e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technolog y and Science
Volume:03/Issue:06/June-2021 Impact Factor- 5.354 www.irjmets.com

E-COMMERCE PRODUCT RATING BASED OCUSTOMER REVIEW MINING

Nandini Sharma*1, Abhijeet Deshwal*2, Prachi Tomar*3, Mrs. Babli Kumari*4,

Prof. Umang Rastogi*5

*1,2,3,4,5Dept. Of Computer Science & Engineering, Meerut Institute Of Engineering And Technology, India.

ABSTRACT

With the rapid increase in spread of digitalization, the number of users of E-commerce sites is increasing day by day. Only with the help of online shopping, many E-commerce enterprises were unable to know whether the customers were satisfied by the services provided by the firm. To resolve this issue, a section called ' Customer reviews ' came into existence. This enabled the E-commerce enterprises and the manufacturers to get their customers' opinion to improve services and merchandise through mining customers' reviews. On the other hand, this also helped the subsequent buyers of a product by providing them additional information about the same. But often there existed conflicting reviews and rating. This model finds a solution to this problem by analysing the aggregate reviews given by the customers and thereby providing a rating to the product.

Keywords : E-Commerce Dataset, Product Rating, Sentiment Analysis.

I. INTRODUCTION

Other people's views and opinions have always been an important source of information for everyone. A lot of goods are available on the Internet and with the apid rise of e-commerce, many individuals are now purchasing products online. In order to increase customer loyalty, demands and online shopping experience, online retailers have started a popular practise to encourage their customers to suggest opinions on the items they have purchased. The Internet has eventually become an excellent means of sharing these views. Any of the web pages that contain such views are astronomically huge and are increasingly growing. An overwhelming number of people write reviews and publish them that are being useful for others as more and more popular users become familiar with the Site. As a consequence, the number of feedbacks that a product gets is rising exponentially. At some big merchant pages, some famous items can get hundreds of ratings. This has now become the most trustworthy types of advertisement. In many cases, a product can have a huge number of feedbacks containing both positive and negative ones. It is practically impossible to read all the feedback in order to gain conclusion and make a judgement. Or in some cases there can occur conflicting reviews and ratings. A 5-starred product can happen to have very poor feedbacks. The reason behind this can be- some people like to review a product whereas some like to rate it. This cannot be a mandate for the users to do both the tasks. So in order to make a unanimous judgement, we have come up with a model that can put things straight. It has a very simple yet complicated functioning. All the feedbacks given by the clients are the input to our algorithm. By assessing all of them, our model will give you a promising rating. And so the customer can decide whether he wants or doesn't want to make his purchase. The method of computing sentiment and opinion has been regarded as a difficult field of study that can be useful for various purposes. Product aspect ranking consists of three major tasks: product aspect recognition, sentiment-based grouping and product aspect ranking.

II. LITERATURE REVIEW

This section includes : (1) online customer analysis of reviews and (2) Online customer review for filtering and weighting.

2.1) online customer analysis of reviews : Collective wisdom is expressed through online customer reviews. These ratings and reviews indicate future results when used effectively [4]. It is important to evaluate customer reviews and ratings to determine customer requirements for market success. The majority of work on the processing of online customer reviews focuses on opinion mining, which seeks to classify the attitudes of reviewers, whether positive or negative, with respect to different characteristics of a product. The positive review indicates great message opinion; and generally, the negative review demonstrates the unfortunate message [5]. For firms, positive reviews are a crucial way to market their product, with customers willing to spend 31% more on a company with excellent reviews, while negative reviews have a devastating effect on the product. Theoretical methods have been developed by many researchers for customer review analyses, Wong and Lam classify customer needs on various auction websites using hidden Markov models and conditional random fields from customer feedback [8] . An automated approach that differentiates between positive and negative customer feedback is proposed by Dave et al [6]. Menon et al. present a method of representation of vector space document deriving customer requirements from consumer reviews and numerical ratings for new product development [7].Wu and Huberman analyze the temporal evolution of large-scale customer reviews, discovering that latter reviews tend to show a large difference from previous ones, which in turn moderates the average review to the less extreme [9]. In different fields of study, including data-driven product design, online customer reviews is used. In order to improve new methods of product design, Liu et al. propose four feature categories that reflect designers' viewpoints. They develop a method that automatically evaluates the helpfulness of online customer reviews from a designer's perspective [3]. Tucker and Kim propose a method that uses customer review data to model and forecast emerging product trends [10].Chen et al. examine how the social standing of reviewers affects consumer reactions to consumer review information. They notice that the consumer reviews highlighted impact product sales more strongly than other reviews[11].

Opinion mining refers to the use of "natural language processing, computational linguistics, and text mining to differentiate" whether or not the motion picture is nice or not depending on the feeling of the message. The problem for the next few decades is the analysis of natural language data. It's an incredibly difficult problem, and when looked at in isolation, sarcasm and other kinds of ironic language are inherently difficult for machines to detect. It's imperative to have a sufficiently sophisticated and rigorous enough approach that relevant context can be taken into account. For example, that would require knowing that a particular user is generally sarcastic, ironic, or hyperbolic, or having a larger sample of the natural language data that provides clues to determine whether or not a phrase is ironic [2].

2.2) Online customer review for filtering and weighting : While reliable reviews are necessary for analyses of customer reviews, inaccurate reviews including null reviews , may impede accurate and objective analyses of customer reviews. The problems associated with unreliable reviews have recently increased as well. Filtering such reviews helps removing null reviews, which improves performance and reduces the bias caused by such reviews. For removing null data isnull() function is used, isnull() function detect missing values in the given series object and then reset_index() method sets a list of integer ranging from 0 to length of data as index. For more reliable reviews we used WordNet from ntlk corpus, WordNet is a lexical database which was developed by Princeton. To find the meanings of terms, synonyms, antonyms, and more, we used WordNet alongside the NLTK module, and after filtering, the model puts weight on all the positive and negative reviews with the help of sentiment analyser and rate the product accordingly. If the positive responses are greater, the rating will shift towards the better side and vice versa.

III. METHODOLOGY

Since many people need a product review before they spend money on it and actually buy it. In certain instances, a product will get an overwhelming amount of customer feedback. In order to draw a consensual and make a decision, it is extremly difficult to read all the reviews, Our model analysis user's reviews and rates a product according to them. The framework can assist many e-commerce firms to develop or sustain their offerings on the basis of consumer feedback and to enhance products on the basis of customer reviews.

Our model basically works in 5 parts:-

1. Data Gathering
2. Data cleaning
3. Data Shaping
4. Training model using data
5. Predictions on new inputs using the trained model

1. Data Gathering :- We downloaded the dataset from Kaggle and are training our model to work upon all the reviews given on that particular dataset.

Link to the dataset:- https://www.kaggle.com/nicapotato/womens-ecommerce-clothing-reviews

	Unnamed: 0	Clothing ID	Age	Title	Review Text	Rating	Recommended IND	Positive Feedback Count	Division Name	Department Name	Class Name
0	0	767	33	NaN	Absolutely wonderful - silky and sexy and comf...	4	1	0	Initmates	Intimate	Intimates
1	1	1080	34	NaN	Love this dress! it's sooo pretty. i happene...	5	1	4	General	Dresses	Dresses
2	2	1077	60	Some major design flaws	I had such high hopes for this dress and reall...	3	0	0	General	Dresses	Dresses
3	3	1049	50	My favorite buy!	I love, love, love this jumpsuit. it's fun, fl...	5	1	0	General Petite	Bottoms	Pants
4	4	847	47	Flattering shirt	This shirt is very flattering to all due to th...	5	1	6	General	Tops	Blouses

2. Data Cleaning :-

2.1. Lemmatization: Lemmatization takes the morphological interpretation of words into consideration. To do so, it is necessary to have detailed dictionaries which the algorithm can look through to link the form back to its lemma. It reduces the inflected words to ensure that the root word belongs to the language. The root word in Lemmatization is called Lemma. A lemma is the form of a collection of words in canonical form, dictionary form, or citation form.

2.2. Removing nosiy data: This process involve elimination of unnecessary and useless data. For example, if the input given by the user is "The product was good". The output after the removal of unnecessary data will be "product, good".

2.3. Removing null data: For removing null data isnull() function is used, isnull() function detect missing values in the given series object and then reset_index() method sets a list of integer ranging from 0 to length of data as index.

```
0    Absolutely wonderful - silky and sexy and comf...
1    Love this dress!  it's sooo pretty.  i happene...
2    I had such high hopes for this dress and reall...
3    I love, love, love this jumpsuit. it's fun, fl...
4    This shirt is very flattering to all due to th...
Name: Review Text, dtype: object
```

3. Data Shaping :-

3.1. Tokenization :- For tokenization, we used punkt. Now with the help of tokenization, we divide the sentence into tokens. The role of tokenizer is to break the text into a lost of sentence which helps in building the model.

3.2. Creating Corpus :- We then created a corpus list in which we added all the words broken down by the tokenizer. Now, apart from the alphabets in lower case (a-z) and the alphabets in upper case(A-Z), we removed everything from the text including numbers, symbols etc. Now we turned all the upper case texts into lower case, tokenized them , lemmatized and sent it back into the corpus list.

3.3. Indexing :- For indexing we used fit_on_text, this method creates the vocabulary index based on word frequency. So if you give it something like, "The cat sat on the mat." It will create a dictionary s.t. word_index["the"] = 1; word_index["cat"] = 2 it is word -> index dictionary so every word gets a unique integer value. 0 is reserved for padding. So lower integer means more frequent word.

3.4. Pad Sequencing :- Pad_sequence is used to ensure that the same length is given to all sequences in a list. This is achieved by padding 0 in the beginning of each sequence until each sequence has the same length as the longest sequence.

```
[[   0    0    0 ...  724  504   24]
 [   0    0    0 ...  320  598   39]
 [   0    0    0 ... 2723  327  226]
 ...
 [   0    0    0 ...   12   11   10]
 [   0    0    0 ...   33 1982  119]
 [   0    0    0 ...  162  315  138]]
```

4. Training model using data :-

4.1. train_test_split :- In Sklearn model selection, train-test-split is a feature to split data arrays into two subsets: for training data and for testing data. 80% of data has been trained and 20% has been tested. The variable 'X' is used for input and 'Y' is used for output. And a random state, 42, has been taken which is responsible for the consistent output, irrespective of the number of times the program is run.

4.2. word_embedding :-In this process, word2vec takes place. Basically a matrix is made in the process in which the words are exchanges with their vector values. Skip words are also included in this matrix.

4.3. Building the model :- Now we build a model and make it sequential and add the embedding layers to it. We ran simple RNN in the model in which the activation value and written sequence are fed. Thereby adding the dropout into it. The model is then run for dense layer. This time giving it a soft max activation. Then we compile our model to check the accuracy and minimize the loss. At last, the model is fitted. In this process, we run epochs = 50 to ensure accuracy.

```
Epoch 1/50
566/566 [==============================] - 46s 79ms/step - loss: 0.3888 - accuracy: 0.5520 - val_loss: 0.3276 - val_accuracy: 0.5959
Epoch 2/50
566/566 [==============================] - 44s 78ms/step - loss: 0.3301 - accuracy: 0.5979 - val_loss: 0.3223 - val_accuracy: 0.6149
Epoch 3/50
566/566 [==============================] - 44s 78ms/step - loss: 0.3142 - accuracy: 0.6116 - val_loss: 0.3254 - val_accuracy: 0.6076
Epoch 4/50
566/566 [==============================] - 44s 78ms/step - loss: 0.3082 - accuracy: 0.6206 - val_loss: 0.3094 - val_accuracy: 0.6209
Epoch 5/50
566/566 [==============================] - 44s 77ms/step - loss: 0.3023 - accuracy: 0.6281 - val_loss: 0.3175 - val_accuracy: 0.6299
Epoch 6/50
566/566 [==============================] - 44s 78ms/step - loss: 0.2914 - accuracy: 0.6447 - val_loss: 0.3195 - val_accuracy: 0.6032
Epoch 7/50
566/566 [==============================] - 44s 78ms/step - loss: 0.2830 - accuracy: 0.6564 - val_loss: 0.3084 - val_accuracy: 0.6143
Epoch 8/50
566/566 [==============================] - 44s 78ms/step - loss: 0.2735 - accuracy: 0.6681 - val_loss: 0.3151 - val_accuracy: 0.6171
Epoch 9/50
566/566 [==============================] - 44s 78ms/step - loss: 0.2639 - accuracy: 0.6817 - val_loss: 0.3188 - val_accuracy: 0.6220
Epoch 10/50
566/566 [==============================] - 44s 77ms/step - loss: 0.2579 - accuracy: 0.6885 - val_loss: 0.3264 - val_accuracy: 0.6319
Epoch 11/50
566/566 [==============================] - 44s 78ms/step - loss: 0.2478 - accuracy: 0.7027 - val_loss: 0.3239 - val_accuracy: 0.6235
Epoch 12/50
566/566 [==============================] - 44s 78ms/step - loss: 0.2404 - accuracy: 0.7120 - val_loss: 0.3336 - val_accuracy: 0.6030
Epoch 13/50
566/566 [==============================] - 44s 77ms/step - loss: 0.2384 - accuracy: 0.7200 - val_loss: 0.3308 - val_accuracy: 0.6127
Epoch 14/50
566/566 [==============================] - 44s 78ms/step - loss: 0.2318 - accuracy: 0.7239 - val_loss: 0.3940 - val_accuracy: 0.6315
Epoch 15/50
566/566 [==============================] - 44s 78ms/step - loss: 0.2258 - accuracy: 0.7299 - val_loss: 0.3582 - val_accuracy: 0.5500
Epoch 16/50
566/566 [==============================] - 44s 78ms/step - loss: 0.2222 - accuracy: 0.7373 - val_loss: 0.3570 - val_accuracy: 0.5986
Epoch 17/50
566/566 [==============================] - 44s 78ms/step - loss: 0.2127 - accuracy: 0.7532 - val_loss: 0.3595 - val_accuracy: 0.5789
Epoch 18/50
566/566 [==============================] - 44s 78ms/step - loss: 0.2078 - accuracy: 0.7598 - val_loss: 0.3687 - val_accuracy: 0.5928
Epoch 19/50
566/566 [==============================] - 44s 78ms/step - loss: 0.1998 - accuracy: 0.7660 - val_loss: 0.3569 - val_accuracy: 0.6079
Epoch 20/50
566/566 [==============================] - 44s 77ms/step - loss: 0.1956 - accuracy: 0.7768 - val_loss: 0.3998 - val_accuracy: 0.6085
Epoch 21/50
566/566 [==============================] - 44s 77ms/step - loss: 0.1899 - accuracy: 0.7830 - val_loss: 0.4048 - val_accuracy: 0.6041
Epoch 22/50
566/566 [==============================] - 44s 78ms/step - loss: 0.1948 - accuracy: 0.7751 - val_loss: 0.4442 - val_accuracy: 0.6081
Epoch 23/50
566/566 [==============================] - 44s 78ms/step - loss: 0.1926 - accuracy: 0.7785 - val_loss: 0.4120 - val_accuracy: 0.6050
```

5. Predictions on new inputs using the trained model :- The model finally works by converting the input review into a rating. The output will be the average rating of all the fed input reviews. With the increase in number of input reviews, the output generated will be more precise.

```
Reviews
['i hate this dress']
['i love this dress']
['Absolutely wonderful - silky and sexy and comfortable']
['size is bad']
['Love this dress! its sooo pretty.']
['i am upset, bad product']
['product was good']
['very bad product']
['Love this dress! it's sooo pretty. ']
['I had such high hopes for this dress and really amazing dress']
['I love, love, love this jumpsuit.']
['This shirt is very flattering']
```

```
Output

print(np.argmax(model.predict(pad_test)))

3
```

IV. RESULT

The model puts weight on all the positive and negative reviews with the help of sentiment analyser and rate the product accordingly. If the positive responses are greater, the rating will shift towards the better side and vice versa.

```
test_rev=['very nice  dress']
test_rev=['i love this dress']
test_rev=['Absolutely wonderful - silky and sexy and comfortable']
test_rev=['size is bad']
test_rev=['Love this dress! its sooo pretty.']
test_rev=['i am upset, bad product']
test_rev=['product was terrible']
test_rev=['very bad product']
test_rev=['Love this dress! its sooo pretty.']
test_rev=['I had such high hopes for this dress and really amazing dress']
test_rev=['I love, love, love this jumpsuit']
test_rev=['This shirt is very tight and not as per the expectation ']
token.fit_on_texts(test_rev)
sequence_test=token.texts_to_sequences(test_rev)
pad_test=pad_sequences(sequence_test,maxlen=max_length)

print(np.argmax(model.predict(pad_test)))

2
```

V. CONCLUSION

The objective of this proposed research is to calculate the average rating of a product from customers' reviews. With the help of the sentiment analyzer, our model put weight on all the positive and negative reviews and then rate the product accordingly. The rating would change towards the better side , if the positive responses are greater and vice versa. The proposed work is composed of five main steps. First, Data Gathering, In this step , We have downloaded dataset and trained our model to work on all the reviews provided on that specific dataset. Second, Data Cleaning, This process involve elimination of redundant and obsolete data, and the removal of null reviews. Third, Data Shaping, In this step we divided the data into tokens. Then we transformed all the reviews into lower case. Then indexed all of it and created a matrix by applying padsequence. And resized matrix and made all of them of the same size. Fourth Training model using data, we used train_test_split , train-test-split is a feature to split data arrays into two subsets for training and for testing and last, Predictions on new inputs using the trained model, In this process the model finally works by converting the input review into a rating. The output will be, all the fed input reviews' average rating. The output generated will be more accurate with the rise in the number of input reviews.

FUTURE SCOPE

After creating this model we have also left some room for future scope. Technology is advancing day by day and there may come a time when users could review a particular product by images, videos and gifs. So to analyse, understand and extract the actual emotions behind those graphics we would need a separate and a more advanced model. That model would understand what the graphic is saying and whether the feedback given is good, bad or neutral. The model would classify those images, videos and gifs separately according to their sentiment and then it would make a decision (rating) after aggregating all the weights of the sentiments concluded. Same condition might occur if there existed audio and sound feedbacks. In such case we might need an advanced model which can classify and segregate moods and sentiments according to the voice message. The model might work upon the tone of the voice and the words spoken. The use of voice to text converter can be done here, which is already available in the market.

VI. REFERENCES

[1] Li Zhuang,Feng Jing,Xiao-Yan Zhu,"Movie Review Mining and Summarization",CIKM’06, November 5–11, 2006, Arlington, Virginia, USA.

[2] Anurag Manni, Naman Jaiswal, Nayan Jaiswal, "Product Rating Based On Review Using Data Mining":International Journal of Advance Research, Ideas and Innovations in Technology,2017.

[3] Liu, Y., Jin, J., Ji, P., Harding, J. A., and Fung, R. Y., 2013, “Identifying Helpful Online Reviews: A Product Designer’s Perspective,” Comput.-Aided Des., 45(2), pp. 180–194.

[4] Asur, S., and Huberman, B. A., 2010, “Predicting the Future With Social Media,” IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT), Toronto, ON, Canada, Aug. 31–Sept. 3, pp. 492–499.

[5] Zhan, J., Loh, H. T., and Liu, Y., 2009, “Gather Customer Concerns From Online Product Reviews—A Text Summarization Approach,” Expert Syst. Appl., 36(2), pp. 2107–2115.

[6] Dave, K., Lawrence, S., and Pennock, D. M., 2003, “Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews,” 12th International Conference on World Wide Web (WWW), Budapest, Hungary, May 20–24, pp. 519–528.

[7] Menon, R., Tong, L. H., Sathiyakeerthi, S., Brombacher, A., and Leong, C., 2004, “The Needs and Benefits of Applying Textual Data Mining Within the Product Development Process,” Qual. Reliab. Eng. Int., 20(1), pp. 1–15.

[8] Wong, T.-L., and Lam, W., 2008, “Learning to Extract and Summarize Hot Item Features From Multiple Auction Web Sites,” Knowl. Inf. Syst., 14(2), pp. 143–160.

[9] Wu, F., and Huberman, B. A., 2010, “Opinion Formation Under Costly Expression,” ACM Trans. Intell. Syst. Technol., 1(1), pp. 1–13.

[10] Tucker, C., and Kim, H., 2011, “Predicting Emerging Product Design Trend by Mining Publicly Available Customer Review Data,” 18th International Conference on Engineering Design, Impacting Society Through Engineering Design, Lyngby/Copenhagen, Denmark, Aug. 15–18, pp. 43–52.

[11] Chen, P.-Y., Dhanasobhon, S., and Smith, M. D., 2008, “All Reviews Are Not Created Equal: The Disaggregate Impact of Reviews and Reviewers at Amazon.com,” SSRN, Rochester, NY, accessed Aug. 22, 2017, https://ssrn.com/abstract¼918083.