COMPARATIVE STUDY OF MACHINE LEARNING ALGORITHMS ON SENTIMENT ANALYSIS OF PRODUCT REVIEWS

Ujwala Baruah, Ratnadeep Das, Amitabha Deb and Shah Alam Mazumder
National Institute of Technology Silchar, Assam-788010, India
ujwala@gmail.com

ABSTRACT: For a very long time, making a machine understand the natural language and process it to perform a creative task has been a problem of high priority to human. Classifying whether a product being sold is worth buying makes it easier for the buyer to make his decision. This also helps the seller to analyse his sales. In this paper, we have done a comparative analysis of dierent machine learning algorithms and studied how well each algorithm works in classifying human sentiments from natural language in text based product reviews.

Keywords: Sentiment analysis Amazon customer reviews Binary classication

I. INTRODUCTION

Sentiment analysis refers to a process which involves extraction and understanding of the sentiments present in a text. The huge amount of information available on the many social media platforms such as Facebook, Instagram, Twitter and online shopping platforms like Amazon,has felicitated a user with an opportunity of expressing his/ her views on any topic, location or a product[6]. Facts and opinions form the main components of any information[3]. The objective statements about any commodity or an event happening around the world are facts. Opinions represent the subjective statements that corresponds to the human sentiments and his perceptions for any event or entity. Sentiment Analysis has become so popular because user opinion or facts, prevalent on the web, are mostly in the form of textual information. Early researchers have mainly emphasized on the categorization of the factual information. However, the search engines,today, can allow a user to search based on keywords. The result of the search can be both facts related to the searched topic or public opinion of the same. People can review products at merchant sites and show interest, praise or criticise them in discussion groups, Internet forums and blogs [1]. These can be termed as the opinions. More examples may include reviews on the online shopping sites like Amazon which can also contain score based ratings or star ratings, based on the quality of the product or the companys service. The reviews can be of a single word, single line, a long paragraph or very long, consisting of several paragraphs. Depending on the words or length of the document, the complexity varies. There can be grammatical, cultural, geographical variants in the document. And it becomes a very dicult task to process the information hidden in the texts. The solution is to rely on the modern techniques of machine leaning and natural language processing [2]. After going through the processing of the natural language, classication of the text has to be done. Classication can mainly be divided into two basic approaches: supervised and unsupervised. In supervised classication,training of the classier is done on labelled data which are similar to the test data. On the other hand, unsupervised classication comprises of assumption of labels based on the distance between the data points [7].

Section II deals with the source and the content of the whole dataset being used. Section III presents dierent techniques starting from acquiring the data, dierent pre-processing steps being applied on the data, the features selection and the method of classication done. It also discusses the experiments and the work done in this paper to improve the accuracy. The results of the experiment are discussed in section IV. Finally, section V concludes with the work done and discusses the future work and improvement possible in the model.

II. SOURCES

The dataset consists of a few million Amazon customer reviews (input texts) and corresponding Binary ratings (output labels). The dataset is publicly available and is taken from Xiang Zhangs Google drive directory which is in.csv format [4]. The dataset contains two les: train.csv and test.csv. The train.csv le contains 3.6M text reviews and ratings for training. The test.csv le consists of 400k text reviews and ratings for testing. Both the les contain three columns of data. The rst column contains the ratings 1 or 2, one being negative rating and two being a positive rating. The second column contains a shorter opinion of the product and the third column contains a longer review of the product.

In this paper, the third column, i.e. the longer review for sentiment analysis was used. We tried to use this column to train the machine to be able to tell how satised a customer is; by analysing his/her posted review.

III. METHODOLOGY

The process of sentiment analysis unfolds in a series of steps. These steps help in the extraction of the nal meaning behind the text that is being analysed. We followed the following steps to come up with our results.

3.1 Data Acquisition

A detailed description about the data was presented in the section II of this paper. The acquired dataset is loaded into our program. The third column contains the reviews which we have used to train our model and then analyse. first column contains the corresponding binary ratings i.e. positive or negative in numbers 1 and 2, 1 being a negative label and 2 being a positive label.

3.2 Data Pre-processing

Carefully Selected pre-processing techniques hae been used to pre-process our data.

- Removing contracted words: Experiments show that the contractions have a bad eect on the accuracy of the model. A list of the contractions is taken and replaced by their uncompressed forms [5]. Contractions like Im, youve, theyre etc. are replaced with their uncompressed counterpart like I am, you have, they are etc.
- Removing stop words: There are some words in the text which do not provide any appreciably dierent meaning to it. These words are removed from text to make it more concise.
- POS-tagging: Keeping only those parts of speech that actually contain sentiments. This was done to help us bring the meaning of the text for the required motive only. Following POS-tags have been used: f\JJR", \JJS", \JJ", \RB", \VB", \VBD", \VBG", \VBN", \VBP", \RBR", \RBS", \VBZ"g. Only these parts of speech are kept in the text for analysis.
- Lemmatization: The text is lemmatized to group the in ected forms of a word together such that analysis can be done treating them as a single item, which can be identied by the words lemma or its dictionary form. We have used the WordNet Lemmatizer() function from the NLTK module.

3.3 Feature Selection

Bag-of-words model: Bag-of-words is a model where the features are the individual words of a sentence, assuming that the words are conditionally independent. The text is converted into a number of feature vectors where each feature represents the existence of one word. Bag-of-words is basically an unordered collection of words, and these words are selected from the texts through calculated feature selection methods.

3.4 Sentiment Classication

The pre-processed text is input to a machine learning algorithm like, SVM, Naive Bayes or even a Neural Network. This trains the feature vectors from the previous step and maps the sentiments in the text to the corresponding binary ratings in the labels.

In the rst phase, we have applied the Support Vector Classier (SVC) which by default uses a Gaussian kernel. This is improved by using a Linear kernel known as Linear SVC. We have also used the Nu SVC where the Nu parameter is used rather than the general parameters. We also tested a Random Forest (RF) Classier to train our classication model. To further improve our classication, we moved on to the Bayesian classiers. We started with the Naive Bayes (NB) Classier. Improvements were done using the Multi-nomial Naive Bayes (MNB) which in our case performs better than the Bernoulli Naive Bayes (BNB) classier. Hence, MNB outperformed all the other classiers used by us.

In the second phase of our work, we have created a model which makes a prediction based on the mode of the votes of individual classiers mentioned above. Mode gives the output as that value which is repeated most of the time in the set.

Algorithm 1 Voted classier
1: Pre-process data.
2: Pass data through dierent machine learning classiers.
3: Acquire decision.
4: The decision is passed through hard voting.(Hard voting is dened as taking the mode of the decision of the individual classiers.)
5: Calculate condence score for each prediction to determine how condently a classier predicts the class according to the following formula:- y $\longleftarrow$ Mo (C1(x);C2(x);:::Cm(x)), where y is the output of the majority voting classier and Ck(x) is classication done by the kth classier.

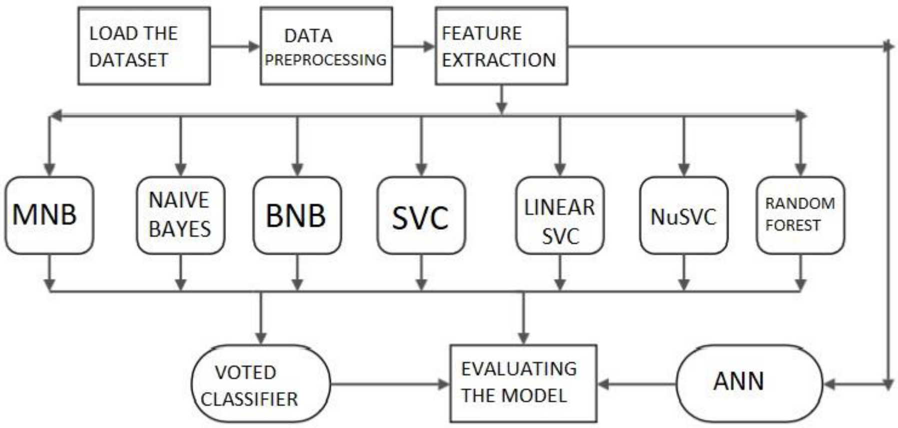

Figure 7.1 Flowchart for sentiment analysis process

In the nal phase, we have trained a neural network. We have used the keras library using tensor flow as back-end for this purpose. The feature extraction is done using the bag of word approach. keras library has an inbuilt class Tokenizer whose task is the conversion of text into a numeric vector. After the conversion, we must make sure that our labels are compatible to fed into the neural network model. Each label is assigned a probability in order to make the prediction. The labels also need be converted to a one hot vector form. This is done using the Scikit-learn. The model is dened using the Keras Sequential model API. We have used just one hidden layer and along with the input and output layers. We have used drop out layer to prevent over fitting.

The condence is calculated as the number classiers voted for a particular prediction. For example, if 4 out of 5 classiers voted for a particular value, the condence would be higher (80%) than compared with if 3 out of 5 would have voted (60%) even though in both the cases, we get the same result.

In this paper, we have made a comparative study of the accuracy of dierent machine learning algorithms while performing the sentiment classication into positive and negative classes. The implementation has been done using a Python program which handles the data eciently. We have used the following python packages along with their functions in brackets:

- Numpy (Numerical library)
- Pandas (Data handling)
- Scikit-Learn (Pre-dened machine learning algorithms)
- NLTK (Natural language processing)
- Keras (Neural network implementing framework)

IV. RESULT AND DISCUSSION

In the rst section we have independently trained the following classiers on the training set and their accuracy is calculated as follows:

Formally, accuracy can be dened as

Accuracy = Number of correct predictions / Total number of Predictions If we consider binary classication, Accuracy is dened as:

Accuracy = (TP + TN)=TP + TN + FP + FN Where TP = True Positives,

TN = True Negatives, FP = False Positives, and FN = False Negatives.

Using the above formula the accuracy on the testing set for the various classiers are:

- MultinomialNB -82.1%
- Naive Bayes-81.3%
- BernoulliNB-80.44%
- SVC-72%
- LinearSVC-80.2%
- NuSVC-80.7%
- RandomForest- 79.1%

The accuracy of the voted classier is 82.3%. Thus, we see that the voted classier has a better accuracy than the individual classiers, also improvement can be done if we assign weights to the various classiers.

Using the neural network model, we have achieved a test accuracy of 87% and a train accuracy of 92%. The reason for the high accuracy is the huge size of the training dataset which allows the neural network to capture the hidden patterns more eciently.

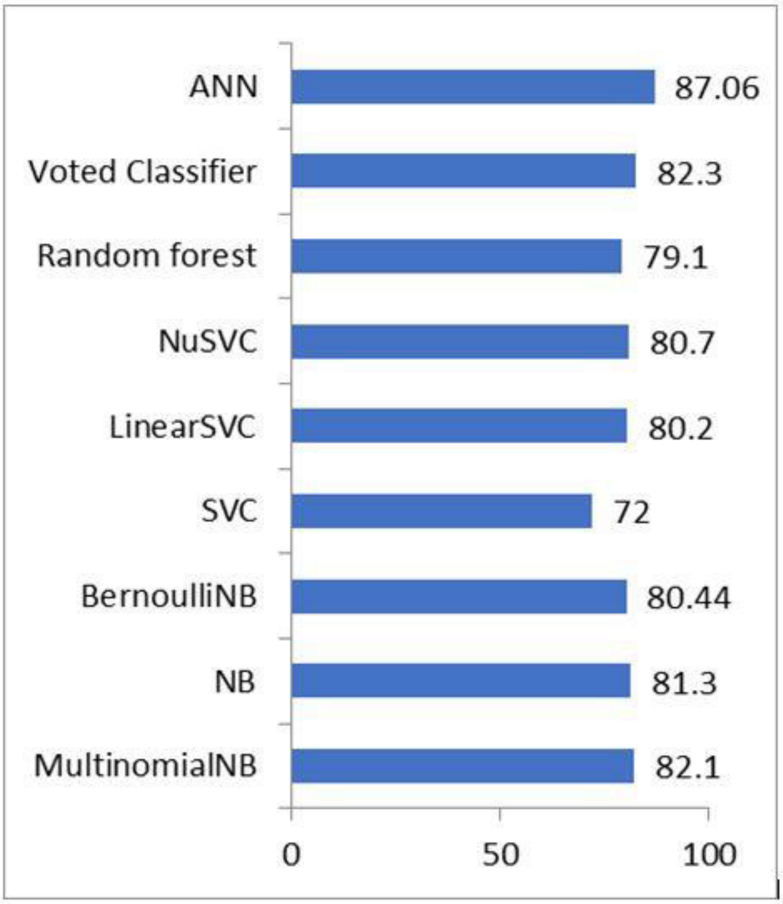

Figure 7.2 Accuracies of the various Classiers.

V. CONCLUSION

In this paper, we have compared some of the tried and tested machine learning algorithms. We have seen that the articial neural network works better than all the other algorithms combined, like the voted classier. We have used our chosen techniques to process the natural language in the reviews. While choosing some other techniques might result in dierent results. Our model cannot correctly classify if there is a tone of sarcasm in the reviews, but this could be analyzed in the future and made some strides in sensing the sarcastic tones in texts. We have used binary classication in our model- positive or negative sentiment. This can be extended to the ve class classication where each class represents each of the star based or 1 to 5 scored rating levels.

REFERENCES

1. Bhatt, A., Patel, A., Chheda, H., Gawande, K.: Amazon review classication and sentiment analysis. International Journal of Computer Science and Information Technologies 6(6), 5107{5110 (2015)
2. Carenini, G., Ng, R. T., Zwart, E.: Extracting knowledge from evaluative text. In: Proceedings of the 3rd International Conference on Knowledge Capture. pp. 11{18. K-CAP '05, ACM, New York, NY, USA (2005). https://doi.org/10.1145/1088622.1088626
3. Dave, K., Lawrence, S., Pennock, D. M.: Mining the peanut gallery: Opinion extraction and semantic classication of product reviews. In: Proceedings of the 12th International Conference on World Wide Web. pp. 519{528. WWW '03, ACM, New York, NY, USA (2003). https://doi.org/10.1145/775152.775226
4. drive.google.com: Textclassicationdatasets google drive. Retrieved on 15/03/2019 from https://drive.google.com/drive/folders/0Bz8a_ Dbh9Qhbfll6bVpmNUtUcFdjYmF2SEpm ZUZUcVNiMUw1TWN6RDV3a0JHT3kxLVhVR2M
5. Dubois, Y., Hassan, H.: Expanding english language contractions in python. Accessed on 15/04/2019 from http://stackoverflow.com/questions/19790188/expanding-english-language-contractions-in-python
6. Shivaprasad, T. K., Shetty, J.: Sentiment analysis of product reviews: A review. In: 2017 International Conference on Inventive Communication and Computational Technologies (ICICCT). pp. 298{301 (2017). https://doi.org/10.1109/ICICCT.2017.7975207
7. Yessenov, K., Misailovic, S.: Sentiment analysis of movie review comments. Methodology 17, 1{7 (2009)