RESEARCH ARTICLE OPEN ACCESS

Sentiment analysis on mobile review

Aditi Linge [1], Bhavya Malviya [2], Digvijay Raut [3], Payal Ekre [4]
Department of Information Technology, Government College of Engineering,
Amravati – Maharashtra - India

ABSTRACT
In this everchanging world, where all the things are slowly converting into different machines, we can say that the need for humans is getting less. In this century, machines are now becoming capable of reading and understanding human emotions. In this project, we have demonstrated how a machine is capable of understanding whether a review on mobile is positive or negative using NLP. Sentiment analysis of product reviews, an application problem, has recently become very popular in text mining and computational linguistics research. Here, we want to study the mobile reviews given by the customers. We use the Logistic Regression algorithm.

Keywords: - Machine Learning, Mobile review, Logistic Regression, N-gram, Support Vector Machine, Random Forest Classifier, Count vectorizer, TF-IDF.

I. INTRODUCTION

Sentiment analysis is an automated process capable of understanding the feelings or opinions that underlie a text. It is one of the most interesting subfields of NLP, a branch of Artificial Intelligence (AI) that focuses on how machines process human language. Sentiment analysis studies the subjective information in an expression, that is, the opinions, appraisals, emotions, or attitudes towards a topic, person, or entity. Expressions can be classified as positive, negative, or neutral.

The aim of this project is to investigate if the sentimental analysis is feasible for the classification of mobile reviews from different buyers and hence strategize the further business strategy. Therefore, we will compare the performance of different classification algorithms on the binary classification (positive vs. negative) of mobile reviews from different customers. This greatly impacts the sales data and customer engagement on a particular mobile.

The objective of this paper is to classify the positive and negative reviews of the customers over different products and build a supervised learning model to polarize large amounts of reviews. Our dataset consists of customers' reviews and ratings which we got from Kaggle. We extracted the features of our dataset and built several supervised models based on them. We used Logistic Regression, Random Forest, and the Support Vector Clustering(svc).

II. RELATED WORK

So far, there are a lot of research papers related to product reviews, sentiment analysis, or opinion mining. For example, Xing Fang and Justin Zhan from North Carolina A&T State University, Greensboro, NC, USA they used algorithms such as Supporting vector machine, Naïve Bayesian, Random Forest which are already existing and supervised algorithm to found sentiment polarity categorization and they used a dataset from Amazon.com. The SVM model takes the most significant enhancement from 0.61 to 0.94 as its training data increased from 180 to 1.8 million. The next model outperforms the Naïve Bayesian model and becomes the 2nd best classifier and the Random Forest model again performs the best for datasets on all scopes.

In paper[2], Wanliang Tan Xinyu Wang Xinyu Xu study the correlation between the Amazon product reviews and the rating of the products given by the customers. For this, they have used both traditional machine learning algorithms including Naive Bayes analysis, Support Vector Machines, K Nearest Neighbor method, and deep neural networks such as Recurrent Neural Network(RNN). They have divided the entire dataset of 34,627 reviews into a training set of size 21000 (60%), a validation set of size 6814 (20%), and a test set of size 6813 (20%). They found that all models perform better with traditional input features than with glove input features. Specifically, LSTM generates the most accurate predictions over all other models.

In 2002, Pang, Lee, and Vaithyanathan tried learning a supervised model for the classification of movies reviews into positive and negative classes with the help of SVM and

Naive Bayes and maximum entropy classification. In this, three algorithm trials got quite good results. In this study, they have tried various features

and it turned out that the machine learning algorithms performed better when a bag of words was used as features in those classifiers.

According to many research works, Naive Bayes, SVM are the two most used approaches in sentiment classification problems.

III. METHODOLOGY

A. Machine Learning Library

Jupyter Notebook [9] is used to implement the machine learning algorithms in this project with the help of other scientific computing libraries - scikit-learn [8], numpy [10], matplotlib [5].

B. Dataset

The dataset used in this project is commonly known as Amazon review Dataset [10], you can also use any review dataset. This data set was created by PromptCloud.

C. Data Pre-processing

We have used a dataset from Kaggle and it extracted 400 thousand reviews of unlocked mobile phones sold on

Amazon.com to find out insights with respect to reviews, ratings, price, and their relationships.

Content

Given below are the fields:

1. Product Title
2. Brand
3. Price
4. Rating
5. Review text

It has 41384 unique values. We found that there are some data points that have been missing when we went through the data. After eliminating those examples, we have 33408 data points in total.

Product Name consists of the name of the mobiles. e.g. Sprint EPIC 4G Galaxy SPH-D7, Brand Name consists of Name of the parent company. e.g. Samsung, price consists of the Price of the mobile phones, (Max: 2598, Min: 1.73, Mean: 226.86), rating consist of Rating of the product ranging between 1-5, Reviews description of the user experience, Review Votes consist of the number of people voted the review (Min: 0, Max: 645, Mean: 1.50).

D. *Data Resampling*:

We have a large dataset and we are trying to make a polarized sentiment analyzer. Due to this, we classified reviews having ratings 1 and 2 as 'negative' reviews or '0' and remaining reviews as 'positive' reviews or '1'. We also found some repeated data and null value data in our dataset, so we dropped these values so that model does not 'underfit' or 'overfit'.

E. *Features:*

We have tried - types of features in the project. The first one is the simplest method. In which we tokenize a collection of text and build a vocabulary of own words. In this method, we made rows for every review and columns for every word in the review. We make a matrix of these rows and columns and count the occurrence of each word. This matrix helps us to better understand the sentiment. Since there are some common words like I, is, the, etc which does not help to understand the sentiment such words are omitted from the matrix.

N-grams are simply all combinations of adjacent words or letters of length n that you can find in your source text. For example, given the word fox, all 2-grams (or "bigrams") are fo and ox. You may also count the word boundary – that would expand the list of 2-grams to #f, fo, ox, and x#, where # denotes a word boundary. The basic point of n-grams is that they capture the language structure from the statistical point of view, like what letter or word is likely to follow the given one. The longer the n-gram (the higher the n), the more context you have to work with. Optimum length really depends on the application – if your n-grams are too short, you may fail to capture important differences. On the other hand, if they are too long, you may fail to capture the "general knowledge" and only stick to particular cases.

F. *Methods*

1. ***Support Vector clustering(svc) :***

   SVC is a part of the Support Vector Machine (SVM)algorithm. SVM is a supervised (requires labelled data sets) machine learning algorithm that is used for problems related to either classification or regression. In our Support Vector Clustering (SVC) algorithm data points are mapped from data space to a high dimensional feature space using a Gaussian kernel. In feature space, we look for the smallest sphere that encloses the image of the data. This sphere is mapped back to data space, where it forms a set of contours that enclose the data points. These contours are interpreted as cluster boundaries. Points enclosed by each separate contour are associated with the same cluster.

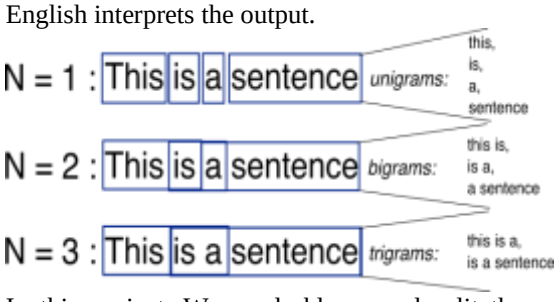
2. ***Random Forest:***

   Random Forest algorithm is a supervised classification algorithm. We can see it from its name, which is to create a forest in some way and make it random. There is a direct relationship between the number of trees in the forest and the results it can get: the larger the number of trees, the more accurate the result. But one thing to note is that creating the forest is not the same as constructing the decision with the information gain or gain index approach.

3. ***Logistic Regression:***

   Logistic regression is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary). Like every regression analysis, logistic regression is a predictive analysis. Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval, or ratio-level independent variables.

   Sometimes logistic regressions are difficult to interpret; the Intellectus Statistics tool easily allows you to conduct the analysis, then in plain English interprets the output.



In this project, We used sklearn and split the dataset into 75 percent training data and 25 percent testing data. We used a random state as '0'. The Random state controls the shuffling applied to the data before applying the split function. This helps the dataset to load in every system in the same manner. Then we transformed this data into a vector and applied our algorithms to it.

IV. RESULTS

The entire dataset of 41384 reviews was pre-processed and got 33408 reviews which were then divided into a training set of size 25056 (75%), a test set of size 8352 (25%).

After converting this training data into a vector we implemented the Random Forest algorithm, Support vector clustering algorithm, and logistic regression algorithm. In the logistic regression algorithm, we used count vectorizer, TFIDF, and N-gram methods. In SVC, we were getting very low accuracy due to underfitting.

Model	Test Accuracy
Random forest classifier	86.71%
Logistic Regression(tfidf)	88.99%
Logistic Regression(countvectorizer)	89.74%
Support vector clustering	89.75%
Logistic Regression(N-gram)	91.04%

V. CONCLUSION AND FUTURE WORK:

We have observed that Logistic regression has the highest accuracy(91.04%) by using the N-gram method.

Support vector clustering has an accuracy of 89.75% and the lowest accuracy percentage is observed in the Random Forest Classifier.

Considering the current issues and challenges of this project, it has a wide scope to extend and improve. In today's world, it is seen that the online world has its own language in the form of various short forms and slang. With a proper dataset, we can also try to train our model to fit in the evolving world. This needs more time and an appropriate date because the trends keep changing. There is scope for improvement even in the accuracy if possible using a variety of algorithms and a larger dataset.

VI. ACKNOWLEDGMENT:

Prof. B. V. Wakode of Information Technology (I.T.), Government College of Engineering, Amravati, has been a source of support and guidance to the authors throughout this research.

REFERENCES

[1] O PANG, LILLIAN LEE, AND SHIVAKUMAR VAITHYANATHAN. THUMBS UP?: SENTIMENT CLASSIFICATION USING MACHINE LEARNING TECHNIQUES. IN PROCEEDINGS OF THE ACL-02 CONFERENCE ON EMPIRICAL METHODS IN NATURAL LANGUAGE PROCESSING VOLUME 10, PAGES 79–86. ASSOCIATION FOR COMPUTATIONAL LINGUISTICS, 2002.

[2]HTTPS://JOURNALOFBIGDATA.SPRINGEROPEN.COM/TRACK/PDF/10.1186/S40537-015-0015-2.PDF

[3]HTTPS://WWW.RESEARCHGATE.NET/PUBLICATION/332622380_SENTIMENT_ANALYSIS_FOR_AMAZONCOM_REVIEWS

[4]HTTPS://WWW.KAGGLE.COM/BENROSHAN/SENTIMENT-ANALYSIS-AMAZON-REVIEWS

[5]HTTPS://STACKOVERFLOW.COM/QUESTIONS/18193253/WHAT-EXACTLY-IS-AN-N-GRAM

[6]HTTPS://MEDIUM.COM/@SYNCED/HOW-RANDOM-FOREST-ALGORITHM-WORKS-IN-MACHINE-LEARNING-3C0FE15B6674

[7]HTTPS://SCIKIT-LEARN.ORG/STABLE/MODULES/GENERATED/SKLEARN.MODEL_SELECTION.TRAIN_TEST_SPLIT.HTML

[8] HTTPS://SCIKIT-LEARN.ORG/STABLE/

[9] HTTPS://JUPYTER.ORG/

[10] HTTPS://NUMPY.ORG/

[11] HTTPS://MATPLOTLIB.ORG/