IRLab_DAIICT at SemEval-2020 Task 12: Machine Learning and Deep Learning Methods for Offensive Language Identification

Apurva Parikh
DA-IICT
Gandhinagar, India
apurvakparikh
@gmail.com

Abhimanyu Singh Bisht
DA-IICT
Gandhinagar, India
bisht2492
@gmail.com

Prasenjit Majumder
DA-IICT
Gandhinagar, India
p_majumder
@daiict.ac.in

Abstract

The paper describes systems that our team IRLab_DAIICT employed for shared task OffensEval 2020: Multilingual Offensive Language Identification in Social Media shared task. We conducted experiments on the English language dataset which contained weakly labelled data. There were three sub-tasks but we only participated in sub-tasks A and B. We employed Machine learning techniques like Logistic Regression, Support Vector Machine, Random Forest and Deep learning techniques like Convolutional Neural Network and BERT. Our best approach achieved a MacroF1 score of 0.91 for sub-task A and 0.64 for sub-task B.

1 Introduction

Since the past few years, rapid improvement in communication technologies has led to an exponential rise in the number of people who have unrestricted access to the internet. This newfound abundance has seen the emergence of the ”Netizen”, and Social Media Platforms like Twitter, Facebook, Instragram etc, the cyber-agoras where Netizens assemble to discuss and debate ideas. But in several cases the exchange on these platforms is not civil and at times the behaviour exhibited by individuals on social media platforms can be hostile, targeting individuals or communities by posting insults or threats. Twitter receives more than 500 million tweets daily[1], which can be classified as offensive or hate speech. This makes the detection of offensive and hateful content on social media platforms an important research question in the field of Natural Language Processing. To promote research in this field several competitions have been organised such as OffensEval 2019 (Zampieri et al., 2019b), HASOC 2019 (Modha et al., 2019), HatEval 2019 (Basile et al., 2019).

OffensEval 2020 (Zampieri et al., 2020) task was organised for five languages English, Arabic, Danish, Greek and Turkish. The organizers proposed a hierarchical task which consists of three sub-tasks as below:

- **Sub-task A: Identification of Offensive Language**, posts are classified as follows:
  - Offensive (OFF): if post contains non-acceptable language or targeted offense.
  - Not Offensive (NOT): post that do not contain profanity or offensive language.
- **Sub-task B: Categorization of Offensive**, posts identified as OFF in sub-task A are further categorized as:
  - Targeted Insult (TIN): posts containing an insult/threat to an individual, group, or others.
  - Untargeted Insult (UNT): posts containing non-targeted profanity and swearing.
- **Sub-task C: Identification of target**,in which posts identified as TIN in sub-task B, are further classified on basis of type of targeted insult.
  - Individual(IND): posts targeting an individual.

[1] https://sproutsocial.com/insights/social-media-statistics/

Proceedings of the 14th International Workshop on Semantic Evaluation, pages 2006–2011
Barcelona, Spain (Online), December 12, 2020.

- Group (UNT): posts targeting group of people.
- Other (OTH): target that does not fall into any of above two category.

All three sub-tasks were organised for English, where as for the other languages only sub-task A was organised. Our team conducted experiments for sub-tasks A and B on English Language.

While experimenting for Sub-task A we discovered that statistical methods i.e TF-IDF with Logistic Regression performed on par with Transformer based model i.e BERT. For this sub-task we achieved 22^{nd} rank out of 85 participants and our submission utilized an Ensemble model. For Sub-task B machine learning approaches like Logistic Regression/Random Forest with TF-IDF, and deep leaning approaches using Convolutional Neural Networks were not able to achieve good results. The best results were given by a generic pre-trained Transformer-based language model, BERT, thereby showing its prowess in categorizing offensive tweets without any explicit feature engineering. For sub-task B we achieved 11^{th} rank out of 43 teams who submitted their results.

The rest of paper is organized as follows. In Section 2 we discuss related work. In Section 3 we briefly discuss about the dataset provided. Proposed methods and obtained results are presented in Section 4. Finally we conclude our work and future works in Section 5.

2 Related Work

Given the pernicious nature of hate speech and offensive content on social media platforms, development of methods to mitigate their proliferation has attracted researcher from many field like Computer Science, Computational Linguistic, Psychology, etc., to come forward and provide solutions. The term hate speech was first coined by (Warner and Hirschberg, 2012). (Davidson et al., 2017) detected hate speech using word n-grams and sentiment lexicon. (Schmidt and Wiegand, 2017) did an extensive study on hate speech, they identified various features for hate speech like words, sentiment, linguistic, knowledge-based features, etc.

Razavi et al. (2010) was the first to work on identifying offensive language using machine learning, they used rule-based methods and statistical features. Recent work in offensive language detection by Zampieri et al. (2019a) have focused on more fine-grained details, i.e, type and target of posts. Recently organized competitions like OffensEval2019 (Zampieri et al., 2019b), HASOC2019 (Modha et al., 2019) also focused on type and target of posts. In OffensEval2019 most of the teams used Deep learning based models like LSTM, Bi-LSTM, CNN with pre-trained embeddings. Transfer Learning using pre-trained language models was a popular methodology amongst top performers, with 7 out of top 10 using BERT (Devlin et al., 2019) for sub-task A. There were few teams who used machine learning algorithms like Logistic Regression and SVM.

3 Dataset

For the second iteration of OffensEval, the organizers provided a weakly labeled dataset with huge amount of data points for training (Rosenthal et al., 2020). Table 1 shows examples of dataset as provided, where Id is index of a post, Post is text, AVG_CONF 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 CONF_STD is the confidences' standard deviation from AVG_CONF for a particular instance. The positive class is OFF for sub-task A, whereas UNT for sub-task B. Table 2 shows statistics of dataset provided. We used external dataset OffensEval2019(Zampieri et al., 2019b) for testing our trained model. The test data was annotated by humans.

4 Experiments and Results

4.1 Text pre-processing

As social media data contains a lot of noise text preprocessing needs to be done so as to aid feature extraction. The provided data was already pre-processed to some extent, the hyperlinks had been replaced with a URL tag and user names (text followed by '@') with a USER tag. We performed further text pre-processing as follows,

Sub-task A

Id	Post	AVG_CONF	CONF_STD
1159533703904800769	most important tweet of the day : Fuck Donald Trump and his whole party! thank you for tuning in!	0.565238	0.187498
1159533707922956289	@USER Omg yes let him enjoy his icecream	0.272727	0.197008

Sub-task B

Id	Post	AVG_CONF	CONF_STD
1159533739871002625	@USER @USER @USER This guys is dumb check his latest tweets he is sick	0.169093	0.180201
1159535368833773570	Looking at a picture that was taken of me today I am fucking fat! Its disgusting	0.583515	0.053105

Table 1: Examples of few data points

Details	#Tweets in Train Data	#Tweets in Test Data
Sub-task A	9075418	3887
Sub-task B	188974	1422

Table 2: Dataset Statics

- The words in a hashtag were retained because the tags are unique and informative, therefore we felt that they may help with categorization.
- All punctuation, numbers, URL tags and emojis were removed.
- Stopwords were removed for some methods with the help of Python package NLTK[2]
- Text was converted to lowercase.

4.2 Sub-task A

Sub-task A was a binary classification problem to predict if a given tweet is Offensive or Not. Main challenge for this task was to decide which class a given tweet belongs to based on the AVF_CONF and the CONF_STD. For sub-task A we hypothesised that if the AVG_CONF $>$ 0.5 then the tweet belongs to the OFF class else the NOT class. Below are methods used for this sub-task.

1. **TF-IDF with Logistic Regression:** For this method we generated the TF-IDF representation for each tweet using Sklearn Python based library with max_features values as 100K top features. For obtained representation we applied Logistic Regression.
2. **TF-IDF with SVM and Random Forest:** For this method we sampled 100K data points using Sklearn train_test_split function with random_state value as 50. For sampled data point we generated their TF-IDF representation with max_feature values as 50K. For obtained features we apllied SVM and Radom Forest classifiers from Sklearn with default values.
3. **BERT:** For this method we sampled 100K data points using Sklearn train_test_split function with random_state value as 50. We fine-tuned a bert-base-uncased for text classification from the HuggingFace[3] transformer library on our sampled data for 3 epochs. Max Number of Tokens was set to 64 and batch size was 64. We experimented with and without stopwords, but with stopwords the model gave us better results on test data.
4. **Ensemble:** We created an ensemble model by doing a majority vote decoding for all four models defined above in case the prediction of the ensemble indeterminate, we select with the prediction provided by the BERT model. This was the method that we submitted as our final submission, which was used for ranking by organizers.

[2] https://www.nltk.org/

[3] https://huggingface.co/

Table 3 shows results obtained by our methods on the test set. Macro F1 was the primary evaluation metric for the competition.

Methods	**F1 (Macro)**
TF-IDF with Logistic Regression	0.90925
TF-IDF with SVM	0.90528
TF-IDF with Random Forest Classifier	0.89677
BERT Classifier fine-tuned for 3 epochs without stopwords	0.90892
Ensemble of all the methods	**0.91039**
BERT Classifier fine-tuned for 2 epochs with stopwords (Post Competition)	**0.91416**
Top Performer	0.92226

Table 3: Results of Sub-task A on Test set

4.3 Sub-task B

Sub-task B was a binary classification problem to predict if a given tweet is a Targeted Insult or an Untargeted Insult. For sub-task B we set two thresholds when developing our models. We conducted experiments with AVG_CONF $>$ 0.5 as Untargeted and AVG_CONF $>$ 0.4 as Untargeted. When we performed tests on our models using the OffensEval 2019 dataset the AVG_CONF $>$ 0.4 threshold gave better results so we kept that threshold while training our models. We used the complete training data to train all the methods listed below.

1. **TF-IDF with Logistic Regression:** Text was represented using TF-IDF with top 40K features. Using the obtained features we trained a Logistic Regression Classifier.

2. **CNN with Glove Embedding:** CNN based text classification utilizes n-gram features, which depend on the width of the convolution kernels, we used kernels of size 2,3 and 4. Also CNNs are faster to train as compared to RNNs, in general. For this method we used pre-trained GloVe embeddings (Pennington et al., 2014) [4] of 200 dimension to represent tokens. Max Sentence length was set to 64. We used the CNN model proposed by (Yoon, 2014). Model was trained for 10 epochs.

3. **BERT:** We fine-tuned a bert-base-uncased for text classification from the HuggingFace transformer library for 3 epochs. When we tested this model on OffensEval 2019 dataset, we were able to achieve the test results of the competitions top performer so we kept this model for our final submission which was used for ranking.

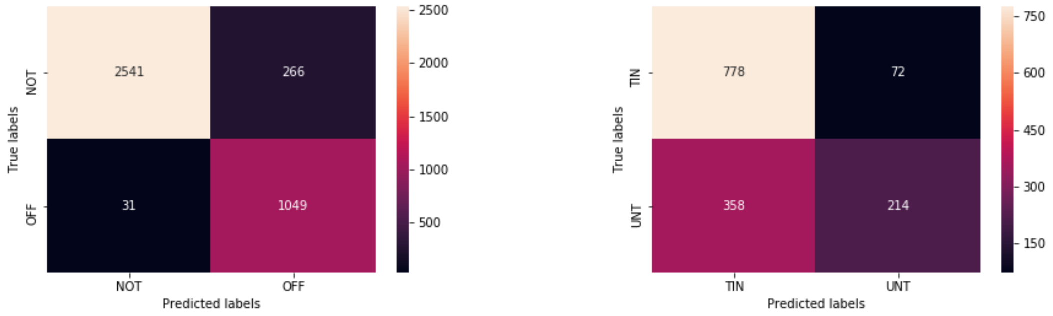
While training models and testing it on OffensEval 2019 data we found that BERT gave best results instead of statistical methods.Table 4 shows results obtained by our methods on test data.

Methods	**Result (MacroF1)**
TF-IDF with Logistic Regression	0.59312
Glove embedding with CNN	0.57280
BERT Classifier	**0.64115**
Top performer	0.74618

Table 4: Results of Sub-task B on Test set

Figure 1 shows the heat-map of predictions by our model that organizers used for evaluation.

[4] http://nlp.stanford.edu/data/glove.6B.zip



(a) Sub-task A: Ensemble

(b) Sub-task B: BERT

Figure 1: Heat map of our predictions used for ranking

5 Conclusion

In this paper we presented methods used to solve two sub-tasks from the OffensEval 2020 track and our results for both tasks. For Sub-task A, feature extraction using TF-IDF and machine learning techniques like Logistic Regression/ Random Forest/SVM, which do not capture polysemy and are unable to deal with OOV words, gave results comparable to SOTA transfer learning technique using BERT, a possible reason for this may be that the models are using the occurrence of particular words to categorize the tweets. For Sub-task B methods that utilize TF-IDF features did not perform as well as transfer learning using BERT, probably because the task was more fine-grained and required the model to capture interactions between words in the tweet and some degree of co-reference resolution. In future we would like to improve result of Sub-task B by extracting some task specific features, also do some more text-prepossessing, include emoticons as they may be used for insulting.

Acknowledgements

We would like to thank the OffensEval 2020 shared task organizers for organizing this interesting shared task and for promptly replying to all our inquiries. We further thank the anonymous reviewers for their feedback.

References

Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, and Manuela Sanguinetti. 2019. SemEval-2019 task 5: Multilingual detection of hate speech against immigrants and women in twitter. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 54–63, Minneapolis, Minnesota, USA, June. Association for Computational Linguistics.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In *Proceedings of the 11th International AAAI Conference on Web and Social Media*, ICWSM '17, pages 512–515.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Sandip Modha, Thomas Mandl, Prasenjit Majumder, and Daksh Patel. 2019. Overview of the HASOC track at FIRE 2019: Hate Speech and Offensive Content Identification in Indo-European Languages. In *Proceedings of the 11th annual meeting of the Forum for Information Retrieval Evaluation*, December.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Amir H. Razavi, Diana Inkpen, Sasha Uritsky, and Stan Matwin. 2010. Offensive language detection using multi-level classification. In *Proceedings of the 23rd Canadian Conference on Advances in Artificial Intelligence*, AI'10, page 16–27, Berlin, Heidelberg. Springer-Verlag.

Sara Rosenthal, Pepa Atanasova, Georgi Karadzhov, Marcos Zampieri, and Preslav Nakov. 2020. A large-scale semi-supervised dataset for offensive language identification. *arXiv preprint arXiv:2004.14454*.

Anna Schmidt and Michael Wiegand. 2017. A survey on hate speech detection using natural language processing. In *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media*, pages 1–10, Valencia, Spain, April. Association for Computational Linguistics.

William Warner and Julia Hirschberg. 2012. Detecting hate speech on the world wide web. In *Proceedings of the Second Workshop on Language in Social Media*, pages 19–26, Montréal, Canada, June. Association for Computational Linguistics.

Kim Yoon. 2014. Convolutional neural networks for sentence classification. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019a. Predicting the type and target of offensive posts in social media. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1415–1420, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019b. Semeval-2019 task 6: Identifying and categorizing offensive language in social media (offenseval). In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 75–86.

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). In *Proceedings of SemEval*.