

# An Analysis of Transformer-based Models for Code-mixed Conversational Hate-speech Identification

Neeraj Kumar Singh<sup>1</sup>, Utpal Garain<sup>1</sup>

<sup>1</sup>Indian Statistical Institute, ISI, Kolkata, India

## Abstract

The current surge in social media usage has resulted in the widespread availability of harmful and hateful content. Such inflammatory content identification in social media is a crucial NLP problem. Recent research has repeatedly demonstrated that context-level semantics matter more than word-level semantics for assessing the existence of hate content. This paper investigates many state-of-the-art transformer-based models for hate content detection in code-mixed datasets. We emphasize transformer-based models since they capture context-level semantics. In particular, we concentrate on Google-MuRIL, XLM-Roberta-base, and Indic-BERT. Additionally, we have experimented with an ensemble of the three mentioned models. Based on substantial empirical evidence, we observe that Google-MuRIL emerges as the top model with macro F1-scores of **0.708** and **0.445** for HASOC shared tasks 1 and 2, placing us 1<sup>st</sup> and 6<sup>th</sup> on the overall leaderboard standings respectively.

## Keywords

Hinglish, BERT, Codemixed Language, HateSpeech, Offensive Tweets

## 1. Introduction

Due to the accessibility of the internet, many people engage in a social media interaction on sites like Facebook, Instagram, Twitter, Sharechat, etc. These platforms are completely free and very user-friendly. The problem arises when a user or group of users share content to spread some propaganda, like hate speech, fake news, racial biases towards a group of people, etc., by using these platforms. These platforms have developed some rules. If these rules are broken, the post might be deleted or the user's account might be temporarily suspended. Manual moderation is not a solution given the volume of content being produced on these sites. Hence, these platforms are looking towards automatic moderation systems. Automated hate-offensive speech identification is a vital task as a result of this issue.

The majority of research on the identification of hate speech is restricted to English. After Mandarin and English, Hindi is the third most widely spoken language in the world. Despite this, it is consistently regarded as a low-resource language because it is mostly represented typographically. There are rarely any efforts made to identify hateful or offensive content in other Indian languages like Marathi, Gujrati, Dravid, Bangla, etc.

---

Forum for Information Retrieval Evaluation, December 9-13, 2022, India

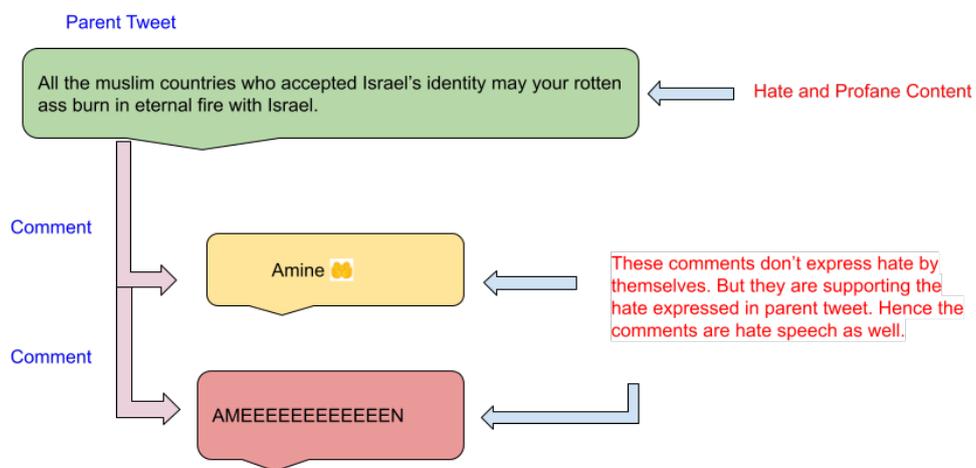
✉ neeraj1909@gmail.com (N. K. Singh); utpal@isical.ac.in (U. Garain)



© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

The difficulty of constructing local languages in universal key inputs has led to a recent trend in code-mixed data as well. The several alternative interpretations of the Hinglish words in various contextual settings make automated categorization challenging in the case of code-mix Indic languages like Hinglish, where Hindi is expressed in romanized English. When we deal with conversational speaking in the Hinglish language, it also gets very challenging. When identifying hate speech, the conversation's context is absolutely essential. Whether one agrees or disagrees with the previous comment or the prevailing philosophy in the discourse, it is possible to develop hatred for a certain target group, as shown in Figure 1. If the context of the parent tweet is taken into consideration, it is possible to determine whether a conversational thread contains hate or inflammatory material, even though it is not always obvious from a single comment or a reply to a comment that it does.



**Figure 1:** Regarding the debate that was occurring in Israel at the time, the parent tweet expresses anger and contempt toward Muslim nations. In the two responses to the tweet, the word "Amine," which in Persian means "honestly," was used. which, in the context of the parent, is encouraging hatred..

Earlier, in HASOC 2020[1], a dataset was launched for the identification of hate speech in Dravidian-CodeMix Tenglish and Manglish (Tamil and Malayalam written using the Roman script), and in HASOC 2021 [2], to detect hate speech in conversational code-mixed Hinglish, a new dataset was launched (Hindi language written using the Roman script instead of the Devanagari script). This time, HASOC [3] has added three projects as a continuation of the prior work. Task 1 is available for Hinglish and German and focuses on identifying hate speech and offensive language. Task 1 involves classifying conversational codemix tweets into two categories: hateful and offensive (HOF) and non-hateful and offensive (NOT). The second aim is to categorise conversational hate speech in the code-mixed language Hinglish into multiple classes. Task 3 focuses on the identification of hate speech and abusive language in Marathi. We participated only in Task 1 and Task 2, Identification of Conversational Hate-Speech in Code-

Mixed Languages (ICHCL). In this study, we show our transformer-based BERT model-based system. The code for this shared task is available at <https://github.com/neeraj1909/HASOC-2022>.

## 2. Related Work

Conversational hate speech detection in the codemix dataset is a challenging task. A majority of work has been done using the various transformer-based pre-trained models for the first time at HASOC 2021 [2, 4, 5, 6]. The transformer-based pre-trained architecture of BERT [7] is used by [8], more specifically mBERT [7] and XLM-Roberta [9]. An ensemble model of the three Transformer based architectures Indic-BERT [10], Multilingual BERT [11], and XLM-Roberta [9] is used by [12]. For feature extraction, [13] used TF-IDF, Word2Vec, Emo2Vec, and Hashtag vector. An ensemble of Random Forest, Multilayer Perceptron, and Gradient Boosting is used for the classification task. Transformer-based XLM-Roberta [9] model for the classification task is used by [14]. Works like [15] used codemix data augmentation using WordNet. They tried several different models, like Logistic Regression(LR) with word-level TF-IDF, Convolutional Neural Network(CNN) with word-level TF-IDF, and finetuned BERT pre-trained model. FastText library is used by [16] for the identification of hate speech. An ensemble of three different Transformer based models Ernie2.0 [17], Twitter Roberta Base Offensive [18], and HateBERT [19] is used by [20].

## 3. Dataset

Datasets [21] have been sampled from Twitter. To lessen the impact of bias, organisers have selected contentious stories on several subjects that are likely to feature hateful, offensive, and profane posts. The controversial stories are as follows:

- Temple-Mosque Controversy
- Covid Controversy
- Common Civil Code
- Hinduphobia
- Namaz on public place
- Farmer Protest
- Historical Hindu Muslim
- Islamophobia
- Russian-Ukrainian conflict etc.

### 3.1. Task-1: Finding Hate-Offensive Content in Conversational Hinglish-German Code-Mixed Languages

Task 1's major goal is the coarse-grained binary classification of conversational hate speech and offensive language, primarily for Hinglish and German. HOF denotes a tweet, comment, or reply that promotes or involves hate speech or other rude or obscene languages, as opposed to a tweet, comment, or reply that is free of any harmful, vulgar, or hateful language. The dataset statistics for Hinglish and German for binary classification are shown in Table 1.

Category	Dataset Size
Hate and Offensive (HOF)	2612
Non Hate-Offensive (NOT)	2609
Total	5221

**Table 1**  
Data distribution for Task 1 (ICHCL Binary Classification)

Category	Dataset Size
Contextual Hate (CHOF)	888
Standalone Hate (SHOF)	1636
Non-Hate (NONE)	2390
Total	4914

**Table 2**  
Data distribution for Task 2 (ICHCL Multi-Class Classification)

### 3.2. Task 2: Classifying Conversational Hate Speech in Hinglish Code-Mixed Languages

For task 2 (multi-class classification), the HOF class is classified into three sub-classes: Standalone Hate (SHOF), Contextual Hate (CHOF), and Non-Hate (NONE). It's not always possible to tell from a single comment or reply to a comment whether or not a conversational thread contains hate-offensive information. It is simple to identify by providing the parent tweet's context. As seen in Figure 1, the response is positive, but only insofar as it expresses hatred for the person who posted the original tweet in the remark. It validates the venom that was written in the comment. It is therefore also hate speech. Table 2 contains the dataset statistics for the conversational code-mixed data for multi-class classification.

### 3.3. Data Preprocessing

We flatten each interaction into a separate parent-comment-reply chain before feeding the data into our model. We concatenate the tweets and add a "[SEP]" token to each tweet, comment, and reply to help users distinguish one from the next. Every instance has a final label applied to it that corresponds to the last comment in the discussion chain. Except for the comma (,), full-stop (.), bar (|), and question mark (?), we removed all punctuation from the tweets. By eliminating URLs, mentions, and new-line characters, we clean up the data. We replaced the emojis with their CLDR short names. For multi-class classification, the target class's frequency is highly imbalanced. For our model to take into account this issue, we have assigned different weights to each class by dividing the frequency of each class by the size of the dataset and subtracting this result from one.

## 4. Methodology

Three different transformer model types were used in our studies; they are explained below:

- **XLM-Roberta**: A masked language model built on transformers and pre-trained on 100 languages. On numerous cross-lingual benchmarks, it produced cutting-edge performance. Facebook AI published this model in 2019.
- **Indic-BERT**: a pre-trained ALBERT model that has been trained on a sizable dataset of 12 Indic languages, including Assamese, Bengali, English, Gujarati, Hindi, Kannada, Malayalam, Marathi, Oriya, Punjabi, Tamil, and Telugu. There are roughly 9 billion tokens in the multilingual corpus.
- **Google MuRIL**[22]: In 2020, Google released it. It is a multi-lingual BERT for Indic languages. MuRIL-BERT was pre-trained in 17 Indic languages, including English, Assamese, Bengali, Gujarati, Hindi, Kannada, Kashmiri, Malayalam, Marathi, Nepali, Oriya, Punjabi, Sanskrit, Sindhi, Tamil, Telugu, and Urdu.

#### 4.1. Binary Classification

Using binary cross-entropy loss, we improved BERT transformers and trained our linear classifier layer on top of them. Each BERT model had a dropout layer on top of which added a fully connected linear layer. The representation of the CLS token served as the input to this fully connected layer.

#### 4.2. Multi-class Classification

In this task, the target class's frequency is highly imbalanced. To deal with this class-imbalance issue, we have used the weights for each class. We have assigned different weights to each class by dividing the frequency of each class by the size of the dataset and subtracting it from one.

The ensemble of the three models together has also been put to the test. For our purposes, we decided to combine the output of the three models using the majority voting process. There are two ways to perform a majority voting system:

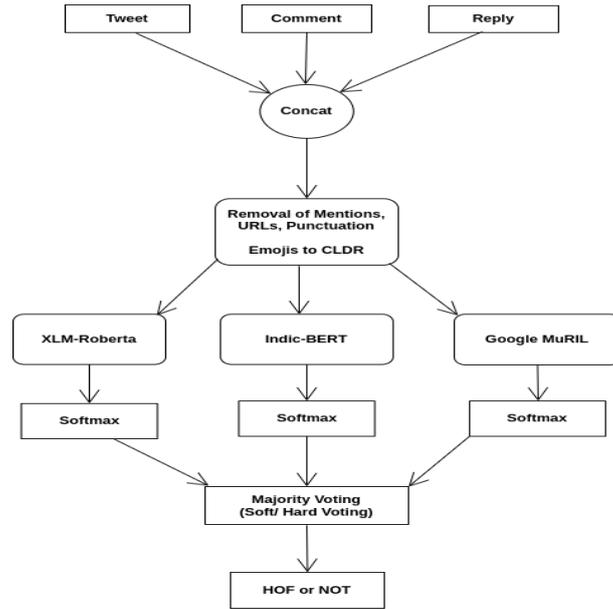
- **Hard Voting Ensemble**: In this instance, the model chooses the prediction class with the most votes out of all of the fine-tuned transformer models.
- **Soft Voting Ensemble**: The model in this case sums the class probabilities from all of the fine-tuned transformer models and selects the class with the highest sum of probabilities.

#### 4.3. Tuning Parameters

We utilised a batch size of 64 and a maximum sequence length of 512 for all models. We employed early halting on the validation loss with patience of 10 epochs to obtain the optimised learnable parameters. A  $2e-5$  initial learning rate Adam optimizer was employed.

### 5. Results

We divided the training data into three groups for evaluation purposes: a train set, a validation set, and a test set, with a ratio of 80:10:10. We track the top model using validation loss at each



**Figure 2:** Ensemble Model for Task 1

Model	Accuracy(%)	Precision	Recall	F1 Macro
<b>Google-MuRIL</b>	0.78	0.78	0.78	0.78
<b>XLM-Roberta</b>	0.72	0.72	0.72	0.72
<b>Indic-Bert</b>	0.71	0.71	0.70	0.70
<b>Soft Voting Ensemble</b>	0.67	0.67	0.66	0.66
<b>Hard Voting Ensemble</b>	0.67	0.67	0.66	0.66

**Table 3**  
Performance on ICHCL Binary Classification Task (Task 1)

epoch. Using the Google-MuRIL BERT, we are achieving the best results in binary code-mixed categorization. In the case of binary classification, we have found that the Google-MuRIL model’s macro F1-score is 5–6% higher than the XLM–Roberta–Base model. We utilised various random seeds to test our models and found that their performance was largely the same.

### 5.1. Results for Training Data

For the binary classification problem, Table 3 compares the results of Google-MuRIL, XLM-Roberta-base, Indic-BERT, and an ensemble of these three models. Table 4 displays the classification outcomes for multi-class categorization.

### 5.2. Results for Test Data

In accordance with the findings from the validation data (in section 5.1), we adjusted the model’s hyperparameters for the entire training set of data. We turned in the five runs for tasks 1 and 2

Model	Accuracy(%)	Precision	Recall	F1 Macro
<b>Google-MuRIL</b>	0.60	0.56	0.56	0.56
<b>XLM-Roberta</b>	0.54	0.54	0.53	0.53
<b>Indic-Bert</b>	0.55	0.55	0.54	0.54
<b>Soft Voting Ensemble</b>	0.55	0.55	0.55	0.55
<b>Hard Voting Ensemble</b>	0.55	0.55	0.55	0.55

**Table 4**  
Performance on ICHCL Multi-Class Classification Task (Task 2)

Task Name	Submission Name	F1 Macro	Precision	Recall
Task 1	binary_muril_ichcl	0.7083	0.7121	0.7091
Task 2	multiclass_muril_ichcl	0.4448	0.5248	0.4575

**Table 5**  
Final results

that were shared. We may deduce from the leaderboard that Google-MuRIL is the best model for both shared tasks, as indicated in Table 5.

## 6. Conclusion

We have compared the outcomes for various transformer-based BERT architectures in this research. In both tasks, we found that Google-MuRIL outperforms all alternative transformer-based systems. It has also been seen that changing the random seed does not change the model performance. We have also observed that the performance of all three BERT models is better than the ensemble model. So, some of the actions will be to speculate on the reason behind them.

## 7. Acknowledgement

This research is funded by the Science and Engineering Research Board (SERB), Dept. of Science and Technology (DST), Govt. of India through Grant File No. SPR/2020/000495.

## References

- [1] T. Mandl, S. Modha, A. Kumar M, B. R. Chakravarthi, 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, 2020, pp. 29–32.
- [2] S. Modha, T. Mandl, G. K. Shahi, H. Madhu, S. Satapara, T. Ranasinghe, M. Zampieri, Overview of the hasoc subtrack at fire 2021: Hate speech and offensive content identification in english and indo-aryan languages and conversational hate speech (2021) 1–3.
- [3] Satapara, Shrey and Majumder, Prasenjit and Mandl, Thomas and Modha, Sandip and Madhu, Hiren and Ranasinghe, Tharindu and Zampieri, Marcos and North, Kai and Pre-

- masiri, Damith, Overview of the HASOC Subtrack at FIRE 2022: Hate Speech and Offensive Content Identification in English and Indo-Aryan Languages, in: FIRE 2022: Forum for Information Retrieval Evaluation, Virtual Event, 9th-13th December 2022, ACM, 2022.
- [4] F. M. P. del Arco, S. Halat, S. Padó, R. Klinger, Multi-task learning with sentiment, emotion, and target detection to recognize hate speech and offensive language (2021).
  - [5] 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: Forum for Information Retrieval Evaluation (Working Notes)(FIRE), CEUR-WS. org, 2021.
  - [6] R. Rajalakshmi, S. Srivarshan, F. Mattins, E. Kaarthik, P. Seshadri, Conversational hate-offensive detection in code-mixed hindi-english tweets (2021).
  - [7] J. D. M.-W. C. Kenton, L. K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of naacL-HLT, 2019, pp. 4171–4186.
  - [8] S. Banerjee, M. Sarkar, N. Agrawal, P. Saha, M. Das, Exploring transformer based models to identify hate speech and offensive content in english and indo-aryan languages, arXiv preprint arXiv:2111.13974 (2021).
  - [9] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, Unsupervised cross-lingual representation learning at scale, arXiv preprint arXiv:1911.02116 (2019).
  - [10] D. Kakwani, A. Kunchukuttan, S. Golla, N. Gokul, A. Bhattacharyya, M. M. Khapra, P. Kumar, Indicnlp suite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for indian languages, in: Findings of the Association for Computational Linguistics: EMNLP 2020, 2020, pp. 4948–4961.
  - [11] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805 (2018).
  - [12] Z. M. Farooqi, S. Ghosh, R. R. Shah, Leveraging transformers for hate speech detection in conversational code-mixed tweets, arXiv preprint arXiv:2112.09986 (2021).
  - [13] A. Hegde, M. D. Anusha, H. L. Shashirekha, An ensemble model for hate speech and offensive content identification in indo-european languages, in: Forum for Information Retrieval Evaluation (Working Notes)(FIRE), CEUR-WS. org, 2021.
  - [14] A. Kadam, A. Goel, J. Jain, J. S. Kalra, M. Subramanian, M. Reddy, P. Kodali, T. Arjun, M. Shrivastava, P. Kumaraguru, Battling hateful content in indic languages hasoc '21, arXiv preprint arXiv:2110.12780 (2021).
  - [15] M. S. Jahan, M. Oussalah, J. Mim, M. Islam, Offensive language identification using hindi-english code-mixed tweets, and code-mixed data augmentation, in: Forum for Information Retrieval Evaluation (Working Notes)(FIRE), CEUR-WS. org, 2021.
  - [16] N. P. Motlogelwa, E. Thuma, M. Mudongo, T. Leburu-Dingalo, G. Mosweunyane, Leveraging text generated from emojis for hate speech and offensive content identification, in: Forum for Information Retrieval Evaluation (Working Notes)(FIRE), CEUR-WS. org, 2021.
  - [17] Y. Sun, S. Wang, Y. Li, S. Feng, H. Tian, H. Wu, H. Wang, Ernie 2.0: A continual pre-training framework for language understanding, in: Proceedings of the AAAI Conference on Artificial Intelligence, volume 34, 2020, pp. 8968–8975.
  - [18] F. Barbieri, J. Camacho-Collados, L. Neves, L. Espinosa-Anke, Tweeteval: Unified benchmark and comparative evaluation for tweet classification, arXiv preprint arXiv:2010.12421

(2020).

- [19] T. Caselli, V. Basile, J. Mitrović, M. Granitzer, Hatebert: Retraining bert for abusive language detection in english, arXiv preprint arXiv:2010.12472 (2020).
- [20] B. Chinagundi, M. Singh, T. Ghosal, P. S. Rana, G. S. Kohli, Classification of hate, offensive and profane content from tweets using an ensemble of deep contextualized and domain specific representations (2021).
- [21] S. Modha, T. Mandl, P. Majumder, S. Satapara, T. Patel, H. Madhu, Overview of the HASOC Subtrack at FIRE 2022: Identification of Conversational Hate-Speech in Hindi-English Code-Mixed and German Language , in: Working Notes of FIRE 2022 - Forum for Information Retrieval Evaluation, CEUR, 2022.
- [22] S. Khanuja, D. Bansal, S. Mehtani, S. Khosla, A. Dey, B. Gopalan, D. K. Margam, P. Aggarwal, R. T. Nagipogu, S. Dave, et al., Muril: Multilingual representations for indian languages, arXiv preprint arXiv:2103.10730 (2021).