International Journal for Modern Trends in Science and Technology, 9(02): 209-213, 2023

ISSN: 2455-3778 online
DOI: https://doi.org/10.46501/IJMTST0902038
Available online at: http://www.ijmtst.com/vol9issue02.html

Offensive Language Detection in Tweets using Various Regression and Classifier Algorithm

Dr.Ch.Surya Kiran[1], M.Jaya Lakshmi Devi[2], K. Triveni[2], M.Prashanth[2], M. Pradeep[2]

[1]Professor, Department of CSE, NRI Institute of Technology, Andhra Pradesh, India

[2]Department of CSE, NRI Institute of Technology, Andhra Pradesh, India

To Cite this Article

Dr.Ch.Surya Kiran, M.Jaya Lakshmi Devi, k. Triveni, M.Prashanth, M. Pradeep. Offensive Language Detection in Tweets using Various Regression and Classifier Algorithm. International Journal for Modern Trends in Science and Technology 2023, 9(02), pp. 209-213. https://doi.org/10.46501/IJMTST0902038

Article Info

Received: 18 January 2023; Accepted: 20 February 2023; Published: 24 February 2023.

ABSTRACT

The goal of this paper is to analyze tweets for inappropriate language using machine learning classification methods. In order to evaluate the efficacy of several well-known classification algorithms and locate the model that is most suited to the task at hand, a training and prediction pipeline has been built. We are going to take a dataset from Hate speech Twitter Annotations and Hate speech and offensive language detection, and then we are going to feed this data as input to a variety of different classifiers and regression models. utilizing matplotlib, we were able to evaluate our method on a dataset consisting of 25K tweets that was made available to the public. We were also able to demonstrate that our method was effective in terms of its distribution through the use of graphical representations. Finally, we were able to tune the best algorithm by taking into account both performance and time complexity. We did this by considering metrics such as accuracy, precision, and recall in both test and training data.

KEYWORDS: - Offensive Language Detection, Naive Bayes, Linear SVM, Attribute Selection, Twitter

1. INTRODUCTION

There are numerous websites online nowadays that use inappropriate language in their written content, and the proliferation of websites like these has the potential to mislead a significant number of individuals in our society. For situations like these, we come up with a simple solution, such as a model that can determine whether or not the provided text contains objectionable language. Our research is called "Offensive language identification in tweets using different regression and classifier methods," and it aims to identify offensive language in tweets. And our model was trained using the database from Hate speech Twitter Annotations and Hate speech and offensive language detection, and these data are provided as input to multiple classifiers and regression models. Furthermore, our model was trained using the database from Hate speech Twitter Annotations. our method was evaluated on a dataset consisting of 25 thousand tweets that was made available to the public. This was done so that we could demonstrate the effectiveness of our approach in the form of a graphical representation using matplotlib. Additionally, we tuned the best algorithm by taking into account both its performance and its time complexity. We did this by considering metrics such as accuracy, precision, and recall in both test and training data. And the model is connected to the UI in such a way that when the user gives the text to check for objectionable content, it displays the status result of the text, indicating whether or not the text is offensive. In

recent years, one of the most significant concerns has been the widespread practice of publishing offensive or abusive information on social media platforms. Because of the widespread use of social media platforms like Facebook and Twitter, this has resulted in a significant increase in the number of issues that have arisen. The primary reason for this is due to the fact that our model will automate and speed up the process of detecting offensive content in order to facilitate the moderation of offensive posts. This will be accomplished by checking and notifying the user as to whether the input that they have entered is offensive or not offensive. The primary goal of this research is to identify offensive language via the use of a variety of machine learning regression and classifier techniques. Also utilized to determine if the text that was supplied by the user was offensive or not offensive by using the most accurate algorithm possible.

2. LITERATURE SURVEY

Despite the fact that the analysis of hate speech in social media is a relatively new research field, it has generated a lot of interest and already has complete events devoted to the issue, in addition to a surge in relevant publications. During this session, some of the most important results from the most recent and pertinent publications will be presented, and then they will be discussed.

As an example, the International Workshop on Semantic Evaluation SemEval-20191 [6] featured challenges that were centered on the identification and classification of foul language in social media. [Citation needed] Identifying objectionable language was the first of the three primary sub-tasks, followed by the automated classification of different sorts of offenses, and finally locating offending content. As part of the first subtask, the communications were evaluated and categorized according to whether or not they were objectionable. If a tweet contains any profane language or language that may be considered objectionable, it was marked as offensive. The results that the Deep Learning BERT [7] produced for this challenge [8] were superior. The second subtask required participants to provide a prediction on the sort of crime. Insult and Untargeted were the two classes that were used for this purpose. A Twitter message was branded as "Untargeted" if it featured unacceptable language but was not directed at a specific person or group, while a Twitter post was labeled as "Insult" if it insulted a specific person or group (swearing). The rule-based method combined with a keyword filter, which may include hashtags, signs, emoticons, and other characteristics, was found to be the most effective solution for this issue [9]. The last subtask focused on the individuals who were the targets of the crimes. Individual was used to describe an offense committed against a single user, Group was used to describe an offense committed against a group of individuals, and Other was used to describe an offense committed against an organization, a scenario, an event, or a problem. The group that had superior overall outcomes was also the one that solved this issue using BERT [10]. Deep learning was also used in this way so that foul language in German texts could be identified [11].

A comparable use was investigated in the paper [12].

They developed a new dataset with the help of the Twitter API in order to classify tweets as either hate speech, offensive language, or none of these. Inside their dataset, they compiled a collection of 85.4 million Twitter samples from about 33,000 different users of the platform. From there, they constructed a collection of twenty-four thousand labeled tweet samples. For the purpose of the classification job, features such as bigrams, unigrams, and trigrams were weighted according to their respective TF-IDF values. A number of additional capabilities were incorporated, such as binary and count indicators for URLs, mentions, retweets, and hashtags.

They put a wide variety of classifiers to the test, including logistic regression, Naive Bayes, decision trees, random forests, and linear support vector machines (Support Vector Machine). Logistic Regression and Linear Support Vector Machines were shown to have a tendency to provide superior outcomes as a consequence of their tests. The top model achieved an F1-score of 0.9 and had an overall accuracy of 0.91. It also had a recall of 0.90. Unfortunately, the classifier could not provide satisfactory results when used to identify instances of hate speech; the accuracy and recall scores for this class were, respectively, 0.44 and 0.61.

It was hypothesized in [13] that Deep Learning may be used to classify the messages on social media. The categories of racism, sexism, or neither were taken into consideration. As part of their tests, they combined a number of different Long Short-Term Memory (LSTM)

models. In their categorization, the characteristics specified a user's propensity towards posting messages in any of the utilized classes, as well as the collection of messages submitted by a user and the subsets that included labeled messages. The approach that was taken was not influenced in any way by the language. Better results in detecting sexist communications were achieved using the suggested approaches (about 0.99 in terms of accuracy and F1-score). Nevertheless, racist messages received worse results, with roughly 0.75 and 0.70 of accuracy and F1-Score respectively. This contrasts with the findings achieved by neutral messages, which offered superior results (0.94 for precision).

The challenge of recognizing hate speech was enlarged in [14] to include the identification of vulnerable community members. Converting words to vector and employing n-grams were two of the strategies that were used in the process of extracting characteristics from the messages. They used the Gated Recurrent Unit (GRU) and many other Recurrent Neural Networks (RNNs) for the process of detecting hate speech. An accuracy of around 0.92 was given by these classifiers. It was suggested in [15] that abusive tweets posted in English may be categorized using something called a Convolutional Neural Network, or CNN. The labels that were used in this work were either derogatory, insulting, or designed to incite hatred. The most accurate outcomes earned a score of 0.83, while the most precise ones scored 0.80.

SVM, bidirectional Long Short-TermMemory (BiLSTM), and CNN were the classification models that were utilized in the study [16] to determine if messages were offensive or not offensive. In the studies, the BiLSTM achieved a higher level of accuracy in its ability to identify objectionable texts (0.81).

The accuracy was 0.83 for the identification of words that was not objectionable.Both the SVM and CNN models achieved a precision of 0.66 and 0.78 respectively when detecting offensive messages, and a precision of 0.80 and 0.87 when detecting messages that were not offensive.

In light of the information that was provided in the works that were discussed earlier, we have determined that there is a deficiency in research about the kinds of characteristics that are used in order to study offensive language.

Also, a better fine tuning approach to the usual classical classification approaches was not thoroughly explored, which may lead to lower results. This can be a problem since poorer findings can lead to poorer outcomes. So, we are going to restate that the primary purpose of this study is to analyze the quality of the features that were employed for this issue in addition to the strategy of fine tuning that was used by traditional classification methods.

3.PROPOSED SYSTEM

The old system gives rise to a number of drawbacks, all of which are addressed by the new model that has been suggested. We used Multinomial NB, Decision tree Classifier, Linear SVC, SGD classifier, Adaboost classifier, Bagging Classifier, Logistic Regression, and k-neighbors classifier to identify offensive tweets posted by a user in a particular data set. After that, we prepared a model for testing with a training dataset that contained an algorithm model that was more accurate.

We would feed it a sentence from the front page as an input, and then the model would use that phrase to determine the level of the output, which would proclaim whether or not the statement was offensive.We used to express the classification in a graphical form with score and algorithms, and we classified summaries of algorithms by utilizing attitudes such as test accuracy, F1 score, precision, recall time, and prediction time.

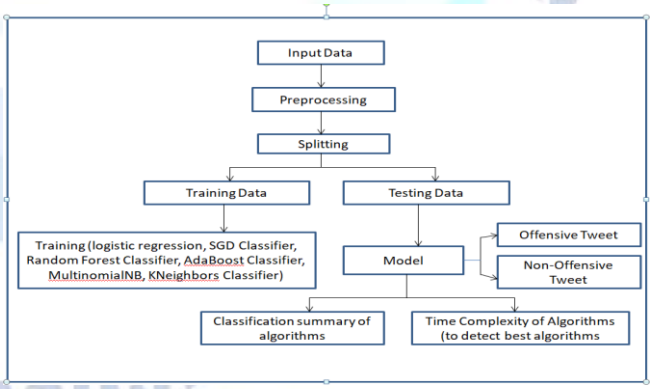

Figure1: Block Diagram for Proposed System

4.RESULTS

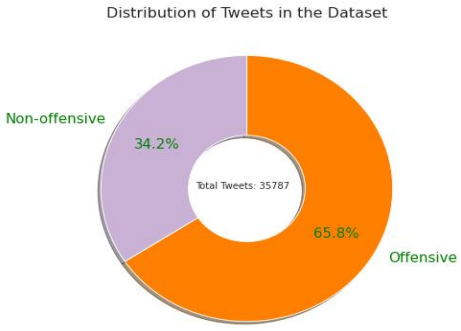

Figure 2: Distribution of tweets in the dataset

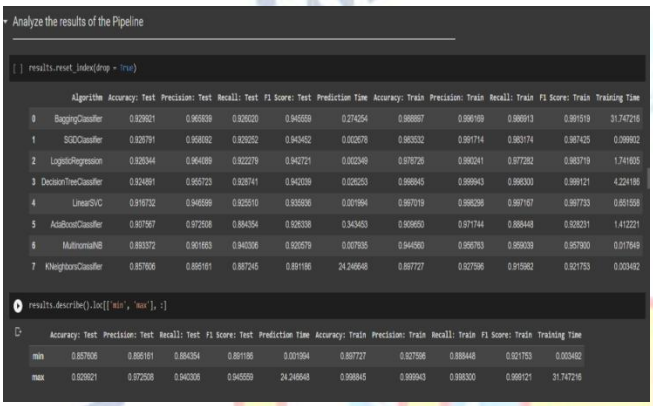

Figure 3: Algorithms analysis result

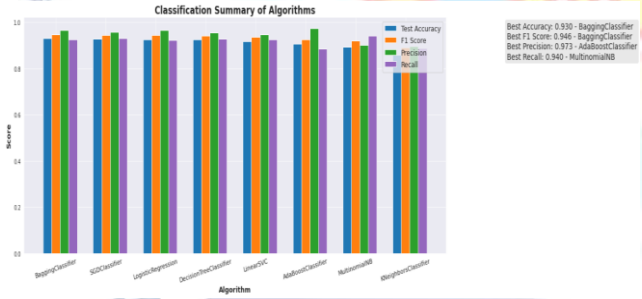

Figure 4: Classification summary of algorithms

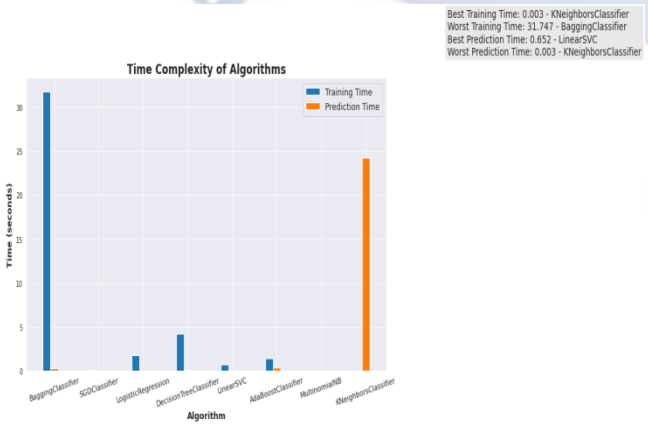

Figure 5: Time complexity of algorithms

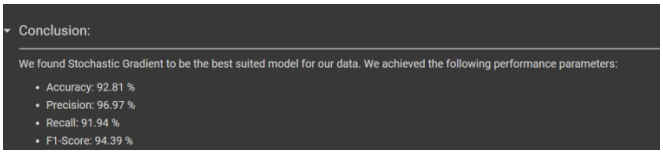

Figure 6: Result of best algorithms

5.CONCLUSIONS:

We construct a Linear Support Vector Machine (SVM) and a Naive Bayes classifier in this study to identify foul language used in tweets. Over the course of the test, it was discovered that the Linear SVM is quite sensitive to the sort of data that is used in the process of training. It was discovered that the procedure of parameter control was made more complex by the data normalization using tags that were used. The experiments also shown that the evaluation sequence of messages has a significant impact on the final outcome of the classifier. This finding is significant owing of the high standard derivation that was seen for the tests that were conducted using various seeds. This is a normal process because the weight regulation and the learning coefficient (alpha) cause the learning to be arranged by the other inputs, which results in an imbalance of the weights if large sequences of messages with the same label are given as input, for example. This happens because the learning is arranged by the other inputs. Thus, the Linear SVM requires a balanced input in order to provide accurate results. The challenge of configuring the parameters for this algorithm turned out to be more difficult than I had anticipated.

The Naive Bayes classifier, on the other hand, was shown to be an effective text classifier. This algorithm's quick execution may be attributed, in part, to its simple design and the relative ease with which it can be put into practice. This method has very excellent performance, where it shown to be better than many other algorithms presented in the literature. [Citation needed] This classifier did not have the issue that was identified with the SVM, the standard derivation that was produced via the testing was quite low, and the average value was extremely near to the best possible result.

Conflict of interest statement

Authors declare that they do not have any conflict of interest.

REFERENCES

[1] F. Del-Vigna, A. Cimino, F. Dell-Orletta, M. Petrocchi, and M. Tesconi, "Hate me, hate me not: Hate speech detection on facebook," in First Italian Conference on Cybersecurity, 2017.

[2] J. Jacobs and K. Potter, Hate crimes: Criminal law & identity politics. Oxford University Press on Demand, 1998.

[3] M. Bouazizi and T. Ohtsuki, "Multi-class sentiment analysis on twitter: Classification performance and challenges," Big Data Mining and Analytics, vol. 2, no. 3, pp. 181–194, Sep. 2019.

[4] G. Jalaja and C. Kavitha, Sentiment Analysis for Text Extracted from Twitter. Singapore: Springer Singapore, 2019, pp. 693–700.

[5] S. Sharma and A. Jain, "Cyber social media analytics and issues: A pragmatic approach for twitter sentiment analysis," in Advances in Computer Communication and Computational Sciences, S. K. Bhatia, S. Tiwari, K. K. Mishra, and M. C. Trivedi, Eds. Singapore: Springer Singapore, 2019, pp. 473–484.

[6] M. Zampieri, S. Malmasi, P. Nakov, S. Rosenthal, N. Farra, and R. Kumar, "Identifying and categorizing offensive language in social media (offenseval)," arXiv preprint arXiv:1903.08983, 2019.

[7] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," arXiv preprint arXiv:1810.04805, 2018.

[8] P. Liu, W. Li, and L. Zou, "Transfer learning for offensive language detection using bidirectional transformers," in Proceedings of the 13th International Workshop on Semantic Evaluation, 2019, pp. 87–91.

[9] J. Han, S. Wu, and X. Liu, "Identifying and categorizing offensive language in social media," in Proceedings of the 13th International Workshop on Semantic Evaluation, 2019, pp. 652–656.

[10] A. Nikolov and V. Radivchev, "Offensive tweet classification with bert and ensembles," in Proceedings of the 13th International Workshop on Semantic Evaluation, 2019, pp. 691–695.

[11] J. Risch, A. Stoll, M. Ziegele, and R. Krestel, "Offensive language identification using a german bert model."

[12] T. Davidson, D. Warmsley, M. Macy, and I. Weber, "Automated hate speech detection and the problem of offensive language," in Proceedings of the 11th International AAAI Conference on Weblogs and Social Media, ser. ICWSM '17, 2017. [13] G. Pitsilis, H. Ramampiaro, and H. Langseth, "Detecting offensive language in tweets using deep learning," arXiv preprint arXiv:1801.04433, 2018.

[13] Z. Mossie and J.-H. Wang, "Vulnerable community identification using hate speech detection on social media," Information Processing & Management, p. 102087, 2019.

[14] P. Mathur, R. Shah, R. Sawhney, and D. Mahata, "Detecting offensive tweets in hindi-english code-switched language," in Proceedings of the Sixth International Workshop on Natural Language Processing for Social Media, 2018, pp. 18–26.

[15] M. Zampieri, S. Malmasi, P. Nakov, S. Rosenthal, N. Farra, and R. Kumar, "Predicting the type and target of offensive posts in social media," arXiv preprint arXiv:1902.09666, 2019.

[16] Z. Waseem and D. Hovy, "Hateful symbols or hateful people? predictive features for hate speech detection on twitter," in Proceedings of the NAACL student research workshop, 2016, pp. 88–93.

[17] 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. 13 825–13 835, 2018.

[18] S. Kotsiantis, D. Kanellopoulos, and P. Pintelas, "Handling imbalanced datasets: A review," GESTS International Transactions on Computer Science and Engineering, vol. 30, no. 1, pp. 25–36, 2006.

[19] J. Wilbur and K. Sirotkin, "The automatic identification of stop words," Journal of information science, vol. 18, no. 1, pp. 45–55, 1992.

[20] I. Rish, "An empirical study of the naive bayes classifier," in IJCAI 2001 workshop on empirical methods in artificial intelligence, vol. 3, no. 22, 2001, pp. 41–46

[21] Y.-W. Chang and C.-J. Lin, "Feature ranking using linear svm," in Causation and Prediction Challenge, 2008, pp. 53–64. [23] G. Forman, "Bns feature scaling: an improved representation over tf-idf for svm text classification," in Proceedings of the 17th ACM conference on Information and knowledge management. ACM, 2008, pp. 263–270