IJITCE

ISSN 2347- 3657

International Journal of Information Technology & Computer Engineering

Email : ijitce.editor@gmail.com or editor@ijitce.com

International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 13, Issue 2, 2025

Using Text Classification for Identifying Harmful Language on Social Media

[1] Mrs.Neha Shireen

[2]MOHAMMED MUJTABA AHMED, [3] MOHD ABDUL MUHAIMIN, [4] MOHAMMED RAYYANUDDIN QURESHI.

[1]Assistant Professor, Department of CSE-AIML, Lords Institute of Engineering & Technology.

[234] Student, Department of CSE-AIML, Lords Institute of Engineering & Technology.

Abstract

Worryingly, foul language is becoming more common in crowdsourced material across different social media sites. To use such rhetoric is to potentially intimidate or offend someone or some group. Researchers have been looking at automatic speech detection and prevention for some time now, and they've produced a variety of supervised approaches and training datasets. Our proposed architecture for text categorization in this work includes eight classifiers, three embedding approaches, a modular cleaning step, and a tokenizer. The results of our studies on the dataset we received from Twitter for the purpose of detecting inflammatory language are encouraging. The three AdaBoost, SVM, and MLP algorithms achieved the greatest average F1-score on the popular TF-IDF embedding approach when hyperparameter tuning was taken into account.

Index Terms—offensive language detection, social media, machine learning, text mining

INTRODUCTION

Throughout the course of their lifespan has increased from 18% in 2017 to 37% in 2019 [63]. Offensive, hostile or threatening speech on the material shared by the audience could vary from mild or implicit bullying to serious and explicit violent threats over victims with particular characteristics such as race, sex, religion, community, etc. The increasing prevalence of cyberbullying in public media is a worldwide concern that has the potential to harm people's online life, as shown by [64]. Modern methods for identifying hate speech or other types of objectionable language take into account context, domain, and platform specifics, but do not take severity into account. So far, many datasets have been made public with the express purpose of testing the accuracy and reliability of these approaches [65]-[67]. Numerous scientific fields have been impacted by deep learning in the last ten years. These include medical imaging, social computing, healthcare, cyber security, natural language processing, and many more. New worries about the psychological and bodily security of social media users have emerged in recent years, coinciding with the seemingly exponential rise of these platforms. A study found that among bullied adolescents in the 12–18 age range, 15% experienced cyberbullying on social media. Our paper presents a modular pipeline for text classification that includes eight classifiers, three embedding approaches, a tokenizer, and a cleaning step. This study's experiment was optimized using a dataset that was based on Twitter. We don't promise that our approach will operate flawlessly on every social media site, but it might point the way for academics in both academia and business in their pursuit of better understanding these platforms. A more generalized application of this paper's findings is the comprehensive examination of identifying online

cyberbullying in online communities. A universal approach cannot be developed because of the unique characteristics of each social media site. For instance, due to the longer average post and conversation duration on Reddit, [68] demonstrates that training a classifier on Reddit is more difficult than Gab. In a classification job, Reddit input is more noisy than Gab. A synopsis of the remaining sections of the paper is provided below. We begin by outlining the experimental design and methods used in Section II. Section IV presents our numerical experiments and remarks, whereas Section III outlines our case study. Section V concludes with some last thoughts. Part II. Procedure The procedures for conducting the tests and cleaning and preparing the dataset are briefly covered in this section. Additionally, these stages are shown in Fig. 1, which will be explained later on. Section A: Preparing Data When training binary classifiers, the initial step is to prepare the data. The following are the data preparation techniques that must be meticulously followed: 1. Extracting pure

International Journal of Information Technology & Computer Engineering
ISSN 2347–3657
Volume 13, Issue 2, 2025

text from the dataset, deleting duplicates and NaNs. 2. Converting to lowercase. 3. Expanding the acronyms. These are the basic cleaning procedures that need to be applied to the data. • Jargon: Twitter users often use slang due to the nature of the microblogging platform. Particularly new slangs that haven't been defined in dictionaries yet pose challenges to text mining methods. As a result, we want to use the reference dictionary1 to standardize the content by eliminating slang and obsolete acronyms. • Techniques for removal: It is common practice on social media to incorporate emoticons, links, hashtags, and user references. In order to normalize the text, preparing the data and eliminating typical patterns selectively is required. 1 Get the latest Twitter moods at https://github.com/goncalopereira/twitter-moods. A. Tokenizer The first step in any text analysis is to tokenize the words and divide the text into smaller pieces, such as paragraphs and sentences. Our system allows us to build bespoke tokenizers at either the sentence or word level, which can then be passed into embedding algorithms. C. Engineering Features In this experiment, we convert text into numerical representation (also called em bedding or vector) using the typical vectorization embedding methods, such as i) Term Frequency-Inverse Document Frequency (TF IDF), ii) Word2Vec, and iii) FastText. Each one is briefly described below. • TF-IDF: Counting the number of times words appear in all texts is one approach to converting words into vectors. This method's overemphasis on frequently occurring terms in the dataset is one of its limitations. In TF-IDF, the weights of common words are assigned according to their relative frequency, as opposed to the word counting technique. • Word2Vec: This method's output is word vectors, and it accepts a compressed text file as input. To generate a decentralized word representation, two model topologies are available. Predicting the current word using the context (window size) is the job of the continuous bag-of-words (CBOW) architecture, whereas the Skip-gram uses the current word to predict words in the specified window. • FastText: Produced by adding vectors that correspond to the words in the text, FastText depicts low-dimensional vector text. In order to incorporate words, FastText uses a Neural Network. When it comes to training and assessment, the FastText model is often compared to other deep learning classifiers that are faster and more accurate [69], [70]. D. Algorithms for Classification For the binary classification challenge, we use eight classifiers in this work. Who are our classifiers? 1. Naïve Bayes (NB), a Gaussian algorithm; 2. Decision Tree (DT); 3. Logistic Regression (LR); 4. Random

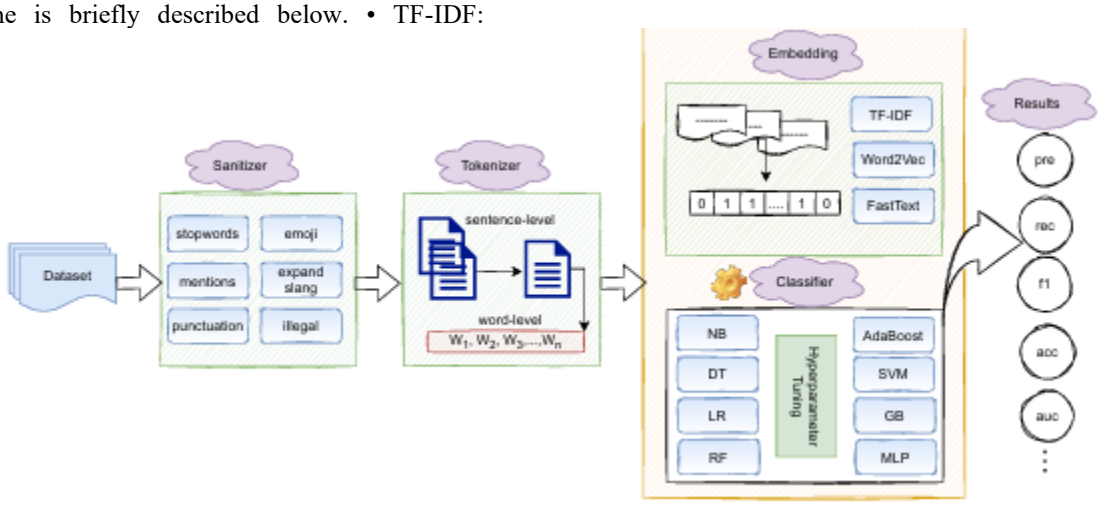

Fig. 1: The modular experimental setting with the flow of data from dataset to results.

The following are the several types of neural networks: RF, AdaBoost, SVM, GB, and MLP. Preliminary findings from the f1 accident and the autism spectrum Staff members were given the task of assigning a category to each tweet. When it came time to tweak the hyperparameters, we used Bayesian optimization as well. Bayesian optimization makes smart choices about the combinations of classifier parameters based on the results of earlier assessments. In addition, by restricting the search

International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 13, Issue 2, 2025

space, this method converges to an ideal set of hy perparameter values with fewer steps.

DATASET

A corpus of around 24,783 tweets with crowdsourced annotations was assembled by Davidson et al.2. The descriptors "hate speech," "offensive language," and "neither" are all represented in this dataset. They start with Hatebase.org's hate speech vocabulary, which contains terms and expressions that internet users have identified as hate speech. They obtained a sample of 33,458 tweets from Twitter users by searching for tweets that included phrases from the lexicon using the Twitter API. They obtained 85.4 million tweets after extracting the timeline for every user. Workers at CrowdFlower (CF) manually coded 25,000 tweets selected at random from this corpus that included phrases from the glossary. CF The five columns in each data file are as follows: Class, Count, Offensive language, Hate Speech, and Neither. The terms "HateSpeech" and "Offensive language" are defined as such for the purposes of this research. The sample is severely skewed, since 20620 out of 24783 tweets (or 83% of all tweets) include profanity or other foul words [71]-[75]. Fig. 2 further illustrates that rude communications are often shorter than average ones.

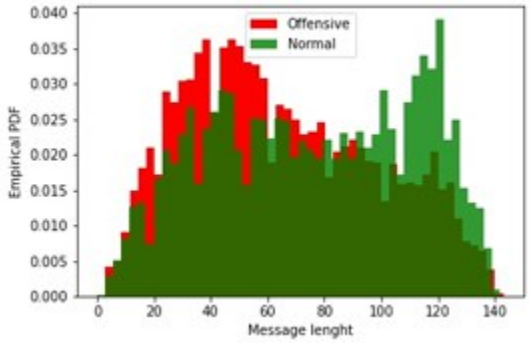

Fig. 2: Offensive/normal messages lengths

RESULTS AND DISCUSSIONS

Section II describes the experimental setting, and Figure 1 shows the results of the investigation. Table I displays the results of the training binary classifiers throughout the dataset, including precision, recall, F1 score, balanced accuracy, and AUC score. It should be mentioned that the results are arranged in decreasing order depending on F1-score.

TABLEI:Reportingperformancemetricsusingeight classifiers

classifier	*embedding*	*pre*	*rec*	*f1*	*acc*	*auc*
AdaBoost	TF-IDF	0.95	0.94	0.95	0.94	0.94
SVM	TF-IDF	0.95	0.95	0.95	0.95	0.93
MLP	TF-IDF	0.95	0.95	0.95	0.95	0.92
DT	TF-IDF	0.94	0.94	0.94	0.94	0.89
RF	TF-IDF	0.94	0.94	0.94	0.94	0.90
LR	TF-IDF	0.93	0.93	0.93	0.93	0.86
SVM	Word2Vec	0.92	0.93	0.93	0.93	0.85
MLP	Word2Vec	0.93	0.93	0.93	0.93	0.88
LR	Word2Vec	0.92	0.93	0.92	0.93	0.85
GB	TF-IDF	0.92	0.92	0.92	0.92	0.81
MLP	FastText	0.92	0.92	0.92	0.92	0.85
SVM	FastText	0.91	0.92	0.91	0.92	0.82
RF	Word2Vec	0.9	0.91	0.9	0.91	0.76
LR	FastText	0.9	0.91	0.9	0.91	0.79
GB	Word2Vec	0.9	0.91	0.9	0.91	0.79
AdaBoost	Word2Vec	0.89	0.9	0.89	0.9	0.78
AdaBoost	FastText	0.88	0.89	0.89	0.89	0.76
GB	FastText	0.89	0.9	0.89	0.9	0.76
RF	FastText	0.89	0.89	0.88	0.89	0.71
NB	Word2Vec	0.88	0.84	0.85	0.84	0.82
DT	Word2Vec	0.85	0.85	0.85	0.85	0.74
NB	FastText	0.86	0.84	0.85	0.84	0.78
DT	FastText	0.83	0.83	0.83	0.83	0.69
NB	TF-IDF	0.88	0.63	0.68	0.63	0.77

Fig.3:F1-scoretrendlinechartsondifferentclassifiersagainst eachembeddingmethod

The highest F1 score achieved during validation for the Adaboost, SVM, and MLP classifiers on TF-ID embedding is 95%, as shown in Table I. However, as a consequence of NBonTF-IDF, the lowest

performance is 68%. Based on the results, it seems that TF-IDF embedding is more effective for Adaboost, SVM, and MLP models, whereas NB and other models perform much worse. Table I shows that, out of the three embeddings, NB performs the poorest as a classifier. One possible explanation is that, when given a class label, the NBalgorithm thinks that features are conditionally independent. however the independence assumption is often disproven in reality. In spite of its strong performance on TF-IDF, DT ranks second worst when it comes to Word2Vec and FastText embeddings. Also shown in Figure 3 is the trend line of several classifiers compared to each embedding technique. When paired with the NB classifier, TF-IDF embeddings have the lowest score, even if they perform well for most of the classifiers alone. In this experiment, NB and DT perform the poorest as classifiers. We employed hyperparameter adjustment in our experiment, as indicated in Section II. When the parameters for the MLP classifier are set to Adaptive and Lu, respectively, the outcome might be favorably affected by the learning rate. Additionally, the values MinimumSampleSplit=2 for DT and GB produced superior results compared to MinimumSampleSplit=

CONCLUSIONS

In this study, we provide a Twitter-centric modular text classification pipeline for use with social media datasets. Our suggested method makes use of a modular design that facilitates the simple integration of various text categorization components. One important thing this research has to offer is a novel modular text classification pipeline that can help with benchmarking by analyzing the best-performing techniques, features, and embeddings from the state-of-the-art.

REFERENCES

[1]. N. Sabetpour, A. Kulkarni, and Q. Li, "OptSLA: an optimization-based approach for sequential label aggregation," in Findings of the Association for Computational Linguistics: EMNLP 2020. On line: Association for Computational Linguistics Nov. 2020, pp. 1335–1340. [Online]. Available: https://aclanthology.org/2020.findings-emnlp.119

[2]. N. Sabetpour, A. Kulkarni, S. Xie, and Q. Li, "Truth discovery in sequence labels from crowds," 2021.

[3]. S. Khorshidi, G. Mohler, and J. G. Carter, "Assessing gan-based approaches for generative modeling of crime text reports," in 2020 IEEE International Conference on Intelligence and Security Informatics (ISI). IEEE, 2020, pp. 1–6.

[4]. S. Khorshidi, M. Al Hasan, G. Mohler, and M. B. Short, "The role of graphlets in viral processes on networks," Journal of Nonlinear Science, vol. 30, no. 5, pp. 2309 2324, 2020.

[5]. O. Jafari, P. Nagarkar, B. Thatte, and C. Ingram, "Satel litener: An effective named entity recognition model for the satellite domain," in Proceedings of the KMIS 2020, vol. 3, 2020, pp. 100–107.

[6]. S. Zhang, O. Jafari, and P. Nagarkar, "A survey on machine learning techniques for auto labeling of video, audio, and text data," arXiv preprint arXiv:2109.03784, 2021.

[7]. M. Saadati, J. Nelson, A. Curtin, L. Wang, and H. Ayaz, "Application of recurrent convolutional neural networks for mental workload assessment using functional near infrared spectroscopy," in Advances in Neuroergonomics and Cognitive Engineering, H. Ayaz, U. Asgher, and L. Paletta, Eds. Cham: Springer International Publishing, 2021, pp. 106–113.

[8]. N. Kalantari, D. Liao, and V. G. Motti, "Characterizing the online discourse in twitter: Users' reaction to mis information around covid-19 in twitter," in 2021 IEEE International Conference on Big Data (Big Data), 2021, pp. 4371–4380.

[9]. A. Esmaeilzadeh, M. Heidari, R. Abdolazimi, P. Ha jibabaee, and M. Malekzadeh, "Efficient large scale nlp feature engineering with apache spark," in 2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC). IEEE, 2022.

[10]. R. Abdolazimi, M. Heidari, A. Esmaeilzadeh, and H. Naderi, "Mapreduce preprocess of big graphs for rapid connected components detection," in 2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC). IEEE, 2022.

[11]. M. Malekzadeh, P. Hajibabaee, M. Heidari, and B. Berlin, "Review of deep learning methods for automated sleep staging," in 2022 IEEE 12th Annual

Computing and Com munication Workshop and Conference (CCWC). IEEE, 2022.

[12]. M. Heidari, J. H. J. Jones, and O. Uzuner, "An empirical study of machine learning algorithms for social media bot detection," in 2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), 2021, pp. 1–5.

[13]. J. Liu, M. Malekzadeh, N. Mirian, T. Song, C. Liu, and J. Dutta, "Artificial intelligence-based image enhancement in pet imaging: Noise reduction and resolution enhance ment," PET clinics, vol. 16, no. 4, pp. 553–576, 2021.

[14]. D. Bamgboje, I. Christoulakis, I. Smanis, G. Chavan, R. Shah, M. Malekzadeh, I. Violaris, N. Giannakeas, M. Tsipouras, K. Kalafatakis et al., "Continuous non invasive glucose monitoring via contact lenses: Current approaches and future perspectives," Biosensors, vol. 11, no. 6, p. 189, 2021.

[15]. M. Malekzadeh, T. Song, and J. Dutta, "Pet image de noising using unsupervised domain translation," in 2021 IEEE Nuclear Science Symposium and Medical Imaging Conference Proceedings (NSS/MIC). IEEE, 2021.

[16]. Y. Soofi and M. Bitaraf, "Output-only entropy-based dam age detection using transmissibility function," Journal of Civil Structural Health Monitoring, pp. 1–15, 2021.

[17]. P. Hajibabaee, F. Pourkamali-Anaraki, and M. Hariri Ardebili, "An empirical evaluation of the t-sne algorithm for data visualization in structural engineering," in 2021 IEEE International Conference on Machine Learning and Applications. IEEE, 2021.

[18]. S. Zad, M. Heidari, J. H. J. Jones, and O. Uzuner, "Emotion detection of textual data: An interdisciplinary survey," in 2021 IEEE World AI IoT Congress (AIIoT), 2021, pp. 0255–0261.

[19]. S. Zad, M. Heidari, J. H. Jones, and O. Uzuner, "A survey on concept-level sentiment analysis techniques of textual data," in 2021 IEEE World AI IoT Congress (AIIoT), 2021, pp. 0285–0291.

[20]. M. Heidari, S. Zad, B. Berlin, and S. Rafatirad, "Ontology creation model based on attention mechanism for a spe cific business domain," in 2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), 2021, pp. 1–5.