IIT Varanasi at HASOC 2019 : Hate Speech and Offensive Content Identification in Indo-European Languages

Akanksha Mishra and Sukomal Pal

Department of Computer Science and Engineering,
Indian Institute of Technology (BHU), Varanasi - 221005, India.
{akanksham.rs.cse17,spal}@itbhu.ac.in
https://cse-iitbhu.github.io/irlab/index.html

Abstract. The track aims to develop a system that identifies hate speech and offensive content in the document and further classifies them into hate speech, offensive content, or usage of profane words. Also, it determines whether hate speech is targetted to some individual or a group. We use bidirectional long short term memory along with attention across all languages (English, German, and Hindi) in the track.

Keywords: Hate Speech, Offensive Content, Indo-European Languages, Bidirectional LSTM, Attention

1 Introduction - Task Description

With the increasing usage of the internet or particularly social media like Twitter and Facebook, social media users take advantage of the anonymity provided on such platforms to spread hate or offensive content for an individual or a group. However, such platforms with broad audiences need to prevent abusive behavior of the users, which they may not do in real life. Such activities by the users are increasing day by day, which makes it difficult for the companies to monitor such contents manually. Due to challenges of handling massive multilingual data, we need to develop an automatic way of handling hate speech and offensive content on social media platforms across all languages.
The track [3] focusses on the identification of hate speech and offensive content on social media platforms. It aims to develop the system for three languages, namely English, German, and Hindi. The track consists of three sub-tasks:-

- **Sub-Task 1:** This sub-task is binary classification problem to determine whether the document consists of hate speech, offensive content or profane words, or not. This sub-task classify the document in one of the two classes for all three languages:-

- **Hate and Offensive Content (HOF):** The document or post contains non acceptable languages which may be in the form of hate speech, offensive content or profane words.

- **Non-Hate and Offensive Content (NOT):** The document or post contains no hate speech or offensive content for an individual or a group.

– **Sub-Task 2:** This sub-task is a multi class classification problem to further classify whether the document or post contains hate speech, offensive content or profane words against an individual or a group. In this sub-task, we consider only those documents or posts which are classified as HOF in the first sub-task. This sub-task classify the document or post in one of the classes for all three languages:-

  - **Hate Speech (HATE):** The document or post which contains hate speech against an individual or a group. It may also contain hate speech for a group due to their political opinion, gender, social status, race, religion or any other equivalent reasons.

  - **Offensive (OFFN):** The document or post which makes social users uncomfortable or upset about anything. The content may also be seen as violent acts or insulting an individual.

  - **Profane (PRFN):** The document or post consists of unacceptable languages which may be cursing or usage of swear words. It doesn't include posts which contains abuse or insult of an individual or a group.

– **Sub-Task 3:** This sub-task also considers only those documents or posts which are classified as HOF in sub-task 1. This sub-task is only for English and Hindi data. This sub-task classify the document or post into one of the categories:-

  - **Targeted Insult (TIN):** The document or post which targets an individual, group or others.

  - **Untargeted (UNT):** The document or post which are not targeting any individual, group or others.

2 Related work

Several shared tasks organized related to offensive content identification for one or the other languages. OffensEval [8] task organized in SemEval-2019 focuses on the identification of offensive content, automatic categorization of offense types, and identification of the target of offensive posts. The shared task used

Offensive Language Identification Dataset (OLID) [7] consists of 14,000 English tweets from Twitter and annotated mainly for offensive language.
The GermanEval [6] shared task on the identification of offensive content deals with the German tweets from Twitter. It focuses on two sub-tasks, mainly binary and 4-way classification. For this task, several machine learning (SVM, Logistic Regression, Decision Trees, and Naive Bayes) and neural network (CNN, LSTM and its variants, GRU, and combination of these) based classifiers were used. N-grams and word embeddings are commonly used features, and SVM, RNN, and LSTM are widely used classifiers in the shared task on aggression identification [2] organized as part of First Workshop on Trolling, Aggression, and Cyberbullying (TRAC-1) at COLING 2018.
Survey on automatic detection of hate speech [1] describes different definitions of hate speech from various sources. Most of the studies have considered this as a binary classification problem; however, some have considered this as a multi-class approach. Machine learning, deep learning, and ensemble-based classifiers are generally used. Frequently used features are TF-IDF, bag of words, N-gram, dictionary, types dependencies, word sense disambiguation techniques, word2vec, paragraph2vec, and several others.

3 Methodology

This section describes the model and architecture followed for the identification of hate speech and offensive content in the document and further segregating them as per the relevant category.

Preprocessing of Data: We preprocessed data by removing all the punctuation symbols using a pre-initialized string, string.punctuation available in the string library. We kept words with hashtags; however, we removed the hash symbols. After that, we removed stop words from the data. Further, we removed all usernames, webpage links, and retweet symbol (RT) in case of Twitter data. After removing non-letters from the data, all the tokens are lemmatized. All the data preprocessing steps mentioned here are done for all the languages.

Model Architecture: The model consists of four layers as explained below:-
Word Representation Layer: We represent each word of a sentence of the document or post in the form of dense vectors. We used two different versions[1] of pretrained glove [4] word embedding. One of the pretrained glove embeddings is based on the common crawl which represents each word in the dimension of 300, and the other one is based on Twitter data which represents each word in the dimension of 200.
Bidirectional LSTM layer: In this layer [5], two copies of hidden layer is created. Vector representation of words is fed to the first hidden layer as the input sequence is and reverse copy of the input sequence is fed to the second hidden

[1] https://nlp.stanford.edu/projects/glove/

layer. The results of two hidden layers is concatenated and fed to the next layer.
Attention Layer: This layer helps in focussing on the important terms in the input by iterating over the input trying to focus on relevant information.
Fully connected layer and output layer: In this layer, all the nodes of the previous layer are connected to all the nodes of the next layer.

4 Experiments

4.1 Dataset

The dataset is created from Twitter and Facebook data and shared by the task organizers in a tab-separated format for three languages, namely English, German, and code-mixed Hindi for all sub-tasks. However, there is no sub-task 3 for the German language. All the instances belonging to NOT category in sub-task 1 will further be classified into NONE category in sub-task 2 and sub-task 3. Figure 1 shows detailed statistics about the dataset.

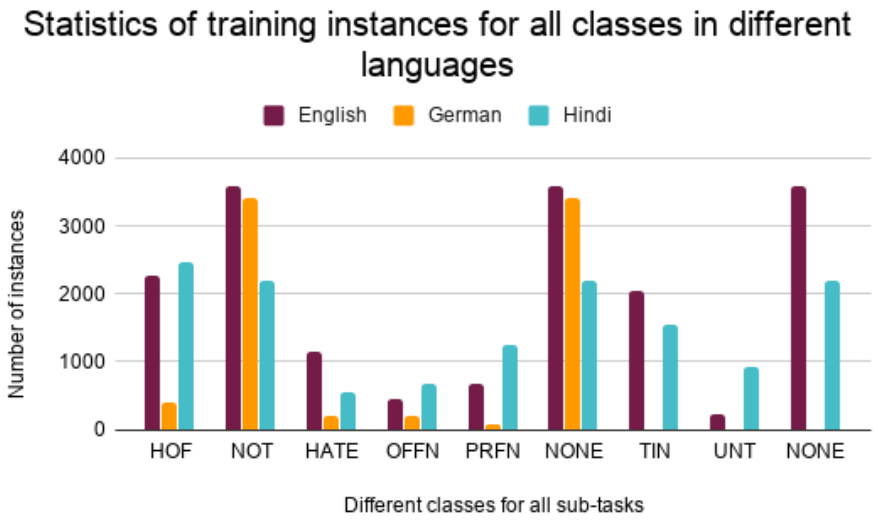

Fig. 1. Statistics of training set

4.2 Experimental Settings

We perform padding of the sentence to make sentences of equal length based on the maximum length of the sentence in the dataset. In the bidirectional LSTM layer, we use recurrent dropout of 0.2 and tanh as an activation function. The

dropout layer, with a rate of 0.3, is used to avoid overfitting of the model. At the output layer, a softmax activation function is used. We use Adam optimizer and categorical cross-entropy loss function for training. Detailed variation of different runs submitted for various sub-tasks for different languages is listed out in table 1.

As mentioned earlier, we have used two different versions of GloVe pre-trained embedding. These versions differ in the sense that they are trained on different datasets. GloVe common crawl embedding is trained by crawling the data on the internet and collecting about 840B tokens, 2.2M vocabulary, and representing each word in a 300-dimensional vector. GloVe twitter pre-trained embedding is trained on twitter dataset and consists of 2B tweets, 27B tokens, 1.2M vocabulary, and representing each word in a 200-dimensional vector. We stopped further iterations as soon as the model starts overfitting. Different epochs for different runs of various sub-tasks is given in table 1. We trained the model with or without NONE category; hence, NONE included indicates whether the model is trained, including the NONE category or not. In case of sub-task 1, there is no NONE category thus it is not applicable (NA) for sub-task 1.

Table 1. Experimental setup for different runs

	Sub-Task	**1**		**2**			**3**		
Language		**Run1**	**Run2**	**Run1**	**Run2**	**Run3**	**Run1**	**Run2**	**Run3**
English	GloVe	Common Crawl	Twitter	Common Crawl	Twitter	Twitter	Common Crawl	Twitter	Twitter
	Dimension	300	200	300	200	200	300	200	200
	#Epochs	5	6	5	6	8	3	4	5
	NONE Included	NA	NA	Yes	Yes	No	Yes	Yes	No
German	GloVe	Common Crawl	Twitter	Common Crawl	Twitter	Twitter	-	-	-
	Dimension	300	200	300	200	200	-	-	-
	#Epochs	4	5	4	5	5			
	NONE Included	NA	NA	Yes	Yes	No	-	-	-
Hindi	GloVe	Common Crawl	Twitter	Common Crawl	Twitter	-	Common Crawl	Twitter	
	Dimension	300	200	300	200	-	300	200	-
	#Epochs	3	4	4	5	-	5	5	-
	NONE Included	NA	NA	Yes	Yes	-	Yes	Yes	-

4.3 Performance Comparison

This section discusses the different metrics evaluated for the track. Detailed results based on macro F1, weighted F1 and accuracy for all subtasks for all languages are given in table 2. For the English language, best performance based on macro F1 score is obtained for Run 2, Run 3, and Run 3 for sub-task 1, sub-task 2, and sub-task 3 respectively. Similarly, in the case of German, Run 2 and Run 3 performs better as compared to other runs for sub-task 1 and sub-task 2 respectively. Moreover, for Hindi language, Run 1, Run 2 and Run 1 outperforms other runs for sub-task 1, sub-task 2 and sub-task 3 respectively.

Table 3, 4 and 5 lists out precision, recall and F1-score for all classes of different languages 'English', 'German' and 'Hindi' respectively.

Table 2. Different metrics for all sub-task for all languages

	Sub-Task	**1**		**2**			**3**		
Language	**Metrics**	**Run1**	**Run2**	**Run1**	**Run2**	**Run3**	**Run1**	**Run2**	**Run3**
English	Macro F1	0.4725	0.4872	0.096	0.07	0.2375	0.1164	0.1169	0.3066
	Weighted F1	0.6137	0.5918	0.0345	0.0268	0.6085	0.0742	0.0745	0.6294
	Accuracy	62	57	10	10	65	21	21	68
German	Macro F1	0.4625	0.5003	0.0677	0.0582	0.2459	-	-	-
	Weighted F1	0.7674	0.7665	0.0188	0.0175	0.7726	-	-	-
	Accuracy	84	81	7	8	84	-	-	-
Hindi	Macro F1	0.7419	0.7062	0.4447	0.4759	-	0.4694	0.4678	-
	Weighted F1	0.7431	0.7068	0.5834	0.5987	-	0.6794	0.6597	-
	Accuracy	74	71	64	62	-	70	69	-

Table 3. Detailed evaluation of different runs for language 'English'

		Run 1			**Run 2**			**Run 3**			
Sub-Task		**Precision**	**Recall**	**F1-score**	**Precision**	**Recall**	**F1-score**	**Precision**	**Recall**	**F1-score**	**Support**
1	HOF	0.21	0.18	0.19	0.24	0.33	0.28	-	-	-	288
	NOT	0.74	0.77	0.75	0.75	0.65	0.70	-	-	-	865
	Macro Avg	0.47	0.47	0.47	0.49	0.49	0.49	-	-	-	1153
	Weighted Avg	0.61	0.62	0.61	0.62	0.57	0.59	-	-	-	1153
2	HATE	0.11	0.77	0.19	0.10	0.81	0.19	0.12	0.06	0.08	124
	NONE	0.00	0.00	0.00	0.00	0.00	0.00	0.74	0.85	0.79	865
	OFFN	0.08	0.07	0.08	0.06	0.03	0.04	0.11	0.01	0.03	71
	PRFN	0.09	0.19	0.12	0.05	0.08	0.06	0.06	0.05	0.06	93
	Macro Avg	0.07	0.26	0.10	0.05	0.23	0.07	0.26	0.24	0.24	1153
	Weighted Avg	0.02	0.10	0.03	0.02	0.10	0.03	0.58	0.65	0.61	1153
3	NONE	0.00	0.00	0.00	0.00	0.00	0.00	0.74	0.88	0.81	865
	TIN	0.21	1.00	0.35	0.21	1.00	0.35	0.17	0.09	0.11	245
	UNT	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	43
	Macro Avg	0.07	0.33	0.12	0.07	0.33	0.12	0.30	0.32	0.31	1153
	Weighted Avg	0.05	0.21	0.07	0.05	0.21	0.07	0.59	0.68	0.63	1153

5 Possible Improvements

Integrating a rule based system with the deep learning based approach may result in improving the accuracy. The system can be used for the identification of hate speech and offensive content on the social media forums.

References

1. Fortuna, P., Nunes, S.: A survey on automatic detection of hate speech in text. ACM Comput. Surv. **51**(4), 85:1–85:30 (Jul 2018). https://doi.org/10.1145/3232676, http://doi.acm.org/10.1145/3232676

Table 4. Detailed evaluation of different runs for language 'German'

		Run 1			Run 2			Run 3			
Sub-Task		**Precision**	**Recall**	**F1-score**	**Precision**	**Recall**	**F1-score**	**Precision**	**Recall**	**F1-score**	**Support**
1	HOF	0.20	0.01	0.01	0.21	0.07	0.11	-	-	-	136
	NOT	0.84	0.99	0.91	0.84	0.95	0.89	-	-	-	714
	Macro Avg	0.52	0.50	0.46	0.53	0.51	0.50	-	-	-	850
	Weighted Avg	0.74	0.84	0.77	0.74	0.81	0.77	-	-	-	850
2	HATE	0.04	0.10	0.06	0.06	0.02	0.04	0.00	0.00	0.00	41
	NONE	0.00	0.00	0.00	0.00	0.00	0.00	0.84	0.99	0.91	714
	OFFN	0.09	0.64	0.17	0.09	0.79	0.17	0.43	0.04	0.07	77
	PRFN	0.03	0.33	0.05	0.02	0.17	0.03	0.00	0.00	0.00	18
	Macro Avg	0.04	0.27	0.07	0.04	0.25	0.06	0.32	0.26	0.25	850
	Weighted Avg	0.01	0.07	0.02	0.01	0.08	0.02	0.75	0.84	0.77	850

Table 5. Detailed evaluation of different runs for language 'Hindi'

		Run 1			Run 2			
Sub-Task		**Precision**	**Recall**	**F1-score**	**Precision**	**Recall**	**F1-score**	**Support**
1	HOF	0.71	0.74	0.73	0.66	0.74	0.70	605
	NOT	0.77	0.74	0.76	0.76	0.68	0.71	713
	Macro Avg	0.74	0.74	0.74	0.71	0.71	0.71	1318
	Weighted Avg	0.74	0.74	0.74	0.71	0.71	0.71	1318
2	HATE	0.35	0.19	0.25	0.30	0.21	0.24	190
	NONE	0.69	0.93	0.79	0.75	0.81	0.78	713
	OFFN	0.30	0.09	0.13	0.46	0.21	0.29	197
	PRFN	0.65	0.56	0.61	0.49	0.75	0.59	218
	Macro Avg	0.50	0.44	0.44	0.50	0.49	0.48	1318
	Weighted Avg	0.57	0.64	0.58	0.60	0.62	0.60	1318
3	NONE	0.71	0.84	0.77	0.67	0.93	0.78	713
	TIN	0.69	0.59	0.63	0.77	0.46	0.57	542
	UNT	0.00	0.00	0.00	0.13	0.03	0.05	63
	Macro Avg	0.47	0.48	0.47	0.52	0.47	0.47	1318
	Weighted Avg	0.67	0.70	0.68	0.69	0.69	0.66	1318

2. Kumar, R., Ojha, A.K., Malmasi, S., Zampieri, M.: Benchmarking aggression identification in social media. In: Proceedings of TRAC (2018)
3. Modha, S., Mandl, T., Majumder, P., Patel, D.: 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 (2019)
4. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: Empirical Methods in Natural Language Processing (EMNLP). pp. 1532–1543 (2014), http://www.aclweb.org/anthology/D14-1162
5. Schuster, M., Paliwal, K.K.: Bidirectional recurrent neural networks. IEEE Transactions on Signal Processing **45**(11), 2673–2681 (1997)
6. Wiegand, M., Siegel, M., Ruppenhofer, J.: Overview of the germeval 2018 shared task on the identification of offensive language (2018)
7. Zampieri, M., Malmasi, S., Nakov, P., Rosenthal, S., Farra, N., Kumar, R.: Predicting the Type and Target of Offensive Posts in Social Media. In: Proceedings of NAACL (2019)
8. Zampieri, M., Malmasi, S., Nakov, P., Rosenthal, S., Farra, N., Kumar, R.: Semeval-2019 task 6: Identifying and categorizing offensive language in social media (offenseval). In: Proceedings of the 13th International Workshop on Semantic Evaluation. pp. 75–86 (2019)