sdutta at ComMA@ICON: A CNN-LSTM Model For Hate Detection

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

Abstract

In today's world, online activity and social media are facing an upsurge of cases of aggression, gender-biased comments and communal hate. In this shared task, we used a CNN-LSTM hybrid method to detect aggression, misogynistic and communally charged content in social media texts. First, we employ text cleaning and convert the text into word embeddings. Next we proceed to our CNN-LSTM based model to predict the nature of the text. Our model achieves 0.288, 0.279, 0.294 and 0.335 Overall Micro F1 Scores in multilingual, Meitei, Bengali and Hindi datasets, respectively, on the 3 prediction labels.

1 Introduction

Identifying aggressive and abusive atrocities on the internet is an important field of study in today's world. Researchers are striving to develop remedial measures to combat such online content.

In order to efficiently carry out these tasks, the research community have proposed several Machine Learning models, to enhance the efficiency of handling large sets of data and accurately assessing them. The extent of accuracy, however, is a point of concern, since ML models are entirely dependent on large, comprehensive training datasets. Models are prone to poor performance due to lack of properly curated datasets. Conventional models and ensembles are more reliable in these cases, as their data is easily interpreted.

The work is designed to identify objectionable and abusive content on online platforms, as either aggressive, gender based or communally charged. The objective of the model is to demarcate the overlapping aspects of the three types of contents being investigated, and also if this intersectionality could be useful to the task. The task includes multilingual datasets to widen the spectrum of potentially abusive content and to challenge the models.

2 Related Works

Important research contributions have been made in the domain of aggression detection in text (Razavi et al., 2010; Kumar et al., 2018, 2020) and offensive language (Nobata et al., 2016). Gender bias and communally charged content detection have been investigated in research work such as Anzovino et al. (2018), Kiritchenko and Mohammad (2018) and Davidson et al. (2017) respectively. Aforementioned works are different in terms of the target subject they investigate. The NLP research fraternity has analysed the pragmatic and structural features of such forms of hate speech (Djuric et al., 2015; Dadvar et al., 2013) and developing systems that could automatically detect and handle these (Waseem et al., 2017; Zampieri et al., 2019).

Although the most prevalent language for predicting model datasets is English, there are some other languages on which works have been reported, for example, in Hindi (Mandla et al., 2021).

However, on a general note, any predictive model built on historical data may inadvertently inherit human biases based on gender or ethnicity (Sweeney, 2013; Datta et al., 2015; Sun et al., 2019).

3 Model Description

The prediction pipeline is described in Figure (1). The task required us to detect aggression, misogyny and communal hatred in text data in multiple languages. Additional challenge was introduced by code mixing and code switching.

We use a CNN-LSTM based neural network for our prediction task. The steps undertaken are presented here.

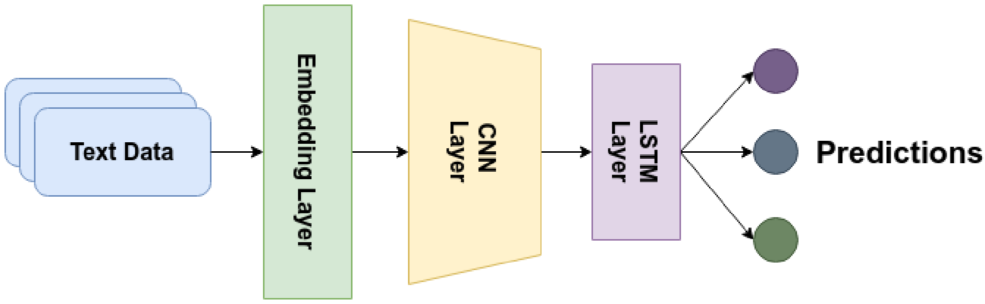
3.1 Text Data Cleaning

The data was cleaned using the following steps:

- **Hashtag, User Handle and URL Removal**: Hashtags and user handles provide redundant

Proceedings of the 18th International Conference on Natural Language Processing: Shared Task on Multilingual Gender Biased and Communal Language Identification, pages 53–57
National Institute of Technology - Silchar, December 16-19, 2021.©2021 NLP Association of India (NLPAI)

Figure 1: Model Diagram



information and these were removed using regular expressions.

- **Punctuation Removal**: Punctuation introduces noise in the text and inflates the vocabulary size. It was also cleaned using regular expressions.

3.2 Word Embedding Vectorization

The word embedding layer converts the sentences into dense word vectors (Mikolov et al., 2013). These provide valuable information to subsequent layers regarding the words.

3.3 CNN - LSTM Model

The combination of CNN and RNN based models (Wang et al., 2016) provides certain advantages. The CNN layer captures global information while LSTM takes care of sequential information.

The CNN layer specializes in identifying informative features from text. The LSTM layer is designed to capture subtle patterns and regularities in sequences. They allow modeling non-markovian dependencies looking at the context window around a focus word, while zooming-in on informative sequential patterns in that window (Goldberg, 2017).

4 Experiments and Results

4.1 Dataset

A multilingual dataset with a total of 12000 samples for training and development and an overall 3000 samples for testing in four Indian languages Meitei, Bangla (Indian variety), Hindi and English, were provided for the task. Each language data was divided into train, validation and test sets. Each data point contains text that is code-mixed with English or their respective varieties of English (i.e. English used in the context of these languages) (Kumar et al., 2021b).

For the task (Kumar et al., 2021a), the contents are categorized broadly into three levels, namely aggression, gender bias and communal bias. The dataset, for each level, is marked at different specific labels or classifications:

- **Level A - Aggression** : This level gives a 3-way classification in between 'Overtly Aggressive'(OAG), 'Covertly Aggressive'(CAG) and 'Non-aggressive'(NAG) text data.

- **Level B - Gender Bias** : At this level the classifier will need to classify the text as 'gendered'(GEN) or 'non-gendered'(NGEN).

- **Level C - Communal Bias** : At the level C, the task is to develop a binary classifier for classifying the text as 'communal' (COM) and 'non-communal'(NCOM).

The task could be approached as three separate classification tasks or a multi-label classification task or a structured classification task. The final submission file contains the labels for each of the three levels as one single predicted tuple.

4.2 Experimental Setup

Figure (1) shows our entire classification model. We create our entire model using Tensorflow (Abadi et al., 2015) and Keras (Chollet et al., 2015). The train, validation and test data was used as is given in (Kumar et al., 2021b).

The random number seed was set to 2833. We selected the maximum sequence length to be of 256 tokens. A vocabulary size of 85000 words was chosen per language for the classification task.

The word embedding dimension was taken to be 50. The Convolution layer gave a 64 dimensional output which was then fed to LSTM layer with `units` hyperparameter set to 100. This output was further fed into the final prediction layer.

Table 1: Predictions by Our Model

Text	Aggression		Misogyny		Communal	
	Actual	Predicted	Actual	Predicted	Actual	Predicted
Chi Chi.A Abar MP. Banglar Lajja	CAG	NAG	GEN	GEN	NCOM	COM
Are kyo apni izzat nilam kar rhi ho	OAG	NAG	GEN	GEN	NCOM	COM
Sunila ekai khangdabi nmaidud khupak thaninge	OAG	CAG	GEN	GEN	NCOM	COM
Aur ye bumbedkar waale bhi bahut madarchod hai	OAG	OAG	GEN	GEN	NCOM	COM

Table 2: Model Scores on Task

Language	Instance F1	Overall Micro F1	Agg. Micro F1	Gen. Micro	Comm. Micro
Multilingual	0.02	0.288	0.376	0.281	0.208
Meitei	0.007	0.279	0.388	0.311	0.138
Bangla	0.006	0.294	0.438	0.339	0.107
Hindi	0.047	0.335	0.44	0.204	0.361

Table 3: Model Scores Comparison on Task

Team Name	Instance F1 Scores			
	Multilingual	Meitei	Bengali	Hindi
Team_BUDDI	0.371	-	-	0.398
Hypers	0.322	0.129	0.223	0.336
Beware Haters	0.294	0.322	0.292	0.289
sdutta	0.02	0.007	0.006	0.047
MUCIC	0.000	0.000	0.000	0.000

We chose Cross Entropy as the loss function for all the 3 prediciton tasks. All other hyperparameters were kept to their default values as is defined in (Chollet et al., 2015).

We trained the model for 12 epochs on a Intel Xeon CPU with Early Stopping enabled. The code[1] was run in the Google Colab environment.

The scores obtained are shown in Table (2).

4.3 Error Analysis

Our model underperforms severely and seems to overfit on certain categories. Some predicitons are shown in Table (1). As is summarized in Table (3), our model provides suboptimal performance in the task compared to other models.

The aggression predictions seem somewhat better than other classes. However, for all the tasks, the performance is not satisfactory.

The main reason for this problem is the huge imbalance in the dataset. The number of data points in one class hugely surpasses other classes. This tends to make the model predict the majority class only. Even enabling early stopping to prevent overfitting gave a poor result due to the high imbalance in this model.

We identified some issues to be cautious of while training on this dataset which are listed below.

- The data is highly imbalanced which can cause severe overfitting. The model will predict only the majority class, which will result in good scores on the train data, but in practice, it will not be beneficial. One can change the loss function to weigh each sample differently during loss calculations. Moreover, a totally different loss function can be used to handle this imbalance.
- There is a lot of code mixing and code switching in this dataset. Code mixing and code switching can inflate the vocabulary size, as there will be multiple representations of the same word. A lot of the texts also contain unicode characters. This further aggravates the problem and can limit the performance of

[1] https://github.com/Dutta-SD/CoMMA_ICON

models in learning good representations of the data. Unicode normalisation can alleviate this problem partially.

These problems severely limit the performance of the model in this dataset. One needs to be aware of these pitfalls before training models.

5 Conclusion

Our model performs moderately on the aggression labels. However, in gender-bias and communally charged labels, it significantly under-performs. Out of the four datasets, the model performs the best on Hindi dataset, but accuracy declines in Meitei and Multilingual datasets.

In the future, we aim to re train the model using sample weighting to obtain better results. We also aim to train using larger models to obtain better results.

References

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

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.

Francois Chollet et al. 2015. Keras.

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.

Amit Datta, Michael Carl Tschantz, and Anupam Datta. 2015. Automated experiments on ad privacy settings: A tale of opacity, choice, and discrimination.

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

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

Yoav Goldberg. 2017. Neural network methods for natural language processing. *Synthesis lectures on human language technologies*, 10(1):1–309.

Svetlana Kiritchenko and Saif M. Mohammad. 2018. Examining gender and race bias in two hundred sentiment analysis systems.

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, Bornini Lahiri, Akanksha Bansal, Enakshi Nandi, Laishram Niranjana Devi, Shyam Ratan, Siddharth Singh, Akash Bhagat, and Yogesh Dawer. 2021a. Comma@icon: Multilingual gender biased and communal language identification task at icon-2021. In *Proceedings of the 18th International Conference on Natural Language Processing (ICON): COMMA@ICON 2021 Shared Task*, Silchar, India. NLP Association of India (NLPAI).

Ritesh Kumar, Enakshi Nandi, Laishram Niranjana Devi, Shyam Ratan, Siddharth Singh, Akash Bhagat, Yogesh Dawer, and Akanksha Bansal. 2021b. The comma dataset v0.2: Annotating aggression and bias in multilingual social media discourse.

Ritesh Kumar, Atul Kr Ojha, Shervin Malmasi, and Marcos Zampieri. 2020. 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.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space.

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.

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.

Tony Sun, Andrew Gaut, Shirlyn Tang, Yuxin Huang, Mai ElSherief, Jieyu Zhao, Diba Mirza, Elizabeth Belding, Kai-Wei Chang, and William Yang Wang. 2019. Mitigating gender bias in natural language processing: Literature review.

Latanya Sweeney. 2013. Discrimination in online ad delivery. *Communications of the ACM*, 56(5):44–54.

Jiang Wang, Yi Yang, Junhua Mao, Zhiheng Huang, Chang Huang, and Wei Xu. 2016. Cnn-rnn: A unified framework for multi-label image classification.

Zeerak Waseem, Thomas Davidson, Dana Warmsley, and Ingmar Weber. 2017. Understanding abuse: A typology of abusive language detection subtasks.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019. Predicting the type and target of offensive posts in social media.