Estimation and Analysis of Usability Scores of Product Reviews

Aditee Vasant Jadhav, Aditi Khurd, Krupa Dhirajlal Vadher and Pranav Sudhir Dixit

Abstract— **The recent advancements in technology have enabled every industry to showcase their products on their websites and due to such large amount of choices, online shopping is turning out to be a popular choice of people, as they can do it easily at home without being have to move out physically. People look at product reviews and then make decisions of their choice. There are huge number of reviews on e-commerce websites even for less popular products. The consumers and businessmen find it a big challenge to process such a big data of reviews. Thus customers find it difficult to go through all reviews and many good reviews get buried down huge number of reviews posted afterwards. To solve the problem of ignoring good reviews, we propose to evaluate the product reviews and calculate the usability scores of the reviews using different NLP techniques. We define usability scores of the reviews as the average normalized score calculated using polarity, subjectivity, readability scores of the review. This project will help fair chance of getting viewed to the good quality reviews. We suggest that assigning usability scores to the reviews will help reviewers to write better reviews and help customers make better buying decisions by looking at better quality reviews.**

I. INTRODUCTION

Online shopping has gained popularity over decades due to ease for customers to buy the products sitting comfortably at home and relying heavily upon other user reviews for the products. The product reviews play a vital role in helping a customer to make buying decisions. Thus, researchers have been attracted towards understanding the role of product reviews in businesses. These product reviews are useful to both decision making of customers and quality improvement for industrial firms. The advancements in technology and increased users of internet has contributed towards a rapid increase in online product reviews being posted and being looked at.

The product reviews can make both positive and negative impact on the customers. This impact of reviews helps in better decision making for the customer. However, the large amount of product reviews make it infeasible for the customer to go through all the reviews and then decide on product purchase. Also, not all reviews contribute a helpful information in decision making process of the customers and if such reviews are displayed on the top pages, the customer may go in dilemma towards product buying decision. Thus, it would be great if a customer could get an insight of helpful reviews that represent an overview about product and its features. Some e-commerce companies like Amazon.com provide an option of voting helpful reviews to the customers. The reviews on Amazon.com are by default sorted by using the helpfulness voting received. Thus, customers can make better buying decision and this helps increasing the product sales. However, there are certain problems in sorting reviews based on the helpfulness voting received. One of the problems is that a very small fraction of product reviews receive helpfulness voting. Another problem is the situation where recently posted reviews do not get votes. One more problem is faced when the reviews are posted at such a speed that some useful reviews may get buried down the pile of huge amount of reviews before receiving any helpfulness voting. It is also worth noticing that the reviews with helpfulness voting are exposed highly to the users and they gain even more visibility to the users. This resembles the real world phenomenon of 'rich getting richer and poor getting poorer'. Another scenario that needs attention is that the customers looking at the reviews with high helpfulness voting are influenced without noticing the review posting date and the context of the review.

Although the product reviews are useful to the customers in the process of purchase decision making, extracting the helpful reviews from a huge dataset is a tedious task. The reviews data is increasing in size day by day at a great speed. Thus, researchers suggested applying machine learning techniques for extracting useful information out of data. Many sentiment mining algorithms and natural language processing techniques have been developed to extract helpful information out of given reviews. Still the huge size of reviews needs to be processed every time we dig into reviews dataset. Also the existing experiments have focused on sentiment mining and not much work is done on measuring the quality of the review. In our project, we focus on measuring the quality of reviews and thereby estimating the helpfulness of the reviews instead of relying solely on helpfulness voting. We propose to calculate the quality of the review in terms of usability score which will be computed based on different techniques of assessing the textual reviews like polarity, subjectivity, readability index, etc. We will make use of these metrics and aggregate them to discover the helpfulness of the reviews.

This report is organized as: Previous research done in this field and how our work differs from others is explained in section II. The details of the methods we will apply and expected results are explained in section III.

II. RELATED WORK

Numerous research work is done for determining the helpfulness of the product reviews. Liu et.al.[1] proposed a system that treats helpfulness of reviews as a binary classification issue. Their system labels reviews as 'favourable' and 'unfavorable' based on vectors of review text. However, they did not use helpfulness voting of the reviews in their system. Ghose et.al. [2] proposed a system that assigns two ranks to the reviews based on either helpfulness voting received or based on effective sales of the product. They analyzed the review data along with its effect on sales. They discovered that the products that receive the reviews that have subjective or objective text tend to have more sales.

Otterbacher et.al. [3] suggested machine learning model for predicting the helpfulness voting of the reviews based on factors like review length, style of writing of the user. The authors claim that use of different metrics for determining quality of review will help in efficient and more accurate determination of helpful reviews. Danescu et.al. [4] proposed a model to establish relationship between ratings of the products and fraction of reviews that receive helpfulness voting. The authors concluded that the helpfulness voting is generally associated with average product rating.

Mudambi et.al. [5] conducted experiments to determine what properties of review text make it more prone to receive helpfulness voting. The authors found out that moderate reviews received more helpfulness votes than extreme reviews. They also discovered that length of review and subjectivity impacted helpfulness of the review in a greater way. Huang et.al. [6] carried out experiments to establish the equation for impact of review length on review helpfulness. The authors found that length of review measured in term of count of words had limited impact on helpfulness of the review. When number of reviews in the system cross certain threshold, the length of review shows lesser impact on helpfulness. The authors suggested that extracting semantic features from reviews might discover new patterns that may contribute towards estimating helpfulness of the reviews.

The literature survey here suggests that not all the reviews on a product are seen by customers and top reviews receive more helpfulness votes and low reviews get comparatively less votes. This elevates the Matthew social effect of "rich getting richer and poor getting poorer". Thus is has become necessity to understand the semantic features of reviews which may contribute towards helpfulness of the reviews. Thus, it would be great if we could compute helpfulness of reviews using semantic features rather than waiting for customers to cast their votes and thereby rank reviews according to calculated helpfulness of reviews.

III. METHODOLOGY

The facts and the issues stated in literature survey makes an observation that the Matthews social effect hampers the estimation of helpfulness of reviews. In this project, we planned to address this issue by calculating the helpfulness of reviews in terms of 'usability score'. The term 'usability score' cab be defined as an aggregated value of different semantic features of the reviews. We first compute different semantic metrics for reviews and final output is the aggregated value of all semantic metrics. The methodology we plan to apply and the metrics we plan to use in our project for computing semantic features of reviews can be listed and described as follows:

A. Semantic features

In this project we focus on first computing different semantic features of text reviews and then we aggregate them together into a new term defined as usability score'. This 'usability score' can be used in place of 'helpfulness' of the review. The semantic features in this project are: subjectivity, polarity, flesch index, entropy, Dale Chall index, lex diversity, helpfulness ratio.

The 'usability score' of review is calculated by taking aggregation of all semantic features stated above. The 'polarity' of review is calculated by subtracting negative score of review from positive score of review. These positive and negative scores can be calculated using existing databases like SentiWordNet. The 'polarity' score of the review will tell if the review states positive information or negative information.

The 'subjectivity' score of the review is defined as the fraction of review statements that express an opinion against total number of statements in the review text. This score is calculated using number of nouns, adjectives, verbs in review text. The 'review length' and 'set length' is calculated by measuring the number of total words in the review and the number of unique words in the review respectively. The 'lex diversity' is computed by dividing 'set length' by 'review length'.

The 'readability' score of the review tells about level of ease of language used in the review. To calculate readability score, we use two different metrics: Flesch index, Dale Chall index. The 'entropy' of the review is the probabilistic model representing amount of information embedded in the review.

B. Usability score

The final output for this project is usability score computed for each review of the products. We define the term usability score of a review as the unique value obtained by taking aggregation of all normalized semantic feature values. While taking aggregation, we must notice the relationship between semantic feature and overall (rating). If the semantic feature is negatively correlated with overall, we multiply it by -1 while calculating aggregation. For example, assume that there are 3 semantic features x, y, z for a review and semantic feature z is negatively correlated with overall, while semantic features x, y are positively correlated with overall. Then usability score $Usability$ is calculated as:

$$Usability = \frac{x + y - z}{3} \quad (1)$$

Similarly, we calculate and analyze all semantic features and then take aggregation depending upon the correlation between each semantic feature and overall value.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

A. Dataset

For this project we plan to use Amazon reviews data which readily available over internet. Two of our teammates had worked on recommendation systems competition for kaggle and they have worked on this dataset. However, they had used star ratings for their experiments and they did not work on text reviews for recommendation of products. They suggested that taking a deep dive into this reviews dataset may help us discover new patterns and hence we decided to work on Amazon reviews dataset. Their previous work focused on product id, user id and start ratings in terms of numeric value. When they worked on this competition, they thought that although they did not work on text reviews, it might have hidden knowledge. When we decided to work on this dataset, we observed that this dataset is very large and just finding vector of words in a review will not be useful for recommendation system. Literature survey also showed that quality of reviews is an important factor. Hence we decided to work in finding helpfulness of reviews from different semantic features.

Amazon reviews dataset is very large and it would be time consuming to work on reviews of each category of products in this data. Hence, we decided to work on a subsection of Amazon reviews dataset: reviews dataset on toys and games. Toys reviews dataset contains field like ASIN (Amazon Standard Identification Number), reviewer id, reviewer name, review time, review text, helpfulness votes, ratings. Out of these attributes, we plan to first focus on computing semantic features from reviews text field only. Then we will try to analyze relation between semantic features and ratings.

B. *Data preprocessing*

For our project, we are using only text reviews from Toys reviews dataset which is a subset of Amazon reviews dataset. The text reviews are created by customers and are heavily dominated by their local language. There may be lot of noise in text reviews. To reduce this noise, we performed certain preprocessing steps on the data as explained later in this subsection.

Before performing data preprocessing, we need to notice that available and downloadable data is in JSON format and loading data from JSON format was quite time consuming. Thus, we decided to first convert JSON to CSV and use CSV for further steps in our project. After data conversion, we visualized the data as explained in later subsection 'Data visualization'. Then we formed following data preprocessing steps.

We first removed missing values by eliminating 'Nan' from dataset. Then we converted all the words in text review to lowercase. We removed the words that do not contain any alphabetical letter from the review. We also removed the words containing numbers from the review text. The punctuation symbols from text review were also removed. Then we discarded the reviews whose length is below certain threshold level. Because less descriptive reviews do not contribute as helpful reviews.

C. *Data visualization*

In this project, we tried to get insight into dataset by visualizing the dataset using different types of plots. We plotted graphs like: bar plot, line plot, scatter plot, histogram, dist plot, count plot, facet and box plot. The dataset we used has shape of (167597, 9). This means that there are 167597 data instances and each data instance has 9 features. The count plot as shown in Fig. 1 tells us that there are more reviews with rating '5' given for that product and less reviews with other ratings. The dataset consists of product reviews from year 2000 to 2015. The scatter plot tells us that initially there were less product with ratings and reviews. The rating of '1' was barely given in early years of toy products. Facet plot as shown in Fig. 2 tells us that there has been less ratings given by customers from year 2000 to year 2010. The number of ratings have increased after year 2010 and the rating values 3,4,5 are more commonly given as compared with the rating values 1,2.

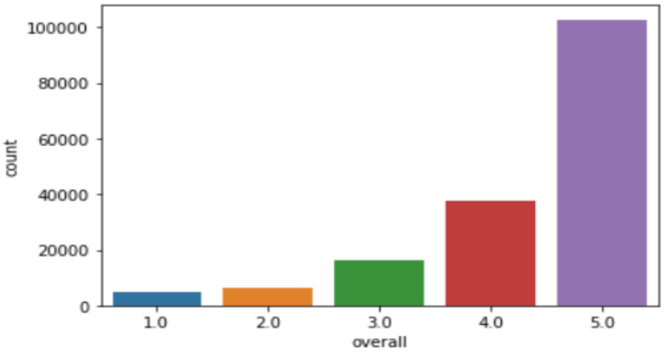

Fig. 1. Count plot of 'overall'

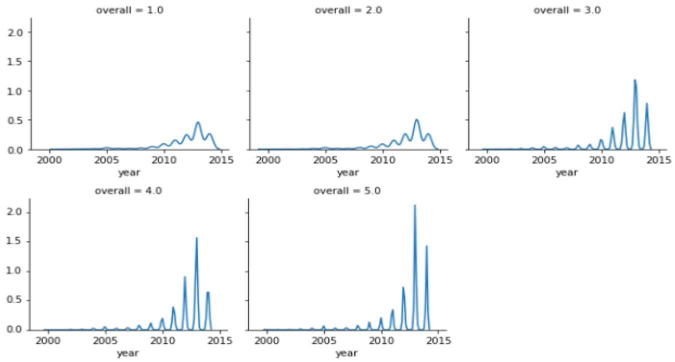

Fig. 2. Faceting of 'overall' vs 'year'

D. *Calculating semantic features*

We studied different ways of computing semantic scores of the reviews dataset and we finally implemented following semantic scores by making efficient use of NLP (Natural Language Processing) libraries available in the python language.

1) Subjectivity: To compute subjectivity of the review, we used pre-defined functions in python library 'TextBlob'. This library helps in calculating sentiment of the input text data. The sentiment is expressed in terms of subjectivity and polarity. Here, we extracted only subjectivity value from the sentiment of input text review. We added extra column 'subjectivity_score' for storing computed subjectivity values.

2) Flesch Index: To compute readability score in terms of flesch index of the review, we used pre-defined functions in python library 'textstat'. This library helps in calculating readability score of the input text data. Flesch kincaid grade is nothing but flesch index in our project. This score tests the ease of reading of any text. The score values may range from 0 to 1000, with -30 being very easy to read and 1000 being very difficult to read. We added extra column 'flesch_index' for storing computed flesch index score values.

3) Polarity: To compute polarity of the review, we used pre-defined functions in python library 'TextBlob'. This library helps in calculating the sentiment of input text data. The sentiment is expressed in terms of subjectivity and polarity. Here, we extracted only polarity value from the sentiment of input text review. The polarity values ranges from -1 to 1 in value. We added extra column 'polarity_score' for storing computed polarity values.

4) Entropy: Entropy is a measure which tells how much information in a particular sentence is provided by the words contained by it. To calculate the entropy of reviews, we used nltk punkt package. Punkt is a sentence tokenizer which seperates the text into the list of words. NLTK already has the pre-trained package of the punkt on the english language. We added extra column 'entropy_score' for storing computed entropy score values.

5) Dale Chall Index: Dale Chall index indicates how easy a text is for school students of each grade to understand. A score less than 4.9 indicates the text can be understood by students of 4th grade or lower while a text with score greater than 10 can only understood by college students. A formula is used to calculate the index which basically calculates the ratio of difficult words to all the words and also the ratio of total words to total sentences in the text. We implemented a Python program to calculate the index. The list of 3000 difficult words is freely available which we used to calculate the ratio of the difficult words in the text.

6) Helpful ratio: A simple helpful ratio can be calculated by finding the helpful votes to the total votes. But we used a different approach. We calculated the total votes received by a reviewer for all his reviews. We also found the total helpful votes received by a reviewer for all his reviews. Finally we calculated the ratio of helpful votes to the total votes. Reviews which received 0 votes get a value of 0/0 which is invalid. Making the helpful ratio as 0 will give the reviews a negative bias as it will mean they are worst in terms of helpfulness. So we used a technique stated in [7] where we add 1 to the helpful votes and 2 to the total votes. By this reviews with no votes get helpful ratio of 0.5 which indicates a neutral value. A score of less than 0.5 represents a bad score while a score above 0.5 indicates a good score.

7) Lex diversity: Lex Diversity is basically the ratio of unique words to total number of words in a review given. This measure gives an idea of lexical sophistication or density of words in the reviews. This score can be used further used to find out the content in the text and eliminate the function words from the review. To calculate the Lexical Diversity of the reviews in our dataset, we used nltk library and wordtokenize() function prominently. Tokens are basically the words in raw form of the review text. The diversity ratio is calculated by taking the ratio of set of words to total words. The calculated lex diversity ratio were added as a new column to the dataset.Higher the lex diversity ratio, more is the number of unique words in the text.

E. Analysis of semantic features

In order to understand the impact of semantic scores towards estimation of usefulness of review, we first need to find correlations between semantic scores and ratings given to the reviews. This way we understand what type of semantically weighted reviews attract more customers and thus increase rating of the product. To understand these concepts, we perform visualization of semantic reviews using different ways and its analysis can be given as:

1) Subjectivity: To find how subjectivity of the review may impact rating of the product, we performed visualization of 'subjectivity scores' using different graphs like: line plot, dist plot, joint plot and statistical equations like: covariance, pearson correlation, spearmans correlation. The experiments we performed show that the pearson correlation between 'overall'(rating) and 'subjectivity score' is 0.07 and the spearmans correlation between them is 0.08. Both the values are positive, which means that 'subjectivity score' is positively correlated with 'overall'. The line plot as shown in Fig. 3 between 'overall' and 'subjectivity score' tells that as 'subjectivity score' increases, the 'overall' also increases. This means that if the review is more subjective, people tend to read it and find it more helpful. This helps in decision making process and thus product gets better ratings.

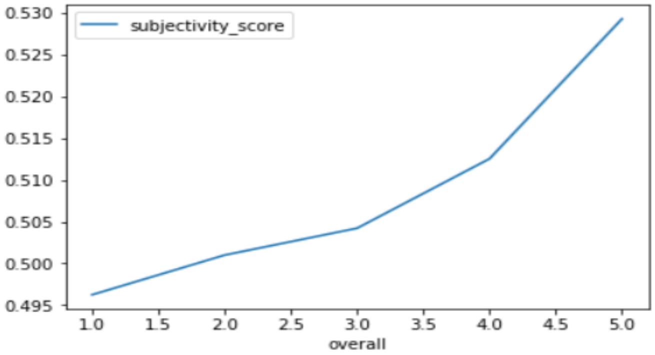

Fig. 3. Line plot between 'subjectivity_score' and 'overall'

2) Flesch Index: To find how readability of the review may impact rating of the product, we performed visualization of 'flesch index' using different graphs like: line plot, dist plot, joint plot and statistical equations like: covariance, pearson correlation, spearmans correlation. The experiments we performed show that the pearson correlation between 'overall'(rating) and 'flesch index' is -0.12 and the spearmans correlation between them is -0.2. The both values are negtive, which means that 'flesch index' is negatively correlated with 'overall'. The line plot as shown in Fig. 4 between 'overall' and 'flesch index' tells that as 'flesch index' increases, the

’overall’ decreases. The higher the flesch index, the difficult is review to understand. This means that if the review is easy to read (low flesch index), people tend to read it and find it more helpful. This helps in decision making process and thus product gets better ratings. If the review is difficult to read (high flesch index), people tend to ignore review. Thus, even if review is genuine, it may not be considered helpful.

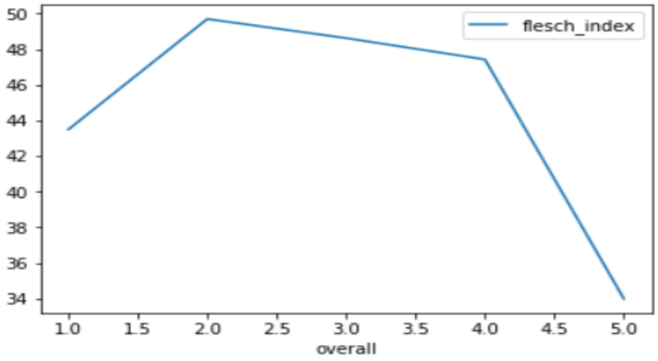

Fig. 4. Line plot between ’flesch_index’ and ’overall’

3) Polarity: To find how polarity of the review may impact rating of the product, we performed visualization of ’polarity scores’ using different graphs like: line plot, dist plot, joint plot and statistical equations like: covariance, pearson correlation, spearmans correlation. The experiments we performed show that the pearson correlation between ’overall’(rating) and ’polarity score’ is 0.290 and the spearmans correlation between them is 0.281. The both values are positive, which means that ’polarity score’ is positively correlated with the column ’overall’. The line plot as shown in the Fig. 5 which is a comparison between ’overall’ and ’polarity score’ tells that as ’polarity score’ increases, the ’overall’ also increases.

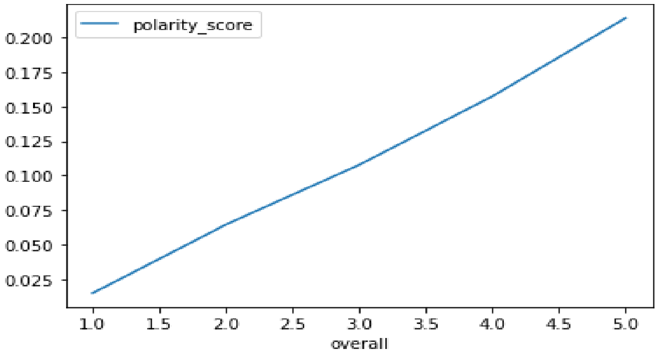

Fig. 5. Line plot between ’polarity_score’ and ’overall’

This means that if the review is more polar, people tend to read it and find it more helpful. This helps in decision making process and thus product gets better ratings.

4) Entropy: To find how entropy of the review may impact rating of the product, we performed visualization of ’entropy scores’ using different graphs like: line plot, dist plot, joint plot and statistical equations like: covariance, pearson correlation, spearmans correlation. The experiments we performed show that the pearson correlation between ’overall’(rating) and ’entropy score’ is -0.117 and the spearmans correlation between them is -0.138. The both values are negative, which means that ’entropy score’ is negatively correlated with the column ’overall’. The line plot as shown in Fig. 6 between ’overall’ and ’entropy score’ tells that as ’entropy’ increases, the ’overall’ decreases. The higher the flesch index, the difficult is review to understand. This means that if the review is easy to read (low flesch index), people tend to read it and find it more helpful. This helps in decision making process and thus product gets better ratings. If the review is difficult to read (high flesch index), people tend to ignore review. Thus, even if review is genuine, it may not be considered helpful.

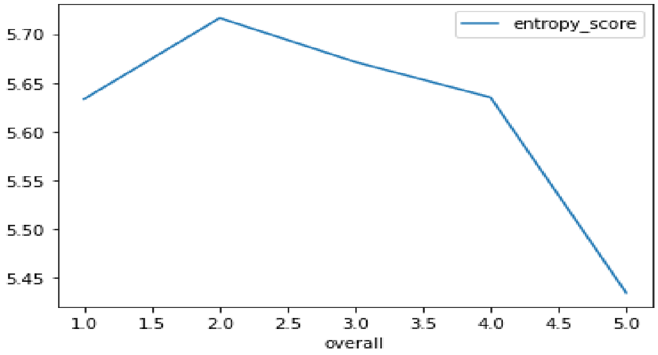

Fig. 6. Line plot between ’entropy_score’ and ’overall’

5) Dale Chall Index: Different types of plots and indexes were calculated to find the relation between Dale Chall index and the rating. A line plot, joint plot and distribution plot was plot to see the relation graphically. Looking at the line plot in Fig. 7 it can be seen that as the score increases the the value of index increases slightly. The analysis also revealed a score of 0.04208 for Pearson Correlation and 0.04215 for Spearmans Correlation. They are very small values tending to 0 indicating no correlation. From the line plot, it can be seen there is not must variance between the average index values and that the best to worst values of the index range from 7.15 to 7.30. A score between 7.0 to 7.9 means that

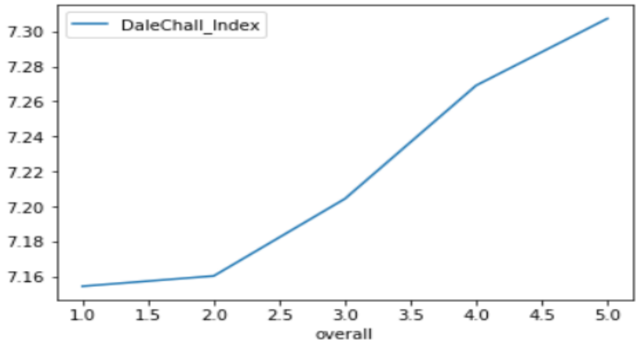

Fig. 7. Line plot between ’Dale Chall index’ and ’overall’

the text is easily understood by an average 9th or 10th grade student. Looking at the joint plot it can be seen that all the rating values have a similar range of Dale Chall index values. Some reviews have index value above 15 have low ratings as they are hard to understand. Apart from that little can be inferred from Dale Chall index.

6) Helpful ratio: Like others we calculated the values and plots for the helpful ratio. Fig. 8 shows the line plot of ratings versus helpful ratio. Even though the plot shows variation it can be observed that the average ratio of all the ratings ranges from 0.66 to 0.75 which is not much of variation. The Pearson correlation is -0.039 and Spearmans correlation is -0.066. Even though both the correlations are negative the value is close to 0 which indicates no correlation. Also looking at the join plot it can be observed that for each value of rating, the helpful ratio ranges from 0 to 1 indicating that any relation between rating and helpful ratio cannot be inferred.

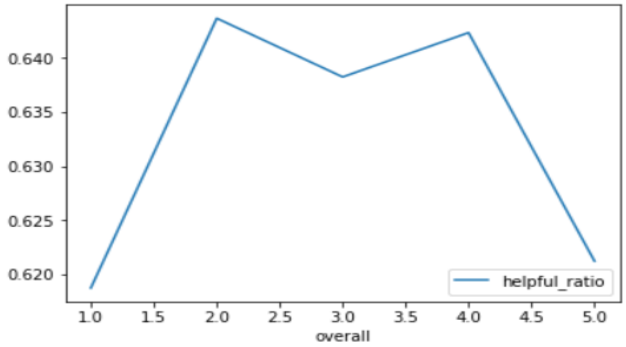

Fig. 8. Line plot between ’Helpful Ratio’ and ’overall’

7) Lex diversity: To further understand the impact of Lex Diversity Ratio on the rating of product, we visualized the computed ratios using matplotlib and seaborn package of python. Fig. 9 shows the line plot distribution of lex diversity ratio across the overall ratings.

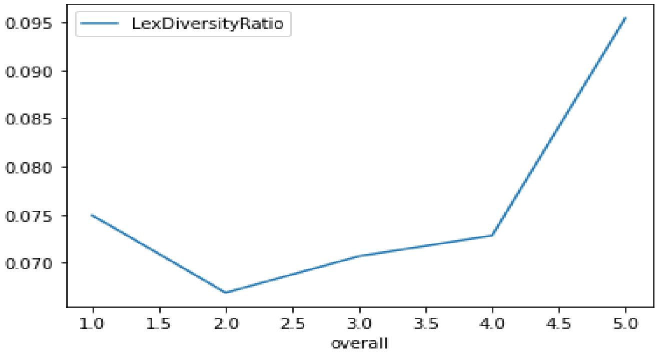

Fig. 9. Line plot between ’Lex Diversity Ratio’ and ’overall’

The Lex Diversity Ratios for our dataset range in between 0.25 to 1.0 . Thus, there was no need to normalize these values. The Pearson correlation between the rating and Lex Diversity Ratio was found to be 0.171 and Spearmans correlation between the ratings and Lex Diversity Ratio was 0.206. The Fig. 10 shows the univariate distribution of Lex Diversity Ratios.

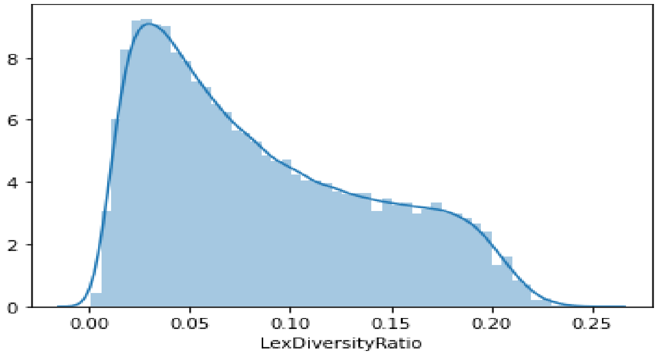

Fig. 10. Dist plot between ’Lex Diversity Ratio’ and ’overall’

F. Normalization of semantic features

Once all semantic features are calculated and analyzed, we are all set to compute final usability score by computing average of them. However, we must notice that the range of values for all semantic features is different. Polarity score lies between -1 and 1. Flesch index lies between 0 and 1000. Some scores lie between 0 and 1. Calculating simple average on all these semantic scores may create bias and semantic features with higher values ranges may dominate semantic features with lower values ranges. Thus, we decided to first normalize all the semantic features such that all semantic features values will lie between 0 and 1. This will remove the bias and calculating average over normalized semantic features will give simple and easy to understand usability score. Thus, we performed normalization on all the semantic features using following steps:

1) Find minimum and maximum value of each semantic feature
2) Calculate normalized score

Consider x^{ij} be value of j^{th} semantic feature for i^{th} review in dataset, min^j be minimum value of j^{th} semantic feature and max^j be maximum value of j^{th} semantic feature. Then the normalized score value of x^{ij} will be given by:

$$normalizedScore^{ij} = \frac{x^{ij} - min^j}{max^j - min^j} \tag{2}$$

G. Usability score

This is last step of project. While calculating $Usability$, we need normalized values of semantic features: subjectivity, polarity, entropy, Flesch index and Dale Chall index. The correlation between these semantic scores and overall is as:

- Positively correlated features: Subjectivity, Polarity, Dale Chall index
- Negatively correlated features: Flesch index, Entropy

Using the information above, we understand that to achieve maximum usability of a review, it should have more subjectivity score, polarity score and dale chall index, and it should have less flesch index and entropy score. Thus, to compute $Usability$, we will take aggregation of all these score. But, we will multiply by -1 to negatively correlated features before computing aggregation. The formula for $Usability$ is given as: Consider $usability^i$ be value of Usability score for i^{th} review in dataset, SUB^i, POL^i, DCI^i, FLI^i, ENT^i be subjectivity score, polarity score, Dale Chall Index, Flesch Index and entropy score of i^{th} review. Then the usability score value of will be given by:

$$usability^i = \frac{SUB^i + POL^i + DCI^i - FLI^i - ENT^i}{5} \tag{3}$$

Since the $Usability$ score is an aggregated value over normalized semantic scores, the value range for $Usability$ score lies between 0 and 1. We can interpret this as: higher the $Usability$ of the review, the more are chances of it being appearing to customers and thus, more people will tend to buy that product. Thus, computing $Usability$ when the review is submitted for a product will help in ordering reviews according to usability. This will enable more useful reviews to appear at top of the list and this will encourage customers to write better and helpful reviews.

The relation between usability score of the reviews and the ratings is given by Fig. 11. This graph shows that as usability of the review increases, the rating of the product also increases. Thus, we conclude that writing useful and semantically meaningful reviews will help in gaining better ratings of the products. The distribution of usability score is given by Fig. 12.

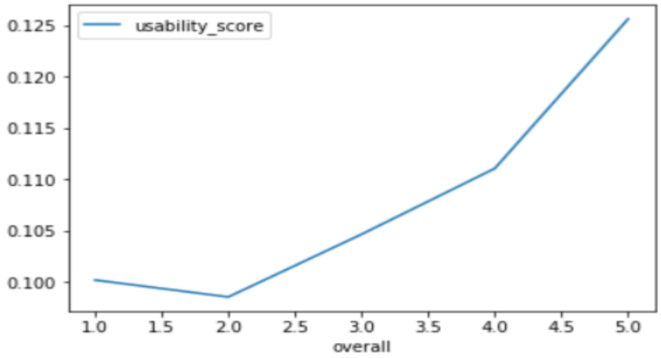

Fig. 11. Line plot between 'Usability score' and 'overall'

V. COMPARISONS

Amazon reviews dataset is one of highly used dataset for machine learning algorithms. We have used a subset of Amazon reviews dataset: Toys and Games dataset which consists of reviews over toys and games sold on Amazon.com. There has been some work done over this dataset. However, the previous work focuses on predicting the ratings of the products based on r=previous ratings. Some approaches also make use of product reviews while predicting the ratings of the products. When we tried to explore this dataset, we observed that some reviews were very short or difficult to understand. Some reviews did not convey exact sentiment. Previously there has been no work done on this dataset to find if review is really helpful or not. Hence, we decided to find ways of exploring the helpfulness of the review. After a long literature survey, we observed that semantic scores like subjectivity, polarity, entropy, readability ease scale help in understanding the sentiment of the review and this may convey semantic value of the review. When we visualized and analyzed data, we observed that these semantic features affect ratings of the product either positively or negatively. Hence, we computed usability of the reviews using the semantic features of the reviews. Such work has not been done previously. Hence, we can say that found a new way computing usability of the reviews and our approach is gives better ordering for the reviews as compared to existing approach where reviews are ordered using helpful voting given by customers. However, existing approach fails to order the reviews which did not receive any helpful voting. In this type of scenario, our approach works well and finds helpfulness of the review in terms of usability score. Hence, our approach solves the issue of missing helpful voting and gives efficient ordering of the reviews.

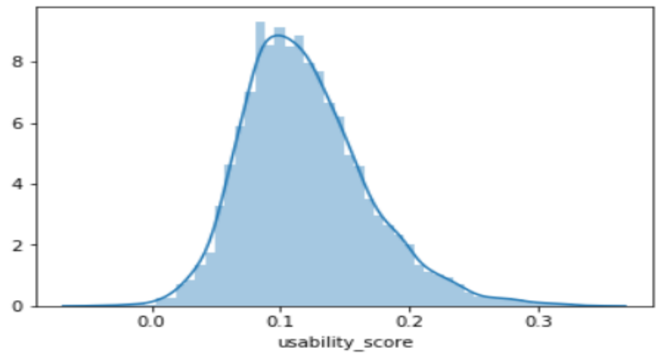

Fig. 12. Distribution plot of 'Usability score'

VI. CONCLUSIONS

Product reviews play a vital role in buying decisions of customers and thus it helps firms to grow their business. However, not all reviews contribute towards this goal and hence we need to establish a meaningful way of computing helpfulness of the reviews, so that customers can see more helpful reviews and make better buying decisions. This will also help users in writing the better reviews. In our project, we computed helpfulness of reviews in term of 'usability score' which is a aggregated representation different semantic features of review text. To define semantic features of review text, we computed subjectivity score, polarity score, flesch index, entropy score, dale chall index, lex diversity, review length and set length. We established correlation between helpfulness of the reviews and semantic features. We observed that reviews with better ratings had higher usability score. This means that if a review is semantically better (in terms of sentiment and ease of readability), then it will attract more customers and thus that product gets better chance to be seen and sold. This will increase in sales of the products.

Thus, if companies organize the review by ordering them in descending fashion, the helpful reviews will be available for user. This will also encourage customers to write better reviews. This work can further be extended to train a machine learning model for predicting usability of the reviews.

VII. CONTRIBUTIONS

The individual contribution towards project completed is as:

- Aditee Jadhav: Code for - Data conversion, Data visualization, Calculating ’subjectivity score’, calculating ’flesch index’, analyzing ’subjectivity score’, analyzing ’flesch index’, normalizing ’subjectivity score’, normalizing ’flesch index’.
- Krupa Vadher: Code for - Data preprocessing, Calculating ’polarity score’, calculating ’entropy score’, analyzing ’polarity score’, analyzing ’entropy score’, normalizing ’polarity score’, normalizing ’entropy score’.
- Pranav Dixit: Code for - Calculating ’Dale Chall index’, calculating ’helpfulness ratio’, analyzing ’Dale Chall index’, analyzing ’helpfulness ratio’, normalizing ’Dale Chall index’, normalizing ’helpfulness ratio’.
- Aditi Khurd: Code for - Calculating ’lex diversity score’, normalizing ’lex diversity score’, analysis of ’lex diversity score’ and Tableau visualizations for various features involved in the Semantic score calculation and analysis of dataset.

ACKNOWLEDGMENT

The authors are grateful to Prof. Rojas Carlos for providing a valuable feedback to our project and his comments have helped us write better project report.

REFERENCES

[1] J. Liu, Y. Cao, C.-Y. Lin, Y. Huang, M. Zhou, "Low-Quality Product Review Detection in Opinion Summarization", Proc. Joint Conf. Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), pp. 334-342, 2007.

[2] A. Ghose and P. G. Ipeirotis, "Estimating the Helpfulness and Economic Impact of Product Reviews: Mining Text and Reviewer Characteristics," in IEEE Transactions on Knowledge and Data Engineering, vol. 23, no. 10, pp. 1498-1512, Oct. 2011.

[3] J. Otterbacher, "Helpfulness in Online Communities: A Measure of Message Quality", CHI ’09: Proc. 27th Int’l Conf. Human Factors in Computing Systems, pp. 955-964, 2009.

[4] C. Danescu-Niculescu-Mizil, G. Kossinets, J. Kleinberg, L. Lee, "How Opinions Are Received by Online Communities: A Case Study on Amazon.com Helpfulness Votes", Proc. 18th Int’l Conf. World Wide Web (WWW ’09), pp. 141-150, 2009.

[5] Mudambi, S.M., Schuff, D., "What Makes a Helpful Review? A Study of Customer Reviews on Amazon.com", (SSRN Scholarly Paper No. ID 2175066), Social Science Research Network, Rochester, NY, 2010

[6] Huang, A. H., Chen, K., Yen, D. C., & Tran, T. P., "A study of factors that contribute to online review helpfulness", Computers in Human Behavior, 48, 17–27, 2015.

[7] B. Nguy, "Evaluate helpfulness in amazon reviews using deep learning.", Stanford University. 2016.