Hate-Alert@DravidianLangTech-EACL2021: Ensembling strategies for Transformer-based Offensive language Detection

Debjoy Saha , Naman Paharia , Debajit Chakraborty,
Punyajoy Saha, Animesh Mukherjee
Indian Institute of Technology, Kharagpur, India
sahadebjoy10@iitkgp.ac.in, namanpaharia.27@gmail.com, debajit15@iitkgp.ac.in
punyajoys@iitkgp.ac.in, animeshm@cse.iitkgp.ac.in

Abstract

Social media often acts as breeding grounds for different forms of offensive content. For low resource languages like Tamil, the situation is more complex due to the poor performance of multilingual or language-specific models and lack of proper benchmark datasets. Based on this shared task "Offensive Language Identification in Dravidian Languages" at EACL 2021, we present an exhaustive exploration of different transformer models, We also provide a genetic algorithm technique for ensembling different models. Our ensembled models trained separately for each language secured the **first** position in Tamil, the **second** position in Kannada, and the **first** position in Malayalam sub-tasks. The models and codes are provided[1].

1 Introduction

Social media platforms have become a prominent way of communication, be it for acquiring information or promotion of business[2]. While we cannot deny the positives, there are some ill consequences of social media as well (Thavareesan and Mahesan, 2019, 2020a,b). Bad actors often use different social media platforms by posting tweets/comments that insult others by targeting their culture and beliefs. In social media, such posts are collectively known as offensive language (Chen et al., 2012). To reduce offensive content, different social media platforms like YouTube have laid down moderation policies and employ moderators for maintaining civility in their platforms. Recently, the moderators are finding it difficult to continue the moderation due to the ever-increasing volume of offensive data. Hence, platforms are looking toward automatic moderation systems. For instance, Facebook is proactively removing a large part of the harmful content from its platform, even before the users report them. There are concerns by different policy-makers that these automatic moderation systems may be erroneous[3]. Situation for countries like India is more complex, as courts often face dilemma while interpreting harmful content and social platforms like Facebook are often unable to take necessary actions[4]. Hence, more effort is required to detect and mitigate offensive language in the Indian social media.

Recently, different shared tasks like HASOC 2019[5] have been launched to understand hateful and offensive language in Indian context but it is limited to Hindi and English mostly. A subtask in HASOC 2020[6] (Chakravarthi et al., 2020c; Mandl et al., 2020) aimed to detect offensive posts in a code-mixed dataset (Jose et al., 2020; Priyadharshini et al., 2020). Extending that task further, the organisers of this shared task have put together a large dataset of 43919, 7772, 20010 posts in three Dravidian languages – Tamil, Kannada, Malayalam respectively, to further advance research on offensive posts in these languages (Chakravarthi and Muralidaran, 2021; Chakravarthi et al., 2021a; Suryawanshi and Chakravarthi, 2021). In this paper, we aim to build algorithmic systems that can detect offensive posts. Contributions of our paper are two-fold. First, we investigate how the current state-of-the-art multilingual language models perform on these languages. Second, we demonstrate how we can use ensembling techniques to improve our classification performance.

[1] `https://github.com/Debjoy10/Hate-Alert-DravidianLangTech`
[2] https://www.webfx.com
[3] https://www.forbes.com/
[4] https://www.npr.org
[5] `https://hasocfire.github.io/hasoc/2019/index.html`
[6] Dravidian-codemix-fire2020

Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages, pages 270–276
April 20, 2021 ©2021 Association for Computational Linguistics

Classifiers	Tamil			Kannada			Malayalam		
	Train	Dev	Test	Train	Dev	Test	Train	Dev	Test
Not-offensive	25425	3193	3190	3544	426	427	14153	1779	1765
Offensive-untargeted	2906	356	368	212	33	33	191	20	29
Offensive-targeted-individual	2343	307	315	487	66	75	239	24	27
Offensive-targeted-group	2557	295	288	329	45	44	140	13	23
Offensive-targeted-other	454	65	71	123	16	14	-	-	-
Not-in-indented-language	1454	172	160	1522	191	185	1287	163	157
Total	35139	4388	4392	6217	777	778	16010	1999	2001

Table 1: Dataset statistics for languages Tamil, Kannada and Malayalam for all splits Train, Dev and Test

2 Related Work

Offensive language has been studied in the research community for a long time, One of the earliest studies (Chen et al., 2012) tried to detect offensive users by using lexical syntactic features generated from their posts. Although, they provided an efficient framework for future research, their dataset was small for any conclusive evidence. Davidson et al. curated one of the largest dataset containing both offensive and hate speech. The authors found that one of the issues with their best performing models was that they could not distinguish between hate and offensive posts. In order to mitigate this, subsequent research (Pitsilis et al., 2018) tried to use deep learning to identify offensive language in English and found that recurrent neural networks (RNNs) are quite effective this task. Recently, the research community has begun to focus on offensive language detection in other low resourced languages like Danish (Sigurbergsson and Derczynski, 2019), Greek (Pitenis et al., 2020) and Turkish (Çöltekin, 2020). In the Indian context, the HASOC 2019 shared task (Mandl et al., 2019) was a significant effort in that direction, where the authors developed a dataset of hate and offensive posts in Hindi and English. The best model in this competition used an ensemble of multilingual transformers, fine-tuned on the given dataset (Mishra and Mishra, 2019). In Dravidian part of HASOC 2020, Renjit and Idicula used an ensemble of deep learning and simple neural networks to identify offensive posts in Manglish (Malayalam in roman font).

Transformer based language models are becoming quite popular in the past few years. Recently, different multilingual models like XLM-RoBERTa (Conneau et al., 2019), multilingual-BERT (Devlin et al., 2018), MuRIL [7] and Indic-BERT (Conneau et al., 2019) have been introduced to facilitate NLP research in different languages. Often in different machine learning pipeline, ensembling different classification outcomes helps in getting better performance (Alonso et al., 2020; Renjit and Idicula, 2020; Mishra and Mishra, 2019). Rather than selecting the models for the ensemble manually, genetic algorithms (GA) are used to optimise the weights of different classifiers, to improve the ensemble performance on the development set. GA-based ensembling techniques have previously been used in the hate speech domain for architecture and hyperparamter search Madukwe et al. (2020).

[7]https://tfhub.dev/google/MuRIL/1

3 Dataset description

The shared task on Offensive Language Identification in Dravidian Languages-EACL 2021 (Chakravarthi et al., 2021b) is based on a post classification problem with an aim to moderate and minimise offensive content in social media. The objective of the shared task is to develop methodology and language models for code-mixed data in low-resource languages, as models trained on monolingual data fail to comprehend the semantic complexity of a code-mixed dataset.

Dataset: The Dravidian offensive code-mixed language dataset is available for Tamil (Chakravarthi et al., 2020b), Kannada (Hande et al., 2020) and Malayalam (Chakravarthi et al., 2020a). The data provided is scraped entirely from the YouTube comments of a multilingual community where code-mixing is a prevalent phenomenon. The dataset contains rows of text and the corresponding labels from the list not-offensive, offensive-untargeted, offensive-targeted-individual, offensive-targeted-group, offensive-targeted-other, or not-in-indented-language. Final evaluation score was calculated using weighted F1-score metric on a held-out test dataset.

We present the dataset statistics in Table 1. Please note that the Malayalam split of the dataset contained no instances of 'Offensive-targeted-other' label, so classification is done using 5 labels only, instead of the original six labels. In order to understand the amount of misspelt and code-mixed words, we compare with an existing pure language vocabulary available in the Dakshina dataset (Roark et al., 2020). We find the proportion of out-of-vocabulary (OOV) words (including code-mixed, English and misspelt words) in the dataset as 85.55%, 84.23% and 83.03% in Tamil, Malayalam and Kannada respectively.

4 Methodology

In this section, we discuss the different parts of the pipeline that we followed to detect offensive posts in this dataset.

4.1 Machine learning models

As a part of our initial experiments, we used several machine learning models to establish a baseline performance. We employed random forests, logistic regression and trained them with TF-IDF vectors. The best results were obtained on ExtraTrees Classifier (Geurts et al., 2006) with 0.70, 0.63 and 0.95 weighted F1-scores on Tamil, Kannada and Malayalam respectively. As we will notice further, these performances were lower than single transformer based model. Hence, the simple machine learning models were not used in the subsequent analysis.

4.2 Transformer models

One of the issues with simple machine learning models is the inability to learn the context of a word based on its neighbourhood. Recent transformer based architectures are capable of capturing this context, as established by their superior performance in different downstream tasks. For our purpose, we fine-tuned different state-of-the-art multilingual BERT models on the given datasets. This includes XLM-RoBERTa (Conneau et al., 2019), multilingual-BERT (Devlin et al., 2018)[8], Indic BERT and MuRIL[9]. We also pretrain XLM-Roberta-Base on the target dataset for 20 epochs using Masked Language Modeling, to capture the semantics of the code-mixed corpus. This additional pretrained BERT model was also used for fine-tuning. In addition, all models were fine-tuned separately using unweighted and weighted cross-entropy loss functions (Mannor et al., 2005). For training, we use HuggingFace (Wolf et al., 2019) with PyTorch (Paszke et al., 2019). We use the Adam adaptive optimizer (Loshchilov and Hutter, 2019) with an initial learning rate of 1e-5. Training is stopped by early stopping if macro-F1 score of the development split of the dataset does not increase for 5 epochs.

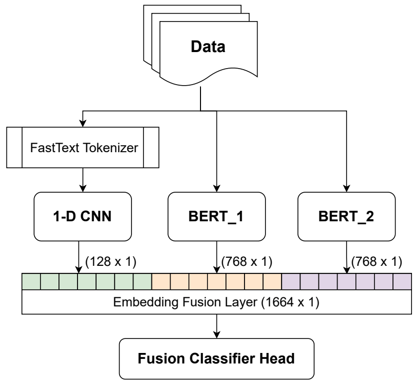

Figure 1: Our fusion model architecture for two BERT models. Note that 768×1 embedding sizes are used for the BERT-base models. Embeddings size of 1024×1 is used for BERT-large models.

4.3 Fusion models

Convolution neural networks are able to capture neighbourhood information more effectively. One of the previous state-of-the-art model to detect hate speech was CNN-GRU (Zhang et al., 2018), We propose a new BERT-CNN fusion classifier where we train a single classification head on the concatenated embeddings from different BERT and CNN models. BERT models were initialised with the fine-tuned weights in the former section and the weights were frozen. The number of BERT models in a single fusion model was kept flexible with maximum number of models fixed to three, due to memory limitation. For the CNN part, we use the 128-dim final layer embeddings from CNN models trained on skip-gram word vectors using FastText (Bojanowski et al., 2017)[10]. FastText vectors worked the best among other word embeddings like LASER (Artetxe and Schwenk, 2019). For the fusion classifier head, we use a feed-forward neural

[8]XLM-Roberta-Base, 270M parameters, trained on data from 100 languages; Multilingual-BERT-Base, 179M parameters, trained on data from the top 104 languages.

[9]Originally released by Google, MuRIL (Multilingual Representations for Indian Languages) is a BERT model pre-trained on code-mixed data from 17 Indian languages https://huggingface.co/simran-kh/muril-cased-temp

[10]https://fasttext.cc/docs/en/unsupervised-tutorial.html

Classifiers	Tamil		Kannada		Malayalam	
	Dev	Test	Dev	Test	Dev	Test
XLMR-base (A)	0.77	0.76	0.69	0.70	0.97	0.96
XLMR-large	**0.78**	**0.77**	0.69	0.71	**0.97**	**0.97**
XLMR-C (B)	0.76	0.76	**0.70**	**0.73**	**0.97**	**0.97**
mBERT-base (C)	0.73	0.72	0.69	0.70	0.97	0.96
IndicBERT	0.73	0.71	0.62	0.66	0.96	0.95
MuRIL	0.75	0.74	0.67	0.67	0.96	0.96
DistilBERT	0.74	0.74	0.68	0.69	0.96	0.95
CNN	0.71	0.70	0.60	0.61	0.95	0.95
CNN + A + C	0.78	0.76	0.71	0.70	**0.97**	**0.97**
CNN + A + B	**0.78**	**0.77**	0.71	0.71	**0.97**	**0.97**
CNN + B + C	0.77	0.76	**0.71**	**0.72**	**0.97**	**0.97**

Table 2: Weighted F1-score comparison for transformer, CNN and fusion models on Dev and Test splits (XLMR-C refers to the custom-pretrained XLM-Roberta-Base Classifier).

ModelSets	Tamil		Kannada		Malayalam	
	Dev	Test	Dev	Test	Dev	Test
Trans	0.80	0.78	0.74	0.73	0.98	0.97
F-models	0.79	0.77	0.73	0.73	0.98	0.97
R-models	0.79	0.78	0.75	0.74	0.97	0.97
Overall	**0.80**	**0.78**	**0.75**	**0.74**	**0.98**	**0.97**

Table 3: Weighted-F1 score comparison for GA-weighted ensemble for transformers category, Fusion models(F-models) and Random seed models(R-models)

network having four layers with batch normalization (Ioffe and Szegedy, 2015) and dropout (Srivastava et al., 2014) on the final layer. The predictions were generated from a softmax layer of dimension equal to the number of classes. We present the details of the pipeline in Figure 1.

4.4 Ensembling strategies

Ensemble of different models often turn out better predictors than using a single classifier. Standard prediction averaging ensembles will not perform well, since some models might be weak predictors in the mix of different models. One of the strategies to reduce the influence of weak models is using weights for different models based on their performance. Genetic algorithm (GA) based techniques (Madukwe et al., 2020) are one of the popular ways to set the weights of different models in an ensemble. Our approach is similar to that introduced in Zhou et al. (2001), except that instead of selecting the models with the highest weights for the final ensemble, we directly use the weights to compute the weighted average ensemble.

Another issue with neural networks is the performance is dependent on the initial random seeds. With pretrained models like BERT, most of the weights are fixed only in the final layer (classification head). Past research (McCoy et al., 2020) has shown that even the initialisation of this final layer can affect the final performance by large margins. Hence, we take 10 different random seeds to train the models and then pass all the models to the GA pipeline. We perform this operation for two of the best models in Table 2.

5 Results and conclusion

We observe that among the individual transformer models, the best performance is obtained using XLM-RoBERTa-large (XLMR-large) in the Tamil dataset and Custom XLM-RoBERTa-base (XLMR-C) in the Kannada dataset. For Malayalam dataset, both the former models perform similarly. The higher performance of XLM-RoBERTa (Artetxe and Schwenk, 2019) models can be attributed to the fact that they are pretrained using a parallel corpus (same corpus in different languages). Further pretraining with our dataset helps in further improvement of the performance in the Kannada dataset. We did not use the XLM-R large model further due to limited GPU space. Next, we note the performance of the fusion models, which perform almost similarly across different combinations.

When we use different random seeds, the performance of multilingual BERT models varied around 2-3% across different languages. For XLM-RoBERTa models the variation was more (around 15-20%). Table 3 shows the ensemble performance of different categories of models and all the models combined. GA-optimised weighted ensembling improves the final model scores by a 1-2% across datasets of different languages which finally helped us to rank higher in the leader board.

In this shared task, we evaluated different transformer based architectures and introduced different ensembling strategies. We found that XLM-RoBERTa models usually perform better than other transformer models, although their performance is highly variable across different random seeds. GA based ensembling helps us in further improving the models. Our immediate next step will be to investigate the reason behind lower performance of IndicBERT and MuRIL which are specifically trained for Indian context.

References

Pedro Alonso, Rajkumar Saini, and György Kovács. 2020. Hate speech detection using transformer ensembles on the hasoc dataset. In *International Conference on Speech and Computer*, pages 13–21. Springer.

Mikel Artetxe and Holger Schwenk. 2019. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Bharathi Raja Chakravarthi, Navya Jose, Shardul Suryawanshi, Elizabeth Sherly, and John Philip McCrae. 2020a. A sentiment analysis dataset for code-mixed Malayalam-English. In *Proceedings of the 1st Joint Workshop on Spoken Language Technologies for Under-resourced languages (SLTU) and Collaboration and Computing for Under-Resourced Languages (CCURL)*, pages 177–184, Marseille, France. European Language Resources association.

Bharathi Raja Chakravarthi and Vigneshwaran Muralidaran. 2021. Findings of the shared task on Hope Speech Detection for Equality, Diversity, and Inclusion. In *Proceedings of the First Workshop on Language Technology for Equality, Diversity and Inclusion*. Association for Computational Linguistics.

Bharathi Raja Chakravarthi, Vigneshwaran Muralidaran, Ruba Priyadharshini, and John Philip McCrae. 2020b. Corpus creation for sentiment analysis in code-mixed Tamil-English text. In *Proceedings of the 1st Joint Workshop on Spoken Language Technologies for Under-resourced languages (SLTU) and Collaboration and Computing for Under-Resourced Languages (CCURL)*, pages 202–210, Marseille, France. European Language Resources association.

Bharathi Raja Chakravarthi, Ruba Priyadharshini, Shubhanker Banerjee, Richard Saldhana, John Philip McCrae, Anand Kumar M, Parameswari Krishnamurthy, and Melvin Johnson. 2021a. Findings of the shared task on Machine Translation in Dravidian languages. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*. Association for Computational Linguistics.

Bharathi Raja Chakravarthi, Ruba Priyadharshini, Navya Jose, Anand Kumar M, Thomas Mandl, Prasanna Kumar Kumaresan, Rahul Ponnusamy, Hariharan V, Elizabeth Sherly, and John Philip McCrae. 2021b. Findings of the shared task on Offensive Language Identification in Tamil, Malayalam, and Kannada. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*. Association for Computational Linguistics.

Bharathi Raja Chakravarthi, Ruba Priyadharshini, Vigneshwaran Muralidaran, Shardul Suryawanshi, Navya Jose, Elizabeth Sherly, and John P. McCrae. 2020c. Overview of the Track on Sentiment Analysis for Dravidian Languages in Code-Mixed Text. In *Forum for Information Retrieval Evaluation*, FIRE 2020, page 21–24, New York, NY, USA. Association for Computing Machinery.

Y. Chen, Y. Zhou, S. Zhu, and H. Xu. 2012. Detecting offensive language in social media to protect adolescent online safety. In *2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Confernece on Social Computing*, pages 71–80.

Ying Chen, Yilu Zhou, Sencun Zhu, and Heng Xu. 2012. Detecting offensive language in social media to protect adolescent online safety. In *2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Confernece on Social Computing*, pages 71–80. IEEE.

Çağrı Çöltekin. 2020. A corpus of turkish offensive language on social media. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 6174–6184.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 11.

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*.

Pierre Geurts, Damien Ernst, and Louis Wehenkel. 2006. Extremely randomized trees. *Mach. Learn.*, 63(1):3–42.

Adeep Hande, Ruba Priyadharshini, and Bharathi Raja Chakravarthi. 2020. KanCMD: Kannada CodeMixed dataset for sentiment analysis and offensive language detection. In *Proceedings of the Third Workshop on Computational Modeling of People's Opinions, Personality, and Emotion's in Social Media*, pages 54–63, Barcelona, Spain (Online). Association for Computational Linguistics.

Sergey Ioffe and Christian Szegedy. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR.

Navya Jose, Bharathi Raja Chakravarthi, Shardul Suryawanshi, Elizabeth Sherly, and John P. McCrae. 2020. A Survey of Current Datasets for Code-Switching Research. In *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)*, pages 136–141.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization.

K. J. Madukwe, X. Gao, and B. Xue. 2020. A ga-based approach to fine-tuning bert for hate speech detection. In *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 2821–2828.

Thomas Mandl, Sandip Modha, Anand Kumar M, and Bharathi Raja Chakravarthi. 2020. Overview of the HASOC Track at FIRE 2020: Hate Speech and Offensive Language Identification in Tamil, Malayalam, Hindi, English and German. In *Forum for Information Retrieval Evaluation*, FIRE 2020, page 29–32, New York, NY, USA. Association for Computing Machinery.

Thomas Mandl, Sandip Modha, Prasenjit Majumder, Daksh Patel, Mohana Dave, Chintak Mandlia, and Aditya 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 forum for information retrieval evaluation*, pages 14–17.

Shie Mannor, Dori Peleg, and Reuven Rubinstein. 2005. The cross entropy method for classification. In *Proceedings of the 22nd International Conference on Machine Learning*, ICML '05, page 561–568, New York, NY, USA. Association for Computing Machinery.

R Thomas McCoy, Junghyun Min, and Tal Linzen. 2020. Berts of a feather do not generalize together: Large variability in generalization across models with similar test set performance. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 217–227.

Shubhanshu Mishra and Sudhanshu Mishra. 2019. 3idiots at hasoc 2019: Fine-tuning transformer neural networks for hate speech identification in indo-european languages. In *FIRE (Working Notes)*, pages 208–213.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *arXiv preprint arXiv:1912.01703*.

Zeses Pitenis, Marcos Zampieri, and Tharindu Ranasinghe. 2020. Offensive language identification in greek. *arXiv preprint arXiv:2003.07459*.

Georgios K Pitsilis, Heri Ramampiaro, and Helge Langseth. 2018. Detecting offensive language in tweets using deep learning. *arXiv preprint arXiv:1801.04433*.

Ruba Priyadharshini, Bharathi Raja Chakravarthi, Mani Vegupatti, and John P. McCrae. 2020. Named Entity Recognition for Code-Mixed Indian Corpus using Meta Embedding. In *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)*, pages 68–72.

Sara Renjit and Sumam Mary Idicula. 2020. Cusatnlp@hasoc-dravidian-codemix-fire2020:identifying offensive language from manglishtweets.

Brian Roark, Lawrence Wolf-Sonkin, Christo Kirov, Sabrina J. Mielke, Cibu Johny, Isin Demirsahin, and Keith Hall. 2020. Processing south asian languages written in the latin script: the dakshina dataset.

Gudbjartur Ingi Sigurbergsson and Leon Derczynski. 2019. Offensive language and hate speech detection for danish. *arXiv preprint arXiv:1908.04531*.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958.

Shardul Suryawanshi and Bharathi Raja Chakravarthi. 2021. Findings of the shared task on Troll Meme Classification in Tamil. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*. Association for Computational Linguistics.

Sajeetha Thavareesan and Sinnathamby Mahesan. 2019. Sentiment Analysis in Tamil Texts: A Study on Machine Learning Techniques and Feature Representation. In *2019 14th Conference on Industrial and Information Systems (ICIIS)*, pages 320–325.

Sajeetha Thavareesan and Sinnathamby Mahesan. 2020a. Sentiment Lexicon Expansion using Word2vec and fastText for Sentiment Prediction in Tamil texts. In *2020 Moratuwa Engineering Research Conference (MERCon)*, pages 272–276.

Sajeetha Thavareesan and Sinnathamby Mahesan. 2020b. Word embedding-based Part of Speech tagging in Tamil texts. In *2020 IEEE 15th International Conference on Industrial and Information Systems (ICIIS)*, pages 478–482.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Ziqi Zhang, D. Robinson, and Jonathan Tepper. 2018. Detecting hate speech on twitter using a convolution-gru based deep neural network.

Zhi-Hua Zhou, Jian-Xin Wu, Yuan Jiang, and Shi-Fu Chen. 2001. Genetic algorithm based selective neural network ensemble. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence - Volume 2*, IJCAI'01, page 797–802, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.