Helpfulness Prediction of Product Assessments using Machine Learning

Kagolanu Trishul, Srinath R. Naidu

Abstract*: Paper Customers express their opinion on products through reviews. Since there will be a lot of reviews that will be posted, only those reviews which are helpful should be made accessible to the customer. Hence, helpfulness of review needs to be predicted. This work categorizes the features into reviewer, review text and review metadata. Machine Learning algorithms Linear Regression and Random Forests are used for prediction of helpfulness using these features. It is observed that rating of a review has the highest influence on predicting helpfulness followed by user average rating deviation, difficult words and positive words. This work defines the features such as stem sim length and lem sim length which are derived from the product description which have performed reasonably well. Using all the features with Random Forests algorithm for prediction gave the best performance in automatically predicting helpfulness.

***Index Terms*: helpfulness prediction, lem sim length, machine learning, random forests, stem sim length.**

I. INTRODUCTION

In today's world, which is the era of internet, people express their opinions about a product, service or event through various platforms. E-commerce is an area where customers give reviews about products or services. Consumers generally have to make purchase decisions based on incomplete information about a product. These reviews are helpful for customers who want to buy a product and to manufacturers or business owners who want to assess the performance and quality of their product. In general, the customer reviews are provided in addition to product descriptions, product suggestions etc.

As there may be large number of reviews present for the products, it would become difficult for the customers to go through all the reviews. Also, as there is no editorial or quality control imposed on these reviews, there is a drastic variation in the quality of the reviews which can range from being of high-quality to extremely pointless. Hence, it is ideal that the customers should read few good quality and helpful reviews than reading all of them. A Helpful customer review can be defined as a judgement and analysis of a product by a customer that expedites the process of deciding to buy the product by other customers [6]. The websites generally ask the readers of a review whether they found the review as a helpful one or as unhelpful. These helpfulness votes are used for determining the helpfulness of a review and in general, it is calculated as the proportion of helpfulness votes received upon the total votes given to that review [11].

However, using the helpfulness votes does not completely solve the problem and this methodology has its own issues. Most of the reviews in the e-commerce websites have very less votes given to them and some are not given any votes at all. The products which are not popular have reviews which might not be seen by many users. The listing of reviews also suffers from Matthew effect [10] which states that a popular review always stays popular and highly voted as it is easily visible to most of the users and they might get inclined to the review and a review at the bottom remains at the bottom because it hardly gets noticed. Also, review posted earlier has a high chance of getting higher number of votes than a recent review which may get fewer votes. Hence, there is a need to automatically predict the helpfulness of a review and rank them according to this prediction so that the customers have access to the most helpful reviews.

A model needs to be developed which can take the review data as input and predict its helpfulness. Machine learning models which can automatically learn from the data and predict helpfulness is to be implemented. This work uses features from all three categories reviewer, review text and review metadata together and formulated features such as stem sim length and lem sim length, which is apparently not done in previous works.

II. LITERATURE SURVEY

Helpfulness prediction of product reviews has seen significant research interest over the years. The process of helpfulness prediction of the reviews is performed in four stages namely Dataset collection, Feature extraction, Prediction and Evaluation. In prediction, generally machine learning algorithms are used and respective metrics are used in evaluation. Such procedure is done works such as [18], [19].

A. *Datasets*

The data needs to be collected for the required analysis. Generally, publicly available datasets are taken, or the data is crawled from the respective websites. Amazon data is a popular choice among researchers for all the related studies.

Revised Manuscript Received on July 5, 2019.

Kagolanu Trishul, *Dept. of Computer Science & Engineering*, Amrita School of Engineering, Bengaluru, Amrita Vishwa Vidyapeetham, India, trishulkagolanu95@gmail.com

Srinath R. Naidu, *Dept. of Computer Science & Engineering*, Amrita School of Engineering, Bengaluru, Amrita Vishwa Vidyapeetham, India.

The Julian McAuley Amazon product review dataset [8,9] was used for the analysis in [1] & [3]. Data was crawled from the amazon website in [1], [2] & [5]. Reviews of products of various categories were obtained and also the metadata information was also obtained in most of these cases.

B. *Regression vs Classification*

The problem of helpfulness prediction can be taken as a regression problem or a classification problem or a combination of both. In the scenario of regression, the helpfulness score is predicted by the model whereas in the scenario of classification, the helpfulness is segregated into classes and the class to which a review belongs to is predicted. This was taken as a complete regression problem by [1] & [3]. In [5] this was taken as a classification problem. From the data, the helpfulness score (P value) was calculated and a threshold 't' was set for each category ranging from "Extremely Helpful" to "Not at all Helpful" as shown in I. In [2], a combination of classification and regression was used. The prediction was performed in two stages with classification followed by regression. In classification stage, reviews are organized into "low-quality" and "high-quality" reviews. In regression stage, the helpfulness score was predicted only for those reviews that fell into the category of "high-quality" reviews.

I. Class labels for helpfulness scores

P value	**Classes (Label)**
> 80%	Extremely Helpful
60% to 80%	Very Helpful
40% to 60%	Somewhat Helpful
20% to 40%	Not very Helpful
<20%	Not at all Helpful

C. *Feature Extraction*

Feature extraction is the most crucial step and the results highly depend on what features are extracted, how they are extracted and how they are represented. The features can be broadly classified as Review Text, Reviewer and Review metadata.

1) *Review Text*

The behaviour of the review and its characteristics can be obtained from the review text. [1] have classified the variable further into linguistics, Psychological, Summary Language and Text complexity. They have proposed pronouns, article words, prepositions and auxiliary verbs in the review text. They have also considered the length of the review as word-count, character-count and sentence-count. [1] have considered readability variables as another parameter they indicate the effort required by the readers of a review to read and understand the review. These include Automated Readability Index (ARI), Gunning Fog Index (GFI Coleman-Liau Index (CLI), Flesch-Kincaid Grade Level (FKGL), Flesch-Kincaid Reading Ease (FKRE). In [2] we see review text variables such as unique word count which is referred as set length, Wrong Words, Lex Diversity which is the proportion of unique words in the review along with Noun, Adjective and verb and One letter Words. Also, they have used Flesch Reading Ease and Dale Chall Reading which indicate the ease of reading of the reviews. [3] have considered psychological variables such as 'Analytic' which is the level of formal, logical and hierarchical thinking, 'Clout' which indicates the level of expertise and confidence in the review, posEmo which is the amount of positive words in the review, negEmo which is the amount of negative words in the review. They have also considered linguistic variables such as WordCount, Words Per Sentence (WPS) which is the average word-count per sentence indicating the conciseness along with readability of sentences in the review and 'Compare' which is the proportion of comparison words such as 'better', 'smaller' etc. and is useful in situations where products or its features need to be compared with other products. [5] have extracted the positive and negative sentiment words from the review text and calculated the features subjectivity, polarity, neg_refs_per_ref, pos_refs_per_ref, senti_diffs_per_ref which are based on these sentiment words.

2) *Reviewer*

Along with the review text of the review, the information regarding the reviewer may also be useful in estimating how helpful the review of the reviewer can be. Profiling the activity of the reviewer is an important aspect of this. [1] have introduced two variables of reviewers namely productivity score and helpfulness per day. [1] have stated the importance of temporal dimension in making a prediction. Reviewer Helpfulness per day is computed as the ratio of aggregate helpfulness attracted by the reviews on the number of days between which the user was active in giving reviews. This parameter represents how frequently a customer writes reviews which have received high helpful votes. This implies that reviewers having high helpfulness per day make more helpful reviews available to the customers that help them making better purchase decisions. Productivity score adds the temporal dimension to the reviews produced by the reviewer.

Similarly, in [5] they have considered Reviewer's ranking, helpful percentage and review No. as the features. Reviewer's ranking is posted on Amazon reviewer's profile page. It is calculated by the overall helpfulness of the reviews taking into consideration the amount of reviews written by the customer. Helpful Percentage, the percentage of helpful votes received by the reviewer's previous reviews. Review No. is the amount of prior reviews the reviewer has written. These features take the history of the reviewer's performance to predict how helpful the upcoming reviews of the reviewer will be. The usage of these features is based on the assumption that the reviews given by a reviewer are almost of the same quality. In [7], they have considered features like reviewer expertise for predicting helpfulness. As the dataset is movie data, the similarity of a movie with the previous movies watched by the reviewer is calculated which indicates the reviewer expertise and is used for predicting the helpfulness of this review.

3) *Review Metadata*

Apart from the review text and reviewer variables, the information obtained from the

metadata of the review also is significantly useful in predicting the helpfulness of a review.

In [2], they have considered the customer question-answer data and product description data for the analysis. The two features extracted are DescSim and QASim. DescSim is the similarity between the review text and the product description. It is calculated as the cosine similarity between the bag of words vectors of product description and review text. Similarly, questions from the question-answer data is taken and the cosine similarity between questions and the review text is calculated which is represented by QASim. [3]&[5] used rating of a review for prediction and considered it as a confirmatory variable determining review helpfulness. Age of a review which is the amount of days since the review was posted is used in [5] for predicting the helpfulness stating that initial reviews tend to be more useful being comprehensive and giving detailed account of the aspects of the product which may leave little room for the newer reviews to contribute and comment about. [7] have pointed out the decay of helpfulness of a review over time. They have considered the Timeliness of a review as a parameter for helpfulness and hypothesized the exponential decay is the behavioral property of helpfulness of a review.

III. IMPLEMENTATION

The following sections describe the procedure followed in implementing the helpfulness prediction system. Firstly, the data is collected and filtered followed by feature extraction. The feature extraction is divided into three categories. The first one is the reviewer features which are the variables extracted based on the reviewer's activity over time. The second category is review text features. Here the features are extracted from the text of the review which mainly focus on the content of text and the way it is written. The third category is review meta data. This includes the data that is present with the review apart from the review text which further describes the review entry and may also have the information of the product on which the review is on. Finally, machine learning is used to predict the helpfulness score where it is taken as a regression problem. The flow of the implementation is described in fig. 1.

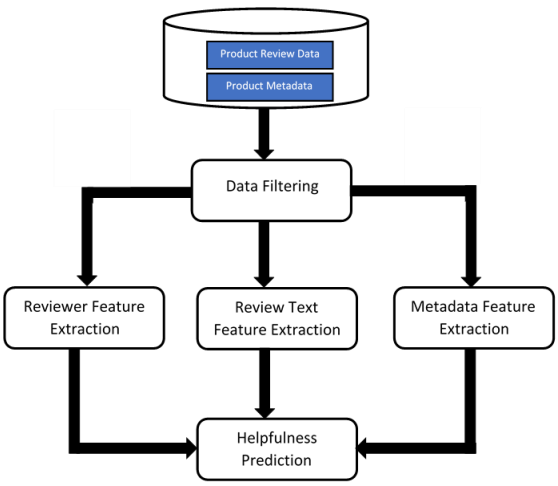

Fig. 1. Architecture of Proposed System

A. *Data Collection*

The data taken is product reviews from Amazon.com. The dataset is extracted from Julian Mcauley, UCSD website [8], [9] and is of category 'Electronics'. Along with that 'Electronics' Metadata is also extracted which contains the metadata about the products on which the reviews are written. After that, only those reviews which received at least 10 votes are considered for analysis. Furthermore, reviews for products which don't have a description are only used for measuring user-related features and are not-at-all used anywhere else. The distribution of helpfulness score in the data is depicted in fig. 2.

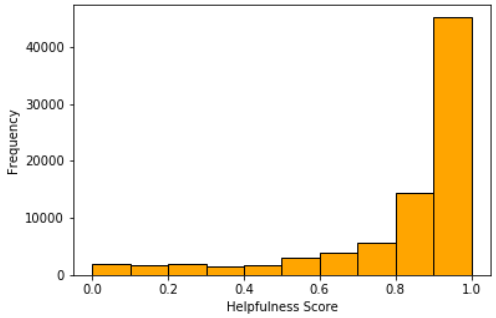

Fig. 2. Distribution of Helpfulness Score over Data

B. *Feature Extraction*

A total of 19 features categorized into reviewer, review text and review metadata are used for prediction. The features are User Average Rating Deviation, User Average Delay, Difficult Words, SMOG Index, Flesch Reading Ease, Coleman-Liau Index, Linsear Write Formula, Gunning Fog Index, Automated Readability Index, Flesch-Kincaid Grade Level, Dale-Chall Readability Score, Review Length, Sentence Count, Words Per Sentence, No. of Positive Words, No. of Negative Words, Stem Sim Length, Lem Sim Length, Rating. The dependent variable here is helpfulness score which needs to be predicted using these features.

1) Reviewer Features

a) User Average Rating Deviation

This is the average of the deviation of a user's rating in his/her reviews from the average rating for a product. This feature is generally used in spam detection [12], [13]. Firstly, for each product the average rating given to it by all the reviews is calculated. Then, the deviation of the rating of a review from this average rating is calculated which is the absolute value of the review rating subtracted by average rating given to the product. After that, for each user, the average value of this rating deviation is calculated. This is calculated for the users over the entire data including training and testing data and also including those reviews for which the product description is unavailable

b) ser Average Delay

This feature represents the average time a reviewer takes to give a review for a product. Firstly, the unix time at which the first review was posted for each product is noted. Then, the time difference of each review with the first review of the product is calculated. Finally, for each reviewer the average of this delay of his/her reviews is calculated and stored.

2) Review Text Features

a) Difficult Words

This feature measures the count of difficult words found in the review.

Difficult words are those words which are not used frequently, being specific and in general may be long [2]. For this the function difficult_words from the package textstat was used. For each word in the review, this method checks for the presence of this word in a 3000 word long 'easy word' list. If it is not present in the list, then it is classified as a difficult word. The count of these difficult words is the required feature.

b) Readability Variables

These variables measure the readability of text by calculating the amount of education required to comprehend the text [1], [17]. The following popular readability variables are used here:

- SMOG Index
- Flesch Reading Ease
- Coleman-Liau Index
- Gunning Fog Index
- Automated Readability Index
- Flesch-Kincaid Grade Level
- Linsear Write Formula
- Dale-Chall Readability Score

c) Other Text Features

Text features such as review length which is the length of a review in number of words, Sentence Count, Words Per Sentence are used in prediction also, the readability variables use them in their formula. Also, No. of Positive Words and No. of Negative Words [14], [15] are considered as well.

3) Review Meta Data Features

a) Stem Sim Length and Lem Sim Length

These features measure the ability of the review to comment on the product or aspects of the product. Here, the goal is to find out to what extent the review talks about the product and its aspects.

The features stem sim length and lem sim length attempt to analyze upto what extent the reviews talk about the product or its aspects and how relevant the review is to the context of the product. They basically measure the ability of the review to comment on the product or its aspects approximately. Product description was used for analysis in studies such as [2]. The process here mainly involves three stages.

The first stage deals with the product description. As the description is in the form of paragraph text, it is split into a list of words. After that, the stopwords are removed. Then, pos-tagging is done. Here, only the Nouns are extracted. Stemming of words is done and this list is stored. Similarly, lemmatization of words is done and the list is stored. This is shown in fig. 3.

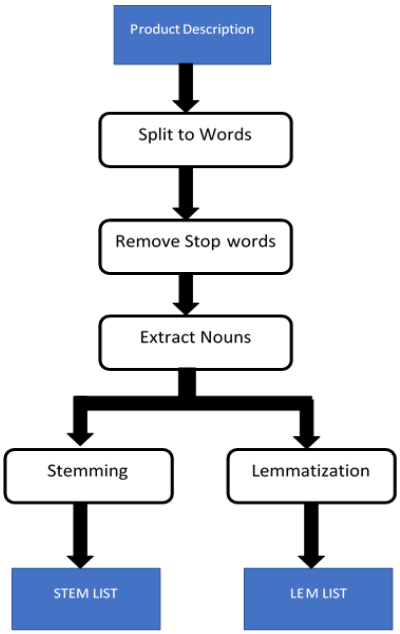

Fig. 3. Word List Extraction from Product Description

The second stage deals with the review text. Firstly, the review text is split into words. Then, the stop words are removed. Finally, stemming and lemmatization of the words is done. The stemmed words list and lemmatized words list are stored separately. This is shown in fig. 4.

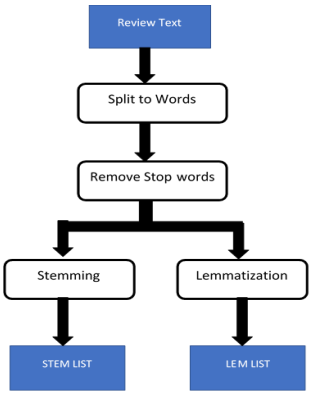

Fig. 4. Word List extraction from review text

The third stage deals with the comparison of the description and the review. Here, a set intersection is done between the stemmed words of review and the stemmed list of the corresponding product description. The length of this

set is the feature stem sim length. Similar procedure is applied for lemmatized words and the feature lem sim length is also extracted.

b) Rating

This is the score given by the reviewer to the product. [3] has asserted its importance in predicting helpfulness of reviews. Here, it ranges between 0 to 5 and indicates the overall opinion of the user towards the product.

C. *Helpfulness Prediction*

After the features are extracted, they are used by a machine learning model to predict the helpfulness score which ranges from 0 to 1. The machine learning models used here are Linear Regression and Random Forests. The Random Forests model is used with attributes 'n_estimators' as 500 which is the number of trees used in the model and random state as 42 which is the seed used in random number generator in the model. The data is split into training data and testing data for the training and testing of the respective models. Scikit-learn package in python [16] is used for this purpose.

D. *Performance Evaluation*

The Evaluation is done through cross-validation where the dataset is divided into 60% training and 40% testing. The performance of the models is evaluated using the metrics Mean Absolute Error (MAE), Mean Squared Error (MSE) and Root Mean Squared Error (RMSE). MAE is the average of absolute values of difference between the predicted and actual scores. In MSE, the values of error which is the difference between predicted and actual score, are squared and averaged. The root of this value is RMSE.

IV. RESULTS

A. *Feature-wise Correlation with helpfulness Analysis*

This section deals with the correlation analysis of each feature with helpfulness score. Higher the absolute value, higher is the correlation and higher the ability to explain the behavior of helpfulness by the feature in a linear fashion which is simple and easy to interpret. The correlation of helpfulness with each feature sorted in descending order of the absolute value of correlation is shown in II. It can be observed that the rating of the review (overall) has the highest correlation with helpfulness with a value of around 0.47. This is followed by user average rating deviation, Number of positive words and difficult words.

II. Feature – Helpfulness Correlation

Feature	Correlation
rating	0.476
user_deviation	-0.331
pos_no	0.224
difficult_words	0.217
review_length	0.197
lem_sim_length	0.172
stem_sim_length	0.171
sentence_count	0.122
neg_no	0.12
smog_index	0.116
dale_chall_readability_score	0.114
flesch_reading_ease	-0.112
automated_readability_index	0.111
flesch_kincaid_grade	0.111
gunning_fog	0.11
wps	0.11
coleman_liau_index	0.108
user_delay	-0.095
linsear_write_formula	0.015

B. Feature-wise Results

Here, the goal is to analyze which feature predicts helpfulness with minimal error. The metrics used here are Mean Squared Error (MSE), Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE). The results represented by these metrics for Linear Regression and Random Forests are shown in III. In the graph represented by fig. 5, the MSE of least of Linear Regression and Random Forests algorithms of the top ten features with least error is shown. It is observed that rating of the review gives the least error which is 0.044. This is followed by no of positive words, difficult words and user average rating deviation. In the graph represented by fig. 6, the MAE of least of Linear Regression and Random Forests algorithms of top ten features with least error is shown. Similar to MSE, it is observed that the rating of the review gives the least error which is 0.144. This is followed by Number of Positive words, User Average Rating Deviation and Difficult words respectively.

Feature-wise Results

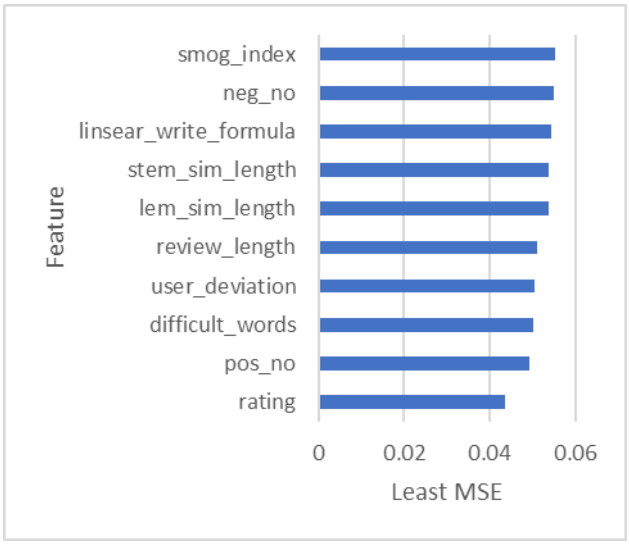

Fig. 5 MSE Top-10 Features

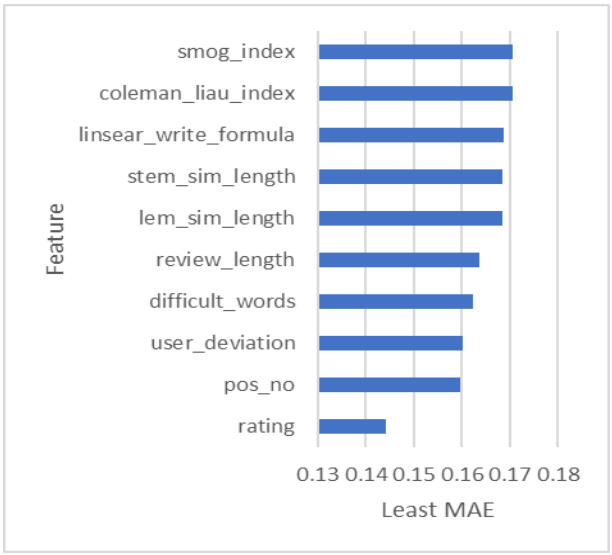

Fig. 6 MAE Top-10 Features

C. Category-wise Results

This section covers the comparative analysis of helpfulness prediction results obtained for each category namely reviewer features, review text and review meta data and for both machine learning algorithms linear regression and random forests for each of these categories. The MAE and MSE for both machine learning models is depicted in IV. In the graph represented by fig. 7 the MSE of least of linear regression and random forests of these categories is shown. In the graph represented by fig. 8 the MAE for the same is depicted. In both the cases the meta data features showed the best performance with and MSE of 0.042 and MAE of 0.1427. This is followed by review text features for MSE where as it is reviewer features (user) for MAE.

III. Category-wise Results

Feature	mae-LR	mse-LR	mae-RF	mse-RF
rating	0.146	0.044	0.144	0.044
pos_no	0.170	0.054	0.160	0.049
difficult_words	0.170	0.054	0.162	0.050
review_length	0.171	0.054	0.164	0.051
lem_sim_length	0.171	0.055	0.168	0.054
stem_sim_length	0.171	0.055	0.169	0.054
sentence_count	0.172	0.056	0.171	0.055
neg_no	0.172	0.056	0.171	0.055
smog_index	0.172	0.056	0.171	0.055
dale_chall	0.172	0.056	0.173	0.057
ARI	0.172	0.056	0.173	0.057
flesch_kincaid_grade	0.172	0.056	0.171	0.055
gunning_fog	0.172	0.056	0.178	0.060
wps	0.172	0.056	0.173	0.057
coleman_liau_index	0.172	0.056	0.171	0.056
linsear_write_formula	0.173	0.057	0.169	0.054
user_delay	0.173	0.056	0.171	0.057
flesch_reading_ease	0.172	0.056	0.173	0.057
user_deviation	0.160	0.051	0.172	0.063

Category	mae-LR	mse-LR	mae-RF	mse-RF
User Features	0.160	0.050	0.164	0.056
Review Text Features	0.165	0.051	0.163	0.050
Meta Data Features	0.146	0.043	0.143	0.043

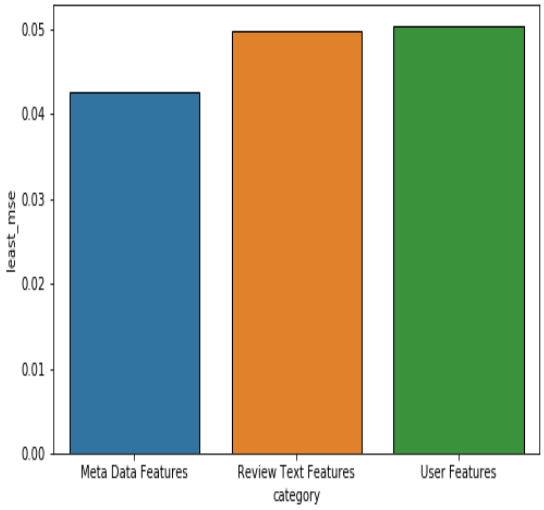

Fig. 7 Category-wise MSE

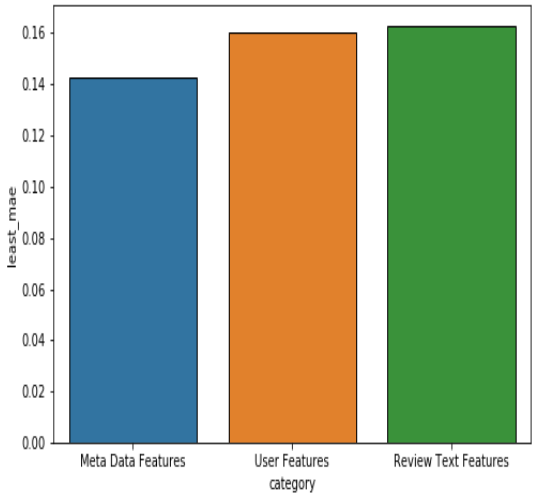

Fig. 8 Category-wise MAE

D. Prediction with entire feature-set

Here, all the features are taken by the algorithms Linear Regression and Random Forests to predict the helpfulness score. The values of MAE, MSE and RMSE are shown in V. The difference between the performance of linear regression and Random Forests in terms of MSE and MAE can be observed through the graphs represented by fig. 9 and fig. 10 respectively.

IV. Prediction Results with total features

Algorithm	**MAE**	**MSE**	**RMSE**
Linear Regression	0.1415	0.0404	0.2011
Random Forest	0.1368	0.0380	0.1951

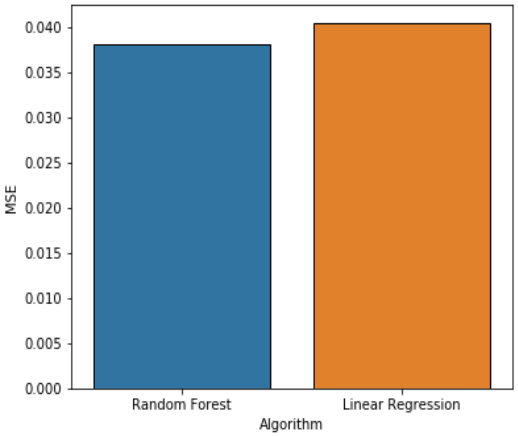

Fig. 9 Machine Learning Algorithm-wise MSE

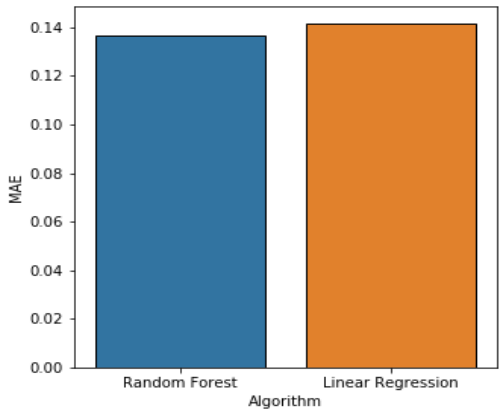

Fig. 10 Machine Learning Algorithm-wise MAE

V. CONCLUSIONS

The purpose of this work is to develop a system which automatically predicts helpfulness of product reviews along with analyzing the factors that influence the helpfulness of reviews. Here, we have divided the features into three categories reviewer, review text and review meta data and evaluated their predicting capabilities individually and combined.

The rating of a review shows immense impact on review helpfulness. This is followed by features such as Number of Positive Words, Number of Difficult Words, Lemmatized Description Similar Words, Stemmed Description Similar Words. Review meta data showed the best performance out of the three categories. Also, in most of the cases Random Forest Algorithm showed the better performance than Linear Regression. Readability features showed high inter-feature correlation among themselves. This is followed by features such as Number of Difficult Words, Number of Positive Words and Length of Review. But Positive Word Number and Difficult Word Number showed better performance than Review Length individually in predicting helpfulness. It can be said that they are a better version of review length consisting only important segments of a review in the context of helpfulness prediction.

The best performance of the system is achieved when all the features are used together and Random Forests algorithm is used as the machine learning algorithm to predict helpfulness score. The system achieved a Mean Squared Error (MSE) of 0.038, Root Mean Squared Error (RMSE) of 0.1951, Mean Absolute Error (MAE) of 0.1368 in predicting helpfulness score of reviews.

VI. FUTURE WORK

This work can be extended and enhanced in several possible ways. Firstly, further features can be explored in all three categories of reviewer, review text and review meta data and their influence in predicting helpfulness can be analyzed. Also, different sets of independent variables can be

compared in predicting helpfulness. Helpfulness received by previous reviews of the reviewer can be used to predict helpfulness of other reviews. Algorithms such as Extreme Gradient Boosting which is an optimized implementation of Gradient Boosting can be utilized. Deep Learning can be used which can reduce the workload of feature extraction but giving an accurate result. This work focused on Electronics data. Similar analysis can be done on data of other categories. Reviews obtaining less than ten votes were filtered-out here. These reviews can also be considered for analysis. Utilizing the relationship among the features for optimizing the system performance can be done. Further, this can also be taken as a classification problem or as a combination of classification and regression in order to get better results. The current problem can be considered as a Big Data problem since the number of reviews involved is large and a lot of processing is involved. Hence utilization of distributed and parallel computing can be explored for faster and efficient performance.

REFERENCES

1. Malik, M. S. I. and Ayyaz Hussain. “An analysis of review content and reviewer variables that contribute to review helpfulness.” Inf. Process. Manage. 54: 88-104, 2018.
2. Saumya, Sunil, Jyoti Prakash Singh, Abdullah M. Baabdullah, Nripendra P. Rana and Yogesh Kumar Dwivedi. “Ranking online consumer reviews.” Electronic Commerce Research and Applications 29: 78-89, 2018.
3. Park, Yoon-Joo “Predicting the helpfulness of online customer reviews across different product types”, Sustainability, 10. 1735, 2018.
4. Y.K. Chua, Alton & Banerjee, Snehasish “Helpfulness of user-generated reviews as a function of review sentiment, product type and information quality”, Computers in Human Behavior, 54. 547-554, 2016.
5. Y. Zhang and D. Zhang, "Automatically predicting the helpfulness of online reviews," Proceedings of the 2014 IEEE 15th International Conference on Information Reuse and Integration (IEEE IRI 2014), Redwood City, CA, pp. 662-668, 2014.
6. Mudambi, S. M., & Schuff, D. “What makes a helpful review? A study of customer reviews on Amazon. com.”, MIS Quarterly, 34(1), 185–200, 2010.
7. Y. Liu, X. Huang, A. An and X. Yu, "HelpMeter: a nonlinear model for predicting the helpfulness of online reviews," 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, Sydney, NSW, pp. 793-796, 2008.
8. R. He, J. McAuley, “Modeling the visual evolution of fashion trends with one-class collaborative filtering.”, WWW, 2016.
9. J. McAuley, C. Targett, J. Shi, A. van den Hengel, “Image-based recommendations on styles and substitutes.”, SIGIR, 2015
10. Merton, R.K., The Matthew effect in science: the reward and communication systems of science are considered. Science 159 (3810), 56–63, 1968.
11. Kim, S.M.; Pantel, P.; Chklovski, T.; Pennacchiotti, M. Automatically assessing review helpfulness.In Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing, Sydney, Australia; pp. 423–430, 22–23 July 2006 .
12. A. Mukherjee, V. Venkataraman, B. Liu, and N. Glance, “What yelp fake review filter might be doing?”, 2014.
13. X. Li, L. Xie, F. Zhang and H. Wang, "Online Deceptive Product Review Detection Leveraging Word Embedding," 2017 IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing, 15th Intl Conf on Pervasive Intelligence and Computing, 3rd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech), Orlando, FL, pp. 867-870, 2017.
14. Minqing Hu and Bing Liu. "Mining and Summarizing Customer Reviews." Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2004), Seattle, Washington, USA, Aug 22-25, 2004.
15. Bing Liu, Minqing Hu and Junsheng Cheng. "Opinion Observer: Analyzing and Comparing Opinions on the Web." Proceedings of the 14th International World Wide Web conference (WWW-2005), Chiba, Japan, May 10-14, 2005.
16. Pedregosa F. , Varoquaux G. , Gramfort A. , Michel V., Thirion B. , Grisel O. , Blondel M. , Prettenhofer P. , Weiss R. , Dubourg V. , V,erplas J. , Passos A. , Cournapeau D. , Brucher M. , Perrot M. , Duchesnay E.,"Scikit-learn: Machine Learning in Python", Journal of Machine Learning Research, Volume 12, 2825-2830, 2011
17. Eltorai, Adam & S Naqvi, Syed & Ghanian, Soha & Eberson, Craig & Weiss, Arnold Peter & Born, Christopher & Daniels, Alan, "Readability of Invasive Procedure Consent Forms", Clinical and translational science,2015.
18. Pandey S., M S., Shrivastava A., “Data classification using machine learning approach”, In: Thampi S., Mitra S., Mukhopadhyay J., Li KC., James A., Berretti S. (eds) Intelligent Systems Technologies and Applications. ISTA 2017. Advances in Intelligent Systems and Computing, vol 683. Springer, Cham, 2018.
19. V. Vishagini and A. K. Rajan, "An improved spam detection method with Weighted Support Vector Machine," 2018 International Conference on Data Science and Engineering (ICDSE), Kochi, pp. 1-5, 2018.

AUTHORS PROFILE

Kagolanu Trishul pursued M. Tech. in Computer Science at Amrita School of Engineering, Bengaluru, Karnataka, India.

Srinath R. Naidu is working as Associate Professor at Amrita School of Engineering, Bengaluru, Karnataka, India. He received Ph.D in the area of statistical timing analysis for digital integrated circuits from Eindhoven University of Technology in 2004