An Approach to Overcome Flaming Problem of Social Networking Sites Using Various Classifiers

¹Dr. Avinash J. Agrawal, ²A.V. Zadgaonkar

^{1,2} Dept. of CSE, Shri Ramdeobaba College of Engineering and Management, Nagpur, 440013, India
Email: agrawalaj@rknec.edu, zadgaonkarav1@rknec.edu

Received: 20th September 2018, Accepted: 11th October 2018, Published: 31st October 2018

Abstract

Natural language Processing (NLP) is a field of artificial intelligence and computational linguistics concerned with interaction of humans with machines. Social Networking sites (SNS) is a widely used communication tool today. Sometimes it gives rise to the problem of flaming. A flaming incident is triggered by comments and action made by user in SNS that ends up damaging reputation or causing negative impact on the target party. The paper proposes an architecture to overcome this problem. The training dataset used contains a total of 17282 tweets already annotated as positive, negative and neutral. The test dataset that has been used contains a total of 4382 tweets already annotated as positive, negative and neutral. Result of machine learning techniques used in the proposed architecture are reported and discussed in this paper.

Keywords

Natural Language Processing, Flaming, Naïve Bayes Classifier, Maximum Entropy Classifier, Feature Selection

Introduction

Social networking sites have achieved huge popularity in the last decade by making the transmission of information [8][9] across the globe very easy for everyone. *Flaming* [1] is one of the current hazards of the online communication. The flaming is caused due to negative comments made by user that can have a huge negative impact on the target. So, in order to overcome this problem, sentiment analysis techniques can be used. In this paper, an architecture has been proposed that describes various techniques for finding polarity (positive, negative and neutral) of the input document. A training set of 17282 tweets already annotated (positive, negative and neutral) was preprocessed [4] by applying tokenization, case folding to lower case, stop word removal and stemming to construct a sentiment dictionary. This dictionary contains the words and positive, negative and neutral frequencies. The dictionary is context dependent as it depends on the training dataset. This is the problem of text classification [1]. Then this dictionary is used for assigning polarity of the input document. The proposed approach describes various method like Naïve Bayes Classifier[1][2], Maximum Entropy [3] and Feature Selection based on Entropy[1] method, that can use the constructed dictionary for sentiment analysis of input. Sentiment Analysis can help us to know the polarity of a given document. First step is to apply pre-processing to the labeled dataset and construct sentiment polarity dictionary for the target company. Second step is to pre-process input document (comment) and then classify its sentiment using constructed dictionary, Classifiers and Feature Selection based on Entropy method.

Proposed Architecture

Figure 1. Proposed Architecture

Fig. 1 describes the proposed architecture of the system. The training dataset that contains 17282 tweets is first pre-processed to form Sentiment Dictionary, Pre-processing involves tokenization, case folding to lower case, stop-word removal and stemming. The sentiment dictionary contains words and their respective positive, negative and neutral frequencies. The input test data is then inputted and pre-processing is applied on it. The pre-processed input data is a

collection of features (words). The positive, negative and neutral frequencies of the features in test data is obtained from the sentiment dictionary and then using all the four methods the probability of the input belonging to all the classes namely positive, negative and neutral class is calculated. The input data is then considered to be classified in the class which has the maximum probability. The four methods that are used to calculate probabilities are Naive Bayes Classifier, Maximum Entropy Classifier, Naive Bayes with Feature selection based on Entropy Method, Maximum Entropy with Feature Selection based on Entropy Method. The output of all the four classification methods is compared with the original classification in the training dataset which is already annotated with appropriate category it belongs to. In this way the calculation of the accuracy of various classification methods has been done.

Dataset Description

Dataset	Training Dataset	Testing Dataset
Positive Tweets	7735	1933
Negative Tweets	7731	1994
Neutral Tweets	1816	455
Total	17282	4382

Table I. Description of Training and Testing Dataset used

The positive tweets are positive comments on a particular movie. Negative tweets are flames regarding the same movie and neutral tweets are random comments on various movies. The aim is to identify the flames that are made regarding the movie.

Pre-Processing of Datasets

The dataset collected from the SNS media contains opinions about the data which are expressed in different ways by individuals. The negative, positive and neutral polarity labeled dataset is highly susceptible to inconsistency and redundancy. In order to improve the quality of result, pre-processing of raw data is necessary as shown in fig. 2. The Apache OpenNLP library is a machine learning based toolkit for the of natural language text. It supports the most common NLP [6][7] tasks such as tokenization, sentence segmentation, part-of-speech tagging, named entity extraction, chunking, parsing and co reference resolution. These tasks are usually required to build more advanced text processing services. Stop Words List consists of English words that are most commonly used and do not contribute much to the sentiment of any sentence. The **Porter** stemming algorithm (or '**Porter stemmer**') is a process for removing the commoner morphological and inflexional endings from words in English. Its main use is as part of a term normalization process that is usually done when setting up Information Retrieval systems. It helps in removing repeated words and punctuation and thus help in improving the quality of the data.

The various pre-processing techniques required are :

A. Tokenization

In Lexical analysis, Tokenization is the process of breaking up a stream of text into words, symbols or other meaningful elements called tokens. In this way, the input data on tokenization will give collection of features. For the training data, these features are used for the construction of sentiment dictionary. Features here refer to words. For test data, these features are accessed in the dictionary to obtain their frequencies in all the three classes, which is then used for the classification of the input.

Figure 2. Preprocessing Steps

B. Case Folding

Case folding is the act of converting words into a (usually lowercase) form that does not necessarily result in the correct spelling. Here case folding is applied so as to convert all the words obtained after tokenization into lower case so that the matching of the words is not affected by the case.

C. Stop Word Removal

Many words will be syntactic words, i.e. they do not contribute to the semantic meaning of the sentence. Such words are required to be removed. Stop Words list is obtained and after tokenization such words that belong to the stop word list are discarded since they only lead to occupying space while not contributing for classification.

D. Stemming

In linguistic morphology and information retrieval, stemming is the process of reducing inflected (or sometimes derived) words to their word stem, base or root form, generally a written word form. Stemming will help in reducing the size of the sentiment dictionary and also help in getting more accurate results.

Sentiment Dictionary

Sentiment Dictionary is constructed using the annotated Training dataset that contains tweets and their annotations as they belong to positive, negative or neutral. Pre-Processing is applied to the dataset. Sentiment Dictionary contains four columns for words and their respective frequencies in all the three classes. This dictionary is referred by all the classifiers to get the respective frequencies (positive, negative and neutral) of all the word in the test document.

Naive Bayes Classifier

A Naive Bayes classifier [1][2] is a simple probabilistic model which is based on the Bayes rule along with a strong independence assumption. The Naive Bayes model involves a simplifying conditional Independence assumption i.e. given a class (positive, negative or neutral), the words are conditionally independent of each other. This assumption does not affect the accuracy in text classification by much but makes really fast classification algorithms applicable for the problem.

In the proposed system a Multinomial Naive Bayes Classifier has been used that first calculates joint probability of words and then assigns texts to the class (Positive/Negative/Neutral) they belong.

Assumptions in Multinomial Naive Bayes Classifier:

- In a document, the occurrence distribution of words conforms to the multinomial distribution.
- Obtain documents from a sequence of words and occurrence of words should be Independent of each other.
- Prior Probability of a document belonging to class c is a uniform distribution.

From Training data let:

$|V|$ = Number of unique words

$V = \{w_1, w_2, \dots, w_{|V|}\}$

$P_{i,c}$ = Occurrence Probability of a word w_i ($1 \leq i \leq |V|$) in class c

$\Theta_c = \{P_{1,c}, P_{2,c}, \dots, P_{|V|,c}\}$ = parameter Vector of c

$n_{i,d}$ = Number of occurrences of a word w_i in document d .

$P(d|\Theta_c)$ = Likelihood of a document d

$$P(d|\Theta_c) = \frac{(\sum n_{i,d})!}{\prod n_{i,d}!}$$

Naive Bayes Classifier outputs c^* as

$c^* = \operatorname{argmax}_c P(d|\Theta_c)$

$c^* = \operatorname{argmax}_c \prod_i P_{i,c}^{n_{i,d}}$

Laplacian Smoothing: If the classifier encounters a word that has not been seen in the training set, the probability of both the classes would become zero and there won't be anything to compare between. This problem can be solved by Laplacian smoothing.

For estimation of parameter $P_{i,c}$

$$P_{i,c} = \frac{n_{i,c} + (\alpha - 1)}{\sum_i n_{i,c} + |V|(\alpha - 1)}$$

$$\frac{n_{i,c} + (\alpha - 1)}{\sum_i n_{i,c} + |V|(\alpha - 1)}$$

In this paper $\alpha=2$ has been used which is well used as Laplacian Smoothing. So

$$c^* = \operatorname{argmax}_c \prod_i \left(\frac{n_{i,c} + (\alpha - 1)}{\sum_i n_{i,c} + |V|(\alpha - 1)} \right)^{n_{i,d}}$$

$$\left(\frac{n_{i,c} + (\alpha - 1)}{\sum_i n_{i,c} + |V|(\alpha - 1)} \right)^{n_{i,d}}$$

The main theoretical drawback of Naive Bayes methods is that it assumes conditional independence among the linguistic features. If the main features are the tokens extracted from texts, it is evident that they cannot be considered as independent, since words co-occurring in a tweet are somehow linked by different types of syntactic and semantic dependencies.

Maximum Entropy Classifier

The idea behind Maximum Entropy classifier [3] is that, given the constraints, most uniform models are most preferable. Maximum Entropy models are feature based models. These features are used to find a distribution over the different classes using logistic regression.

Goal: Given a set of features, a set of their corresponding functions that measures the contribution of each feature to the model known as weighting Factor, as $\lambda_i (i = 1, 2, \dots, k)$ and a set of constraints our aim is to find the probability distribution that maximizes the relative Entropy. By maximizing entropy, it is ensured that no biases are introduced into the system.

$$p(c|d, \lambda) = \frac{\exp \sum_i \lambda_i f_i(w_i, c)}{\sum_c \exp \sum_i \lambda_i f_i(w_i, c)}$$

Where $f_i(w_i, c) = \{$

1, if $w_i \in c$

0, otherwise

The Major advantages of using Maximum Entropy Classifier are as follows:

- Accuracy
- Consistency - This Method shows consistency in results and if priors are used results also over a period of time.
- Performance/Efficiency - handle huge amounts of data
- Flexibility - The classifier is flexible of having many different type of data in a unified platform that are classified accordingly.

Entropy Based Feature Selection

It is possible that dictionary may contain words (eg. I, we, us) that are not informative in case sentiment analysis. So, in order to remove these words, entropy method can be used, assuming that these words are uninformative. Let $|C|$ be the number of classes, $n_{i,c}$ is the number of occurrences of word w_i in class c . N_i is the number of w_i in all classes. $H(i)$ is the entropy of word w_i which is calculated as:

$$H(i) = - \frac{1}{\log |C|} \sum_c \frac{n_{i,c}}{N_i} \log \frac{n_{i,c}}{N_i}$$

$0 \leq H(i) \leq 1$

When the occurrences probabilities of a word w_i in respective classes are equal to each other i.e. there are no bias based on occurrences probabilities present between classes, $H(i)$ gets highest value 1. On contrary, when a word w_i occurs in only one class i.e. its occurrence probably is causing maximum bias between classes, the value of entropy $H(i)$ is lowest i.e. 0. So, those words are used whose Entropy $H(i)$ is equal to or lower than the threshold value α , i.e. $H(i) \leq \alpha$.

Thus if we have $\alpha=1$ then we use words that occur only in one class and if we set $\alpha=0$, all features will be used.

A. Naïve Bayes classifier with feature selection based on Entropy Method

All the features (words) in the input test sentence do not provide good bias to any class. So to improve the accuracy of the calculation only half of the features present in the input is used for the classification through Naive Bayes Classifier. These features are selected on the basis of entropy. The entropy of all the features in the input sentence is calculated and the features are arranged according to the increasing order of entropy. Half of the features from the starting of this increasing order array are only selected for the classification. It is found out on the experimental basis that this method increases the accuracy of the classification.

Figure 3. Naive Bayes Classifier with Feature Selection based on Entropy Method

Result & Analysis

Out of 4383 tweets, Naive Bayes Classifier was able to correctly classifier 3122 tweets. Maximum Entropy Classifier was able to correctly classify 3203 tweets, Naive Bayes Classifier with feature Selection based on Entropy Method was able to correctly classify 3310 tweets and Maximum Entropy Classifier with feature selection based on Entropy Method was able to correctly classify 3452 tweets.

Accuracy is calculated using the following formula:

$$\text{Accuracy} = \frac{\text{Correctly Classified Tweets}}{\text{Total Tweets}} * 100$$

So Accuracy of all the four classifiers obtained is as follows:

Sr.	Classifier	Accuracy(%)
01	Naive Bayes Classifier	71.26
02	Maximum Entropy Classifier	73.11
03	Naive Bayes Classifier With Feature Selection based on Entropy Method	75.60
04	Maximum Entropy With Feature Selection	79.00

Table 2. Accuracy of all the Four Classifiers

Graph in fig 4 shows the number of correctly classified tweets using all the four classifiers:

Figure 4. Graphical Representation of Results

Conclusion

Classification using the Naive Bayes Classifier has minimum accuracy. Its accuracy improves by using the feature selection method. The classification accuracy of Maximum Entropy Classifier is better than the Naive Bayes Classifier but still less than the Naive Bayes Classifier with Feature Selection based on Entropy Method whose accuracy is further less than the Maximum Entropy Classifier with Feature Selection based on Entropy Method. So it is found out that maximum Entropy Classifier with Feature Selection based on Entropy Method is best for classification of a given input tweet and hence is best for the detection of flames.

References

- [1] Shun Yoshida, Jun Kitazono. Seiichi Ozawa- Takahiro Sugaxvara- Tatsuya Haga. and Shogo Nakatnura. 'Sentiment Analysis for Various SNS Media using Naive Bayes Classifier and us application to flaming Detection “, 978- I-4799-4540-5/14/ IEEE, 20 14
- [2] Eric N, Forsyth, and Craig H. Martell, "Lexical and Discourse Analysis of Online chat Dialog” International Conference on Semantic Computing IEEE, 2007.
- [3] Yang Li, Bin_Xing Fang and Li Guo, "A Novel Online Spam Filler Bused on URLs and maximum entropy Model", IEEE, 2006.
- [4] Madhuri A. Tayal, Dr. M M Raghuwansh and Dr. Latesh Malik. 'Knowledge Representation predicate logic Implementation using Sentence-T.vpefor Natural Languages' 2013 International conference on circuits. Power and Computing Technologies IEEE- 201 13
- [5] Saravanan M, Divya and Kumares V S, "Probing of Geospatial Stream Data to Report Disorientation 2013 IEEE Recent Advances in Intelligent Computational Systems (RAICS).2013
- [6] Avinash J Agrawal, O G Kakde " Semantic Analysis of natural language using domain ontology for Information Access from International Journal of Intelligent System and Applications published by IEEE in 2013
- [7] Avinash J Agrawal , Snehal Dixit, on “survey on Spam Delection ”International Journal of Computational And communication Technology Volume 4 in 2013
- [8] A Shende, A J Agrawal, OG Kakde " Domam Specific /named Entity Recognition using Supervised Approach International Journal of Convutational Linguistics(IJCL) 3 (I). 67-78, 20 12
- [9] N R Kasture. A Agrawal, "A Supervised Word Sense Disambiguity method using ontology and context knowledge”, Computer Engineering and Intelligence System 3(8).2012