bitsa_nlp@LT-EDI-ACL2022: Leveraging Pretrained Language Models for Detecting Homophobia and Transphobia in Social Media Comments

Jaidi Nithin Reddy and **Poonam Goyal**
Birla Institute of Technology and Science, Pilani, India
nithin.reddy@pilani.bits-pilani.ac.in
poonam@pilani.bits-pilani.ac.in

Abstract

Online social networks are ubiquitous and user-friendly. Nevertheless, it is vital to detect and moderate offensive content to maintain decency and empathy. However, mining social media texts is a complex task since users don't adhere to any fixed patterns. Comments can be written in any combination of languages and many of them may be low-resource.

In this paper, we present our system for the LT-EDI shared task on detecting homophobia and transphobia in social media comments. We experiment with a number of monolingual and multilingual transformer based models such as mBERT along with a data augmentation technique for tackling class imbalance. Such pretrained large models have recently shown tremendous success on a variety of benchmark tasks in natural language processing. We observe their performance on a carefully annotated, real life dataset of YouTube comments in English as well as Tamil.

Our submission achieved ranks 9, 6 and 3 with a macro-averaged F1-score of 0.42, 0.64 and 0.58 in the English, Tamil and Tamil-English subtasks respectively. The code for the system has been open sourced[1].

1 Introduction

Twenty first century social media has become the epicenter of polarized opinions, arguments, and claims. The ease of information access not only benefits fruitful discussions but also facilitates phenomena such as hate speech and cyber bullying.

Recently organized workshops and shared tasks have fostered discussions around detection of hate speech, toxicity, misogyny, sexism, racism and abusive content (Zampieri et al., 2020; Mandl et al., 2020). While research in processing and classifying offensive language in social media is vast (Pamungkas et al., 2021), there is very little work on detecting sexual orientation discrimination in particular. More so, compared to resource-rich languages such as English and Japanese, Indic languages such as Tamil and Malayalam are scarce in well-annotated data. Although advancements in large multilingual models have promoted cross-lingual transfer learning in Indic languages (Dowlagar and Mamidi, 2021), there have not been any visible attempts to censor homophobia and transphobia. The perception of the subject matter as being taboo prohibits advancements in data collection, annotation and analysis.

Curbing sensitive online content is imperative for preventing harm to mental health of the community as well as avoiding divide between minorities. These reasons have contributed towards the need of moderating social media comments spreading any form of hatred towards the LGBTQIA+ population.

While both - the detection of homophobia/transphobia and the corresponding research in Indic languages - is underserved and low-resource, another factor contributing to the difficulty in processing social media texts is code-mixing - a phenomena in which multilingual speakers switch between two or more languages in a conversation with the aim to be more expressive. Popular language models tend to perform adversely when applied to code-mixed text and hence newer techniques need to be adopted to handle this situation (Doğruöz et al., 2021).

The pretraining and fine-tuning paradigm has taken extensive advantage of transformer based large multilingual models which perform well in cross-lingual scenarios. In this paper we explore the performance of a number of such models when fine-tuned on a dataset for detecting homophobia and transphobia. Surprisingly, our experiments also show that these multilingual models exhibit reasonably accurate performance on code-mixing tasks, even without any previous exposure to code-mixing during pretraining.

[1] The code for this task is available at github.com/vitthal-bhandari/Homophobia-Transphobia-Detection.

The remainder of the paper is organized as follows: Section 2 talks about the previous related work in this domain. Section 3 gives a detailed explanation of the methods used in the system and Section 4 describes the corresponding experimental settings. We mention the detailed results in Section 5, conduct an ablation study in Section 6 and conclude our discussion with Section 7.

2 Related Work

To the best of our knowledge no prior work identifying either homophobia or transphobia directly exists in recent literature. However, offensive language detection, in general, in Dravidian languages has been the focus of multiple research works in the past (Chakravarthi et al., 2021a; Mandl et al., 2020).

Baruah et al. (2021) at HASOC-Dravidian-CodeMix-FIRE2020 trained an SVM classifier using TF-IDF features on code-mixed Malayalam text and an XLM-RoBERTa based classifier on code-mixed Tamil text to detect offensive language in Twitter and YouTube comments. Sai and Sharma (2020) fine-tuned multilingual transformer models and used a bagging ensemble strategy to combine predictions on the same task.

Saha et al. (2021) developed fusion models by ensembling CNNs trained on skip-gram word vectors using FastText along with fine-tuned BERT models. A neural classification head was trained on the concatenated output obtained from the ensemble.

A number of approaches have been deployed to tackle code mixing in Indic languages as well, since multilingual transformer models lack the complexity to extract linguistic features directly from code switched text. Vasantharajan and Thayasivam (2021) used a selective translation and transliteration technique to process Tamil code-mixed YouTube comments for offensive language identification. They converted code-mixed text to native Tamil script by translating English words and transliterating romainzed Tamil words. Similar technique was used by Upadhyay et al. (2021) and Srinivasan (2020).

3 Methodology

This shared task was formulated as a multiclass classification problem where the model should be able to predict the existence of any form of homophobia or transphobia in a YouTube comment. The entire pipeline consists of two main components - a classification head on top of different popular models based on the transformer architecture, and a data augmentation technique for oversampling the English dataset. These components have been explained in further detail ahead.

3.1 Transformer-based Models

Since its introduction in 2017, the Transformer architecture and its variants have set a new state of the art across several NLP tasks. Various pre-trained language models (PLMs) based on the Transformer architecture were experimented with in this task as mentioned below.

BERT (`bert-base-uncased`) uses the encoder part of the Transformer architecture and has been pretrained on the Book Corpus and English Wikipedia using a masked language modeling (MLM) and next sentence prediction (NSP) objective (Devlin et al., 2018).

mBERT or multilingual BERT (`bert-base-multilingual-cased`) isa BERT model that has been pretrained on 104 languages across Wikipedia and has shown surprisingly good cross-lingual performance on several NLP tasks.

XLM-RoBERTa (`xlm-roberta-base`) has been pretrained on 2.5TB of massive multilingual data using the MLM objective. It beat mBERT on various cross-lingual benchmarks (Conneau et al., 2019).

IndicBERT is pretrained on a large-scale corpora of 12 Indian languages. It outperforms mBERT and XLM-RoBERTa on a number of tasks, while having 10 times fewer parameters to train (Kakwani et al., 2020).

HateBERT is obtained by re-training BERT on RAL-E, a large-scale dataset of reddit comments from banned communities. It outperforms BERT on three English datasets for offensive, abusive language and hate speech detection tasks. (Caselli et al., 2021).

3.2 Data Augmentation

Data augmentation is an important technique to build robust and more generalizable models. There are a number of techniques in NLP, each suitable to a certain task that can be used to augment the data (Feng et al., 2021).

For this task (in English), Surface Form Alteration as exhibited by *Easy Data Augmentation* (EDA) was utilized (Wei and Zou, 2019). EDA

Class	English			Tamil			Tamil-English		
	Train	**Dev**	**Test**	**Train**	**Dev**	**Test**	**Train**	**Dev**	**Test**
Homophobic	157	58	61	485	103	135	311	66	88
Transphobic	6	2	5	155	37	41	112	38	34
Non-anti-LGBT+ content	3001	732	924	2022	526	657	3438	862	1085
Total		4946			4161			6034	

Table 1: Detailed split of the multilingual dataset of YouTube comments

produces new data samples by randomly deleting, inserting or swapping the order of words in a sentence. It can also perform synonym replacement for any word selected at random. These four simple, yet effective, operations make EDA easy to use.

4 Experimental Setup

In this section we review the setup needed to reproduce the experiments.

4.1 Datasets

The dataset for the task was provided by the organizers (Chakravarthi et al., 2021b). It is a collection of 15,141 multilingual YouTube comments classified as being one of Homophobic, Transphobic, or Non-anti-LGBT+ content. The split of the dataset is shown in Table 1.

4.2 Preprocessing

Two different preprocessing methods were adopted. First, punctuation symbols were removed, since social media comments are highly informal and tend to contain large number of punctuation symbols which may dilute the system performance.

In addition, de-emojification was carried out to replace emojis in the text with corresponding English expressions using the Python `emoji` package. Table 2 displays a sample de-emojification example.

I love it 💗 💗 💗
i love it growing heart growing heart growing heart

Table 2: Depiction of de-emojification on a sample English YouTube comment

4.3 EDA Parameters

As is visible from Table 1, the dataset is highly imbalanced in its split. The Homophobia class constitutes slightly less than 10% of the data, while only 2.9% comments were labeled as being Transphobic. Hence both these classes were subject to oversampling by means of EDA. The class Non-anti-LGBT+ content was downsampled to mitigate the imbalance.

Augmentation was only applied to English comments.

The parameter α (indicating the percent of words in a sentence that are changed) was kept as default ($= 0.1$). However the argument n_{aug} (specifying the number of augmentations to be produced for each sample) was chosen to be 16 and 32 for Homophobia and Transphobia classes respectively.

GT	I have to experience like that. So sad
RD	i to experience like so sad
SR	i have to experience like that so pitiful
RI	i have to experience like that distressing so sad
RS	experience have to i like that so sad

Table 3: Depiction of data augmentation on a sample English YouTube comment. GT: ground truth, RD: random deletion, SR: synonym replacement, RI: random insertion, RS: random swapping

The final classwise split of the training data is shown in Table 4.

Class	**Final size**
Homophobic	2826
Transphobic	204
Non-anti-LGBT+ content	1500

Table 4: Classwise split of the training data after EDA augmentation

4.4 Baseline Methods

We provide baselines for all three tracks based on a simple feature extraction approach.

We use the `[CLS]` token associated with the final hidden state of the transformer model as feature vector for a linear regression classifier.

To extract the hidden state from the checkpoint, we use BERT base model for the English track and mBERT for the other tracks.

4.5 Setup

The experiments were run on a Google Colab Pro notebook with Tesla P100 GPU.

For the all tasks, the maximum sequence length was set to 128 and batch size to 32. The learning rate and the number of epochs were set to $2e-5$ and 3 respectively for the English and Tamil track and $3e-5$ and 5 respectively for the code-mixed track. The choice of EDA parameters was based on suggestions given in the original paper whereas the model hyperparameters were selected based on popular successful configurations.

5 Results

The metric used to rank system performances is macro-averaged F1-score. It is calculated as the (unweighted) arithmetic mean of all the per-class F1-scores.

$$\text{Macro-averaged F1-score} = \frac{1}{N}\sum_{i=1}^{N} F1_i$$

where i is the class index and N is the number of classes

Tables 5, 7 and 9 list the macro-averaged Precision, macro-averaged Recall and macro-averaged F1-score for various PLMs tested on English, Tamil and code-mixed Tamil-English development dataset respectively.

Similarly Tables 6, 8 and 10 list the corresponding metrics achieved by the final submissions on English, Tamil and Tamil-English test dataset as released by the organizers.

The tables also provide baseline metrics for each track based on the method explained in Section 4.4.

5.1 English

Model	P	R	F1
BERT embeddings + LR	0.40	0.47	0.42
BERT base cased	0.46	0.46	0.461
XLM-RoBERTa	0.49	0.40	0.42
hateBERT	0.50	0.44	0.461
mBERT	0.48	0.45	**0.462**

Table 5: Performance of various PLMs on augmented, preprocessed English development dataset

Model	P	R	F1
mBERT	0.43	0.42	0.42

Table 6: Performance of best peforming system (*mBERT*) on preprocessed English test dataset

5.2 Tamil

Here we investigate the performance of some popular multilingual models that were trained on Tamil language.

Model	P	R	F1
mBERT embeddings + LR	0.71	0.59	0.63
IndicBERT	0.48	0.47	0.47
XLM-RoBERTa	0.47	0.55	0.50
mBERT	0.77	0.71	**0.72**

Table 7: Performance of various PLMs on preprocessed Tamil development dataset

Model	P	R	F1
mBERT	0.69	0.61	0.64

Table 8: Performance of best peforming system (*mBERT*) on preprocessed Tamil test dataset

5.3 Tamil-English

For the code-mixed task, we analyze the performance of the same set of multilingual models that were experimented with on the Tamil task.

Model	P	R	F1
mBERT embeddings + LR	0.61	0.47	0.51
IndicBERT	0.39	0.41	0.40
XLM-RoBERTa	0.40	0.43	0.41
mBERT	0.67	0.52	**0.54**

Table 9: Performance of various PLMs on preprocessed Tamil-English development dataset

Model	P	R	F1
mBERT	0.61	0.56	0.58

Table 10: Performance of best peforming system (*mBERT*) on preprocessed Tamil-English test dataset

6 Ablation Study

In this section we discuss the effect of preprocessing and data augmentation (DA) on the model performance.

The dataset as described in Section 4.4 is highly skewed towards the *Non-anti-LGBT+ content* class. Hence it makes sense to compare the performance of a majority classifier with that of the models submitted for evaluation.

We train a dummy classifier based on most-frequent strategy and tabulate the results (macro-averaged Precision, Recall and F1-score) in Table 11. We deliberately use the un-augmented version of preprocessed English dataset to show the show the performance of the majority classifier without handling class imbalance.

	P	R	F1
English	0.31	0.33	0.32
Tamil	0.26	0.33	0.29
Code-mixed	0.30	0.33	0.31

Table 11: Performance of dummy majority classifier on the dataset

The poor performance is a consequence of the extreme class imbalance which we aim to solve by data augmentation. However, not all DA techniques prove to be effective for all NLP tasks. Thus we also analyze the effect of preprocessing and DA on the performance of transformer models.

Table 12 analyzes the efficacy of EDA as a DA technique for handling class imbalance in our English dataset. It also divides a line between the performance of the model on the stock dataset v/s one that has been preprocessed.

	Setting	P	R	F1	Rel.
English	base	0.52	0.40	0.43	
	+PRE	0.40	0.43	0.41	↓
	+DA	0.52	0.37	0.39	↓
Tamil	base	0.73	0.75	0.74	
	+PRE	0.70	0.73	0.72	↓
Code-mixed	base	0.43	0.42	0.43	
	+PRE	0.71	0.56	0.60	↑

Table 12: Performance of mBERT on the stock version of the dataset as it is (base), preprocessed dataset (+PRE) and augmented but non-preprocessed English dataset (+DA)

We observe that preprocessing (de-emojification in all three tracks and de-punctuation in the case of only English) does not increase the macro-averaged F1 score for English and Tamil. Infact it reduces the score by a small margin. However, we notice a significant improvement in the case of code-mixing.

We also observe that EDA is not an efficient DA technique as it fails to handle the class imbalance. Transformer models were able to successfully predict with higher precision and recall in the absence of any augmentation and with limited samples.

7 Conclusion and Future Work

Homophobia and transphobia have not been the focus of many umbrella hate speech detection tasks. We examined the ability of pretrained large transformer-based models to detect homophobia and transphobia in a corpus of YouTube comments written in English and Tamil. Experimental results demonstrated that multilingual BERT performed the best on both language tasks, and the code-mixed task as well, without being exposed to any code-mixing beforehand. This can be attributed to its capability for zero-shot cross-lingual transfer when fine-tuned on downstream tasks.

From Section 6 we also observed that the effect of preprocessing was largely dependent on the choice of language setting. This makes sense considering the difference in underlying language constructs. Tamil, for instance, does not make use of standard English-based punctuation marks. On the other hand, we conclude that the choice of an effective DA techniqe depends on the underlying task and the data source. Social media data often lacks linguistic purism and hence, token perturbations such as those introduced by EDA did not help.

In the future, we would like to adopt a more aggressive DA technique such as that involving text generation (text In-filling, generating typos) or an auxilliary dataset (kNN, LM decoding). We would also like to evaluate the effect of translation and transliteration on code-mixed text classification.

Acknowledgments

We would like to acknowledge the efforts of the workshop organizers in effecting positive social change through AI by conducting such shared tasks. We also thank the reviewers for their time and insightful comments.

References

Arup Baruah, Kaushik Amar Das, Ferdous Ahmed Barbhuiya, and Kuntal Dey. 2021. Iiitg-adbu@ hasoc-dravidian-codemix-fire2020: Offensive content detection in code-mixed dravidian text. *arXiv preprint arXiv:2107.14336*.

Tommaso Caselli, Valerio Basile, Jelena Mitrović, and Michael Granitzer. 2021. HateBERT: Retraining BERT for abusive language detection in English. In *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*, pages 17–25, Online. Association for Computational Linguistics.

Bharathi Raja Chakravarthi, Ruba Priyadharshini, Navya Jose, Anand Kumar M, Thomas Mandl, Prasanna Kumar Kumaresan, Rahul Ponnusamy, Hariharan R L, John P. McCrae, and Elizabeth Sherly. 2021a. 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*, pages 133–145, Kyiv. Association for Computational Linguistics.

Bharathi Raja Chakravarthi, Ruba Priyadharshini, Rahul Ponnusamy, Prasanna Kumar Kumaresan, Kayalvizhi Sampath, D. Thenmozhi, S. Thangasamy, Rajendran Nallathambi, and John P. McCrae. 2021b. Dataset for identification of homophobia and transophobia in multilingual youtube comments. *ArXiv*, abs/2109.00227.

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*.

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.

A. Seza Doğruöz, Sunayana Sitaram, Barbara E. Bullock, and Almeida Jacqueline Toribio. 2021. A survey of code-switching: Linguistic and social perspectives for language technologies. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1654–1666, Online. Association for Computational Linguistics.

Suman Dowlagar and Radhika Mamidi. 2021. A survey of recent neural network models on code-mixed indian hate speech data. In *Forum for Information Retrieval Evaluation*, FIRE 2021, page 67–74, New York, NY, USA. Association for Computing Machinery.

Steven Y Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard Hovy. 2021. A survey of data augmentation approaches for nlp. *arXiv preprint arXiv:2105.03075*.

Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, Gokul N.C., Avik Bhattacharyya, Mitesh M. Khapra, and Pratyush Kumar. 2020. IndicNLPSuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for Indian languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4948–4961, Online. Association for Computational Linguistics.

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.

Endang Wahyu Pamungkas, Valerio Basile, and Viviana Patti. 2021. Towards multidomain and multilingual abusive language detection: a survey. *Personal and Ubiquitous Computing*.

Debjoy Saha, Naman Paharia, Debajit Chakraborty, Punyajoy Saha, and Animesh Mukherjee. 2021. Hate-alert@DravidianLangTech-EACL2021: Ensembling strategies for transformer-based offensive language detection. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, pages 270–276, Kyiv. Association for Computational Linguistics.

Siva Sai and Yashvardhan Sharma. 2020. Siva@ hasoc-dravidian-codemix-fire-2020: Multilingual offensive speech detection in code-mixed and romanized text. In *FIRE (Working Notes)*, pages 336–343.

Anirudh Srinivasan. 2020. MSR India at SemEval-2020 task 9: Multilingual models can do code-mixing too. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 951–956, Barcelona (online). International Committee for Computational Linguistics.

Ishan Sanjeev Upadhyay, Nikhil E, Anshul Wadhawan, and Radhika Mamidi. 2021. Hopeful men@LT-EDI-EACL2021: Hope speech detection using indic transliteration and transformers. In *Proceedings of the First Workshop on Language Technology for Equality, Diversity and Inclusion*, pages 157–163, Kyiv. Association for Computational Linguistics.

Charangan Vasantharajan and Uthayasanker Thayasivam. 2021. Towards offensive language identification for tamil code-mixed youtube comments and posts. *SN Computer Science*, 3(1).

Jason Wei and Kai Zou. 2019. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6383–6389, Hong Kong, China. Association for Computational Linguistics.

Marcos Zampieri, Preslav Nakov, Sara Rosenthal, Pepa Atanasova, Georgi Karadzhov, Hamdy Mubarak, Leon Derczynski, Zeses Pitenis, and Çağrı Çöltekin. 2020. SemEval-2020 task 12: Multilingual offensive language identification in social media (OffensEval 2020). In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1425–1447, Barcelona (online). International Committee for Computational Linguistics.