kubapok@LT-EDI 2024: Evaluating Transformer Models for Hate Speech Detection in Tamil

Jakub Pokrywka and **Krzysztof Jassem**
Adam Mickiewicz University
Faculty of Mathematics and Computer Science
{firstname.lastname}@amu.edu.pl

Abstract

We describe the second-place submission for the shared task organized at the Fourth Workshop on Language Technology for Equality, Diversity, and Inclusion (LT-EDI-2024). The task focuses on detecting caste/migration hate speech in Tamil. The included texts involve the Tamil language in both Tamil script and transliterated into Latin script, with some texts also in English. Considering different scripts, we examined the performance of 12 transformer language models on the dev set. Our analysis revealed that for the whole dataset, the model google/muril-large-cased performs the best. We used an ensemble of several models for the final challenge submission, achieving 0.81 for the test dataset.

1 Introduction

This paper deals with hate speech detection in the Tamil language, which is an official language in Sri Lanka and Singapore. It is also the official language of the Indian state of Tamil Nadu and the union territory of Puducherry. The language is spoken by groups of citizens of Malaysia, Mauritius, Fiji, and South Africa. The current number of Tamil speakers is estimated at 75 million. The Tamil language belongs to the family of 24 Dravidian languages, spoken by approximately 250 million people. The Tamil alphabet consists of 246 characters: 12 vowels, 18 consonants, and 216 vowel–consonant combinations. Being spoken in India, a country with a caste-based social system, the Tamil language may suffer from hate speech referring not only to religion, ethnicity, gender, sexual orientation, or political affiliation, but also to caste and migration.

In this paper, we describe a submission to caste/migration hate speech detection task organized at LT-EDI-2024 (Rajiakodi et al., 2024). Our approach, which relied on an ensemble of several models, achieved second place in the competition, with a 0.81 F1-score. Besides the research related strictly to the contest, we examine the performance of 12 up-to-date models that are most suitable for this task. We evaluate each model's performance separately on Tamil, Latin, and combined scripts, finding that the model's performance is different based on the script.

2 Related work

A contest on hate speech detection in the Dravidian languages, called HASOC 2021, was organized in 2021 (Chakravarthi et al., 2021). The data were collected from YouTube comments and posts. The contest consisted of two tasks differing in the nature of the data: the first task was based on Tamil only, while the second task was based on a data set combining Tamil and Malayalam. The winning solution for Task 1 achieved a 0.86 F1-score. The winning solution in the Tamil track for Task 2 achieved a 0.68 F1-score. The HASOC 2021 shared task gave rise to a number of papers, such as Rajalakshmi et al. (2023); Pradeep et al. (2021) and Subramanian et al. (2022). The papers report submissions with F1-scores ranging from 0.66 to 0.84.

3 Caste/Migration Hate Speech Detection Challenge

The Caste/Migration Hate Speech Detection task is a part of the Fourth Workshop on Language Technology for Equality, Diversity, and Inclusion (LT-EDI-2024) (Rajiakodi et al., 2024). The main objective of the challenge is to develop a text classifier in the Tamil language that can determine whether a given social media text contains hate speech related to caste or migration. The competition's evaluation metric is the macro average F1-score, and the participants are provided with training (train) and development (dev) datasets.

Proceedings of the Fourth Workshop on Language Technology for Equality, Diversity, Inclusion, pages 196–199
March 21, 2024 ©2024 Association for Computational Linguistics

4 Dataset analysis

We examined the train and dev datasets and discovered that the texts could be classified into three primary categories:

1. Tamil language written in Tamil script
2. Tamil language transliterated into Latin script
3. English language

There are also comments that may contain a mixture of Tamil and English language. We observed that in both the training and development datasets, 51% of the texts are in Tamil script, and the remaining 49% in Latin script. The test dataset has an even split of 50% for both Tamil and Latin scripts. If more than half of the characters in a comment are non-Latin, we classify the comment as Tamil script. Table 1 shows the number of samples labeled as caste/migration hate speech. The average comment lengths in characters for the train, dev, and test datasets are 133, 134, and 129, respectively.

Dataset	HS	not-HS	Overall
train	2052	3303	5355
dev	351	594	945
test	-	-	1575

Table 1: Breakdown of datasets by label. HS stands for caste/migration hate speech comments, and not-HS stands for comments with a lack of such hate speech.

5 Evaluation of transformer models for Tamil hate speech

We utilized HuggingFace's Transformers library to fine-tune the selected encoder language models. We used the standard Trainer class and set the learning rate to 2e-5, batch size to 16, weight decay to 0.01, and warmup ratio to 0.1. We trained for 30 epochs and calculated the F1-score on the dev set after each epoch. The best model based on this metric was selected for evaluation. We used two A100 80GB model cards and tested the following HuggingFace model cards:

- distilbert-base-uncased (eng) (Sanh et al., 2019)
- bert-base-cased (eng) (Devlin et al., 2018)
- roberta-base (eng) (Liu et al., 2019)
- roberta-large (eng) (Liu et al., 2019)
- bert-base-multilingual-cased (eng) (Devlin et al., 2018)
- xlm-roberta-base (multilingual) (Conneau et al., 2019)
- xlm-roberta-large (multilingual) (Conneau et al., 2019)
- microsoft/mdeberta-v3-base (multilingual) (He et al., 2021)
- monsoon-nlp/hindi-bert (hindi) (mon)
- l3cube-pune/hindi-roberta (hindi) (Joshi, 2022)
- google/muril-base-cased (17 indian langs) (Khanuja et al., 2021)
- google/muril-large-cased (17 indian langs) (Khanuja et al., 2021)
- l3cube-pune/tamil-bert (tamil) (Joshi, 2022)

These can be accessed at the following URLs: `https://huggingface.co/modelcard` (change `modelcard` to the proper name). The language of each model is given in parentheses.

Tables 2, 3 and 4 show the means and standard deviations of scores from five runs on the whole dev dataset and on the Tamil and Latin parts of that dataset.

Based on the F1-scores, it is evident that the google/muril-large-cased model performs the best overall for the entire dataset, although other multilingual models also perform well. This holds true even for the Tamil script, where the performance of the sole English language model decreases. For Latin script, the English models, multilingual models and certain Hindi models perform equally well. It is surprising to note that in all cases, the F1-score for the English version of the roberta-large model is inferior to that of roberta-base. We also found that the F1-score of the google/muril-base-cased model is lower by approximately 0.05 than that of google/muril-large-cased.

6 Submission to the challenge

Because we conducted the model evaluations described in the previous section after the competition was over, we could not use this knowledge for the final submission to the challenge. However, for the final submission, we followed the same training process using an ensemble of the following model cards: l3cube/pune-kannada-bert, microsoft/mdeberta-v3-base, and xlm-roberta-large. We combined the train and dev datasets and

model	**F1-Score**	**Precision**	**Recall**	**AUROC**	**Accuracy**
bert-base-cased	0.66 ± 0.01	0.70 ± 0.01	0.62 ± 0.02	0.79 ± 0.01	0.76 ± 0.00
roberta-base	0.71 ± 0.01	0.72 ± 0.01	0.70 ± 0.02	0.82 ± 0.01	0.79 ± 0.01
roberta-large	0.67 ± 0.03	0.69 ± 0.02	0.65 ± 0.04	0.78 ± 0.01	0.76 ± 0.02
bert-base-multilingual-cased	0.72 ± 0.00	0.75 ± 0.01	0.70 ± 0.01	0.84 ± 0.00	0.80 ± 0.00
xlm-roberta-base	0.72 ± 0.01	0.76 ± 0.02	0.70 ± 0.02	0.84 ± 0.01	0.80 ± 0.01
xlm-roberta-large	0.74 ± 0.01	0.76 ± 0.02	0.72 ± 0.01	0.84 ± 0.01	0.81 ± 0.01
microsoft/mdeberta-v3-base	0.73 ± 0.01	0.75 ± 0.03	0.71 ± 0.02	0.84 ± 0.00	0.80 ± 0.01
monsoon/nlp-hindi-bert	0.57 ± 0.01	0.55 ± 0.02	0.59 ± 0.04	0.70 ± 0.01	0.67 ± 0.01
l3cube/pune-hindi-roberta	0.65 ± 0.14	0.70 ± 0.04	0.63 ± 0.19	0.80 ± 0.07	0.77 ± 0.05
google/muril-base-cased	0.71 ± 0.01	0.74 ± 0.03	0.69 ± 0.03	0.81 ± 0.01	0.79 ± 0.01
google/muril-large-cased	$\mathbf{0.76} \pm 0.01$	$\mathbf{0.78} \pm 0.02$	$\mathbf{0.74} \pm 0.02$	$\mathbf{0.85} \pm 0.01$	$\mathbf{0.82} \pm 0.01$
l3cube/pune-tamil-bert	0.71 ± 0.01	0.71 ± 0.02	0.72 ± 0.03	0.82 ± 0.01	0.79 ± 0.01

Table 2: Evaluation of models on the whole dev dataset. The best results are highlighted in bold.

model	**F1-Score**	**Precision**	**Recall**	**AUROC**	**Accuracy**
bert-base-cased	0.54 ± 0.02	0.62 ± 0.02	0.48 ± 0.05	0.69 ± 0.01	0.71 ± 0.01
roberta-base	0.66 ± 0.02	0.68 ± 0.03	0.64 ± 0.02	0.78 ± 0.01	0.76 ± 0.02
roberta-large	0.60 ± 0.06	0.63 ± 0.04	0.58 ± 0.08	0.72 ± 0.03	0.73 ± 0.03
bert-base-multilingual-cased	0.69 ± 0.01	0.72 ± 0.01	0.66 ± 0.01	0.82 ± 0.00	0.79 ± 0.01
xlm-roberta-base	0.71 ± 0.01	0.74 ± 0.02	0.68 ± 0.02	0.83 ± 0.01	0.80 ± 0.01
xlm-roberta-large	0.74 ± 0.01	$\mathbf{0.76} \pm 0.03$	0.71 ± 0.02	0.85 ± 0.01	0.81 ± 0.01
microsoft/mdeberta-v3-base	0.73 ± 0.02	0.75 ± 0.03	0.71 ± 0.03	0.84 ± 0.01	0.81 ± 0.02
monsoon/nlp-hindi-bert	0.54 ± 0.01	0.46 ± 0.02	0.66 ± 0.05	0.64 ± 0.01	0.59 ± 0.02
l3cube/pune-hindi-roberta	0.61 ± 0.15	0.68 ± 0.04	0.59 ± 0.19	0.78 ± 0.07	0.75 ± 0.04
google/muril-base-cased	0.70 ± 0.02	0.73 ± 0.03	0.68 ± 0.05	0.81 ± 0.02	0.79 ± 0.01
google/muril-large-cased	$\mathbf{0.75} \pm 0.01$	0.75 ± 0.02	$\mathbf{0.76} \pm 0.02$	$\mathbf{0.86} \pm 0.01$	$\mathbf{0.82} \pm 0.01$
l3cube/pune-tamil-bert	0.71 ± 0.01	0.71 ± 0.02	0.72 ± 0.03	0.83 ± 0.01	0.79 ± 0.01

Table 3: Evaluation of models on the Tamil script part of the dev dataset. The best results are highlighted in bold.

model	**F1-Score**	**Precision**	**Recall**	**AUROC**	**Accuracy**
bert-base-cased	$\mathbf{0.76} \pm 0.01$	0.76 ± 0.01	$\mathbf{0.76} \pm 0.02$	0.86 ± 0.01	0.82 ± 0.00
roberta-base	0.75 ± 0.01	0.75 ± 0.01	$\mathbf{0.76} \pm 0.02$	0.86 ± 0.01	0.81 ± 0.00
roberta-large	0.73 ± 0.01	0.74 ± 0.02	0.72 ± 0.01	0.82 ± 0.02	0.80 ± 0.01
bert-base-multilingual-cased	0.75 ± 0.01	0.77 ± 0.01	0.74 ± 0.02	$\mathbf{0.87} \pm 0.00$	0.81 ± 0.00
xlm-roberta-base	0.74 ± 0.01	0.77 ± 0.02	0.72 ± 0.02	0.84 ± 0.01	0.81 ± 0.01
xlm-roberta-large	0.74 ± 0.01	0.76 ± 0.02	0.73 ± 0.03	0.84 ± 0.01	0.81 ± 0.01
microsoft/mdeberta-v3-base	0.73 ± 0.01	0.74 ± 0.03	0.72 ± 0.05	0.84 ± 0.01	0.80 ± 0.01
monsoon/nlp-hindi-bert	0.62 ± 0.02	0.75 ± 0.03	0.53 ± 0.05	0.74 ± 0.02	0.75 ± 0.01
l3cube/pune-hindi-roberta	0.68 ± 0.13	0.73 ± 0.06	0.67 ± 0.19	0.82 ± 0.07	0.78 ± 0.05
google/muril-base-cased	0.72 ± 0.01	0.74 ± 0.04	0.70 ± 0.03	0.82 ± 0.01	0.79 ± 0.01
google/muril-large-cased	$\mathbf{0.76} \pm 0.01$	$\mathbf{0.80} \pm 0.02$	0.72 ± 0.03	0.85 ± 0.01	$\mathbf{0.83} \pm 0.01$
l3cube/pune-tamil-bert	0.72 ± 0.01	0.72 ± 0.04	0.71 ± 0.03	0.82 ± 0.01	0.78 ± 0.01

Table 4: Evaluation of models on the Latin script part of the dev dataset. The best results are highlighted in bold.

used different new train/dev splits for each model. The model achieved an F1-score of 0.81 on the challenge test set, securing second place, behind the leader with 0.82.

7 Conclusions

We conducted an evaluation of several English, multilingual, and Hindi encoder language models for a classification task in the Tamil language. This task was a part of the Fourth Workshop on Language Technology for Equality, Diversity, and Inclusion. Our post-competition study revealed that the most effective model was google/muril-large-cased. All types of language models performed well on the Latin script portion of the dataset, which may result from the fact that some of the texts were in the English language. Our approach, which relied on an ensemble of selected models, achieved second place in the competition.

8 Limitations

The content of this paper is based on brief comments, primarily in Tamil. The origin, description, and annotation scheme of the text are explained in detail in (Rajiakodi et al., 2024). It is worth noting that the methods used in this study may not be easily scalable to other domains or text lengths. Furthermore, our assumption that Tamil script texts are those in which over half of the characters are non-Latin is merely heuristic and may not hold true in all cases.

References

Hindi Bert. https://huggingface.co/monsoon-nlp/hindi-bert. Accessed: 2023-12-15.

Bharathi Raja Chakravarthi, Prasanna Kumar Kumaresan, Ratnasingam Sakuntharaj, Anand Kumar Madasamy, Sajeetha Thavareesan, Bhavukam Premjith, K R Sreelakshmi, Subalalitha Chinnaudayar Navaneethakrishnan, John, Patrick McCrae, and Thomas Mandl. 2021. Overview of the HASOC-DravidianCodeMix shared task on offensive language detection in Tamil and Malayalam. In *Fire*.

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. *CoRR*, abs/1911.02116.

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.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. Debertav3: Improving DeBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing.

Raviraj Joshi. 2022. L3Cube-HindBERT and DevBERT: Pre-trained BERT transformer models for Devanagari based Hindi and Marathi languages. *arXiv preprint arXiv:2211.11418*.

Simran Khanuja, Diksha Bansal, Sarvesh Mehtani, Savya Khosla, Atreyee Dey, Balaji Gopalan, Dilip Kumar Margam, Pooja Aggarwal, Rajiv Teja Nagipogu, Shachi Dave, Shruti Gupta, Subhash Chandra Bose Gali, Vish Subramanian, and Partha Talukdar. 2021. Muril: Multilingual representations for Indian languages.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Ronak Pradeep, Xueguang Ma, Rodrigo Nogueira, and Jimmy Lin. 2021. Scientific claim verification with VerT5erini. In *Proceedings of the 12th International Workshop on Health Text Mining and Information Analysis*, pages 94–103, online. Association for Computational Linguistics.

Ratnavel Rajalakshmi, Srivarshan Selvaraj, Faerie Mattins R., Pavitra Vasudevan, and Anand Kumar M. 2023. HOTTEST: Hate and offensive content identification in Tamil using transformers and enhanced stemming. *Computer Speech Language*, 78:101464.

Saranya Rajiakodi, Bharathi Raja Chakravarthi, Rahul Ponnusamy, Prasanna Kumar Kumaresan, Sathiyaraj Thangasamy, Bhuvaneswari Sivagnanam, and Charmathi Rajkumar. 2024. Overview of Shared Task on Caste and Migration Hate Speech Detection. In *Proceedings of the Fourth Workshop on Language Technology for Equality, Diversity and Inclusion*, Malta. European Chapter of the Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108.

Malliga Subramanian, Rahul Ponnusamy, Sean Benhur, Kogilavani Shanmugavadivel, Adhithiya Ganesan, Deepti Ravi, Gowtham Krishnan Shanmugasundaram, Ruba Priyadharshini, and Bharathi Raja Chakravarthi. 2022. Offensive language detection in Tamil YouTube comments by adapters and cross-domain knowledge transfer. *Computer Speech Language*, 76:101404.