Sentiment Analysis of Amazon Product Reviews

***Abstract*— The aim of this mini-project is to perform text analysis like sentiment analysis on Customer Reviews of various products from e-commerce Amazon. The purpose of this project is to use various aspects of text analysis to analyze the customer reviews, ratings data for the various products and draw insights to the overall sentiment towards the products, and finally after analyzing the sentiment from review text and rating, try to predict whether the customer recommended the purchased product.**

Index Terms—Text Analysis, Sentiment Analysis, TF-IDF, cosine similarity, co-occurrence network

I. Introduction

In recent years E-commerce businesses are more interested to understand consumer's opinions and experience regarding the various products that they buy from the online platforms. The product reviews give the companies an insight to the success of the products and scopes for improvisation. It is very important for the companies to have good quality control; an easy way to monitor the quality of the products, or the standard of services is to scrutinize the customers' reviews. These e-commerce companies often analyze these reviews for a better understanding of the market and a customer's preferences. The product reviews play an important role for the consumers as well, most of us check the ratings and reviews online before deciding for purchase a particular item. If a particular product has a satisfactory overall rating and good reviews, it has better chance to be successful or a best seller.

The aim of my work is to use various text analysis techniques to analyze the reviews, ratings and try to retrieve the polarity (i.e., positive, or negative sentiment) and use this information to predict whether the consumer recommended the product. Text analysis has been used to extract the features from the dataset and supervised machine learning models like Random Forest and Naive Bayes has been used for prediction.

II. Data and resources

I have used the dataset that is available online, link has been provided in the references section [6]. This dataset comprises 5000 reviews of various products purchased from Amazon e-commerce platform. This dataset is in the form of a comma separated excel document, the main columns that are relevant for my work are as follows:

- name: Indicates the name of the product. Datatype is String
- reviews.doRecommend: Indicates whether the consumer recommends the product. Datatype is boolean
- reviews.rating: Indicates the rating provided by the consumer. Datatype is Integer
- reviews.text: This contains the review written by the consumer for the product. Datatype is string

I have used VADER (Valence Aware Dictionary and sEntiment Reasoner) which is a lexicon and rule-based sentiment analysis tool to determine the overall polarity of a text.

For analyzing the intensity of emotions of positive and negative feedbacks I have used the NRC Emotion Intensity Lexicon. This is a list of English words with real-valued scores of intensities for eight basic emotions (anger, anticipation, disgust, fear, joy, sadness, surprise, and trust).

III. Methodology

Mostly NTLK (Natural Language Toolkit) and its various libraries have been used for text analysis in this project.

A. Data Pre-Processing

The review text might contain various emoticons, so I have converted them to words indicating the emoticons e.g.: '\:\)' is 'happy', '\:\D' is 'laugh', '\:\(' is 'sad'.

NLTK packages have been used to pre-process the reviews text, below are the packages used:

1. sent_tokenize: Used to split a text into individual sentences
2. word_tokenize: Split a sentence into words. I have also converted text to lower case to resolve any ambiguity.
3. WordNetLemmatize: Aim to return the base or dictionary form of a word
4. nltk.corpus Stopwords: Aim to remove English stopwords i.e., words that do not add much meaning to the text. e.g.: the, she, has etc.

B. Basic Analysis of Dataset

Some basic analyses of Dataset have been performed after the cleanup of the reviews text to get an overview.

Out of the 5000 reviews the bar chart (Fig 1) shows the distribution of ratings for all the reviews. As depicted out of 5000 reviews around 4500 reviews are rated either 4 or 5 which is considered positive.

The frequency distribution in Fig 2 shows the top 10 most common words used from the review text. "great", "love", "tablet" are the top three most common words used in the reviews.

I have built word clouds for 4 products as shown in Fig 3. The word clouds display the most prominent and frequent words used in the review texts for each product. They assist in providing an easy and quick visualization of the textual data by characterizing the overall sentiment and emotion.

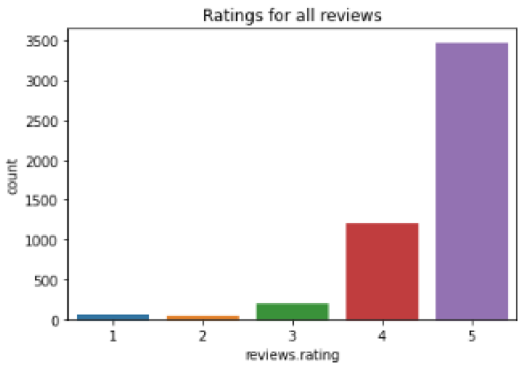

Fig 1. Bar Chart showing rating distribution

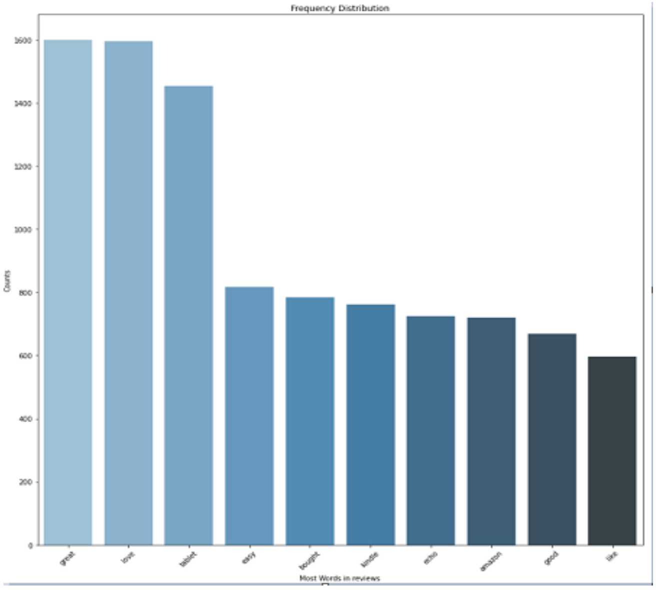

Fig 2. Frequency distribution of most common words

Fig 3. Word Cloud for reviews for 4 products

Next I have identified one word "connect" from the list of words that was generated from the review texts and using networkx I have built the co-occurrence network of the word as shown in Fig 4. Co-occurrence networks are graphical representation of the frequency of appearance of words together. This kind of representation helps to reconstruct and quantify the context of words in text, in other words it indicates how the words appear together in text.

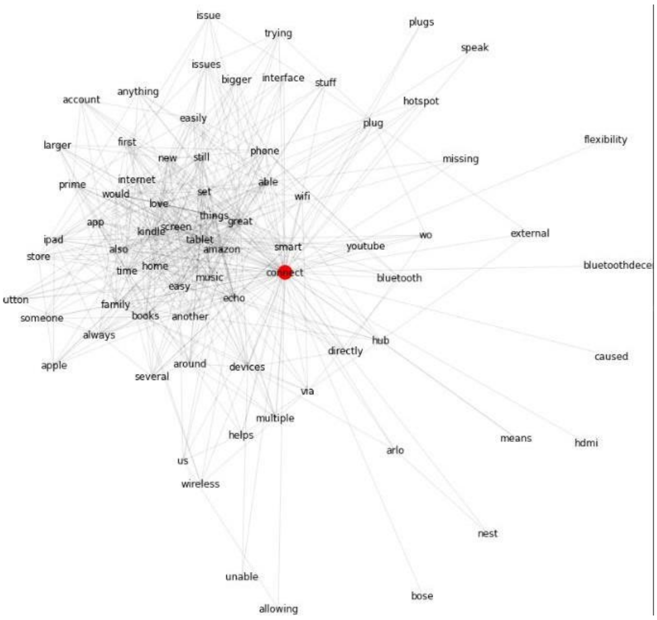

Fig 4. Co-occurrence network of the word "connect".

Using various centrality measures, I have identified the top central nodes (words) as shown in Fig 5. Degree centrality indicates how well connected a node is, depicts semantic richness. Closeness centrality measures how close the node is to all other nodes; it captures how easily the word relates to other words in a semantic/syntactic set of connected words. Page-rank measures the importance of a nodes based on the number and quality of links to other nodes. The words: "tablet", "great", "love", "amazon" are most central words.

	Degree	Closeness	PageRank
tablet	0.119599	0.483540	0.007784
great	0.109761	0.484083	0.007163
amazon	0.090085	0.467155	0.006004
love	0.078897	0.468044	0.004785
like	0.078318	0.462765	0.005199
kindle	0.077739	0.461567	0.004859

Sorted by Degree Centrality

	Degree	Closeness	PageRank
great	0.109761	0.484083	0.007163
tablet	0.119599	0.483540	0.007784
love	0.078897	0.468044	0.004785
amazon	0.090085	0.467155	0.006004
like	0.078318	0.462765	0.005199
kindle	0.077739	0.461567	0.004859

Sorted by Closeness Centrality

	Degree	Closeness	PageRank
tablet	0.119268	0.483525	0.007775
great	0.110019	0.484656	0.007183
amazon	0.089981	0.467383	0.006000
like	0.078227	0.463038	0.005195
kindle	0.077457	0.461676	0.004845
love	0.079191	0.468440	0.004805

Sorted by Page-Rank

Fig 5. Centrality of words

By Calculating the cosine similarity between the TF-IDF vectorized values of the review texts I have compared the similarity of review texts between 3 products: Amazon Fire Tablet, Kindle and Echo. From Fig 6. we can see the reviews for the products are very similar. These are all electronic products and are quite like each other in terms of features hence the reviews are also quite similar.

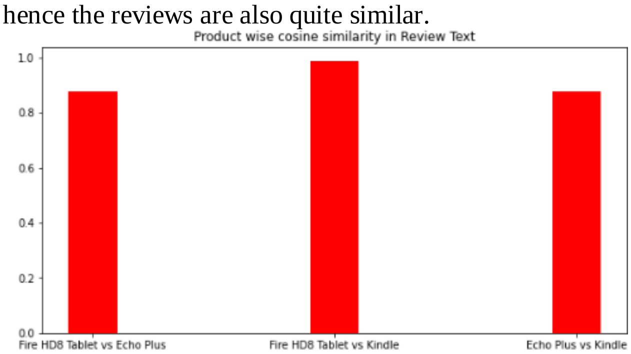

Fig 6. Similarity of reviews

C. SENTIMENT ANALYSIS

The review ratings range from 1 to 5, to find the intensity of the sentiments I have classified the reviews as positive if ratings are 4 and 5 and negative if ratings range from 1 to 3. NRC Emotion Intensity Lexicon has been used to determine the emotions (anger, anticipation, disgust, fear, joy, sadness, surprise, and trust) for positive and the negative reviews. Fig 7. shows the intensity of the basic emotions of the positive and negative reviews.

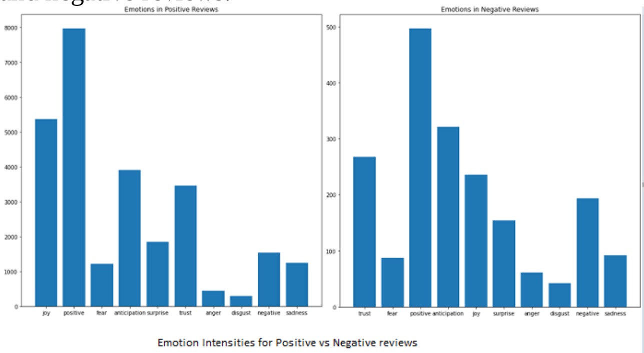

Fig 7. Emotion Intensities

Fig 8. Shows the distribution of the recommendations by all users.

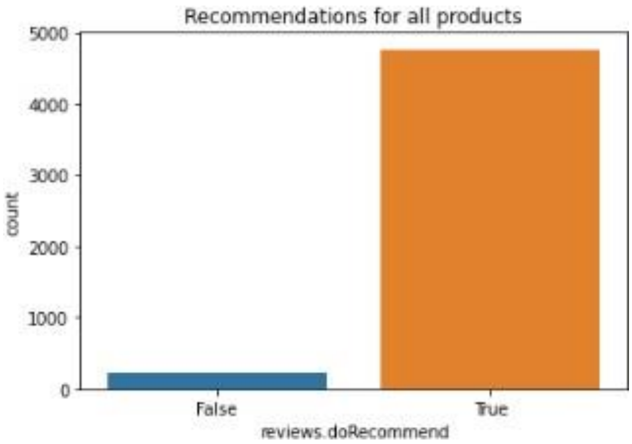

Fig 8. Recommendations

The aim of the next part is to predict whether the consumer recommended the product by analyzing the sentiments of the customers' review texts. At first the features are extracted using text analysis methods and then the features are used to build supervised model to predict the outcome. The two algorithms used for building the model are Random Forest and Naive Bayes.

To extract features two separate methods have been used and the results have been compared.

I. Method 1

I have used VADER Sentiment Intensity Analyzer on each of the customer review texts to determine the polarity_scores in following categories: Positive, Negative, Neutral, Compound. I have also converted the column : "reviews.doRecommend" from categorical to two numerical classes 0 (False: doesn't recommend) and 1 (True: recommends). The polarity scores and the review ratings are used as features for building my model. The features dataset has been partitioned for training set and test set in 7:3 ratio. And the partitioning has been done using k fold stratified partitioning method so that the training and test datasets contain the False and True recommendations categories in the same proportion as the original dataset. At first the training dataset is used to train the models and the trained model is used to predict whether the customer recommended. This prediction is then compared with the actual prediction decision. It is found that Random Forest model works better (accuracy score=0.98 or 98%) than the Naïve Bayes method (accuracy score=0.94 or 94%).

Fig 9. shows the heatmap of the confusion matrix of both the models.

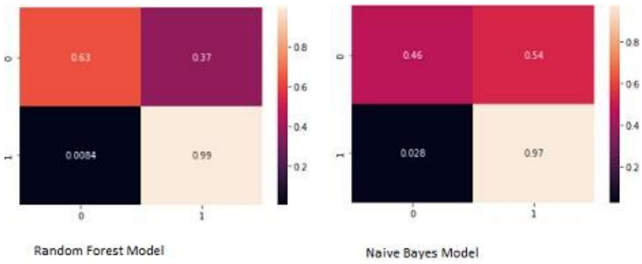

Fig 9. Confusion Matrix plot of the two models

II. Method 2

Data Pre-Processing like sentence tokenize, word tokenize, word lemmatize and stopwords removal has been done on the individual review texts and TF-IDF method is used for extracting the features (vectorized values) from the pre-processed reviews.

The TF-IDF vectorized values are used as the dataset for the recommender model. Again, the dataset is split into training and test dataset using k fold stratified partitioning method. The same Random Tree and Naïve Bayes algorithms are used for the two models. In this case, Naïve Bayes method performs better than the Random Tree model in predicting the False class. The overall accuracy score for the models are: Random Forest (0.95 or 95%), Naïve Bayes (0.93 or 93%). The confusion Matrices is illustrated in Fig 10.

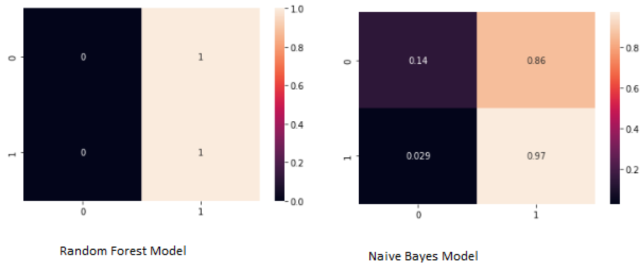

Fig 10. Confusion Matrix plot of the two models

IV. CONCLUSION

To conclude, I have built two type of models using Random Tree and Naive Bayes Classifier and tried to predict the class of recommendation by using features extracted from the review texts by two methods, first, using polarity of sentiments and second using the TF-IDF vectorized values. It is seen that the accuracy is better for the first method i.e., using polarity scores of review text to predict recommendation class.

V. REFERENCES

I have referenced below textbooks, papers for my mini project:

[1] Applied Text Analysis with Python - by Benjamin Bengfort, Rebecca Bilbro, Tony Ojeda

[2] Nandal, N., Tanwar, R. and Pruthi, J., 2020. Machine learning-based aspect level sentiment analysis for Amazon products. Spatial Information Research, 28(5), pp.601-607.

[3] Sharma, S. K., Chakraborti, S. and Jha, T. 2019. Analysis of Book Sales Prediction at Amazon Marketplace in India: A Machine Learning Approach. Information Systems and e-Business Management, 17(2–4), pp. 261–284..

[4] M. S. Elli and Y.-F. Wang. Amazon reviews, business analytics with sentiment analysis.

[5] Bing Liu. Sentiment Analysis and Opinion Mining, Morgan & Claypool Publishers, May 2012

[6] Data resource: https://www.kaggle.com/datasets/datafiniti/consumer-reviews-of-amazon-products