Sentiment Analysis and Homophobia detection of Code-Mixed Dravidian Languages leveraging pre-trained model and word-level language tag

Supriya Chanda[a], Anshika Mishra[b] and Sukomal Pal[a]

[a]*Indian Institute of Technology (BHU), Varanasi, INDIA*
[b]*Vellore Institute of Technology Bhopal, Madhya Pradesh , INDIA*

Abstract

Social media platforms have seen a significant rise in user engagement in recent years. More and more people are expressing their views and ideas on social platforms. There is an ardent need to develop an accurate system to classify text based on sentiments. In this paper, our team IRLab@IITBHU presents a solution architecture submitted to the shared task "Sentiment Analysis and Homophobia Detection of YouTube Comments in Code-Mixed Dravidian Languages" organized by DravidianCodeMix 2022 at Forum for Information Retrieval Evaluation (FIRE) 2022. to reveal how sentiment is expressed in code-mixed scenarios. For task A: we used mBERT model and word-level language tag to classify YouTube comments into positive, negative, neutral, or mixed emotions. And for Task B: we performed basic preprocessing steps and built mBERT model to identify homophobia, transphobia, and non-anti-LGBT+ content from the given corpus. For Task A, our proposed system achieved the best result, securing the first rank for Malayalam-English and Kannada-English code-mixed datasets with the F_1 score of 0.72 and 0.66 respectively.

Keywords

Code Mixed, Kannada, Malayalam, Tamil, mBERT, Sentiment Analysis, Homophobia,

1. Introduction

The evolution of social media networks has given people the liberty to share and access information with ease and in no time. Expressing one's ideas, opinions and views have never been as easy as it is today. Analysis from Kepios [1] shows that over 4.7 billion people worldwide are active social media users, equating to 59% of the world's population. With so much data available at our disposal, it is a necessity to analyze and retrieve useful information from it, which can help us make better decisions.

But the real-world data may not always be perfect like texts from textbooks. Many multilingual speakers tend to combine different languages together in a conversation. India is a land of many languages, and people from different parts of the country speak different languages. There is a popular aphorism that goes like *Kos-kos par badle paani, chaar kos par baani* (The language spoken in India changes every few kilometers, just like the taste of the water). The Indian constitution recognises 22 official languages. With so many different languages, India is home to a large multilingual community. People often tend to switch between the languages to better express their thoughts and ideas. This phenomenon is commonly known as code-mixing. Code-Mixed texts are also written in non-native

FIRE'22: Forum for Information Retrieval Evaluation, December 9-13, 2022, India
supriyachanda.rs.cse18@itbhu.ac.in (S. Chanda); anshika.mishra2019@vitbhopal.ac.in (A. Mishra); spal.cse@itbhu.ac.in (S. Pal)
https://cse-iitbhu.github.io/irlab/supriya.html (S. Chanda); https://cse-iitbhu.github.io/irlab/spal.html (S. Pal)
0000-0002-0877-7063 (S. Chanda); 0000-0001-8743-9830 (S. Pal)

[1]https://kepios.com/reports

scripts. People are generally inclined to write texts in roman-scripts. We have systems trained on monolingual data, but those specialized systems will fail to give us a satisfactory result on code-mixed data. Hence we need a model which can analyze code-mixed textual data available to us.

Sentiment analysis refers to a sub-task in Natural Language Processing which uses computational methods to analyze, process, and better understand the emotions of the users behind a text or interaction. It categorizes users' opinions into various classes of sentiments. It allows organizations to gain insights from a vast volume of unstructured data and help them remodel their strategies that are better focused on the target market. Companies like Amazon and Flipkart can analyze their product reviews to understand the customer's responses and leverage that information to improve their performance.

Given the freedom to express views and beliefs publicly, there comes a challenge to regulate the offensive content present on the internet. A certain section of the community, like LGBTQ+, is being targeted and subjected to many forms of abuse on social media platforms. In a survey of more than 1,100 LGBT+ people for Galop's Online Hate Crime Report 2021 [2], 64% reported experiencing anti-LGBT+ violence or abuse. The daily experience of violence or abuse was reported by 16% of the respondents. Verbal abuse was the most prevalent (92%) and was followed by online abuse (60%) in frequency. Recent studies have shown how this style of abuse can hamper their mental state and hurt them. Hence there is a need for a system to identify the contents which are homophobic or transphobic in nature.

1.1. Task Descriptions

DravidianCodeMix organized the shared task on Sentiment Analysis and Homophobia detection of YouTube comments in Code-Mixed Dravidian Languages [1]. The shared task included two different tasks. Task A analyzes the sentiment of code-mixed text in three Dravidian languages (Tamil-English, Malayalam-English, and Kannada-English). Task A's objective was to divide the code-mixed data into five categories: positive, negative, unknown state, mixed feelings, and not in the intended language. We competed in all three Dravidian languages. However, the Tamil-English language was excluded due to technical issues and a mismatched submission file. Task B identifies homophobia, transphobia, and non-anti-LGBT+ social media text in three monolingual languages (English, Tamil, and Malayalam) and one code-mixed language (Tamil-English). Task B's objective was to categorize the code-mixed material into homophobic, transphobic, and non-anti-LGBTQ+ content. Tasks A and B, which involved 5 and 3 classes, were treated as multi-class problems.

In this paper, we propose a method to analyze the real-world data and perform sentiment analysis on Code-Mixed datasets of YouTube comments in Dravidian languages (Tamil-English, Malayalam-English, and Kannada-English). Along with the identification of Homophobic and transphobic speech contents at different levels.

The rest of the paper is arranged in the following fashion. First, we discuss the related work in section 2. Section 3 describes the dataset. The proposed methodology, which includes pre-processing and model architecture, is described in Section 4. In Section 5, we report our results and analysis. Finally, we conclude in Section 6.

2. Related Work

This section summarizes earlier work on sentiment analysis and homophobia detection.

[2] https://galop.org.uk/wp-content/uploads/2021/06/Galop-Hate-Crime-Report-2021-1.pdf

Sentiment analysis is one of the top study fields targeted to analyze people's feelings and views on a specific subject. Numerous studies have been conducted in various languages, focusing on monolingual languages. Code-Mixed languages, however, are an exception to this rule. Few code-mixed language pairings have been attempted in the past. In addition to sentiment analysis, the Forum for Information Retrieval Evaluation (FIRE) has carried out many code-switching tasks. The tasks include Code-Mixed Question Answering, sentiment analysis for code-mixed Indian languages [2] (ICON 2017), POS tagging for code-mixed Indian social media (ICON 2016), and code-mixed entity extraction. SemEval 2020 organized a Sentiment Analysis in a Code-Switched Data competition [3], including tweets in Hindi-English and Spanish-English pairs. In the Dravidian Languages, there is a severe lack of data for experimentation on code-mixed data.

There are not many datasets that combine Kannada and English for sentiment analysis. A Kannada-English [4] code-mixed dataset was created for emotion prediction. A Tamil-English [5] code-mixed dataset was created as a part of a shared task on Sentiment Analysis of Indian Languages (SAIL). The data was extracted from Twitter.

Recently, it was widely noted worldwide that postings or comments on social media involving hate often devolve into violence. Numerous methods have been developed to identify hate speech, and automatic hate speech recognition has attracted much interest. SemEval, HASOC, and HatEval are just a few of the latest shared tasks used to identify hate speech on social media. Hate speech language detection can be formulated as a classification task. In general, it is a binary classification problem that determines if something is hateful or not. In the future, though, if there is a more detailed description of hate speech, it may be handled as a multi-class classification problem. The methods are divided into two categories. One is the use of classifiers in machine learning models such as SVM [6, 7], logistic regression [8], and random forest [9]. A different strategy is based on deep learning. The transformer-based model utilized a pre-trained model (BERT, mBERT [10], Roberta) and then fine-tuned it for the particular downstream goal of getting cutting-edge outcomes across several languages.

Comments that are homophobic or transphobic are typically seen to be hate speech directed toward LGBT+ people. Concerns about this occurrence are mounting. The first dataset on homophobia and transphobia in multilingual comments in Tamil, English, and Tamil-English was created by Chakravarthi et al. [11]. The dataset provided by this study included a high-quality, expert homophobic and transphobic content categorization from multilingual YouTube comments. On that dataset, they deployed some traditional Machine Learning and Deep Learning models as a baseline, and in 2022 they arranged a shared task at an ACL workshop [12] to further the study of homophobic and transphobic content identification. They claimed that the pre-trained XLM Roberta model delivered the best results for the task.

3. Datasets

The organizers provided two datasets of Training, Development, and testing for two separate shared tasks. The first share task is sentiment detection of code-mixed text in Dravidian languages. There was three pair of code-mixed datasets, Tamil-English, Malayalam-English, and Kannada-English. The training dataset consists of 35,656 Tamil-English [13] and 6,212 Kannada-English [14] and 15,880 Malayalam-English [15] YouTube video comments. Table 1 summarizes the statistics of the training, development, and testing data sample collection and their distribution by class.

Detecting homophobia, transphobia, and non-anti-LGBT+ content from the provided corpus is the goal of the second share task [11]. There were three single monolingual datasets for Tamil, English, and Malayalam, as well as one code-mixed dataset for Tamil and English. Table 2 summarizes the

Table 1
Data Distribution for sentiment detection of code-mixed text in Dravidian languages

Tamil - English				
Class	**Training**	**Development**	**Test**	**Total**
Positive	20070	2257	73	22400
Negative	4271	480	338	5089
not-Tamil	1667	176	0	1843
Mixed_feelings	4020	438	101	4559
unknown_state	5628	611	137	6376
Total	35656	3962	649	40267
Kannada - English				
Class	**Training**	**Development**	**Test**	**Total**
Positive	2823	321	374	3518
Negative	1188	139	157	1484
not-Kannada	916	110	110	1136
Mixed_feelings	574	52	65	691
unknown_state	711	69	62	842
Total	6212	691	768	7671
Malayalam - English				
Class	**Training**	**Development**	**Test**	**Total**
Positive	6421	706	780	7907
Negative	2105	237	258	2600
not-Malayalam	1157	141	147	1445
Mixed_feelings	926	102	134	1162
unknown_state	5279	580	643	6502
Total	15880	1766	1962	19616

statistics of the training, development, and testing data sample collection and their distribution by class.

4. Proposed Methodology

4.1. Data Pre-processing

Data preprocessing was kept to a bare minimum to keep it adaptable for both the shared tasks. We saw in the task report [16], which was previously given that eliminating continuous repetitions did not result in any appreciable performance differences. This year, we limited the repeated characters to two contiguous repeats. This reduces the whole sequence length to below 512. Then, we eliminate hashtags, punctuation, URLs, and numbers and mentions that do not have a clear semantic significance. We replace emojis with their proper semantic text. And stripped off any white spaces and extra spaces.

Table 2
Data Distribution for Homophobia detection in Dravidian languages

Tamil - English			
Class	**Training**	**Development**	**Test**
Non-anti-LGBT+ content	3438	862	1085
Homophobic	311	66	88
Transphobic	112	38	34
Total	3861	966	1207df
Tamil			
Class	**Training**	**Development**	**Test**
Non-anti-LGBT+ content	2022	526	352
Homophobic	485	103	271
Transphobic	155	37	26
Total	2662	666	649
English			
Class	**Training**	**Development**	**Test**
Non-anti-LGBT+ content	3001	732	924
Homophobic	157	58	61
Transphobic	6	2	5
Total	3164	999	990
Malayalam			
Class	**Training**	**Development**	**Test**
Non-anti-LGBT+ content	2434	692	971
Homophobic	491	133	182
Transphobic	189	41	60
Total	3114	866	1213

4.2. Model Architecture

The most well-known NLP technology in recent memory is likely word embedding. It captures a word's semantic characteristics. We used `bert-base-multilingual-cased` (mBERT) pre-trained models [3] to get a vector as an embedding for the sentence that we can use for classification.

mBERT is A transformer architecture that is an encoder-decoder network that uses self-attention on the encoder side and attention on the decoder side. The models are pre-trained on large text corpora such as Wikipedia and produce state-of-the-art results with necessary fine-tuning on several downstream tasks. The contextual language representation model BERT (Bidirectional Encoder Representations from Transformers) has been used for the downstream task of code-mixed language identification. Multilingual BERT or mBERT (`bert-base-multilingual-cased`[4]) is pre-trained on cased text in the top 104 languages with the largest Wikipedias and has a total 179M parameters with 12 transformers blocks, 768 hidden layers and 12 attention head. This model takes a special

[3]https://huggingface.co/transformers/pretrained_models.html
[4]https://github.com/google-research/bert/blob/master/multilingual.md

[CLS] token as input first, followed by a sequence of words as input. It then passes the input to the next layer. [CLS] here stands for Classification. Each layer applies self-attention and passes the result through a feedforward network to the next encoder.

For the implementation, HuggingFace's transformers library was utilized. A Python package called HuggingFace transformers offers pre-trained and adaptable transformer models that may be used for various NLP tasks. The implementation environment is the PyTorch library, which supports GPU computation. The mBERT models were run using Google Colab. We trained our classifier across 2-4 epochs with a batch size of 32. The `AdamW` optimizer is used, and the dropout value is set at 0.1. The learning rate is 2e-5. For tokenization, we utilized the hugging face transformers' pre-trained BERT tokenizer. We utilized the HuggingFace library's `BertForSequenceClassification` module for tinkering and sequence classification.

Initially, we considered task A as a multi-class classification problem. But after carefully studying the dataset description provided by the organizers, we realized that the given tag "Not in intended Language" is independent of the remaining tags. According to the original description of the dataset, any statement which does not contain words from the specified language is to be labeled as not-<language>. In this manner, the above Task A could be considered as a multi-label classification where statements labeled as not-<language> can further be classified into positive, negative, neutral, or mixed emotions. But according to the given shared task description, we approached it as a multi-class classification problem and simultaneously classified these comments into two classes, namely language and not-<language>.

To do a binary classification between language and not-<language>, we deployed our language identification (LID) module, which provides word-level language identification for the text. To find the LID for each word, we used Googletrans [5] which is a free and unlimited python library that implemented Google Translate API. It uses the Google Translate Ajax API to make calls to such methods as detect and translate. Based on the LID of the words, we built a rule-based system that labels the statement as not-<language> if the given statement does not contain any word in the specified language else, label it as language. After that, we utilized a function similar to the OR function that depends on the output of the LID module. The mBERT prediction will remain the same if the output is <language>. Otherwise, not-<language> will be used in its replacement.

After that, we combined the predictions from two modules, mBERT and the Rule-based system (See Figure 1).

For task B, we had to develop a system to predict whether the given comments are homophobic/transphobic in nature. We used the mBERT model to classify the comments into two classes Homophobic and non homophobic.

5. Results and Analysis

In this section, we present the evaluation of our model and submitted results for both tasks: sentiment analysis and homophobic identification for Dravidian languages.

The performance of our proposed models is examined using evaluation measures including accuracy, recall, macro averaged F1-score, and weighted average F1-score for sentiment analysis of code-mixed text in Dravidian languages. The organizers gave the test data for the three Dravidian languages. Based on the training and validation data, we fine-tuned our model, and we then submitted our prediction file for the test data. First, we have improved the mBERT model to create the system and forecast the sentiment polarity for the Tamil-English, Malayalam-English, and Kannada-English

[5] https://pypi.org/project/googletrans/

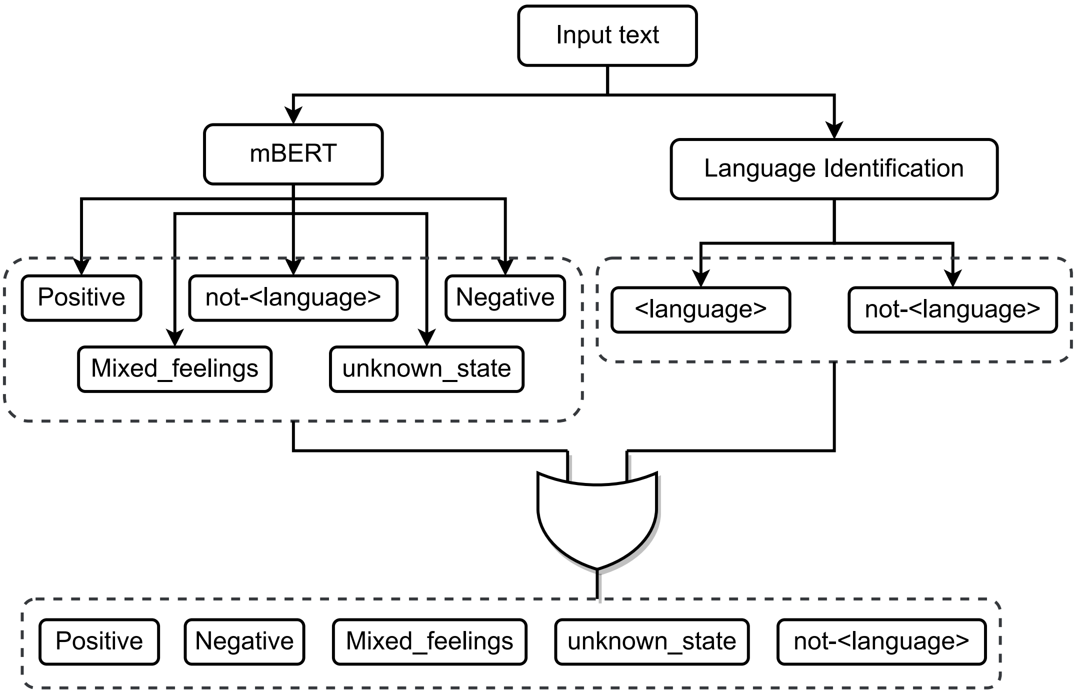


Figure 1: Model Architecture for Task A

languages, as indicated in this study [17]. For all three language pairings, we have demonstrated positive outcomes. Following that, we utilized a rule-based approach that divided everything into two categories: language and not-<language>. After receiving this binary classification, we combine the predictions with the mBERT prediction and then provide the final prediction. Table 3 and 4 display the performance of the test outcomes for the mBERT model and mBERT + Ruled-based systems for the two languages. For Kannada-English and Malayalam-English test data we got 0.66 and 0.72 F_1 score respectively and stand top of the rank list for both of the language pairs. Here we observed that our approach outperforms existing mBERT based model for both language pairs. We can find the class wise performance improvement in table 5 for Kannada-English and table 6 for Malayalam-English pair.

Table 3
Precision, recall, Weighted F_1-scores for Task A on Kannada-English test data and rank list

	Kannada - English			
Team Name	Precision	Recall	Weighted F_1 score	Rank
IRLab@IITBHU (mBERT)	0.60	0.64	0.61	-
IRLab@IITBHU (mBERT + Ruled-based systems)	0.67	0.67	0.66	1/13

For Homophobic detection our proposed model did not perform as expected. And we could only achieve the F_1 score of 0.333 for Tamil-English (See Table 7), 0.289 for Tamil (See Table 8), 0.337 for English (See Table 9) and 0.427 for Malayalam (See Table 10).

We also looked at the confusion matrix, which is depicted in Figure 2. This was a crucial analytical tool that allowed us to examine which classes are misclassified by which classes. Figures 2a and 2b demonstrate how well our model categorizes not-Kannada and not-Malayalam. Only one non-Malayalam language qualifies as positive, while the remaining three are classified as unknown_state.

Table 4
Precision, recall, Weighted F_1-scores for Task A on Malayalam-English test data and rank list

	Malayalam - English			
Team Name	Precision	Recall	Weighted F_1 score	Rank
IRLab@IITBHU (mBERT)	0.69	0.70	0.69	-
IRLab@IITBHU (mBERT + Ruled-based systems)	0.72	0.73	0.72	1/11

Table 5
Precision, recall, F_1-scores, and support for all experiments on Kannada-English (Task A) test data

	mBERT			**mBERT + Ruled-based systems**			
	Precision	Recall	F_1-score	Precision	Recall	F_1-score	support
Mixed_feelings	0.15	0.03	0.05	0.21	0.28	0.24	65
Negative	0.61	0.61	0.61	0.71	0.54	0.61	157
Positive	0.71	0.78	0.74	0.77	0.76	0.76	374
not-Kannada	0.61	0.70	0.65	0.71	1.00	0.83	110
unknown_state	0.38	0.37	0.37	0.39	0.24	0.30	62
macro avg	0.49	0.50	0.49	0.56	0.56	0.55	768
weighted avg	0.60	0.64	0.61	0.67	0.67	0.66	768
Accuracy	0.64			0.67			

Table 6
Precision, recall, F_1-scores, and support for all experiments on Malayalam-English (Task A) test data

	mBERT			**mBERT + Ruled-based systems**			
	Precision	Recall	F_1-score	Precision	Recall	F_1-score	support
Mixed_feelings	0.40	0.16	0.23	0.43	0.25	0.32	134
Negative	0.55	0.53	0.54	0.63	0.56	0.59	258
Positive	0.74	0.81	0.77	0.77	0.82	0.79	780
not-malayalam	0.83	0.71	0.77	0.81	0.97	0.89	147
unknown_state	0.71	0.74	0.72	0.72	0.74	0.73	643
macro avg	0.64	0.59	0.61	0.67	0.67	0.66	1962
weighted avg	0.69	0.70	0.69	0.72	0.73	0.72	1962
Accuracy	0.70			0.73			

Figure 2d shows that our approach could not categorize the transphobic class. There were only six labels of transphobic in the training data. Therefore, this is probably the reason for that. If we see all four confusion matrices for task B (Figure 2c, 2d, 2e and 2f), We discovered that our model predominantly misclassified data as non-anti-LGBT+. Furthermore, the enormously unbalanced dataset may be the cause. Consequently, our model overfitted and labeled the majority of the classes as non-anti-LGBT+.

Table 7
Evaluation results of Task B (Homophobia/Transphobia Detection) on Tamil-English test data and rank list

	Tamil - English	
Team Name	macro-F_1 score	Rank
mucs	0.580	1/8
IRLab@IITBHU	0.333	6/8

Table 8
Evaluation results of Task B (Homophobia/Transphobia Detection) on Tamil test data and rank list

	Tamil	
Team Name	macro-F_1 score	Rank
mucs	0.366	1/6
IRLab@IITBHU	0.289	4/6

Table 9
Evaluation results of Task B (Homophobia/Transphobia Detection) on English test data and rank list

	English	
Team Name	macro-F_1 score	Rank
BharatNLP	0.493	1/8
IRLab@IITBHU	0.337	5/8

Table 10
Evaluation results of Task B (Homophobia/Transphobia Detection) on Malayalam test data and rank list

	Malayalam	
Team Name	macro-F_1 score	Rank
Nitk	0.974	1/9
IRLab@IITBHU	0.427	8/9

5.1. Error Analysis

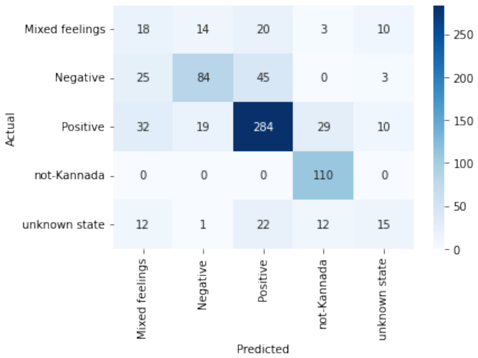
Table 11 lists some instances where our top model made incorrect predictions. In the Gold column of the table, the projected sentiment is in contrast with the anticipated sentiment from the gold standard dataset. It appears that the feeling we anticipated was accurate. When we processed the test dataset via our rule based system we observed that many instances were incorrectly annotated and were not in alignment to the class description provided by the organizer.

Table 11
Error Analysis

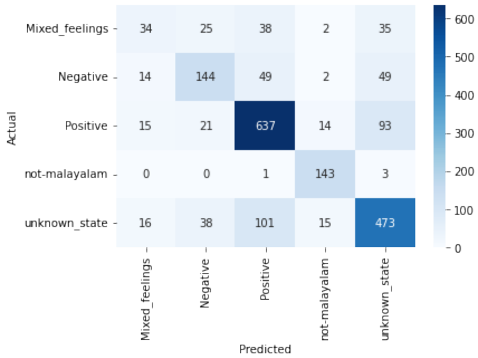
Sample text from dataset	Gold	Predicted
Govt should take action	neutral	not-Tamil
Police pls take action	neutral	not-Tamil
Government take this type of peoples civiliar actions	neutral	not-Tamil

6. Conclusion

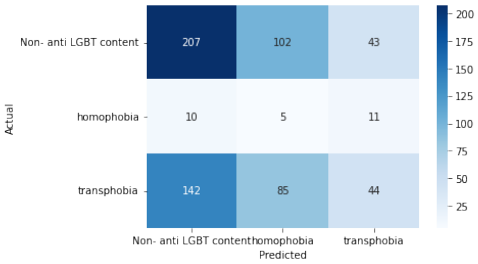
This paper presents the methodology for identifying the sentiment polarities and Homophobia detection from YouTube social media comments in Tamil, Malayalam, and Kannada code-mixed languages. Our group only employed a few preprocessing methods. We experimented with a pre-trained Multilingual BERT model with input variations for shared tasks A and B in all languages. It is evident from the evaluation that optimizing mBERT architecture receives better scores. For job A, our model mBERT + Ruled-based systems works admirably in two languages. Our model has achieved the leading position for both Kannada-English and Malayalam-English language pairings. Our model under



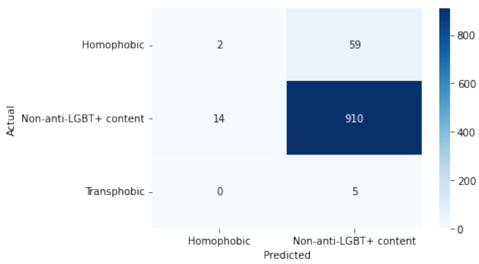
(a) mBERT + Ruled-based systems



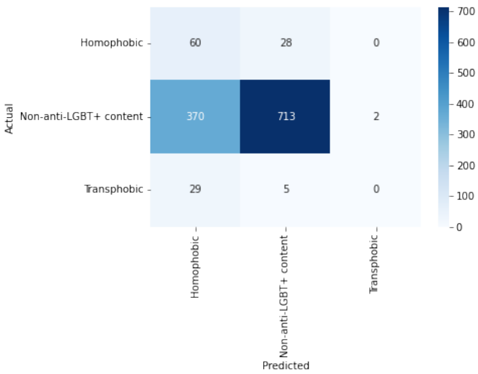
(b) mBERT + Ruled-based systems



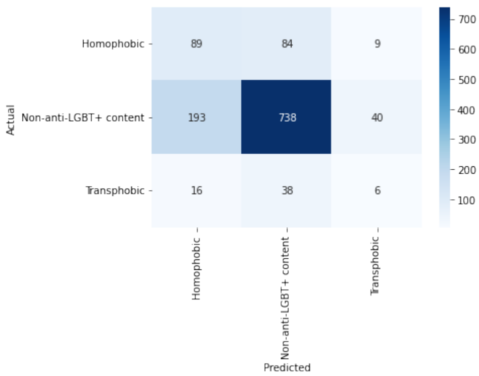
(c) mBERT



(d) mBERT



(e) mBERT



(f) mBERT

Figure 2: Confusion matrices for all submissions on the corpus test set. (a) Task A (Kannada-English), (b) Task A (Malayalam-English), (c) Task B (Tamil), (d) Task B (English), (e) Task B (Tamil-English), (f) Task B (Malayalam)

performed on task B. In Tamil, English, English-Tamil, and Malayalam, we placed fourth, fifth, sixth, and eighth rank, respectively. After addressing the class imbalance, we may use several deep learning methods to enhance task B's performance (homophobia detection). To prevent miss-classification, we will expand this work to additional languages and enhance efficiency by managing the indirect code-mixed comments.

References

[1] K. Shumugavadivel, M. Subramanian, P. K. Kumaresan, B. R. Chakravarthi, B. B, S. Chinnaudayar Navaneethakrishnan, L. S.K, T. Mandl, R. Ponnusamy, V. Palanikumar, M. Balaji J, Overview of the Shared Task on Sentiment Analysis and Homophobia Detection of YouTube Comments in Code-Mixed Dravidian Languages, in: Working Notes of FIRE 2022 - Forum for Information Retrieval Evaluation, CEUR, 2022.

[2] B. G. Patra, D. Das, A. Das, Sentiment Analysis of Code-Mixed Indian Languages: An Overview of SAIL Code-Mixed Shared Task @ICON-2017, 2018. arXiv:1803.06745.

[3] P. Patwa, G. Aguilar, S. Kar, S. Pandey, S. PYKL, B. Gambäck, T. Chakraborty, T. Solorio, A. Das, Semeval-2020 task 9: Overview of sentiment analysis of code-mixed tweets, 2020. arXiv:2008.04277.

[4] A. R. Appidi, V. K. Srirangam, D. Suhas, M. Shrivastava, Creation of corpus and analysis in code-mixed Kannada-English Twitter data for emotion prediction, in: Proceedings of the 28th International Conference on Computational Linguistics, International Committee on Computational Linguistics, Barcelona, Spain (Online), 2020, pp. 6703–6709. URL: https://aclanthology.org/2020.coling-main.587. doi:10.18653/v1/2020.coling-main.587.

[5] B. G. Patra, D. Das, A. Das, R. Prasath, Shared task on sentiment analysis in indian languages sail tweets - an overview, in: Proceedings of the Third International Conference on Mining Intelligence and Knowledge Exploration - Volume 9468, MIKE 2015, Springer-Verlag, Berlin, Heidelberg, 2015, p. 650–655. URL: https://doi.org/10.1007/978-3-319-26832-3_61. doi:10.1007/978-3-319-26832-3_61.

[6] G. Kovács, P. Alonso, R. Saini, Challenges of hate speech detection in social media, SN Comput. Sci. 2 (2021) 95.

[7] A. Saroj, S. Chanda, S. Pal, IRlab@IITV at SemEval-2020 task 12: Multilingual offensive language identification in social media using SVM, in: Proceedings of the Fourteenth Workshop on Semantic Evaluation, International Committee for Computational Linguistics, Barcelona (online), 2020, pp. 2012–2016. URL: https://aclanthology.org/2020.semeval-1.265. doi:10.18653/v1/2020.semeval-1.265.

[8] Z. Waseem, D. Hovy, Hateful symbols or hateful people? predictive features for hate speech detection on Twitter, in: Proceedings of the NAACL Student Research Workshop, Association for Computational Linguistics, San Diego, California, 2016, pp. 88–93. URL: https://aclanthology.org/N16-2013. doi:10.18653/v1/N16-2013.

[9] T. Davidson, D. Warmsley, M. Macy, I. Weber, Automated hate speech detection and the problem of offensive language, Proceedings of the International AAAI Conference on Web and Social Media 11 (2017) 512–515. URL: https://ojs.aaai.org/index.php/ICWSM/article/view/14955.

[10] S. Chanda, S. Ujjwal, S. Das, S. Pal, Fine-tuning pre-trained transformer based model for hate speech and offensive content identification in english indo-aryan and code-mixed (english-hindi) languages, in: P. Mehta, T. Mandl, P. Majumder, M. Mitra (Eds.), Working Notes of FIRE 2021 - Forum for Information Retrieval Evaluation, Gandhinagar, India, December 13-17, 2021, volume 3159 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2021, pp. 446–458. URL: http://ceur-ws.org/Vol-3159/T1-44.pdf.

[11] B. R. Chakravarthi, R. Priyadharshini, R. Ponnusamy, P. K. Kumaresan, K. Sampath, D. Thenmozhi, S. Thangasamy, R. Nallathambi, J. P. McCrae, Dataset for identification of homophobia and transophobia in multilingual youtube comments, arXiv preprint arXiv:2109.00227 (2021).

[12] B. R. Chakravarthi, R. Priyadharshini, T. Durairaj, J. McCrae, P. Buitelaar, P. Kumaresan, R. Ponnusamy, Overview of the shared task on homophobia and transphobia detection in social media

comments, in: Proceedings of the Second Workshop on Language Technology for Equality, Diversity and Inclusion, Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 369–377. URL: https://aclanthology.org/2022.ltedi-1.57. doi:10.18653/v1/2022.ltedi-1.57.
[13] B. R. Chakravarthi, V. Muralidaran, R. Priyadharshini, J. P. McCrae, 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), European Language Resources association, Marseille, France, 2020, pp. 202–210. URL: https://www.aclweb.org/anthology/2020.sltu-1.28.
[14] A. Hande, R. Priyadharshini, B. R. Chakravarthi, 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, Association for Computational Linguistics, Barcelona, Spain (Online), 2020, pp. 54–63. URL: https://www.aclweb.org/anthology/2020.peoples-1.6.
[15] B. R. Chakravarthi, N. Jose, S. Suryawanshi, E. Sherly, J. P. McCrae, 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), European Language Resources association, Marseille, France, 2020, pp. 177–184. URL: https://www.aclweb.org/anthology/2020.sltu-1.25.
[16] S. Chanda, S. Pal, Irlab@ iitbhu@ dravidian-codemix-fire2020: Sentiment analysis for dravidian languages in code-mixed text (2020).
[17] S. Chanda, R. Singh, S. Pal, Is meta embedding better than pre-trained word embedding to perform sentiment analysis for dravidian languages in code-mixed text?, in: FIRE, 2021.