An Efficient BERT Based Approach to Detect Aggression and Misogyny

Sandip Dutta[†], **Utso Majumder**[†], **Sudip Kumar Naskar**[‡]
[†]Department of ETCE, [‡]Department of CSE
Jadavpur University
Kolkata, India
{sandip28dutta, utso1201, sudip.naskar}@gmail.com

Abstract

Social media is bustling with ever growing cases of trolling, aggression and hate. A huge amount of social media data is generated each day which is insurmountable for manual inspection. In this work, we propose an efficient and fast method to detect aggression and misogyny in social media texts. We use data from the Second Workshop on Trolling, Aggression and Cyber Bullying for our task. We employ a BERT based model to augment our data. Next we employ Tf-Idf and XGBoost for detecting aggression and misogyny. Our model achieves 0.73 and 0.85 Weighted F1 Scores on the 2 prediction tasks, which are comparable to the state of the art. However, the training time, model size and resource requirements of our model are drastically lower compared to the state of the art models, making our model useful for fast inference.

1 Introduction

With the rise of social media, there has also been a huge surge of online criticism and trolling. Mitigation of these kinds of online bullying has been an important problem and has been studied for a long time. However, the amount of information generated on social media is too large for humans to wade through. Machine Learning (ML) and Deep Learning (DL) models have achieved great success in the task of text categorization. However, even as larger models with higher accuracy are being built, the importance of the quality of data has still not reduced. Larger models will give sub-optimal results if the training data is not of good quality. But the results of large models are more difficult to interpret. Traditional ML models have the advantage of producing results that are relatively easier to interpret.

In this work, we combine a large deep learning model and traditional ML approaches for the purpose of text classification. Our experimental data is comprised of social media texts from YouTube comments and the task is to predict the presence of aggression and misogyny from the data. First the data is analyzed to reveal the class distribution issue with the data. Next we develop a Bidirectional Encoder Representations from Transformer (BERT) (Devlin et al., 2019) based text data augmentation pipeline to fix the class distribution. This augmented data helps to reduce class imbalance. For the classification part, we develop a Tf-Idf and XGBoost based classification model to classify the text.

Our model[1] achieves a score very close to the state of the art. However, since the augmentation task is essentially a one-time process, our model is simpler and faster. The main classification pipeline does not require high-end computational resources during inference, which makes our model even more efficient.

2 Related Works

So far, significant contributions have been made in the domain of aggression detection in text (Razavi et al., 2010; Kumar et al., 2018, 2020b). Other aspects of of the task have been investigated in areas such as trolling (Cambria et al., 2010; de la Vega and Ng, 2018), misogyny (Anzovino et al., 2018), cyberbullying (Dadvar et al., 2013; Xu et al., 2012), racism (Greevy, 2004), offensive language(Nobata et al., 2016) and hate speech(Djuric et al., 2015; Davidson et al., 2017). All these works are diverse in terms of the target subject they investigate. The works have mainly been conducted on English datasets, however there are some other languages on which works have been reported, for example, in Hindi (Mandla et al., 2021),Spanish (Garibo i Orts, 2019),Chinese (Su et al., 2017), etc.

[1]Source Code : https://github.com/Dutta-SD/AggDetect

Proceedings of the 18th International Conference on Natural Language Processing, pages 493–498
Silchar, India. December 16 - 19, 2021.

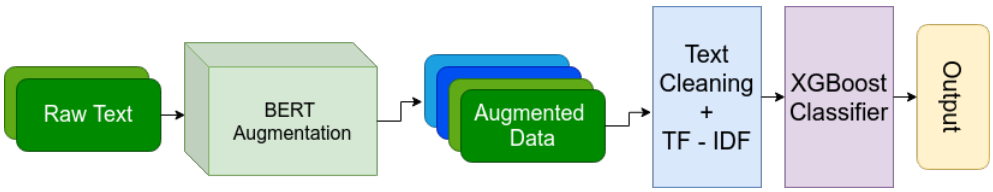

Figure 1: Model Pipeline

3 Model Description

3.1 BERT based Text Data Augmentation

The dataset had a skewed distribution, with the majority of the data belonging to one category. This imbalance can cause models to always predict the majority class. To counter this effect, we employed a BERT based masked word prediction model for data augmentation. The process of augmentation is presented in Algorithm (1).

Algorithm 1 BERT based Data Augmentation - Masked Word Prediction

1: Select one text belonging to the minority class.
2: Filter out the stop words from the text as they do not contribute to the overall sentiment of the text.
3: Randomly select one token and replace it with a special `[MASK]` token. The model fills this token with the word that it deems to be most likely in that context.
4: We take the top `k` predictions for augmenting, where '`k`' is a hyper parameter. We choose k as per the dataset to ensure equal class distribution.
5: Repeat the above steps for other texts in the minority class to make the distribution even for all the classes.

The dataset is augmented once and stored. This essentially being an offline one time process, makes our overall model faster. This process need not be repeated during inference, which makes our net model size smaller.

3.2 Text Data Cleaning

To clean the texts, we use the following steps:

- **Remove Punctuation** – Punctuation marks are replaced with empty string.
- **Remove Stop Words** – We use the default stop words list of `nltk` package and replace the stop words with the null character. We exclude words like "no", "not", "ain't" since they change the semantic nature of a sentence; therefore these words are retained.
- **Stemming** – `Porter Stemmer` from the `nltk` package is used to reduce the vocabulary size and noise.

3.3 Tf–Idf Vectorization

Tf–Idf Vectorization is a scheme that converts a given text into a vector of numerical values. Classification task is made easier by using Tf–Idf, since taking the log of the inverse count of t term reduces the value of unimportant words occurring more frequently in the document. Tf–Idf returns a sparse vector for each text. This helps reduce memory consumption and training time.

3.4 XGBoost Text Classifier

XGBoost (Chen and Guestrin, 2016) or Extreme Gradient Boosting is a very popular algorithm for classification. The model is preferred because of the following reasons.

- **Sparsity Aware Computation** – XGBoost supports computations on sparse data, which helps avoid unnecessary overheads and results in faster training.
- **Approximate Splitting Algorithm** – BERT based data augmentation (cf. Section 3.1) of our pipeline increases the amount of data in our dataset by data augmentation. XGBoost provides approximate methods for splitting the data which saves memory.
- **Regularized Learning Objective** – XGBoost provides regularized objective function which controls model complexity and prevents over-fitting.

4 Experiments and Results

4.1 Dataset

The English dataset (Bhattacharya et al., 2020) we used for our experiments consists of texts scraped from YouTube comments section (Kumar et al.,

Table 1: Experiments on Various Classifiers and Weighted F1 Scores (Other Hyper-parameters are Default Values)

Model	Sub Task A Score	Sub Task B Score
Multi Layer Perceptron Classifier (50 iterations)	0.533	0.720
Random Forest Classifier	0.561	0.740
LinearSVC	0.620	0.773
XGBoost Classifier ($\gamma = 0.2$)	0.729	0.850
Final Model – XGBoost Classifier ($\gamma = 0.1$)	**0.735**	**0.852**

Table 2: Results obtained (Weighted F1 Score)

Team Name	Sub Task A Score	Sub Task B Score
Julian (Risch et al., 2020)	0.802	0.851
abaruah (Baruah et al., 2020)	0.728	0.870
sdhanshu (Safi Samghabadi et al., 2020)	0.759	0.857
Our best model	**0.735**	**0.852**

2020a). The task consists of 2 sub tasks - 'Sub Task A' and 'Sub Task B'.

Sub Task A is to detect the level of aggression in the text and for this sub task the dataset contains 3,375 Non Aggressive (NAG), 453 Covertly Aggressive (CAG, Disguised aggressive content e.g, sarcasm), and 435 Overly Aggressive (OAG, directly aggressive) samples.

Sub Task B addresses misogyny identification and for this the dataset comes with 3,954 Non Misogynistic (NGEN) and 309 Misogynistic (GEN) samples.

An additional validation set was provided with 836 NAG, 117 CAG and 113 OAG samples for Sub Task A and 993 NGEN and 73 GEN samples for Sub Task B. The testset comprised of 690 NAG, 286 OAG and 224 CAG samples for Sub Task A and 1,025 NGEN and 175 GEN samples for Sub Task B.

For both the sub tasks, the ratio of samples indicates a huge data imbalance. This was fixed by using BERT based augmentation and then prediction was done using the model as described in Section 3.2 – 3.4.

4.2 Experimental Setup

Figure 1 shows the entire classification pipeline. Hyper-parameter `k` (cf. Algorithm 1) was set to 2 for our model to augment each data point 2 times. This entire process was repeated 2 times to even out the distribution.

A number of experiments were performed on Random Forest, SVM and Multi Layer Perceptron using scikit-learn (Pedregosa et al., 2011). The BERT masked word prediction augmentation model was moved to the GPU but not finetuned. All hyper-parameters were set as per Devlin et al. (2018). It took about 10 minutes to finish the entire data augmentation process on a Nvidia Tesla K80 GPU.

Next, the Tf-Idf + XgBoost pipeline was trained on an Intel Xeon CPU. The `gamma` hyperparameter of XGBoost model was set to 0.1 for regularization. All other hyper parameters were kept to their default values. The entire process of cleaning, training, validation and predicting on test data took about 2 minutes to complete. We evaluated our models using weighted F1 score.

4.3 Results

The results of our experiments are reported in Table 1. Among our models, XgBoost with $\gamma = 0.1$ yielded the best results for both the subtasks.

Table 2 presents a comparison of our best model with the best models on this task reported in the literature. Risch et al. (2020) obtained the highest scores on Sub Task A and they used bagging of multiple BERT models for prediction. Their entire pipeline required 7 hours to train on a Nvidia 1080 Ti GPU. Baruah et al. (2020) reported the best results on the Sub Task B and they used Transformer based models which demand heavy computational resources. Another top performing model (Safi Samghabadi et al., 2020) took about 6 hours to train on an Nvidia Tesla P40 GPU.

Our model achieves nearly comparable results to the state of the art for this task but requires only a small fraction of the computational resources and time that the top models need. To train the model, the total time required is about 30 minutes, which

Table 3: Predictions by Our Model (Some Texts are Truncated)

Text	Aggression		Misogyny	
	Actual	Predicted	Actual	Predicted
watch sandeep's interview in film companion fat guy with salt and pepper hair (rajiv masand)	NAG	CAG	GEN	NGEN
maria malhotra yes, but i am more of dying on the interviewers face. it looked like a square. I was rolling on the ground	NAG	OAG	NGEN	NGEN
sesh r kotha gulo just heart touching....	NAG	NAG	NGEN	NGEN
where the hell is that sexuall predater cunt. she must be exposed by judiciary, media, parliament	OAG	OAG	NGEN	GEN

Table 4: Model performance (Weighted F1 Score) with different values of hyperparameter `k`

Value of k	Sub Task A Score	Sub Task B Score
0 (No Augmentation)	0.232	0.354
2 (Our Experiment)	0.735	0.852
4	0.551	0.672

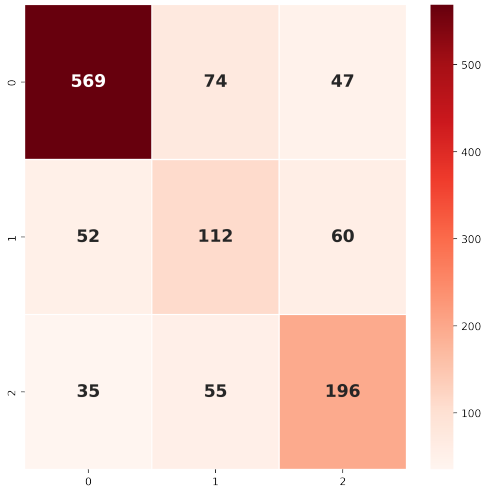

Figure 2: Confusion Matrix Sub Task A (0: NAG, 1: CAG, 2: OAG, Y-Axis: True Label; X-Axis: Predicted Label)

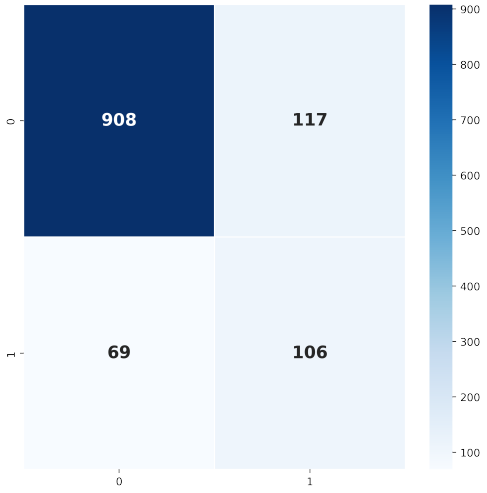

Figure 3: Confusion Matrix Sub Task B (0: NGEN, 1: GEN, Y-Axis: True Label; X-Axis: Predicted Label)

is drastically less compared to the state of the art. The storage requirement for our model is also lower. Our models occupy about few MBs (`.pkl` files), whereas other models take hundreds of MBs.

4.4 Analysis

Table 4 shows the performance of the classifier based on different values of the hyperparameter `k`. It is evident from the table that with no augmentation (`k` = 0), the model performs poorly. With `k` = 2, which makes the class distribution almost even, the model gives the best performance. For higher values of `k`, the class distribution becomes skewed and we get deteriorated performance. Thus best performance is obtained when there is almost equal distribution of all the classes; the model can then learn better representations of the data.

Figure 2 and Figure 3 show the confusion matrix for Sub Task A (Aggression Detection) and Sub Task B (Misogyny Detection), respectively.

We see that our model detects non aggressive and overtly aggressive categories well. However, it shows some problem in detecting aggression in covert forms such as in sarcasm. For misogyny detection, our model shows some errors.

Table 3 shows some predictions from our model where the model predictions match or differ from the actual labels. As has also been reported in Safi Samghabadi et al. (2020), there are some labelling errors in the dataset itself. Our model predictions seem to be more appropriate than ground truth values in those cases.

5 Conclusion

In this work, a fast and efficient pipeline to detect aggression and misogyny from social media texts was developed. The text data is cleaned and analysed and data augmentation was performed using a BERT based model. Various models were experimented upon and Tf-Idf + XgBoost model performs the best among them. Weighted F1 Score of 0.735 and 0.852 was produced by our model, however with drastically reduced computational requirements.

In future, we aim to perform the training on a larger dataset to obtain better results. This would enable us to develop a fast and efficient system to tackle the problem of hatred on social media sites. There is scope of improvement in specific areas like improving the pipeline to detect the subcategory of covertly aggressive better, and to detect misogynous content slightly better. We can extend the reach of the present work to processing multilingual datasets as well.

References

Maria Anzovino, Elisabetta Fersini, and Paolo Rosso. 2018. Automatic identification and classification of misogynistic language on twitter. In *International Conference on Applications of Natural Language to Information Systems*, pages 57–64. Springer.

Arup Baruah, Kaushik Das, Ferdous Barbhuiya, and Kuntal Dey. 2020. Aggression identification in English, Hindi and Bangla text using BERT, RoBERTa and SVM. In *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, pages 76–82, Marseille, France. European Language Resources Association (ELRA).

Shiladitya Bhattacharya, Siddharth Singh, Ritesh Kumar, Akanksha Bansal, Akash Bhagat, Yogesh Dawer, Bornini Lahiri, and Atul Kr. Ojha. 2020. Developing a multilingual annotated corpus of misogyny and aggression. In *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, pages 158–168, Marseille, France. European Language Resources Association (ELRA).

Erik Cambria, Praphul Chandra, Avinash Sharma, and Amir Hussain. 2010. Do not feel the trolls. *ISWC, Shanghai*.

Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794.

Maral Dadvar, Dolf Trieschnigg, Roeland Ordelman, and Franciska de Jong. 2013. Improving cyberbullying detection with user context. In *European Conference on Information Retrieval*, pages 693–696. Springer.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Nemanja Djuric, Jing Zhou, Robin Morris, Mihajlo Grbovic, Vladan Radosavljevic, and Narayan Bhamidipati. 2015. Hate speech detection with comment embeddings. pages 29–30.

Edel Greevy. 2004. *Automatic text categorisation of racist webpages*. Ph.D. thesis, Dublin City University.

Ritesh Kumar, Guggilla Bhanodai, Rajendra Pamula, and Maheshwar Reddy Chennuru. 2018. Trac-1 shared task on aggression identification: Iit (ism)@ coling'18. In *Proceedings of the first workshop on trolling, aggression and cyberbullying (TRAC-2018)*, pages 58–65.

Ritesh Kumar, Atul Kr. Ojha, Bornini Lahiri, Marcos Zampieri, Shervin Malmasi, Vanessa Murdock, and Daniel Kadar, editors. 2020a. *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*. European Language Resources Association (ELRA), Marseille, France.

Ritesh Kumar, Atul Kr Ojha, Shervin Malmasi, and Marcos Zampieri. 2020b. Evaluating aggression identification in social media. In *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, pages 1–5.

Thomas Mandla, Sandip Modha, Gautam Kishore Shahi, Amit Kumar Jaiswal, Durgesh Nandini, Daksh Patel, Prasenjit Majumder, and Johannes Schäfer. 2021. Overview of the hasoc track at fire 2020: Hate speech and offensive content identification in indo-european languages.

Chikashi Nobata, Joel Tetreault, Achint Thomas, Yashar Mehdad, and Yi Chang. 2016. Abusive language detection in online user content. In *Proceedings of the 25th international conference on world wide web*, pages 145–153.

Òscar Garibo i Orts. 2019. Multilingual detection of hate speech against immigrants and women in Twitter at SemEval-2019 task 5: Frequency analysis interpolation for hate in speech detection. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 460–463, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Amir H Razavi, Diana Inkpen, Sasha Uritsky, and Stan Matwin. 2010. Offensive language detection using multi-level classification. In *Canadian Conference on Artificial Intelligence*, pages 16–27. Springer.

Julian Risch, Robin Ruff, and Ralf Krestel. 2020. Offensive language detection explained. In *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying (TRAC@LREC)*, pages 137–143. European Language Resources Association (ELRA).

Niloofar Safi Samghabadi, Parth Patwa, Srinivas PYKL, Prerana Mukherjee, Amitava Das, and Thamar Solorio. 2020. Aggression and misogyny detection using BERT: A multi-task approach. In *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying (TRAC-2020)*, pages 126–131, Marseille, France. European Language Resources Association (ELRA).

Hui-Po Su, Zhen-Jie Huang, Hao-Tsung Chang, and Chuan-Jie Lin. 2017. Rephrasing profanity in Chinese text. In *Proceedings of the First Workshop on Abusive Language Online*, pages 18–24, Vancouver, BC, Canada. Association for Computational Linguistics.

Luis Gerardo Mojica de la Vega and Vincent Ng. 2018. Modeling trolling in social media conversations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Jun-Ming Xu, Kwang-Sung Jun, Xiaojin Zhu, and Amy Bellmore. 2012. Learning from bullying traces in social media. In *Proceedings of the 2012 conference of the North American chapter of the association for computational linguistics: Human language technologies*, pages 656–666.