International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

COPY RIGHT

ELSEVIER SSRN

IJIEMR Transactions, online available on 3rd Aug 2019. Link :http://www.ijiemr.org/downloads.php?vol=Volume-08&issue=ISSUE-08

Title **AN EFFECTIVE SCHEME FOR DETECTING HATEFUL AND OFFENSIVE EXPRESSIONS ON TWITTER**

Volume 08, Issue 08, Pages: 204–209.

Paper Authors

KOLUSU NAVEEN, CH. PHANI KUMAR

Sri Vani Educational Society Group of Institutions, Chevuturu

USE THIS BARCODE TO ACCESS YOUR ONLINE PAPER

To Secure Your Paper As Per UGC Guidelines We Are Providing A Electronic Bar Code

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

AN EFFECTIVE SCHEME FOR DETECTING HATEFUL AND OFFENSIVE EXPRESSIONS ON TWITTER

KOLUSU NAVEEN[1], CH. PHANI KUMAR[2]

[1]M.Tech, Dept of CSE, Sri Vani Educational Society Group of Institutions, Chevuturu

[2]Assistant Professor, Dept of CSE, Sri Vani Educational Society Group of Institutions, Chevuturu

Abstract: Toxic online content has become a major issue in today's world due to an exponential increase in the use of internet by people of different cultures and educational background. Differentiating hate speech and offensive language is a key challenge in automatic detection of toxic text content. In this paper, we propose an approach to automatically classify tweets on Twitter into three classes: hateful, offensive and clean. Using Twitter dataset, we perform experiments considering n-grams as features and passing their term frequency-inverse document frequency (TFIDF) values to multiple machine learning models. We perform comparative analysis of the models considering several values of n in n-grams and TFIDF normalization methods. After tuning the model giving the best results, we achieve 95.6% accuracy upon evaluating it on test data. We also create a module which serves as an intermediate between user and Twitter.

Keywords: hate speech, offensive language, machine learning, twitter

1. INTRODUCTION

In the past 10 years, we have seen an exponential growth in the number of people using online forums and social networks. Every 60 seconds, there are 510,000 comments generated on Facebook [1] and around 350,000 tweets generated on Twitter [2]. The people interacting on these forums or social networks come from different cultures and educational backgrounds. At times, difference in opinions lead to verbal assaults. Moreover, unchecked freedom of speech over the web and the mask of anonymity that the internet provides in cites people to use racists slurs or derogatory terms. This can lower the self esteem of people, leading to mental illness and a negative impact on the society as a whole. Furthermore, toxic language can take various forms, such as cyberbullying, which was one of the major reasons behind suicide [3]. This issue has shown to be increasingly important in the last decade and detecting or removing such content manually from the web is a tedious task. So there is a need of devising an automated model that is able to detect such toxic content on the web.In order to tackle this issue, firstly we must be able to define toxic language. We broadly divide toxic language into two categories: hate speech and offensive language. Similar

approach was used in the studies [4] and [5]. According to Wikipedia, hate speech is defined as "any speech that attacks a person or group on the basis of attributes such as race, religion, ethnic origin, national origin, gender, disability, sexual orientation, or gender identity." We define offensive language as the text which uses abusive slurs or derogatory terms.In this paper, we propose an approach to devise a machine learning model which can differentiate between these two aspects of toxic language. We choose to detect hate speech and offensive text on Twitter platform. By using publicly available Twitter datasets we train our classifier model using n-gram and term frequency-inverse document frequency (TFIDF) as features and evaluate it for metric scores. We perform comparative analysis of the results obtained using Logistic Regression, Naive Bayes and Support Vector Machines as classifier models. Our results show that Logistic Regression performs better among the three models for n-gram and TFIDF features after tuning the hyper parameters. We also make use of Twitter Application Programming Interface (API) to fetch public user tweets from Twitter for detecting tweets containing hate speech or offensive language. Additionally, we create a module which serves as an intermediate between the user and Twitter.Social media has become a new standard of communications in the last years. Every year more and more people actively participate in the content creation, sometimes under the shield of anonymity. Social media has become a complex communication channel in which usually offensive contents are written. Supervising the content and banning offensive messages currently is a subject of high interest for social media administrators. Offensive speech can be addressed to individuals or groups due to the race, sexuality, religion and some other characteristics. In this task two of these characteristics will be used as target for offensive speech, women and immigrants. This problem will be considered as an Author Profiling task, since the main 1 alt.qcri.org/semeval2019/ goal is building a system which would ideally detect author whose content is offensive to women and/or immigrant. Author Profiling is widely studied and some new ideas arise from time to time. We have developed a new representation method for text that reduces the dimensionality of the information for each author to 6 characteristics per class. This representation, Frequency Analysis Interpolation, is used to codify the texts for each user and this codified information is used as input data to support vector machines with linear kernel. In a Big Data environment, reducing the number of characteristics from thousands to 6 per class allows an efficient way to deal with high volumes at high speed.

2. RELATED WORK

Various machine learning approaches have been made in order to tackle the problem of toxic language. Majority of the approaches deal with feature extraction from the text. Lexical features such as dictionaries [6] and bag-of-words [7] were used in some studies. It was observed that these features fail to understand the context of the sentences. N-gram based approaches were also used which shows comparatively better results [8].Although lexical features perform well in

detecting offensive entities, without considering the syntactical structure of the whole sentence, they fail to distinguish sentences' offensiveness which contain same words but in different orders [10]. In the same study, the natural language process parser, proposed by Stanford Natural Language Processing Group, was used to capture the grammatical dependencies within a sentence.There have been several studies on sentiment-based methods to detect abusive language published in the last few years. One example is the work [10] which applies sentiment analysis to detect bullying in tweets and use Latent Dirichlet Allocation (LDA) topic models [11] to identify relevant topics in these texts. Also studies have been conducted for Detection of harassment on Web 2.0 [12] More recently, distributed word representations, also referred to as word embeddings, have been proposed for a similar purposes [13]. Deep learning techniques are recently being used in text classification and sentiment analysis using paragraph2vec approach [14]. Convolutional Neural Network (CNN) based classification, which refers to the generation of a CNN for text classification, is being used as seen in [15], where they experimented with a system for Twitter hate-speech text classification based on a deep-learning, CNN model.The analysis of subjective language on OSN has been deeply studied and applied on different fields varying from sentiment analysis [10] [11] [12] to sarcasm detection [6] [7] or detection of rumors [13] etc. However, relatively fewer works (compared to the aforementioned topics) have been addressed to the hate speech detection. Some of these works targeted sentences in the world wide web such as the work of Warner et al. [5] and Djuric et al. [14]. The first work reached an accuracy of classification equal to 94% with an F1 score equal to 63.75% in the task of binary classification, and the second reached an accuracy equal to 80%.

Gitari et al. [15] extracted sentences from some major "hate sites" in United States. They annotated each of the sentences into one of three classes: "strongly hateful (SH)", "weakly hateful (WH)", and "non-hateful (NH)". They used semantic features and grammatical patterns features, run the classification on a test set and obtained an F1-score equal to 65.12%.

Nobata et al. [16] used lexicon features, n-gram features, linguistic features, syntactic features, pretrained features, "word2vec" features and "comment2vec" features to perform the classification task into two classes, and obtained an accuracy equal to 90%. Nevertheless, some other works targeted the detection of hateful sentences in Twitter. Kwok et al. targeted the detection of hateful tweets against black people. They used unigram features which gave an accuracy equal to 76% for the task of binary classification. Obviously, the focus on the hate speech toward a specific gender, ethnic group, race or other makes the collected unigrams related to that specific group. Therefore, the built dictionary of unigrams cannot be reused to detect hate speech towards other groups with the same efficiency. Burnap et al. [3] used typed dependencies (i.e., the relation between words) along with bag of words (BoW) features to distinguish hate speech utterances from clean speech ones.

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

3. ARCHITECTURE

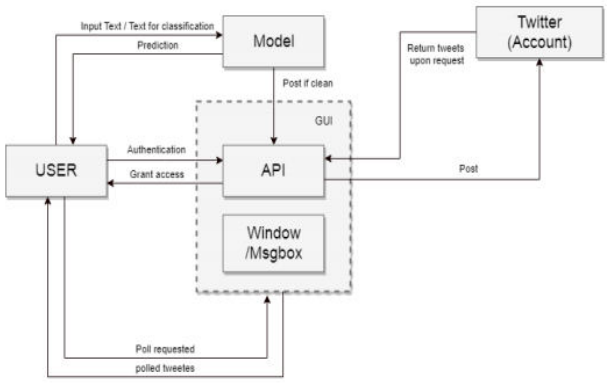

Fig. 1. Architecture of the system interfacing with Twitter through Twitter API

4. PROPOSED APPROACH

The review on the related work done in this field shows that the models trained after extracting N-gram features from text give better results [8]. Also, the TFIDF approach on the bagof-words features also show promising results [7]. Based on the review of features and the prominent classifiers used for text classification in the past work, we decided to extract ngrams from the text and weight them according to their TFIDF values. We feed these features to a machine learning algorithm to perform classification. Given the set of tweets, the aim of this work is to classify them into three categories: hateful, offensive and clean.

A. Data

The dataset that we have generated is a combination of three different datasets. The first dataset is publicly available on Crowdflower1 , which was used in [4] and [5]. This dataset contains tweets that have been manually classified into one of the following classes: “Hateful”, “Offensive” and “Clean”. The second dataset is also publicly available on Crowdflower2 , which consists the tweets with same classes as described previously. The third dataset is published on Github3 and used in the work [4] and [16]. It consists of two columns: tweet-ID and class. In this dataset, tweets corresponding to the tweet-ID are classified into one of the following three classes: “Sexism”, “Racism” and “Neither”.

B. Data Preprocessing

In the data preprocessing stage, we combine the three datasets used for this work. The tasks involves removal of unnecessary columns from the datasets and enumerating the classes. For the third dataset, we retrieve the tweets corresponding to the tweet-ID present in the dataset. We use Twitter API for this purpose. The classes “Sexism” and “Racism” in this dataset are both considered as hate speech according to the definition.

We convert the tweets to lowercase and remove the following unnecessary contents from the tweets:

- Space Pattern
- URLs
- Twitter Mentions
- Retweet Symbols
- Stopwords

We use the Porter Stemmer algorithm to reduce the inflectional forms of the words. After combining the dataset in proper format, we randomly shuffle and split the dataset into two parts: train dataset containing 70% of the samples and test dataset containing 30% of the samples.

C. Feature Extraction

We extract the n-gram features from the tweets and weight them according to their TFIDF values. The goal of using TFIDF is to reduce the effect of less informative tokens that appear very frequently in the data corpus. Experiments are performed on values of n ranging from one to three. Thus,

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

we consider unigram, bigram and trigram features. The formula that is used to compute the TFIDF of term t present in document d is:

tf idf(d, t) = tf(t) ∗ idf(d, t)

Also, both L1 and L2 (Euclidean) normalization of TFIDF is considered while performing experiments. L1 normalization is defined as:

$$v_{norm} = \frac{v}{|v_1| + |v_2| + ... + |v_n|}$$

where n in the total number of documents. Similarly, L2 normalization is defined as:

$$v_{norm} = \frac{v}{\sqrt{v_1^2 + v_2^2 + ... + v_n^2}}$$

We feed these features to machine learning models.

D. Model

We consider three prominent machine learning algorithms used for text classification: Logistic Regression, Naive Bayes and Support Vector Machines. We train each model on training dataset by performing grid search for all the combinations of feature parameters and perform 10-fold cross-validation. The performance of each algorithm is analyzed based on the average score of the cross-validation for each combination of feature parameters. The performance of these three algorithms is compared. Further, the hyperparameters of two algorithms giving best results are tuned for their respective feature parameters, which gives the best result. Again, 10-fold cross validation is performed to measure the results for each combination of hyperparameters for that model. The model giving the highest crossvalidation accuracy is evaluated against the test data. We have used scikit-learn in Python for the purpose of implementation.

5. CONCLUSION

In this paper, we proposed a solution to the detection of hate speech and offensive language on Twitter through machine learning using n-gram features weighted with TFIDF values. We performed comparative analysis of Logistic Regression, Naive Bayes and Support Vector Machines on various sets of feature values and model hyperparameters. The results showed that Logistic Regression performs better with the optimal ngram range 1 to 3 for the L2 normalization of TFIDF. Upon evaluating the model on test data, we achieved 95.6% accuracy. It was seen that 4.8% of the offensive tweets were misclassified as hateful. This problem can be solved by obtaining more examples of offensive language which does not contain hateful words. The results can be further improved by increasing the recall for the offensive class and precision for the hateful class. Also, it was seen that the model does not account for negative words present in a sentence. Improvements can be done in this area by incorporating linguistic features.

REFERENCES

[1] Zephoria.com, 2018. [Online]. Available: https://zephoria.com/top-15-valuable-facebook-statistics/. [Accessed: 22-Jun- 2018].

[2] “Twitter Usage Statistics - Internet Live Stats”, Internetlivestats.com, 2018. [Online]. Available: http://www.internetlivestats.com/twitterstatistics/. [Accessed: 22- Jun- 2018].

[3] S. Hinduja and J. Patchin, “Bullying, Cyberbullying, and Suicide”, Archives of

Suicide Research, vol. 14, no. 3, pp. 206-221, 2010.
[4] H. Watanabe, M. Bouazizi and T. Ohtsuki, "Hate Speech on Twitter: A Pragmatic Approach to Collect Hateful and Offensive Expressions and Perform Hate Speech Detection", IEEE Access, vol. 6, pp. 13825-13835, 2018.
[5] W. Warner and J. Hirschberg "Detecting hate speech on the World Wide Web," in Proc. Second Workshop Language Social Media, pp. 19– 26, June 2012.
[6] M. Bouazizi and T. Ohtsuki, "A pattern-based approach for sarcasm detection on Twitter," IEEE Access, Vol. 4, pp. 5477–5488, 2016.
[7] D. Davidov, O. Tsur, and A. Rappoport, "Semi-supervised recognition of sarcastic sentences in Twitter and Amazon," In Proc.14th Conf. on Computational Natural Language Learning, pp. 107–116, July 2010.
[8] C. Nobata, J. Tetreault, A. Thomas, Y. Mehdad and Y. Chang, "Abusive Language Detection in Online User Content", Proceedings of the 25th International Conference on World Wide Web - WWW '16, 2016.
[9] E. Greevy and A. Smeaton, "Classifying racist texts using a support vector machine", Proceedings of the 27th annual international conference on Research and development in information retrieval - SIGIR '04, 2004.
[10] J. M. Soler, F. Cuartero, and M. Roblizo, "Twitter as a tool for predicting elections results," in Proc. IEEE/ACM ASONAM, pp. 1194–1200, Aug. 2012.
[11] S. Homoceanu, M. Loster, C. Lofi, and W-T. Balke, "Will I like it? Providing product overviews based on opinion excerpts," in Proc. IEEE CEC, pp. 26–33, Sept. 2011.
[12] U. R. Hodeghatta, "Sentiment analysis of Hollywood movies on Twitter," in Proc. IEEE/ACM ASONAM, pp. 1401–1404, Aug. 2013.
[13] Z. Zhao, P. Resnick and Q. Mei, "Enquiring Minds: Early Detection of Rumors in Social Media from Enquiry Posts," in Proc. Int. Conf. World Wide Web, pp. 1395–1405, May 2015.
[14] N. Djuric, J. Zhou, R. Morris, M. Grbovic, V. Radosavljevic, and N. Bhamidipati, "Hate Speech Detection with Comment Embeddings," in Proc. WWW'15 Companion, pp. 29–30, May 2015.
[15] Njagi Dennis Gitari, Z. Zuping, Hanyurwimfura Damien, and Jun Long, "A Lexicon-based Approach for Hate Speech Detection," in , pp., Apr. 2015.
[16] Chikashi Nobata, Joel Tetreault, Achint Thomas, Yashar Mehdad, and Yi Chang, "Abusive Language Detection in Online User Content," in Proc. WWW'16, pp. 145–153, Apr. 2016.

AUTHOR'S PROFILE:

Mr. Kolusu Naveen is a student of Sri Vani Educational Society Group of Institutions, Chevuturu, Krishna District, (A.P).Pin-521229. Presently he is pursuing his M.Tech [C.S.E] from this college.

Mr. CH. Phani Kumar, M.Tech well known Author and excellent teacher. He is currently working as Assistant Professor, of Sri Vani Educational Society Group of Institutions, Chevuturu, Krishna District, (A.P) Pin-521229.

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org