc.

For the implementation of these logical rules, we created HiveQL queries which are as follows:

a) Same reviewer posting reviews frequently about the same product

If the same reviewer posts multiple reviews about the same product, it is possible that he is paid to do so, either to promote a particular brand or to defame it. For this rule, we gave a constraint that no more than two reviews can be written by one reviewer for the same product. So here by using asin and reviewerID, we designed the following query which removes the reviews that satisfy this rule.

*val result = hiveContext.sql("insert overwrite table elecprodnew select * rom elecprodnew where (asin, reviewerID) in(select asin, reviewerID from elecprodnew group by asin, reviewerID having count(*) < 3)")*

b) Many reviews being posted about the same product at the same time

It is possible that there is a machine which is posting multiple reviews about the same product at the same time
because it is highly unlikely that many humans write about the same product at the same second. So for this rule, we gave constraint that no more than two reviews can be written for one product at the same time. Here by using asin and unixReviewTime, we designed the following query which removes the reviews that satisfy this rule.

*val result = hiveContext.sql("insert overwrite table elecprodnew select * from elecprodnew where (asin, unixReviewTime) in (select asin, unixReviewTime from elecprodnew group by asin, unixReviewTime having count(*) < 3)")*

c) A particular user gives only the highest or the lowest rating to every product

A consumer may like one product and dislike the other. However, it is not possible that someone loves all the products he purchases or hates all of them. For this rule, a constraint is given that same reviewer would not give 5.0 rating for more than 10 products and 1.0 rating to more than 5 products. So here by using asin and overall fields, we designed the following query which removes the reviews that satisfy this rule.

*val df = sqlContext.sql("insert overwrite table elecprodnew select * from elecprodnew where (reviewerID, overall) not in (select reviewerID, overall from elecprodnew group by reviewerID, overall having count(*) > 10 and overall='5.0')")*

*val df = sqlContext.sql("insert overwrite table elecprodnew select * from elecprodnew where (reviewerID, overall) not in (select reviewerID, overall from elecprodnew group by reviewerID, overall having count(*) > 5 and overall='1.0')")*

d) Multiple references to other people such as my family, my sister, etc.

Any genuine review will be focused on the product, but someone who is writing a fake review will try to create some random story. So such references can be identified, and the reviews should be removed. For this rule, we gave the constraint that a review should not contain more
than one reference to the third person. So here by using the reviewText field, we
designed the following query which removes the reviews that satisfy this rule.

val references = Array("my family", "my brother", "my sister", "my husband", "my mother", "my wife", "my father")

*for (r1 <- references) val df = sqlContext.sql("insert overwrite table elecprodnew select * from elecprodnew where reviewText not in(select reviewText from elecprodnew LATERAL VIEW explode (split (reviewText, ' "+r1+" ')) t1 As word group by reviewText having count(*) ≥ 2)")*

After removing the fake reviews, we get our second database, which consists of only genuine reviews.

5.2 Opinion Mining

Then we move to our second process which is Opinion Mining. Opinion Mining is performed on the genuine reviews database. It consists to two sub-processes: NLP and Sentiment Analysis.

Firstly, Natural Language Processing (NLP) is performed on the reviews. The sub-processes include Removing Stop-words, Part-of-Speech (POS) Tagging and Stemming and Lemmatization. The Stanford Core-NLP library is used for this process.

The second sub-process under Opinion Mining is Sentiment Analysis. The lexicon-based approach is adopted for performing sentiment analysis. A

sentiment dictionary Sentiwordnet is used to get the positive and negative sentiment scores of different words in English and then, the average sentiment score of each review is calculated by adding the sentiment scores of all the synsets (words) in that review. The formula is as shown below:

Average Review ∑ Sentiment Score of each Synset in Review

Sentiment = Score ∑ No. of Synsets in each review

After completing Opinion Mining, the third database of Review rating is created which consists of the reviews with their average sentiment scores.

5.3 Product Rating

This database is the input to the last process of Clustering. Since Amazon follows the 5-star rating scheme, we also have to rate the products out of 5. For this, we need to create 5 different clusters depending on the average sentiment scores of the reviews.

Three different algorithms are used for clustering: K-means, Bisecting K-means and Gaussian Mixture. Out of these, the Bisecting K-means outperformed the remaining regarding speed. It was faster than the rest. Hence, we use the results provided by this algorithm for our final analysis.

After clustering, each review is assigned a cluster. Depending on the cluster of review, each review has been given a rating between 1 and 5 and the average was taken to get the final rating for each product. We calculated the final rating of each product using the formula shown below:

Final Rating ∑ (Cluster Number * No. of Reviews in the same cluster)

For Each=Product ∑ Total No. of Reviews for the product

In the end, the final database having the Final Product Rating was obtained.

6. Result Analysis

Analysis-1: Comparison of review ratings for single product

Firstly, we have compared the newly generated rating with the rating provided by the user after writing a review. 10 reviews for a particular product are selected randomly from the dataset, and the two values are compared.

Graph 1: Comparison of user rating and system generated a rating for 10 reviews of a particular product.

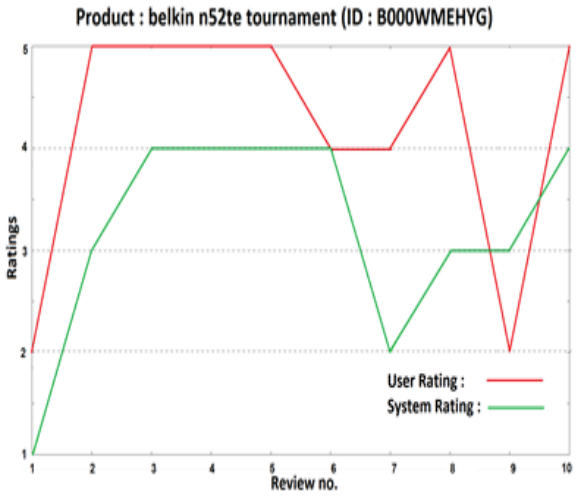


Figure 3. Analysis – 1 : Electronics Category - Game Hardware

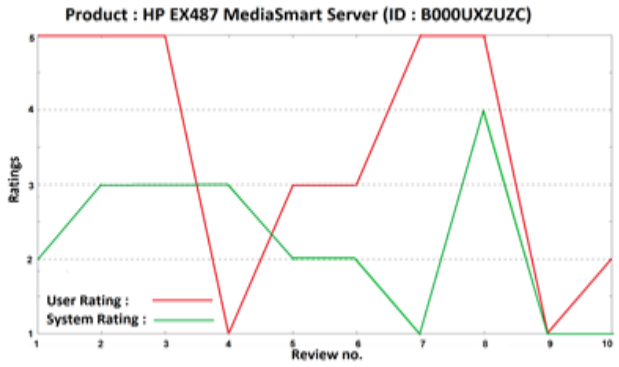


Figure 4. Analysis – 1 : Electronics Category – Servers.

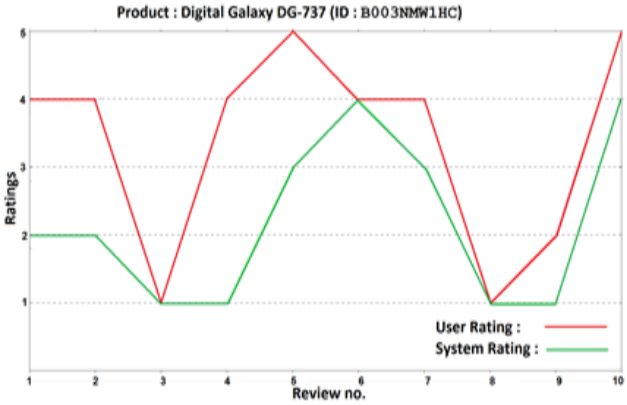


Figure 5. Analysis – 1 : Electronics Category - Video Projectors

Interpretation from the graph :

From the above 3 graphs we understand that for most of the reviews, the user rating is equal to or varies slightly from the system rating. This means that our system is quite accurate. For example, it is possible that the user rates a product with 5 stars and our system gives 4 stars to the product.

However, there are some reviews for which the rating given by the user differs largely from that provided by our system. Such a case arises when the user writes short comments or uses words with only slight positive/negative polarity even when he really likes/dislikes the product. For example, the user may really like a product and give 5 stars to it but only write 'good' in the comment. Instead, if some other words are used like 'amazing,' 'best' or 'awesome', then the difference between the user and system ratings can be reduced.

Additionally, if sarcasm is used by the user, then such a case is not handled by our system while performing sentiment analysis and this can also result in varying ratings.

Analysis-2 : Comparison of final ratings for different products

Next, a comparison is carried out between the overall rating given to the product by all the users and the final rating provided by our system. For the category Game Hardware, there are only 3 different products, but for other two categories, 10 products are selected randomly. The average of the user ratings of all reviews is calculated and compared with the average of the system ratings generated for all reviews after opinion mining. The histogram below helps to clearly interpret the results.

Graph 2 : Comparison of user rating and system generated rating for different products in each category

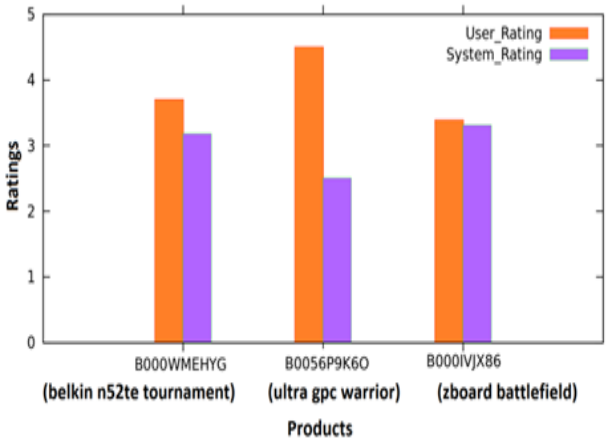
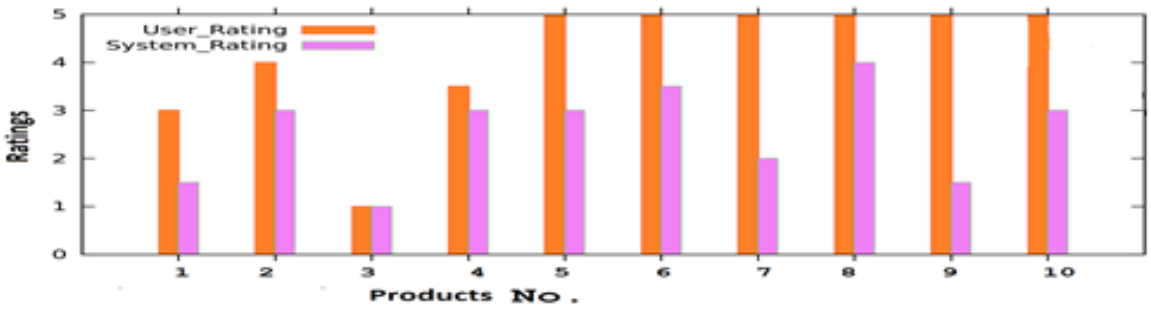
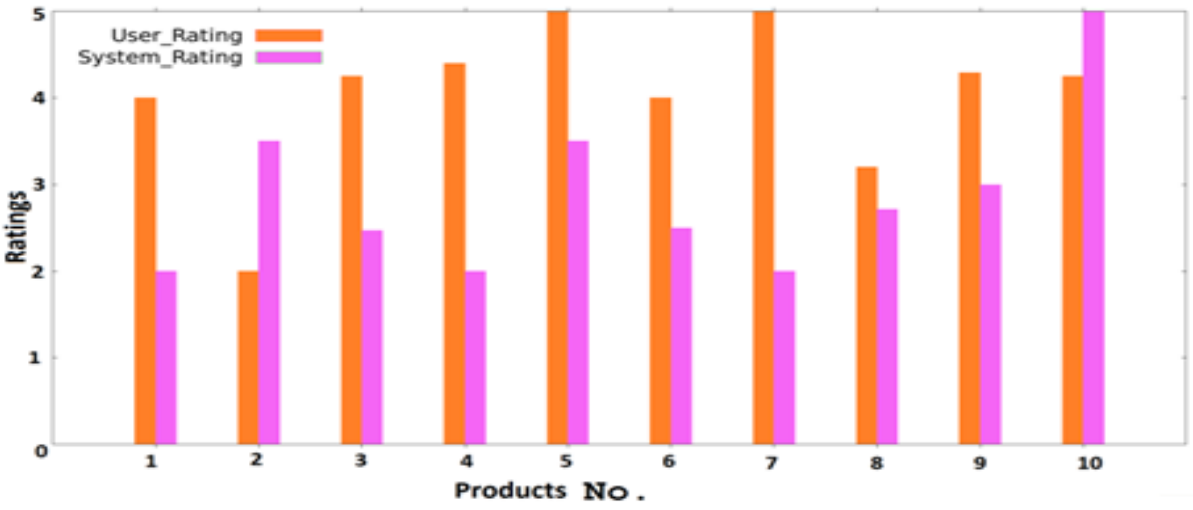


Figure 6. Analysis – 2 : Electronics Category - Game Hardware



NO	Product ID	Product Title
1	B009AEMNGG	Lenovo TS130 110571U Server
2	B0002UEDLA	HP Proliant DL380 G4
3	B000309VKC	HP PROLIANT BL20P G3
4	B001J54B4Q	Thecus NAS M3800 Network Attached Media Server
5	B002XF38J0	Fanqua / BS652/9 Bay/usb+audio
6	B006C6E3B8	HP ProLiant ML110 G7
7	B007G0PKZM	Dell PowerEdge 840 Xeon
8	B00456Q54K	MacBook Pro
9	B009AEMMF8	Lenovo TS130 110568U Server
10	B00006HTE2	Cisco MEM-7100-FLD128M

Figure 7. Analysis – 2 : Electronics Category - Servers



NO	Product ID	Product Name
1.	B00COTRP9Y	VPL-EW246
2.	B00FWSL9VO	SUNOAD HD DVB-S2
3.	B00AYDY3UG	Optoma DW312
4.	B003JH7PTA	ViewSonic PJD5152 SVGA
5.	B001RTS53W	Vivitek D935VX 3000 Lumen XGA DLP Projector
6.	B00KSTMVT6	ICODIS CB-100 DLP LED Projector
7.	B004MB6AS4	PowerLite 905 3000 Lumens
8.	B0007U743O	Kwik-Draw Projector
9.	B000EI5QWA	Optoma HD72 720p DLP Home Theater Projector
10.	B00A8F2R4Y	Epson 5020UBe Home Cinema Wireless 3D HDMI

Figure 8. Analysis – 2 : Electronics Category - Video Projectors

Interpretation from the graph :

From the above 3 graphs, we can see that there are some products for which the user rating and system rating have a small difference between them. This difference may be caused by the approach of opinion mining and the libraries used in our project. This indicates that any customer can directly buy those products by looking at the overall rating provided by Amazon website without even looking at the user reviews.

However, it is also visible that many products have a significant variation between the user and system ratings. So here we can interpret that for that product, there are some reviews which are not considered genuine and are removed by our system during fake review analysis before performing opinion mining. Since only genuine reviews are selected, our system generates more dependable results for the customer to refer and decide whether to buy a product or not. Hence, if there is a large difference between the user and our system generated ratings, any first-time customer can understand that he should not rely simply on the ratings given by Amazon website and instead refer to the ratings given after our analysis or start reading the reviews on the Amazon website to know the details of the product and decide if a purchase should me made or not.

7. Conclusion

This paper gives a new approach for Opinion Mining of Amazon reviews. Our system first removes fake reviews and then performs opinion mining on only genuine reviews to rate the products. This makes our system more reliable when compared to any other approach. Hence the system should be considered better than the existing ones used for analysis. Since we have used the Apache Spark framework instead of Hadoop, the speed of processing our data has also increased, and analysis takes lesser time. Hence our system is efficient too.

8. References

[1] N. Nodarakis, S. Sioutas, A. Tsakalidis, and G. Tzimas. Large-Scale Sentiment Analysis On Twitter with Spark. Mar 15, 2016.

[2] Enock Kanyesigye , Sumitra Menerea," Sentiment Analysis Of Reviews Using Hadoop". 2016.

[3] J. McAuley, R. Pandey, J. Leskovec Knowledge Discovery and Data Mining, 2015.

[4] J. McAuley, C. Targett, J. Shi, A. van den Hengel SIGIR, 2015

[5] Eman M.G. Younis, Faculty of Computer and Information Minia University, Egypt, "Sentiment Analysis and Text Mining for Social Media

Microblogs using Open Source Tools: An Empirical Study".February 2015

[6] Poobana S, Sashi Rekha k, "Opinion Mining From Text Reviews Using Machine Learning Algorithm ".3, March 2015

[7] Mrs. Uma Gurav, Prof. Dr. Nandini sidnal, "Opinion mining for reputation evaluation on unstructured Big Data " . 4, April 2015

[8] Spark. The apache software foundation: Spark homepage. http://spark.apache.org/, 2015. [Online; accessed 27-December-2015]

[9] Sunil B. Mane, Y. Sawant, S. Kazi, and V. ShindeReal.Time Sentiment Analysis of Twitter Data Using Hadoop,College of Engineering, Pune. 2014

[10] Anju Gahlawat. Big Data Analysis using R and Hadoop. September 2014

[11] Pravesh Kumar Singh, Mohd Shahid Husain, "METHODOLOGICALSTUDY OFOPINION MINING AND SENTIMENT ANALYSIS TECHNIQUES". February 2014

[12] Kalyankumar B Waddar, K Srinivasa, "OPINION MINING IN PRODUCT REVIEW SYSTEM USING BIG DATA TECHNOLOGY HADOOP".Jul 5,2014

[13] Julia Kreutzer And Neele Witte, Opinion Mining Using SentiWordNet, Semantic Analysis, Uppsala University. 2013/14

[14] Arjun Mukherjee, Vivek Venkataraman, Bing Liu, Natalie Glance, "Fake Review Detection: Classification and Analysis of Real and Pseudo Reviews" .2013

[15] Nitin Jindal and Bing.Opinion Spam and Analysis.Department of Computer Science, University of Illinois at Chicago.Feb 12, 2008

[16] Bo Pang, Lillian Lee, "Opinion Mining and Sentiment Analysis" . 2008

[17] Bing Liu, Minquing hu, "Mining and summarizing Customer Reviews". 2004

[18] K. Waddar and K. Srinivasa. OPINION MINING IN PRODUCT REVIEW SYSTEM USING BIG DATA TECHNOLOGY HADOOP

[19] B. Pang, L. Lee, and S. Vaithyanathan. Sentiment classification using machine learning techniques.

[20] Maria Soledad Elli, Yi-Fan Wang, Amazon Reviews, business analytics with sentiment analysis

