Implementation of Sentiment Analysis Using Optimized Classifier

DEVANSH ARORA[1], SHAHRUKH ANSARI[2], ABHISHEK KUMAR[3], NEETU GARG[4]
[1, 2, 3] *Student, Department of Computer Science & Engineering, Maharaja Agrasen Institute of Technology, Affiliated to G.G.S.I.P University*
[4] *Assistant Professor, CSE*

Abstract- Emotion recognition (ERC) in speech is a challenging task that has recently gained popularity for its potential applications. Words and phrases reflect people's views on products, services, governments and events on social media. The main aim of sentiment analysis is to categorize the emotion behind a text as positive or negative. There is significant upsurge in adopting sentiment analysis techniques in various business organizations and government projects as it helps analyze the audience's emotion towards a particular aspect. Using the knowledge obtained through sentiment analysis can prove to be beneficial for organizations in terms of sales and popularity. Naive Bayesian classifiers greatly simplify mastery by assuming that skills are independent of a particular class. We study the effect of distributional entropy on type error and show that low-entropy feature distributions provide accurate overall performance of naive Bayes. We also show that Naive Bayes works correctly for positive near-functional functional dependencies .So, it achieves overall good performance and otherwise perfectly fair functionality (as expected) and functionally structured functionality (which is unexpected). Another unexpected result is that the accuracy of Naive Bayes does not always immediately correlate with the level of functional dependence. We plan to improve this.

I. INTRODUCTION

The rapidly growing popularity of online social communication and electronic media-based companies continues to push researchers to study sentiment analysis of a given text. Today, businesses like to use text on social media to gauge public opinion about their customers and products. possessed of this rating is used to make decisions or improve the quality of our services and products. Applications of sentiment analysis cover are as such as event organizing, electoral campaigning, health surveillance, information processing and consumer commodities. The ubiquity of the Commerce on the Internet by organization around the world has taught us that our business operations and social occupation systems are shaped by capricious web texts. Computing performance is being accelerated by the advent of machine learning techniques. This work focuses on improving existing text classifiers used for sentiment analysis such as: Naive Bayes Algorithm. The "Machine Learning Techniques for Sentiment Analysis" chapter of this paper provides the intuition behind the sentiment classification task by leveraging its modeling of the classifier. The driving-force of this conundrum is in the Chapter Problem Motivation. How we solve this conundrum is in the Chapter Solution. Results of experimentation and a consideration of the effectiveness of the classifier are summarized in "Conclusion".

II. LEVEL OF SENTIMENT

With therapid increase in popularity of social media and easy procurement of high availability digital data, the interests of many young researchers in the field of mathematics and computer science have been piqued. This is because in the last decade of the 20th century, there was scarcity of digital words, sentences and paragraphs that reflected opinions of people. Now, researchers are itching to define levels of granularity in texts and paragraphs so that the sentiment of a person who authored them can be deciphered accurately. Written text can be divided into three portions namely Document Portions, Sentence Potions and Word Portions. At the fourth portion, deep convoluted neural networks are used to define granularity.

1. Machine learning techniques for sentiment analysis

The impact that social media and other networking sites have on people currently is enormous. People have become liberal in expressing their opinions on the internet freely and instantly. With one click of a button, their thoughts and views are out there in the world for everyone to see. This sheer availability of data on the internet fuels researchers' desire to feed it into the realm of sentiment analysis so that they can mine interesting knowledge.

Once sentiment analysis is employed to learn how a person feels about a particular product or a service, business organizations can then find ways to improve their product or service to achieve better sales. Therefore, this immediate and automatic determination of sentiments from a large data set of reviews collected from customers is the prime goal of business organizations to pursue success.

There have been many techniques powered by machine learning that are implemented to increase the accuracy of sentiment analysis in the field of natural language processing. Our work in this paper attempts to make use of machine learning to create a model for sentiment analysis with a higher accuracy through better optimization and pre-processing techniques.

2. Naïve bayes used for sentiment classification

The emotional dichotomy can be figured out by analyzing whether the person agrees positively or negatively with what he or she has written or said. The Naive Bayes machine learning algorithm helps in determining this emotional dichotomy. The way the Naive Bayes Classifier works is that it takes a digital text as its input and then classifies the sentiment of the text into the categories: positive, negative or neutral. The core logic of this algorithm makes use of conditional probabilities to determine the category in which the sentiment of the text would fall into.

The advantage of using this algorithm is that it doesn't require a huge dataset for training. It works just as well with a smaller dataset containing either/both continuous or discrete data. The general steps that are followed in this simple Naive Bayes algorithm are as follows: Firstly, raw data is scraped from the web and is then pre processed so that the final data that we end up making our model upon is clean. We make sure that the text doesn't contain any irrelevant data like HTML tags, numbers, foreign words and special characters. The next step is to create a dictionary for the training set where the key is a word and the value is the category of the sentiment that the word represents. The initial marking of sentences into positive, negative and neutral marks is done manually by thorough inspection and analyzation of the text's sentiment so that our final dataset is accurately marked. Let there be a word 'y' from a test sentence (unknown phrase). Let the 'n' words from the document be x1, x2, x3, ... xn. Hence, the conditional probability that a given data point 'y' is in the n-word category of the training set is given by:

$$P(y/x_1, x_2, \ldots\ldots x_n) = P(y) \times \prod_{i=1}^{n} \frac{P(x_i/y)}{P(x_1, x_2, \ldots\ldots x_n)}$$

Fig -1: Probability equation

III. PROBLEM MOTIVATION

Sentiment analysis is critical for every marketing department to gain brand insights. Used for social listening, brand reputation monitoring, reviews study, market survey, customer experience study, and other functions. Natural language processing (NLP) and named entity recognition (NER) are both used in sentiment analysis to identify and distinguish in your data. The Aspect Based Analysis of Sentiment way enables companies to extract highly detailed insights from any data source to reveal insights such as patient notes, EMRs, customer call logs, and more .increase. However, there are some problems organizations face while performing analysis of sentiment such as Tone, Polarity, Sarcasm, Emojies, Idioms, Negations, Comparative sentences, Employee bias. And some big problems such as what if a word in a review was not present in the training dataset at all.

IV. SOLUTION

Let there be a sentence "I like this product" which does not already exist in our training data. Now let P(I like this product| Positive) be the probability of occurrence of the event, the event being the text" I like this product", given that this sentence has a positive sentiment.

If we follow the Naïve Bayes approach to calculate the conditional probability, the answer returned will always be 0 since the sentence "I like this product" is either not present or there does not exist an exact match for it in our current training dataset. This method is ineffectual since it does not produce any estimation for the sentences that do not appear in the training data set a sit always dumbly returns the value 0. That is why this approach is called "naive". The simplicity of the Naive Bayes algorithm arises from the fact that it assumes that every individual word of a sentence is independent of other words. This algorithm ignores the context in which the words are used in. Only those sentences with words matching character by character are considered equal. For instance, the pair of sentences " This product is great." and" This product is great." Are same in the eyes of Naïve Bayes classifier and the pair of sentences" This product is great." And " I think this product is great." Are different. Another approach to deal with this issue would be to count the frequency of individual words instead of only counting the number of occurrences of a particular sentence. This would modify the initial equation. Thus, the following formula shows how to calculate P (I like this product| positive):

$$P(I\ like\ this\ product \mid Positive) = P(I \mid Positive) * P(like \mid Positive) * P(this \mid Positive) * P(product \mid Positive)$$

Fig -2: Modified equation that takes frequency of words into consideration

According to this approach, P (like|Positive) is calculated by dividing the number of occurrences of the word" like" in the positive text by the total number of words in the positive text. This leads to another problem which may arise when the word in the given sentence is encountered for the first time, i.e., the training dataset does not have a record of a word. This means that the frequency of the word would be zero. Hence the probability P (like | Positive) turns out to be zero. Now, if we multiply this factor with all the other probabilities, the eventual result will also become zero. This method does not provide us with any applicable information. To deal with this, probabilities of individual words are used. By adding one in the numerator so that the probability is not zero. All the number of possible words is added to the divisor so that probability cannot ever exceed 100% For example, if your training data has 15 unique words (regardless of label), add 15 to the denominator.

$$P(like \mid Positive) = \frac{3+1}{11+15}$$

Fig-3: Optimized equation of Naive Bayes

CONCLUSION

Naive Bayes is an effective algorithm and can provide us with great accuracy but relies on simple assumption. This technique leads upto 85% accuracy, which is an great jump from accuracy without using any optimization or pre-processing.

REFERENCES

[1] Paolo Romeo: "Twitter Sentiment Analysis: a Comparison of Available Techniques and Services", Master The- sis, Technical University of Madrid, 2020.

[2] Nor Saradatul Akmar Zulkifli and Allen Wei Kiat Lee: "Sentiment Analysis in Social Media Based on English Language Multilingual Processing Using Three Different Analysis Techniques", International Conference on Soft Computing in Data Science, SCDS, Springer, 2019.

[3] Chandni, N. Chandra, S. Gupta and R. Pahade, "Sentiment Analysis and its Challenges" International Journal of Engineering Research Technology (IJERT), pp. 968-970, 2015.

[4] M. Saraee and A. Bagheri, "Feature Selection Methods in Persian Sentiment Analysis," in International Conference on Application of Natural Language to Information Systems, 2013