Identifying Offensive Language in English Tweets using Machine Learning and Deep Learning Methods

Arif Shahriar
ashahri1@ualberta.ca

Alvina Awwal
awwal@ualberta.ca

1 Task

Our paper is oriented around the SemEval 2020 Task 12 which was "Multilingual Offensive Language Identification in Social Media". We chose this particular task because since the onset of COVID and the decrease in outdoor activities, social media has become the general population's sanity. Twitter has always been one of the most popular methods of communication among the ordinary citizens, celebrities and even important people of the government. Statistics show that in 2020, 500 million tweets were sent per day with 152 million daily active users on the platform (Whitney, 2020). Our focus on this proposal is using English tweets to identify offensive language on Twitter data.

Social media and its possible anonymity and unrestricted access has always been a comforting blanket for a vast majority of people. Although this cloak has both advantages and disadvantages, on online platforms, it is usually impacted negatively in the form of open abusive, racial or hateful comments against other users. This gives rise to an important and increasingly necessary research area in the field of Natural Language Processing (NLP) where more and more work is now being done in order to help identify, classify and come up with solutions to reduce these occurrences online. This is critical because words hold a lot of weight and when thrown randomly and carelessly at innocent people from behind screens, it can impact lives and the societies severely.

2 State-of-the-art

We will be covering two tasks in our project. The first task focuses on identifying offensive language, which classifies tweets as offensive or not. The second task categorizes identified offensive tweets as targeted offences or non-targeted insults. Quite a lot of research has been previously done on similar tasks but we concentrated primarily on a few papers from this SemEval 2020 competition since those are recent and gave admirable results. Parikh et al. (2020) demonstrated that for English tweets, a transformer-based pre-trained model BERT yielded the best results in both tasks against other machine learning and deep learning models such as SVM, CNN, Logistic Regression, etc. Another paper by Anwar et al. (2020) used an ensemble of seven machine and deep learning models combined and achieved good results on the 5 languages that they based their project on, with the highest macro F1 score in the English language.

Among all the teams that participated on this exact task, the state-of-the-art outcome on task 1 was 0.9204 with the use of an ensemble of ALBERT models of different sizes as well a different team's RoBERTa-large that was fine tuned on the SOLID dataset which was given to them. For task 2, an ensemble method got the best F1 score of 0.7462 (Zampieri et al., 2020). Ensembling methods seem to perform the best for these kinds of NLP related tasks. Ensemble is a method used in machine learning (ML) and statistics which works with multiple learning algorithms together in order to increase the predictive performance than what would be obtained from a single model. The classifiers combined with varied weights give better and more stable results through weighted average ensemble.

3 Methodology

For our tasks, we will be using both machine learning and deep learning approaches to label the tweets in two classes. Firstly, we will be applying preprocessing steps to remove inconsistency and errors like empty rows, stop words, urls from

raw data (Bedi, 2018). In addition to that, tokenization and lemmatization will be implemented to separate the words, lemmas, or stems. Dataset will be divided into train and test sets for training and evaluation purposes. For word vectorization, term frequency and inverse document frequency method (TF-IDF) will be applied, which keeps track of total occurrences of certain words to recognize their importance in the given context[1]. This method will eventually produce a feature vector. Moreover, it will increase the performance of the algorithm as data becomes more interpretable by models. A logistic regression mode will be trained on the derived feature values from word vectorization method. Additionally, a discriminative classifier model SVM will be used for the first task which tries to draw a plane with maximum margin to separate data values from binary classes (VanderPlas, 2016). Besides the traditional machine learning algorithms, a pre-trained bi-directional model will be used for task 1 which is computationally less expensive. Bidirectional Encoder Representations from Transformers model (BERT) utilizes masked language models to arbitrarily mask few words from provided data and then predicts the vocabulary for masked tokens using context from both left and right direction (Devlin et al., 2018). This pre-trained model has outperformed task-specific architectures in eleven state-of-the-art NLP tasks. In addition to that, we intend to use a convolutional neural network model (CNN) which performs a special operation called convolution to extract meaningful information by traversing learnable filters through word embeddings. Instead of using vectors like typical ML algorithms, available pre-defined embeddings like Glove will be used to find similarities between words that belong to the same category (Choubey, 2020). This model will also apply pooling layers which reduces the dimensionality of the dataset and simultaneously preserves valuable information in data.

Our solutions will be evaluated against the benchmarks in this particular project and topic. The methods we plan on using have already been mentioned above and we aim on running those as best as we can and achieve good results. Ensembling at the end to give an even better outcome is also our goal. The metrics that will be used for evaluation are the F1 scores, precision and recall.

[1]https://en.wikipedia.org/wiki/Tf–idf

Tweet_id	Tweet	avg	std
11595337130442-34241	@kind_honest HELL YES! His grinned and thumbs up are disgusting!	0.84	0.15

Table 1: Example of input data with tweet

Tweet_id	Tweet	TaskA_label
A0	@USER @USER He's an evil law breaker that should be in prison with his criminal heartless family.!	OFF

Tweet_id	Tweet	TaskB_label
A0	@USER it means go away with your sorry ass kkkkk should be in prison with his criminal heartless family.!	TIN

Table 2: Example output of Tasks

4 Available Code

We have also found some code on certain architectures which we plan on using for our project such as BERT and SVM and have successfully been able to run those on a small dataset using Spyder IDE and Google Colab[2] (Bedi, 2018). Those codes need some modifications and experimenting of hyperparameters to better suit our research.

5 Available Data

Examples of input and output data are provided in Table 1 and 2.

The data that we will be using for this task is the SOLID Dataset which consists of 9 million English tweets for the training set and around 4 thousand tweets for the test set (Rosenthal et al., 2021). The file given for the training data is a tsv file with 3 columns- id; average; std. ID refers to the tweet IDs, Average is the average of the confidences predicted by several supervised models for a specific instance to belong to the positive class for that subtask and Std is the confidences' standard deviation from average confidences for a particular instance. Due to hardware limitations, we have been able to scrape around 12 thousand tweets using the Twitter API. The test data for each task is in 2 tsv files. For task 1, one file contains the IDs and tweets and the other the corresponding ID and label of whether the tweet is offensive or not in the form OFF or NOT. Task 2 contains a tsv file with IDs and tweets and the other whether a tweet ID is TIN- targeted or UNT- untargeted.

6 Repository URL

https://github.com/UOFA-INTRO-NLP-F21/f2021-proj-arif-shahriar-anik

[2]https://github.com/ThilinaRajapakse/BERT_binary_text_classification

References

Talha Anwar and Omer Baig. 2020. Tac at semeval-2020 task 12: Ensembling approach for multilingual offensive language identification in social media. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 2177–2182.

Gunjit Bedi. 2018. A guide to text classification(nlp) using svm and naive bayes with python.

Vijay Choubey. 2020. Text classification using cnn.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Apurva Parikh, Abhimanyu Singh Bisht, and Prasenjit Majumder. 2020. Irlab_daiict at semeval-2020 task 12: Machine learning and deep learning methods for offensive language identification. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 2006–2011.

Sara Rosenthal, Pepa Atanasova, Georgi Karadzhov, Marcos Zampieri, and Preslav Nakov. 2021. Solid: A large-scale semi-supervised dataset for offensive language identification. *Proceedings of the Findings of the Association for Computational Linguistics: ACL-IJCNLP*, pages 915–928.

Jake VanderPlas. 2016. In-depth: Support vector machines.

Margot Whitney. 2020. 40 twitter statistics marketers need to know in 2020.

Marcos Zampieri, Preslav Nakov, Sara Rosenthal, Pepa Atanasova, Georgi Karadzhov, Hamdy Mubarak, Leon Derczynski, Zeses Pitenis, and Çağrı Çöltekin. 2020. Semeval-2020 task 12: Multilingual offensive language identification in social media (offenseval 2020). *arXiv preprint arXiv:2006.07235*.