CSE/LING567 Project
1-FOUR-ALL Sentence Classification

Anirudh Manjunath UB#:5028 9832
Vivek Adithya Srinivasa Raghavan UB#: 5029 0568
Department of Computer Science and Engineering
State University of New York at Buffalo
am354@buffalo.edu
vivekadi@buffalo.edu

Fall 2018

Abstract

Sentence Classification, also known as Text Classification, is one of the most interesting tasks in the domain of Natural Language Processing. Classification is a core problem in many applications such as Sentiment Analysis, Spam Detection, Abuse Detection etc. In today's world, business decisions are majorly driven by the ability to infer underlying class of a textual document automatically. This saves vast amount of human effort, time and resources.
In this project, 4 different types of Sentence Classifications tasks, namely 1. Abusive content detection, 2. Fake News detection, 3. Sarcasm Detection and 4. Sentiment Detection are performed for different datasets and their results are discussed. All the tasks are primarily performed using Multinomial Naive Bayes classification method. Procedures such as TF-IDF(term frequency - inverse document frequency) and Count Vectorization are used to generate the features for model training and testing.

1 Introduction

The 4 different Sentence Classification tasks performed in this project and the motivation behind the same are as follows:

1.1 Abusive content detection : With the advent of internet and wide spread penetration of Social Media Forums across the world, there is an immediate necessity to monitor harmful content such as abusive language which include usage of swear words, profanity, hate-speech etc. By employing proactive analysis and detection of abusive content, organizations can monitor and control the quality of the language used in any particular communication channel.

1.2 Fake news detection : Millions of people are dependent on the information they get from the print media, Television news channels, social media and other websites to stay updated every single day. More often than not, even most of these "original" news providers often source their content from various other intermediate organizations which may or may not be an expert or trusted authority. This gives rise to the question of whether one such piece of

news item is real or fake.
Due to the severe implications of fake news that can cause potential damage in terms of reputation, legal complications and financial loss, the need of the hour for such organizations is to be able to detect the credibility of news articles. This can be performed efficiently by identifying and comparing the underlying patterns between ”True” and ”Fake” news articles, by the virtue of their sentence structure, choice of words and and usage of a particular set of repetitive words. This can be done by detecting the contradiction between the objective polarity which is usually negative and the sarcastic characteristics conveyed by the author which is usually positive.

1.3 Sarcasm Detection: Sarcasm is a popular form of expression that is often used to make an ironic remark or a bitter gibe when the actual content reflects a contrasting opposite indeed. Sarcasm Detection finds an important application in the areas of affective computing and sentiment analysis. This because of the very reason that such expressions can entirely flip the polarity of a sentence, thus leading to wrong or contrasting interpretations.

1.4 Sentiment Analysis and Detection: In today’s world, it is very important for businesses and other organizations to be able to understand and interpret what their customers/patrons are feeling/saying about them. One particular application of Sentiment Analysis is in the domain of e-commerce, where the plethora of customer feedback in the form of ratings, reviews and testimonials can be harnessed to drive business decisions and provide immediate attention to important tasks. The use-case discussed in this project() is that of customer reviews gathered from a website called Yelp, which offers a platform for customers and patrons to voice out their opinions regarding restaurants, hotels, and other establishments that offer similar services.
By understanding the tone and type of the reviews that customers the post regarding a particular company/product/service, businesses can take proactive action to either improve and incorporate changes. The task here boils down to categorizing the customer reviews as either being a ”positive” feedback or a ”negative” feedback.

2 Previous Work

In (Davidson et al., 2017)[Thomas Davidson, Dana Warmsley, ICWSM 2017], they have used TFIDF as the feature generator. Experimentation was done with different models such as decision trees, logistic regression, linear SVMs. They obtained predictions with about 91% accuracy. An important thing to note is that, in the paper, they have classified Hate speech and offensive language as different while we have categorized both of them as Offensive. We may have obtained better results if we hadn’t combined.

Previous works in Fake News Detection had less than 500 samples which is impractical to use as a benchmark. In Fake News Detection, (Wang, 2017) “Liar, Liar Pants on Fire”: A New Benchmark Dataset for Fake News Detection[William Yang Wang, 2017], logistic regression and a bi-directional long short-term memory network model is used. The best results they obtained were around 30%.

For sarcasm detection, (Ptáček et al., 2014)Sarcasm Detection on Czech and English Twitter[Tomas Ptacek, Evan Habernal], the discussion was done with respect to usage of SVM and MaxEntropy classifiers. Two Baseline methods along with POS Tagging and other pipelining features were experimented. Sufficiently high accuracies were achieved, but at the cost of lot of computational effort, which nullifies the need for the same.

3 Proposed Method

The methods proposed in this project are a combination of a type of feauture-generation method followed by a type of model-generation method that takes the features generated by the former to train the model and predict unseen datapoints using the model.

3.1 Feature Generation

A. TF-IDF : TF-IDF (Term Frequency - Inverse Document Frequency) is a technique of calculation of weights if a particular word in reference to the sentence in which it appears and other sentences in the same or different document.
Here, TF refers to Term Frequency, which is the ratio of number of times a particular term/word appears in a document and the total number of words in the document. Thus, this value can be used to estimate how popular a certain word is, in a document.
IDF in this technique refers to logarithmically scaled inverse fraction of the documents that contain the word. This inverse fraction is usually obtained by dividing the total number of documents by the number of documents containing the word, and obtaining the logarithm of the resulting quotient.
Hence, TF-IDF value for a particular word in a document is the product of the TF value and the IDF value of the respective word in its document context.

B. Count Vectorizer : Count Vectorizer is a method that is used to convert a collection of text documents into a matrix of token counts. That is, count vectorization involves computation of word count of each unique word in the whole document as a vector. This results in as many vectors as the number of sentences in the document and each vector will consist of as many token counts as the number of the unique tokens in a particular document. Thus, will result in every individual sentence being transformed into a set of attributes, each having a constant number of attributes irrespective of the word count of the particular sentence.

3.2 Model Generation

A. Multinomial Naive Bayes Classifier : Multinomial Naive Bayes Classifier or MNB Classifier belongs to a family of simple "probabilistic classifiers" based on applying Bayes' theorem with strong (naive) independence assumptions between the features. MNB classifier is widely used for tasks like text categorization/classification, particularly for methods that use the word frequencies as the input features. MNB Classifier works in linear time, by computing weights using evaluation of closed-form based on the input features(word frequency). Hence it naturally takes a much shorter time than compared to other algorithms that perform similar tasks. Multinomial Naive Bayes classifier is very suitable for classification task with discrete features such as word frequency.

B. Passive Aggressive Classifier : Passive Aggressive Classifier or PAC belongs to a family of online learning algorithms. As the name suggests, PAC algorithms behaves passively when it encounters a correct classification, that is, when the loss function value is zero and aggressively when it encounters a misclassification, by making use of an aggressive update rule to look for new weights. Since it is a type of online learning algorithm, it can be used to classify massive data streams, with ability to account for new datapoint updates, thus suitable for applications which need models that can react quickly to newly generated data. PAC is known to fit a better model for tasks such as Text Classification, which often involve constant addition of new documents and a need to adapt to growing change in the social trends concerned with usage of slangs, expressions etc.

4 Experiments and Results

Two different feature generators and three different classification models were used and the results of the same are summarized as below:

Count Vectorizer			
Task Type	Multinomial Naive Bayes	Passive Aggressive Classifier	Average task Accuracy
Offensive Word Detection	91.124	94.987	93.055
Fake News Detection	73.403	69.752	71.577
Sarcasm Detection	87.970	83.212	85.591
Sentiment Analysis	92.061	92.758	92.409
Average Method Accuracy	86.140	85.177	

Fig 1. Results of all Classification Tasks using Count Vectorizer as the feature generator

TF-IDF			
Task Type	Multinomial Naive Bayes	Passive Aggressive Classifier	Average task Accuracy
Offensive Word Detection	84.705	94.853	89.779
Fake News Detection	76.402	68.970	72.686
Sarcasm Detection	86.360	83.535	84.948
Sentiment Analysis	92.167	94.000	93.084
Average Method Accuracy	84.908	85.339	

Fig 2. Results of all Classification Tasks using TFIDF Vectorizer as the feature generator

Analysis for Count Vectorizer : It is seen that for Offensive Word Detection task, PAC has a slighlty better accuracy than MNB. For Fake News Detection and Sarcasm Detection, Multinomial Naive Bayes Classifier has a better accuracy. Sentiment Analysis accuracy results for both PAC and MNB are almost similar.

Analysis for TFIDF Vectorizer : It is seen that for Offensive Word Detection task, PAC has a better accuracy than MNB by a huge margin. For Fake News Detection and Sarcasm Detection, again, Multinomial Naive Bayes Classifier has a better accuracy, with results Fake News-

MNB being even better. Sentiment Analysis accuracy results shows that Passive Aggressive Classifier has outperformed MNB by a small margin.

5 Conclusion and Future Perspectives

We have shown that during preprocessing, we remove the stop words and apply TFIDF or count vectorizer. To improve the efficiency, we can use POS tagging and n-grams. To improve the results further, we can use models such as SVMs, CNNs. It should be noted that neural networks do not require feature engineering as the learning of the model is different from the classifiers we have used.

References

Davidson, T., Warmsley, D., Macy, M., and Weber, I. (2017). Automated Hate Speech Detection and the Problem of Offensive Language. (Icwsm):512–515.

Ptáček, T., Habernal, I., and Hong, J. (2014). Sarcasm Detection on Czech and English Twitter. *Proceedings of the 25th International Conference on Computational Linguistics: Technical Papers (COLING 2014), Citeseer.*

Wang, W. Y. (2017). "Liar, Liar Pants on Fire": A New Benchmark Dataset for Fake News Detection. 72(3):430–431.