Natural Language Processing (2025), **31**, pp. 393–414
doi:10.1017/nlp.2024.28

CAMBRIDGE UNIVERSITY PRESS

ARTICLE

Hate speech detection in low-resourced Indian languages: An analysis of transformer-based monolingual and multilingual models with cross-lingual experiments

Koyel Ghosh and Apurbalal Senapati

Department of Computer Science and Engineering, Central Institute of Technology, Kokrajhar, Assam, India
Corresponding author: Koyel Ghosh; Email: ghosh.koyel8@gmail.com

(Received 25 December 2022; revised 31 August 2023; accepted 25 October 2023; first published online 27 August 2024)

Special Issue on '**Natural Language Processing Applications for Low-Resource Languages**'

Abstract

Warning: This paper is based on hate speech detection and may contain examples of abusive/ offensive phrases.

Cyberbullying, online harassment, etc., via offensive comments are pervasive across different social media platforms like ™Twitter, ™Facebook, ™YouTube, etc. Hateful comments must be detected and eradicated to prevent harassment and violence on social media. In the Natural Language Processing (NLP) domain, the most prevalent task is comment classification, which is challenging, and language models based on transformers are at the forefront of this advancement. This paper intends to analyze the performance of language models based on transformers like BERT, ALBERT, RoBERTa, and DistilBERT on the Indian hate speech datasets over binary classification. Here, we utilize the existing datasets, i.e., HASOC (Hindi and Marathi) and HS-Bangla. So, we evaluate several multilingual language models like MuRIL-BERT, XLM-RoBERTa, etc., few monolingual language models like RoBERTa-Hindi, Maha-BERT (Marathi), Bangla-BERT (Bangla), Assamese-BERT (Assamese), etc., and perform cross-lingual experiment also. For further analyses, we perform multilingual, monolingual, and cross-lingual experiments on our **H**ate **S**peech **Assamese** (HS-Assamese) (Indo-Aryan language family) and **H**ate **S**peech **Bodo** (HS-Bodo) (Sino-Tibetan language family) dataset (HS dataset version 2) also and achieved a promising result. The motivation of the cross-lingual experiment is to encourage researchers to learn about the power of the transformer. Note that no pre-trained language models are currently available for Bodo or any other Sino-Tibetan languages.

Keywords: hate speech detection; multilingual; monolingual; cross-lingual; transformer

1. Introduction

The Cambridge Dictionary defines hate speech as follows: "*Hate speech is a public speech that expresses hate or encourages violence towards a person or group based on race, religion, sex, or sexual orientation.*"[a] It has been estimated that half of the world's population uses social media[b] and that users spend over 12½ trillion hours per year online.[c] This current trend is rapidly expanding. Social violence, including riots, has been caused by aggressive online behavior such as false news,

[a] https://dictionary.cambridge.org/dictionary/english/hate-speech
[b] https://datareportal.com/reports/digital-2021-global-overview-report
[c] https://datareportal.com/reports/digital-2022-global-overview-report

abusive comments, and hostile online communities (Laub 2019). Governments worldwide are enacting anti-hate speech laws.[d] As a result, online platforms like ™Twitter, ™Facebook, etc., are becoming increasingly aware of the issue and working to prevent the spread of hate speech, sexual harassment, cyberbullying, and other forms of abuse.

Most hate speech detection research is conducted using European language.[e] Except for the publication of datasets and the improvement in accuracy, very little is done to study Indian languages further. According to Rajrani and Ashok (2019), the Eighth Schedule of the Indian Constitution lists 22 languages and over 1,000 living languages from different linguistic families. Users on social media platforms in India often post in their native languages, which might make it difficult to detect hate speech computationally because of improper syntax or grammar. Because of this incident, we investigated hate speech in online comments. Hate speech detection in text is difficult for machine learning techniques. Researchers increasingly adopt advanced transformer models to improve performance in fields like NLP, information retrieval, and others that deal with language. Named entity recognition (Luoma and Pyysalo 2020), question answering (McCarley *et al.* 2019), token classification (Ulčar and Robnik-Šikonja 2020), text classification (Sun *et al.* 2019), etc., are the significant famous field of NLP researchers. Hate speech detection is closely related to text classification.

India is diverse in language, so many language-specific studies and research have been done. Hate speech is subjective and context-dependent most of the time, so in the Indian context, it is a very challenging problem. Most Indian languages are under-resourced. This study aims to detect hate content in comments written in Hindi, Marathi, Bangla, Assamese, and Bodo gathered from social media platforms. We used HASOC (Hate Speech and Offensive Content Identification)[f] and HS-Bangla (Hate Speech Bangla)[g] datasets, employing binary classification methods. Hindi, Marathi, and Bangla are almost moderately spoken languages in India, and they have the advantage in the computational fields where annotated data is required. Regardless, this is not sufficient; at least they are available. However, some low-resource languages suffer simultaneously due to a shortage of annotated data. Considering the scarcity, we have prepared an Assamese and a Bodo dataset for the hate speech detection task, which will be publicly available shortly. Both languages are one of the 22 scheduled languages in India, primarily spoken in the Northeastern region of India–Assamese shares the Bangla-Assamese script, which has evolved from the Kamrupi script. On the other hand, Bodo pronounced as Boro, is mainly spoken in India's Northeastern part of the Brahmaputra valley. The language is part of the Sino-Tibetan language family under the Assam-Burmese group and shares the Devanagari script. The 2011 Indian Census[h] (Census 2011a, 2011b) estimates a total of 1,482,929 Bodo speakers, including 1,454,547 native speakers. Assamese is spoken by 15,311,351 people, which is a huge number. In NLP research, very few resources are available on Assamese and Bodo, which leads to less advancement.

We take advantage of the multilingual and monolingual language models based on transformers, which have brought attention to low-resource languages. Multilingual and monolingual language models are pre-trained on BERT (Bidirectional Encoder Representations from Transformers) (Devlin *et al.* 2018), RoBERTa (Robustly optimized BERT) (Liu *et al.* 2019), ALBERT (A Lite BERT) (Lan *et al.* 2019), DistilBERT (Distilled version of BERT) (Sanh *et al.* 2019a). We utilize few pre-trained multilingual models like m-BERT (Devlin *et al.* 2018), MuRIL-BERT (Khanuja *et al.* 2021), and monolingual models as well such as NeuralSpaceHi-BERT (Jain *et al.* 2020), RoBERTa-Hindi, Indic-BERTt (Kakwani *et al.* 2020), Maha-BERT (Joshi 2022a), Maha-RoBERTa (Joshi 2022b), XLM-RoBERTa (Conneau *et al.* 2019), Bangla-BERT (Sarker 2020), Assamese-BERT (Joshi 2023) etc. We fine-tuned different multilingual and monolingual

[d] https://www.un.org/en/hate-speech/united-nations-and-hate-speech/international-human-rights-law
[e] https://hatespeechdata.com/
[f] https://hasocfire.github.io/hasoc/2019/index.html
[g] https://www.kaggle.com/naurosromim/bengali-hate-speech-dataset
[h] https://censusindia.gov.in/census.website/data/census-tables

https://doi.org/10.1017/nlp.2024.28 Published online by Cambridge University Press

pre-trained BERT models for all the languages. Later, an analysis-based study compares the performance of different pre-trained language models on Hindi, Marathi, Bangla, Assamese, and Bodo datasets. In our case, the monolingual models are only pre-trained in Hindi, Marathi, Bangla, and Assamese data. Next, we follow a cross-lingual experiment (Litake *et al.* 2022) which includes only monolingual models, since Hindi-Marathi-Bodo shares the Devanagari script, where Assamese-Bangla shares similar scripts except for few characters (Ghosh *et al.* 2012). The experiments are a hurricane test on the dataset to see whether the experiments give a promising result or not. We found a promising result, which is another contribution of this paper. So, our experiment can help researchers to work with under-resource languages.

Our focus in this work is :

- We prepare two hate speech text datasets in Assamese and Bodo languages, consisting of two classes: "NOT" and "HOF."
- Subsequently, we fine-tuned pre-trained language models on three existing (Hindi, Marathi, and Bangla) and two new datasets (Assamese and Bodo) to evaluate their performance in handling a new language.
- To assess the performance of language models, we employed mono and multilingual variants of transformer-based language models (TLMs).
- A comprehensive comparison among all models has been conducted for the mentioned languages. Around forty-one experiments were carried out: twelve for Hindi and Marathi each, seven for Bangla, and five for Bodo and Assamese individually.
- In the end, a cross-lingual experiment between Hindi-Marathi-Bodo and Assamese-Bangla is done where monolingual Marathi models, i.e., Maha-BERT, Maha-RoBERTa, RoBERTa-Base-Mr, and Maha-AlBERT, perform well on the Hindi dataset. NeuralSpaceHi-BERT, RoBERT-Hindi, and DistilBERTHi, which are monolingual Hindi models, also perform well in the case of the Marathi dataset.

The flow of the paper is structured as follows. Section 2 covers the relevant literature on detecting hate speech in Indian languages. Section 3 describes the existing as well as new datasets. Section 4 explains detailed methodology like preprocessing steps, transformer-based models, experimental settings, etc. Section 5 presents the results and findings obtained from the comprehensive experiments conducted. Finally, conclusions are drawn in Section 6.

2. Related work

Extensive research exists in hate speech detection for European languages, but there is a significant gap regarding Indian languages. This scarcity arises from the limited availability of publicly accessible datasets for NLP tasks, including hate speech detection. Creating datasets for hate speech detection is particularly demanding as it requires extensive groundwork and data preprocessing, such as cleaning the data, ensuring annotator agreement on hate speech identification, and transforming raw social media content into valuable training data. In this section, we shall discuss the existing research specific to Indian languages to address this challenge.

2.1 Hindi

In 2019, a shared task called HASOC (Mandl *et al.* 2019) released its first collection of datasets focusing on hate speech detection in Indian languages like Hindi and Marathi. HASOC is a collaborative effort organized by FIRE,[i] the Forum for Information Retrieval Evaluation. HASOC

[i] http://fire.irsi.res.in/fire/2022/home

https://doi.org/10.1017/nlp.2024.28 Published online by Cambridge University Press

(Indo-Aryan Languages) included subtask A, a binary classification task for identifying hateful text in Hindi, which is relevant to our objective. The winning team QutNocturnal (Bashar and Nayak 2020) for subtask A achieved good results in identifying hateful text. They used a convolutional neural network (CNN) with Word2Vec embeddings and achieved Marco-F1 of 0.8149 and weighted F1 of 0.8202. LGI2P (Mensonides *et al.* 2019) team also achieved strong results on subtask A. Their system utilized a fastText model to learn word representations in Hindi, followed by BERT for classification. This approach produced a Marco-F1 score of 0.8111 and a weighted F1 score of 0.8116. The HASOC also included another task (subtask B) focusing on classifying the type of hate speech, such as whether it's profane or abusive (multiclass). The team 3Idiots (Mishra and Mishra 2019) achieved promising results on subtask B using BERT, obtaining Marco-F1 and weighted F1 scores of 0.5812 and 0.7147, respectively. Subtask C focused on identifying whether hate speech targets a specific group or individual, i.e., targeted or untargeted (multiclass). Team A3-108 (Mujadia, Mishra, and Sharma 2019) achieved the best results on this task, with a Marco-F1 score of 0.5754. Their approach relied on Adaboost (Freund and Schapire 1997), a machine learning algorithm that outperformed other options like Random Forest (RF) and Support Vector Machines (SVMs) for this specific task. Interestingly, combining these three classifiers using a technique called "ensemble with hard voting" yielded even better results. They utilized TF-IDF to extract features from word unigrams (individual words) and character sequences of varying lengths (2 g to 5 g), including tweet length as a feature. Subtask A (binary class) and subtask B (multiclass) are offered with a new Hindi dataset in HASOC 2020 (Mandl *et al.* 2020). Team NSIT_ML_Geeks (Raj, Srivastava, and Saumya 2020) utilize CNN and Bidirectional long short-term memory (BiLSTM) to beat other teams with the Marco-F1 score of 0.5337 in subtask A and 0.2667 in subtask B. Nohate (Kumari) fine-tuned BERT model and gained Marco-F1 0.3345 in subtask B. In HASOC-2021, on a newly published Hindi dataset (Modha *et al.* 2021), the best submission was performed macro F1 0.7825 in subtask A where authors fine-tuned multilingual BERT (m-BERT) upto 20 epochs with a classifier layer added at the final phase. The second team also fine-tuned (m-BERT) and scored macro F1 0.7797. NeuralSpace (Bhatia *et al.* 2021) got macro F1 0.5603 in subtask B. They use an XLM-R transformer, vector representations for emojis using the system Emoji2Vec, and sentence embeddings for hashtags. After that, three resulting representations were concatenated before classification. In other independent work, authors (Bhardwaj *et al.* 2020) prepared the hostility detection dataset in Hindi and applied the pre-trained m-BERT model for computing the input embedding. Later, classifiers such as SVM, RF, Multilayer perceptron (MLP), and Logistic Regression (LR) models were employed. In coarse-grained evaluation, SVM achieved the highest weighted F1 score of 84% whereas LR, MLP, and RF scores of 84%, 83%, and 80%, respectively. In fine-grained evaluation, SVM displays the highest F1 scores across three hostile dimensions: Hate (47%), Offensive (42%), and Defamation (43%). Logistic Regression outperforms other models in the Fake dimension, achieving an F1 score of 68%. Authors (Ghosh *et al.* 2023c) presented a multitasked framework for hate and aggression detection on social media data. They used a transformer-based approach like XLM-RoBERTa. Ghosh and Senapati (2022) present an analysis of multi and monolingual language models on three Indian languages like Hindi, Marathi, and Bangla.

2.2 Marathi

Team WLV-RIT (Nene *et al.* 2021) uses XLM-R Large model with a simple softmax layer for fine-tuning on Marathi dataset of HASOC-2021 (Modha *et al.* 2021) and secure the first rank. dataset named OffensEval 2019 (Zampieri *et al.* 2019) and Hindi data released for HASOC 2019 (Mandl *et al.* 2019) are used for the performance. The authors established that transfer learning from Hindi is better than learning from English, with a score of 0.9144 (macro F1). The second team scores 0.8808, fine-tuning LaBSE transformer (Feng *et al.* 2020) on the Marathi and Hindi datasets. LaBSE transformer (Glazkova *et al.* 2021) outperforms XLM-R in the monolingual settings, but

https://doi.org/10.1017/nlp.2024.28 Published online by Cambridge University Press

XLM-R performs better when Hindi and Marathi data are merged. The first huge Marathi hate dataset on text is presented by L3Cube-MahaHate (Velankar *et al.* 2022) with 25,000 distinct tweets from ^{TM}Twitter, later annotated manually, and labeled them into four major classes, i.e., hate, offensive, profane, and not. Finally, they use CNN, Long short-term memory (LSTM), and Transformers. Next, they study monolingual and multilingual variants of BERT like Maha-BERT, Indic-BERT, m-BERT, and XLM-RoBERTa, showing that monolingual models perform better than their multilingual counterparts. Their Maha-BERT (Joshi 2022a) model provides the best results on L3Cube-MahaHate Corpus.

2.3 Bangla

Karim *et al.* (2020) prepared a dataset with 35,000 hate statements (political, personal, geopolitical, and religious) in Bangla and analyzed the data by combining multichannel CNN and LSTM-based networks. Later, more than 5,000 labeled examples were added to the previous dataset, and an extended version, i.e., DeepHateExplainer (Karim *et al.* 2021), was published. Authors used an ensemble method of transformer-based neural architectures to classify them into political, personal, geopolitical, and religious hates. They achieved F1 scores of 78%, 91%, 89%, and 84% for political, personal, geopolitical, and religious hates. In the paper (Romim *et al.* 2021), they prepared a Bangla Hate Speech corpus with 30,000 comments labeled with "1" for hate comments; otherwise, "0." Authors (Mandal, Senapati, and Nag 2022) produced a political news corpus and then developed a keyword or phrase-based hate speech identifier using a semi-automated approach. Authors (Romim *et al.* 2022) created a Bangla dataset that includes 50,200 offensive comments. Here, they did binary classification and multilabel classification using BiLSTM and SVM.

2.4 Assamese

In the NLP field, Assamese is a very low-resourced language. Some recent works have been done on the Assamese language. In Nath *et al.* (2023), authors present AxomiyaBERTa, which is a novel BERT model for Assamese, a morphologically rich low-resource language of Eastern India. Authors (Das and Senapati 2023) present a co-reference Resolution Tagged Data set in the Assamese dataset applying a semi-automated approach as co-reference resolution is an essential task in several NLP applications. Authors (Laskar *et al.* 2023) work on English to Assamese translation using the transformer-based neural machine translation. From the source-target sentences, they extract alignment information and they have used the pre-trained multilingual contextual embeddings-based alignment technique. In the paper Laskar *et al.* (2022), the authors investigate the negation effect for English to Assamese machine translation.

2.5 Bodo

The NLP language technologies remain challenging due to resource constraints, research interest, and the unavailability of primary research tools. Historically, Bodo has a rich literature with a large corpus of oral history in the form of stories, folk tales, etc. Scholars suggested the presence of a lost Bodo script called "Deodhai." Following the script movement, Bodo adopted the use of the Devanagari script. Bodo is one of the low-resource languages out of the scheduled languages of India. Various studies in the field of NLP have recently been undertaken. Bodo Wordnet (Bhattacharyya 2010) is another one of such first initiatives. With the initial efforts on the development of language tools and corpus undertaken by the Government of India. Datasets currently available are mainly due to efforts by Technology Development for Indian Languages (TDIL-DC). One such initiative is English to Indian Language Machine Translation (EILMT) consortia, under

https://doi.org/10.1017/nlp.2024.28 Published online by Cambridge University Press

which the tourism domain English-Bodo parallel corpus consisting of 11,977 sentences was made and released for research purposes. EILMT Consortium developed an English-Bodo Agriculture text corpus consisting of 4,000 sentences and a Health Corpus of 12,383 parallel pairs. Indian Languages Corpora Initiative phase-II (ILCI Phase-II) project initiated by MeitY, Government of India resulted in the creation of 37,768 sentences of Agriculture & Entertainment domain for the Hindi-Bodo language pair. Low-resource languages are limited in terms of resources and language technologies to build the corpus. Recently, Narzary *et al.* (2022) proposed a methodology to utilize available freely accessible tools like Google Keep to extract monolingual corpus from old books written in Bodo. The majority of the NLP research for Bodo is towards the problem of machine translation (MT) with the objective of building English to Bodo or vice versa MT system. One such work (Narzary *et al.* 2019) English to Bodo MT system for Tourism domain achieved BLEU score of 17.9. The landscape of other NLP tasks remains challenging due to the absence of an annotated corpus.

3. Dataset

We used some existing datasets for our experiments, and later, we created an Assamese and a Bodo hate speech dataset for the experiments.

3.1 Existing datasets

Here, we use three publically available datasets i.e., HASOC-Hindi (Mandl *et al*. 2019), HASOC-Marathi (Modha *et al*. 2021), and HS-Bangla (2021) (Romim *et al*. 2021).

3.1.1 HASOC-Hindi (2019) (Mandl et al. 2019)

We use the HASOC-Hindi dataset, which was published in 2019. The entire dataset was collected from ™Twitter and ™Facebook with the help of different hashtags and keywords. Annotators tagged the data with two classes: hate & offensive (*HOF*) and not hate (*NOT*). *HOF* implies that a post contains hate speech, offensive language, or both. *NOT* means the absence of hate speech or other offensive material in the post. This is a shared task data, so training and test data are available separately.

3.1.2 HASOC-Marathi (2021) (Modha et al. 2021)

This dataset is based on the released MOLD dataset (Gaikwad *et al.* 2021). MOLD contains data collected from ™Twitter. Authors used the hashtag #Marathi with 22 typical Marathi curse words and search terms for politics, entertainment, and sports. More than a total of 2,547 tweets were collected and were annotated by six native annotators. The final MOLD dataset contains 2,499 annotated tweets after removing non-Marathi tweets.

3.1.3 HS-Bangla (2021) (Romim et al. 2021)

Researchers gather vast amounts of data from social media (™Twitter, ™Facebook pages and groups, ™LinkedIn), Bangla articles from various sources, including a Bangla Wikipedia dump, Bangla news articles like Daily Prothom Alo, Anandbazar Patrika, BBC, news dumps of TV channels (ETV Bangla, ZEE news), blogs, and books. The scraped text corpus consists of 250 million articles. This dataset consists of 30,000 instances, where 10,000 instances belong to the *hate* category, and 20,000 instances belong to *non-hate*. Hate comments are additionally categorized as political, gender abusive, personal, religious, or geopolitical hate.

https://doi.org/10.1017/nlp.2024.28 Published online by Cambridge University Press

Table 1. Class-wise distribution for HASOC-Hindi (2019), HASOC-Marathi (2021), and HS-Bangla(2021) dataset

	HOF/hate		NOT/non-hate		
Datasets	Train	Test	Train	Test	Total
HASOC-Hindi (2019)	2,469	605	2,196	713	5,983
HASOC-Marathi (2021)	669	207	1,205	418	2,499
HS-Bangla (2021)	8,000	2,000	16,000	4,000	30,000

(a)

text	task_1
क्लिक करें और पढ़ें इस स्मार्टफोन के बारे में. Translation - (Click and read about this smartphone.)	NOT
अबे दल्ले कौन बोलता है कि तू पत्रकार है। भड़वा और दलाल है तू।।Translation - (Hey Dalle, who says that you are a journalist. You are a bhadva and a pimp.)	HOF

HASOC-Hindi (2019) dataset sample

(b)

text	task_1
सुशिक्षित माणूस सुसंस्कारित असेलच असे नाही. Translation - (An educated person is not necessarily a cultured person.)	NOT
अरे मुखा तेच तर सांगतोय महापौरांच्या. Translation - (Oh fool, that's what the mayor is saying.)	HOF

HASOC-Marathi (2021) dataset sample

(c)

sentence	hate	category
খুব তাড়াতাড়ি রায় কার্যকর করা হোক এইটাই দাবি । Translation - (This is the demand that the judgment should be implemented very soon)	0	crime
তোর কপালে জুতা মারি শালার পুত । Translation - (Son of a rascal, I will thrash you with a shoe on your forehead.)	1	sports

HS-Bengali (2021) dataset sample

Figure 1. Dataset samples of (a) HASOC-Hindi (2019), (b) HASOC-Marathi (2021), and (c) HS-Bangla (2021) datasets, respectively.

Table 1 shows the training and test dataset statistics along with *HOF* (hate) and *NOT* (Not hate) class distribution for Hindi, Marathi, and Bangla datasets.

Figure 1 shows some snaps of the datasets, and the task is the binary classification for this paper, i.e., detecting whether a sentence or text conveys hate or not. For HASOC-Hindi (2019) and HASOC-Marathi (2021), classes are *HOF* and *NOT* whereas in HS-Bangla (2021), classes are *1* (hate) and *0* (non-hate).

3.2 New corpus creation

We have created our hate speech dataset in the Assamese and Bodo languages, i.e., (HS-Assamese and HS-Bodo). This dataset is the extended version 2 and well-updated NEIHS (version 1) (Ghosh *et al*. 2023a, 2023b) datasets. We briefly discuss the data generation process in this section.

3.2.1 Dataset collection

Assamese and Bodo data have been collected mainly from ™Facebook and ™YouTube. We observed that the comment sections of political, news, celebrity, entertainment-based, etc. ™Facebook pages or ™YouTube channels are the most toxic. So, we target specific uploads and

https://doi.org/10.1017/nlp.2024.28 Published online by Cambridge University Press

Table 2. Class distribution analysis for Training and test set HS-Assamese and HS-Bodo datasets, respectively

	HOF		NOT		
Datasets	Train	Test	Train	Test	Total
HS-Assamese	2,347	608	1,689	401	5,045
HS-Bodo	1,130	266	885	216	2,497

fetch the hate and not hate comments using open-source scrapper tools.[j] On the internet, data are primarily English transliteration and contain unwanted symbols –, ’, (,), etc. We cleaned the data using an automatic or manual approach (if required) and translated the sentences. Finally, some native speakers annotate the comments either *HOF* or *NOT*. Sentences with *HOF* that include hate words are considered hate-offensive statements, while sentences that convey formal information, suggestions, or questions are considered non-hate sentences.

3.2.2 Dataset annotation

Two native speakers annotate the data for each language. Both annotators are young adults (aged 19 to 24). The annotators are all Central Institute of Technology undergraduate students, Kokrajhar. The annotation team manually assigns binary labels (hate and non-hate) to all Bodo comments to indicate the presence or absence of hateful content. When two students disagreed on a label, a third student with experience in social media research made the final call. Hate speech is a highly subjective issue. As a result, defining what constitutes hate speech is difficult. As a result, we’ve established specific strict guidelines. These regulations are based on the community standards of ™Facebook[k] and ™YouTube.[l] Comments with the following aims should be marked as hate. (a) *Profanity*: Comments that contain profanity, cussing, or swear words are marked as hate. (b) *Sexual orientation*: Sexual attraction (or a combination of these) to people of the opposite sex or gender, to people of the same sex or gender, to both sexes, or to people of more than one gender. (c) *Personal*: remarks on clothing sense, content selection, language selection, etc. (d) *Gender chauvinism*: Comments in which people are targeted because of their gender. (e) *Religious*: Comments in which the person is criticized for their religious beliefs and practices. For example, comments challenging the use of a turban or a burkha (the veil), (f) *Political*: Comments criticizing a person’s political beliefs. For instance, bullying people for supporting a political party. (g) *Violent intention*: Comments containing a threat or call to violence.

3.2.3 Dataset analysis

We summarize class distribution statistics of the HS-Assamese and HS-Bodo datasets in Table 2. Out of 2,497 comments in our HS-Bodo dataset, 1,396 contain hate speech, and for HS-Assamese, out of 5,045 comments, 2,955 are hate, and 2,090 are non-hate. As a result, our data set is slightly skewed in favor of containing hate speech. We split the dataset into a training set and test set by 80:20. In the HS-Bodo training set, 1,130 comments are hate out of 2,015 comments, and 266 are hate out of 482 in the test set. Figure 2 shows the HS-Assamese and HS-Bodo datasets sample.

[j] https://github.com/kevinzg/facebook-scraper
[k] https://web.facebook.com/communitystandards/
[l] https://www.youtube.com/howyoutubeworks/policies/community-guidelines/

https://doi.org/10.1017/nlp.2024.28 Published online by Cambridge University Press

(a)

text	task_1	Hate category
চাল্লা কি মানুহ Translation - (What kind of man is he?)	NOT	non-hate
দেখাত জেহাদি জেহাদি লাগে Translation - (You look like a Jihadi.)	HOF	Personal Attack
মুঠ আপ সমর্থক--৮০% মিঞা(তাৰে ৫০% ফেক নামত কমেণ্ট দিয়ে বাকী ৩০% নিজ নামত) গল ৮০% মিঞা। বাকী থাকিল ২০% বদন(এইকেইটা হল কংগ্রেছ অখিল সমর্থক)" Translation - (Total AAP supporters-- 80% Miya(50% of them are fake accounts and the other 30% is their real accounts), after 80% Miya's. The remaining ones are Badan (These are the supporters of Congress and Akhil)	HOF	Political Hate

HS-Assamese dataset sample

(b)

text	task_1	Hate category
आंग सानो जाय बाराद्राय बखियो बे जेबो मावनो रोङा Translation - (Those who constantly abuse others will achieve nothing.)	NOT	non-hate
हनै मालाय हारसा हिनजावनि खिबु सुग्राफोरा मिनिसोदों Translation - (Look, these women bum cleaners are laughing.)	HOF	Gender attack
सोरबा माबा मोनसे खामानि मावनो थांनायाव मानि हेंथा गिखफोर ? Translation - (Why was there always a barrier when they were going to work for the good ?)	NOT	non-hate

HS-Bodo dataset sample

Figure 2. Samples of (a) HS-Assamese and (b) HS-Bodo datasets where hate comments are tagged as *HOF* and otherwise *NOT*.

4. Methodology

Our experiment is mainly done on several transformer-based BERT models. We utilize three publicly available datasets: HASOC-Hindi (2019), HASOC-Marathi (2021), and HS-Bangla (2021). Later, two new hate speech datasets on the Assamese language (HS-Assamese) and Bodo language (HS-Bodo) were created for all the experiments.

4.1 Problem definition

The task aims to classify a given text as either *HOF* or *NOT*. Our dataset is D, consists of p texts, represented as $\{t_1, t_2, t_3, \ldots, t_i, \ldots t_p\}$, where t_i denotes the i^{th} text and p is the total number of texts present in the dataset. Each text t_i consists of m words, denoted by $t_i = \{w_{i,1}, w_{i,2}, w_{i,3}, \ldots, w_{i,k}, \ldots, w_{i,m}\}$, where $w_{i,k}$ indicates the k^{th} word in the i^{th} text. The dataset D is defined as $D = \{(t_i, y_i)\}_{i=1}^{p}$, where the i^{th} text t_i is labeled as either *HOF* or *NOT*, denoted as y_i. Thus, $D = \{(t_1, y_1), (t_2, y_2), (t_3, y_3), \ldots, (t_i, y_i), \ldots, (t_p, y_p)\}$, where each tuple consists of the text (t_i) and label (y_i) corresponding to the text. This hate speech detection is a binary classification task, and the goal of the task is to maximize the value of the function

$$\text{argmax}_{\theta} \left(\Pi_{i=1}^{p} \left(P(y_i | t_i; \theta) \right) \right), \tag{1}$$

where t_i represents a text with an associated label y_i, which is to be predicted. θ is the model parameter that needs to be optimized. The approach is to develop a classifier for a task where texts must be organized into two classes. First, two parts of datasets are there: one is training, and another one is test datasets. These two datasets aim to train the classifier on the training dataset and assess its performance on the test dataset. The model differentiates between the two classes by examining the processed text data. During the learning phase, the algorithm adjusts its internal settings based on the training data, improving its ability to make accurate predictions. These adjustments are driven by the differences between the two classes in the training data. The classifier becomes trained during the learning process, which can now identify the given text data class. This system is tested on new, unseen text to assure reliability.

4.2 Preprocessing

Any deep learning or transformer model needs cleaned and noise-free data. So, preprocessing is necessary to enhance performance. Researchers use almost similar preprocessing approaches for the same category languages. Datasets include raw comments with punctuation, URLs, emojis, and unwanted characters. In most circumstances, the following actions are employed.

https://doi.org/10.1017/nlp.2024.28 Published online by Cambridge University Press

Normalization. Existing emojis removal, undesirable characters, and stop-words from sentences.

Punctuation removal. Punctuations are removed except ".", "?" and "!" as these are considered delimiters to tokenize each sentence.

Label encoding. Labels (task_1) for HASOC-Hindi (2019), HASOC-Marathi (2021), HS-Assamese, and HS-Bodo are labeled as *NOT* and *HOF*. We encode these labels into a distinctive number. *NOT* is converted to *0* and *HOF* to *1*, where we leave the HS-Bangla (2021) dataset as it is labeled with the numeral already.

We followed the steps mentioned above for HASOC-Marathi (2021), HS-Bangla (2021), HS-Assamese, and HS-Bodo datasets. We perform preprocessing strategies as mentioned in paper (Bashar and Nayak 2020) for HASOC-Hindi (2019), like URL occurrence with `xxurl`, replacing person occurrence (e.g., @someone) with `xxatp`, source of modified retweet with `xxrtm`, source of not modified retweet with `xxrtu`, fixing the repeating characters (e.g., goooood), removed familiar invalid characters (e.g., $< br =>$, $< unk >$, $@ - @$, etc.) and a lightweight stemmer for Hindi language (Ramanathan and Rao 2003) for stemming the words.

4.3 Transformer-based Language Models (TLMs)

4.3.1 Input representation

After basic preprocessing step our texts are $\{t_1, t_2, t_3, \ldots, t_i, \ldots t_p\}$. Each word is then tokenized and three embeddings token, segment, and position embeddings are combined to obtain a fixed-length vector. For every model separate tokenizer is used, like BERT, RoBERTa, ALBERT, DistilBERT uses WordPiece (Wu *et al.* 2016), Byte Pair Encoding (Shibata *et al.* 1999), SentencePiece (Kudo and Richardson 2018), and WordPiece correspondingly. Later, [*CLS*] is added for classification, and [*SEP*] separates input segments. Figure 3(a) shows the input representations for TLMs. So, the preprocessed text t_i having m words: $t_i = \{w_{i,1}, w_{i,2}, w_{i,3}, \ldots, w_{i,k}, \ldots, w_{i,m}\}$. Now, a word embedding layer, position embedding, and segment embedding convert each token into its vector representation.

$$w_{i,k} = \{WordEmbedding + PositionEmbedding + SegmentEmbedding\}, \tag{2}$$

4.3.2 Transformer

For our task, transformers play a vital role. So, a brief introduction to the transformer is needed. Vaswani *et al.* (2017) represent transformers, a sequence-to-sequence architecture (Seq2Seq) is a type of neural network designed to transform a given sentence's sequence of words into a different sequence. Seq2Seq models excel at translation tasks, converting a sequence of words from one language into another. These models comprise an encoder and a decoder. The Encoder transforms the input sequence into a higher-dimensional space (an n-dimensional vector).

In our task, hate speech detection, which is more similar to text classification, only uses a transformer encoder block. Figure 3(b) shows the transformer encoder block. The transformer models below (BERT, RoBERTa, ALBERT, DistilBERT, etc.) are trained as language models. These language models, like BERT, RoBERTa, etc., include two stages: (1) pre-training and (2) fine-tuning. The pre-training involves two self-supervised tasks; one is Masked Language Modeling (MLM), and the other is Next Sentence Prediction (NSP). MLM is to predict randomly masked input tokens, and NSP predicts whether two sentences are adjacent or not. In the pre-trained phase, these models have been trained extensively on vast quantities of unprocessed text using a self-supervised approach. Self-supervised learning involves the model calculating its objective based on input data, indicating that human labeling of data is not required. At this phase, while this model gains a statistical comprehension of the trained language, its practical utility for specific tasks is limited. To address this, the broad pre-trained model undergoes transfer learning. The

https://doi.org/10.1017/nlp.2024.28 Published online by Cambridge University Press

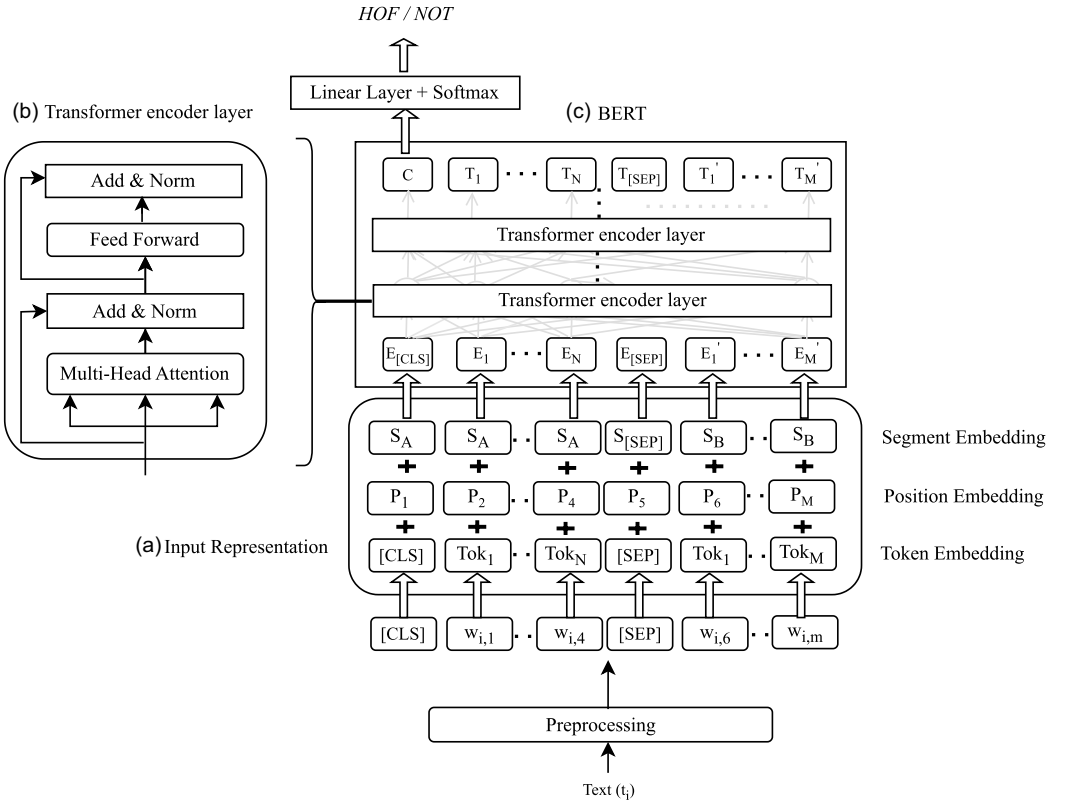

Figure 3. Architecture of hate speech detection model which includes (a) input representation, (b) transformer encoder block, and (c) BERT model.

model is refined in this phase using supervised techniques involving human-provided labels for a particular task. In our experiments, we use several existing pre-trained TLMs and fine-tune them for the specific task i.e., hate speech detection. During fine-tuning, one or more fully connected layers are added on top of the last transformer layer with the *softmax* activation function. Figure 3(c) shows the fine-tuning part of the model, and at the final transformer layer, we include a linear layer with *softmax*.

4.3.3 BERT

Google developed BERT, a transformer-based technique for NLP. BERT can generate embeddings with specific contexts. It generates vectors almost identical for synonyms but distinct when used in different contexts. During training, it learns the details from both sides of the word's context. So, it is called a bidirectional model. We tested Hindi, Marathi, and Bangla-BERT datasets to compare monolingual and multilingual BERT.

1. **m-BERT** (Devlin *et al.* 2018) is prepared with Wikipedia content in 104 top languages, including Hindi, Bangla, and Marathi, utilizing a masked language modeling (MLM) objective using the largest Wikipedia as the training set.
2. **MuRIL-BERT** (Khanuja *et al.* 2021), MuRIL is a BERT model that has already been trained on 17 Indian languages and their transliterated counterparts, including monolingual segments and parallel segments.

https://doi.org/10.1017/nlp.2024.28 Published online by Cambridge University Press

3. **NeuralSpaceHi-BERT** (Jain *et al.* 2020), thanks to its extensive pre-training on the 3 GB monolingual OSCAR corpus made available by neuralspace-reverie, this is ready to use. Text classification, POS tagging, question answering, etc., were all fine-tuned.
4. **Maha-BERT** (Joshi 2022a) uses L3Cube-MahaCorpus and other publicly accessible Marathi monolingual datasets to fine-tune a multilingual BERT (bert-base-multilingual-cased) model.
5. **Bangla-BERT** (Sarker 2020) bangla-Bert-Base was pre-trained using OSCAR and the Bangla Wikipedia Dump Dataset with the help of MLM.
6. **Assamese-BERT** (Joshi 2023) is a monolingual BERT model trained on publicly available Assamese monolingual datasets.

4.3.4 RoBERTa

BERT can benefit from more time spent training on a large dataset. Using a character-level BPE (Byte Pair Encoding) tokenizer, RoBERTa, a self-supervised transformer model trained on raw texts, beats BERT by 4%-5% in natural language inference tasks. However, RoBERTa employs a byte-level BPE tokenizer, which takes advantage of a standard encoding format.

1. **XLM-RoBERTa (base-sized model)** (Conneau *et al.* 2019) is a multilingual RoBERTa model that has been pre-trained on 2.5 TB of cleaned CommonCrawl data in 100 different languages. In contrast to XLM multilingual models, it does not rely on *lang* tensors to identify the language being used and select it appropriately from the input ids.
2. **Roberta-Hindi**[m] is RoBERTa transformer base model, which was pre-trained on a large Hindi corpus (a combination of MC4, OSCAR, and indic-nlp datasets) and released by flax-community.
3. **Maha-RoBERTa** (Joshi 2022b) is a multilingual RoBERTa (xlm-roberta-base) model fine-tuned on publicly available Marathi monolingual datasets and L3Cube-MahaCorpus.
4. **RoBERTa-Base-Mr** is a RoBERTa Marathi model, which was pre-trained on *mr* dataset of C4 (Colossal Clean Crawled Corpus) (Raffel *et al.* 2019) multilingual dataset.

4.3.5 ALBERT

As a lightweight alternative to BERT for self-supervised learning, Google AI released ALBERT.

1. **Indic-BERT** (Kakwani *et al.* 2020) is a multilingual ALBERT model containing 12 major Indian languages (including Hindi, Marathi, Bangla, Assamese, English, Gujarati, Oriya, Punjabi, Tamil, Telugu, Kannada, and Malayalam) was recently released by Ai4Bharat. This model was trained on large-scale datasets.
2. **Maha-ALBERT** (Joshi 2022a) is a Marathi ALBERT model trained on L3Cube-MahaCorpus and Marathi monolingual datasets made available to the public.

4.3.6 DistilBERT

DistilBERT is a lightweight transformer model that is tiny, fast, and cheap, thanks to its training on the BERT base. This version of BERT has 40% fewer parameters and runs 60% faster than the previous version while retaining over 95% of its performance on the GLUE language understanding benchmark.

1. **m-DistilBERT** (Sanh *et al.* 2019b) is trained using all 104 of Wikipedia's language versions.
2. **DistilBERTHi**[n], using OSCAR's monolingual training dataset, this DistilBERT language model has already been pre-trained.

[m] https://huggingface.co/flax-community/roberta-hindi
[n] https://huggingface.co/neuralspace/indic-transformers-hi-distilbert

https://doi.org/10.1017/nlp.2024.28 Published online by Cambridge University Press

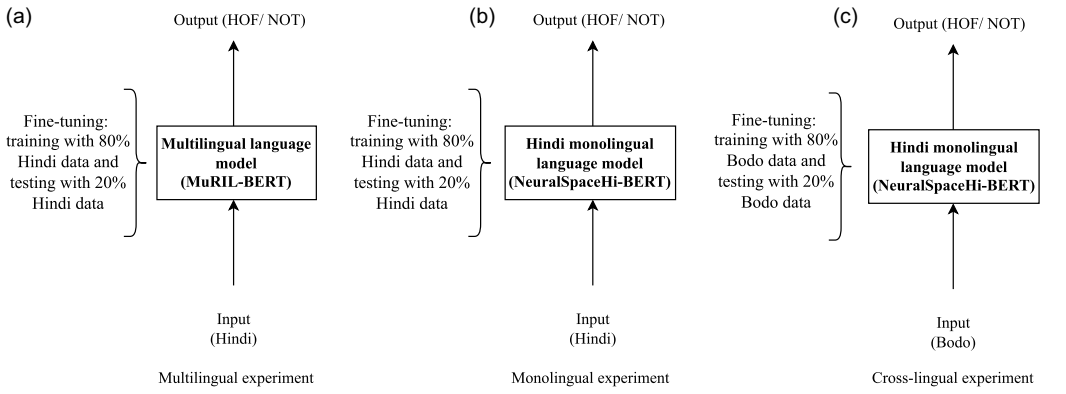

Figure 4. Experiments of hate speech detection model which includes (a) Multilingual experiment, (b) Monolingual experiment, and (c) Cross-lingual experiment.

4.4 Experiments

Our main motive for all the experiments is to fine-tune the existing pre-trained TLMs models with Hindi, Marathi, Bangla, Assamese, and Bodo task-specific datasets. We performed three experiments multilingual, monolingual, and cross-lingual experiments. All three experiments and Forty-one sub-experiments have been performed with different pre-trained models.

4.4.1 Multilingual experiment

This is an experiment where existing multilingual pre-trained TLMs are fine-tuned on task-specific datasets. In our case, we fine-tune existing multilingual pre-trained TLMs like m-BERT, MuRIL-BERT, XLM-RoBERTa, Indic-BERT, m-DistilBERT on Hindi, Marathi, Bangla, Assamese, and Bodo datasets. Note that, existing multilingual pre-trained TLMs are not trained on the Bodo dataset previously. Figure 4(a) shows a very basic overview of multilingual experiments with an example.

4.4.2 Monolingual experiment

Here, existing monolingual pre-trained TLMs are fine-tuned on task-specific datasets where pre-trained models and task-specific datasets belong to the same language. In our case, we fine-tune existing monolingual Hindi pre-trained TLMs like NeuralSpaceHi-BERT, Roberta-Hindi, and DistilBERTHi on the Hindi dataset. Existing monolingual Marathi pre-trained TLMs like Maha-BERT, Maha-RoBERTa, RoBERTa-Base-Mr, and Maha-AlBERT are fine-tuned on the Marathi dataset. Existing monolingual Bangla pre-trained TLM like Bangla-BERT is fine-tuned on the Bangla dataset. Lastly, existing monolingual Assamese pre-trained TLM like Assamese-BERT is fine-tuned on the Assamese dataset. Note that no monolingual Bodo pre-trained BERT model is available. So, we skip this experiment for the Bodo dataset only. Figure 4(b) shows a monolingual experiment with an example.

4.4.3 Cross-lingual experiment

In this experiment, existing monolingual pre-trained TLMs are fine-tuned to task-specific datasets where pre-trained models and task-specific datasets belong to different languages. We are considering the same language family and the same script for this experiment (Hindi—Marathi and Assamese—Bangla). In the case of Hindi and Marathi, both belong to the same Indo-Aryan language family and share the same script, i.e., Devnagri. Assamese and Bangla belong to the

https://doi.org/10.1017/nlp.2024.28 Published online by Cambridge University Press

Table 3. Hyperparameters for all the experiments

Hyperparameter	configuration
Learning-rate	1e-5
Epochs	10, 20
Max seq length	512
Batch size	3, 8

same language family, i.e., Indo-Aryan language family, and share almost the same script, i.e., Bangla-Assamese script, except for two letters. In the case of Bodo, which belongs to the Sino-Tibetan language family but shares the same script as Hindi and Marathi, we did the cross-lingual experiment on Hindi, Marathi, and Bodo too. For example, We fine-tune NeuralSpaceHi-BERT for Marathi and Bodo data, whereas we fine-tune Bangla-BERT for the Assamese language and Assamese-BERT for the Bangla data. Figure 4(c) shows a Cross-lingual experiment with an example.

4.5 Experimental setup

We execute all experiments with the same hyperparameter combination (Table 3) due to memory and *GPU* issues and pick the best result. We use Python-based libraries like Huggingface, PyTorch, and TensorFlow[o] at different stages of our implementations. We utilize the *GPU* of *Google Colab* for all our experiments.

5. Result and analysis

A total of forty-one models are ready, and now, an evaluation of their performance is required. We calculate precision, recall, and weighted F1 scores on the test set of Hindi, Marathi, Bangla, Assamese, and Bodo datasets. Table 4 represents the results of TLMs trained on the HASOC-Hindi (2019), HASOC-Marathi (2021), HS-Bangla (2021), HS-Assamese and HS-Bodo datasets, where simple, star (∗), and double stars (∗∗) indicate multilingual, monolingual, and cross-lingual models correspondingly. We offer both a weighted F1 score and an accuracy score for model evaluation due to the dominant issue of imbalanced class distribution in classification problems. So, a weighted F1 score is a more suitable metric to believe for the imbalanced class distribution scenario.

5.1 Evaluation metrics

We use two class precisions (P_{NOT}, P_{HOF}), recalls (R_{NOT}, R_{HOF}), F1 scores ($F1_{NOT}$, $F1_{HOF}$) to evaluate the models then calculate weighted precision (W_P), recall (W_R), and F1 score (W_{F1}) here. At last, we calculate *Accuracy*.

$$P_{NOT} = \frac{True_{NOT}}{True_{NOT} + False_{HOF}} \tag{3}$$

$$P_{HOF} = \frac{True_{HOF}}{True_{HOF} + False_{HOF}} \tag{4}$$

[o]https://huggingface.co/transformers/, https://pytorch.org/, https://www.tensorflow.org/

https://doi.org/10.1017/nlp.2024.28 Published online by Cambridge University Press

Table 4. Calculations of precision, recall, F1 score, and accuracy of various TLMs on HASOC-Hindi (2019), HASOC-Marathi (2021), HS-Bangla (2021), HS-Assamese, and HS-Bodo datasets, respectively

Models on	Precision			Recall			F1 score			Accuracy
HASOC-Hindi	0	1	w.avg.	0	1	w.avg.	0	1	w.avg.	
m-BERT	0.8078	0.7797	0.7949	0.8275	0.8016	0.8156	0.8175	0.7904	0.8050	0.8050
MuRIL-BERT	0.8695	0.8362	0.8542	0.8266	0.7851	0.8075	0.8475	0.8098	**0.8301**	**0.8308**
NeuralSpaceHi-BERT∗	0.8611	0.8278	0.8458	0.8263	0.7867	0.8081	0.8433	0.8067	0.8264	0.8270
Maha-BERT∗∗	0.8681	0.8297	0.8504	0.8080	0.7570	0.7845	0.8369	0.7916	0.8161	0.8171
XLM-RoBERTa	0.8218	0.7977	0.8107	**0.8492**	**0.8280**	**0.8394**	0.8352	**0.8125**	0.8247	0.8247
Roberta-Hindi∗	0.8485	0.8147	0.8329	0.8231	0.7851	0.8056	0.8356	0.7996	0.8190	0.8194
Maha-RoBERTa∗∗	**0.8892**	**0.8534**	**0.8727**	0.8138	0.7603	0.7892	**0.8498**	0.8041	0.8288	0.8300
RoBERTa-Base-Mr∗∗	0.8246	0.7906	0.8089	0.8155	0.7801	0.7992	0.8200	0.7853	0.8040	0.8042
Indic-BERT	0.7489	0.7198	0.7355	0.7864	0.7603	0.7744	0.7671	0.7394	0.7543	0.7541
Maha-AlBERT∗∗	0.8232	0.7913	0.8085	0.8221	0.7900	0.8073	0.8226	0.7906	0.8079	0.8080
m-DistilBERT	0.7812	0.7487	0.7662	0.7991	0.7685	0.7800	0.7900	0.7584	0.7754	0.7754
DistilBERTHi∗	0.8064	0.7781	0.7934	0.8261	0.8000	0.8141	0.8161	0.7888	0.8035	0.8034
Models on HASOC-Marathi										
m-BERT	0.9019	0.8110	0.8717	**0.9240**	**0.8502**	**0.8995**	0.9128	0.8301	0.8854	0.8848
MuRIL-BERT	0.8995	0.7878	0.8625	0.8805	0.7536	0.8384	0.8898	0.7703	0.8502	0.8512
NeuralSpaceHi-BERT∗∗	0.9066	0.8115	0.8751	0.9066	0.8115	0.8751	0.9066	0.8115	0.8751	0.8752
Maha-BERT∗	0.9234	0.8415	0.8962	0.9125	0.8212	0.8822	**0.9179**	**0.8312**	**0.8891**	**0.8896**
XLM-RoBERTa	0.8588	0.7242	0.8142	0.8734	0.7487	0.8320	0.8660	0.7336	0.8221	0.8224
Roberta-Hindi∗∗	0.9354	0.8540	0.9084	0.8886	0.7632	0.8470	0.9113	0.8060	0.8764	0.8784
Maha-RoBERTa∗	0.9306	0.8520	0.9045	0.9067	0.8067	0.8735	0.9184	0.8287	0.8886	**0.8896**
RoBERTa-Base-Mr∗	**0.9688**	**0.8960**	**0.9446**	0.8100	0.5410	0.7209	0.8823	0.6746	0.8135	0.8272
Indic-BERT	0.8708	0.6785	0.8071	0.7964	0.5507	0.7150	0.8319	0.6079	0.7577	0.7648
Maha-AlBERT∗	0.9138	0.8095	0.8792	0.8761	0.7391	0.8307	0.8945	0.7726	0.8541	0.8560
m-DistilBERT	0.8588	0.6878	0.8021	0.8233	0.6280	0.7586	0.8406	0.6565	0.7796	0.7824
DistilBERTHi∗∗	0.9066	0.7989	0.8709	0.8793	0.7487	0.8360	0.8927	0.7729	0.8530	0.8544
Models on HS-Bangla										
m-BERT	0.9303	0.8630	0.9078	0.9155	0.8362	0.8890	0.9228	0.8493	0.8983	0.8980
MuRIL-BERT	0.9225	0.8507	0.8985	**0.9406**	**0.8835**	**0.9215**	**0.9314**	**0.8667**	**0.9098**	**0.9095**
XLM-RoBERTa	**0.9463**	**0.8975**	**0.9300**	0.9047	0.8249	0.8781	0.9250	0.8596	0.9032	0.9023
Indic-BERT	0.9042	0.8030	0.8704	0.9300	0.8515	0.9038	0.9169	0.8265	0.8867	0.8876
Bangla-BERT∗	0.9333	0.8685	0.9117	0.9207	0.8456	0.8956	0.9269	0.8568	0.9035	0.9033

https://doi.org/10.1017/nlp.2024.28 Published online by Cambridge University Press

Table 4. Continued

	Precision			Recall			F1 score			Accuracy
	0	1	w.avg.	0	1	w.avg.	0	1	w.avg.	
Assamese-BERT**	0.9401	0.8584	0.9092	0.9107	0.8556	0.8856	0.9169	0.8468	0.8931	0.8910
m-DistilBERT	0.8698	0.7290	0.8228	0.9055	0.7941	0.8683	0.8872	0.7601	0.8448	0.8466
Models on HS-Assamese										
m-BERT	0.6312	0.7623	0.7023	**0.7632**	0.6143	0.6923	**0.6921**	0.6956	0.6912	0.6921
MuRIL-BERT	0.6334	**0.7643**	0.7046	0.7623	0.6321	0.6912	0.6913	0.6917	0.6934	0.6923
m-BERT-uncased	0.6423	0.7313	0.6912	0.6911	0.6834	0.6841	0.6616	0.7019	0.6812	0.6808
Bangla-BERT**	0.6123	0.7332	0.6745	0.7142	0.6324	0.6709	0.6639	0.6746	0.6745	0.6717
Assamese-BERT*	**0.6719**	0.7628	**0.7221**	0.6118	**0.8016**	**0.7317**	0.6421	**0.7823**	**0.7225**	**0.7306**
Models on HS-Bodo										
m-BERT	**0.9423**	**0.8445**	**0.8965**	0.7912	**0.9521**	**0.8734**	**0.8623**	**0.8934**	**0.8713**	**0.8812**
MuRIL-BERT	0.9323	0.8345	0.8865	0.7812	0.9421	0.8634	0.8523	0.8834	0.8613	0.8712
m-BERT-uncased	0.9012	0.8411	0.8745	**0.8054**	0.9232	0.8643	0.8541	0.8812	0.8623	0.8623
NeuralSpaceHi-BERT**	0.8821	0.8013	0.8434	0.7443	0.9123	0.8223	0.8012	0.8523	0.8321	0.8312
Maha-RoBERTa**	0.8719	0.8228	0.8421	0.7718	0.9016	0.8417	0.8221	0.8623	0.8425	0.8436

$$R_{NOT} = \frac{True_{NOT}}{True_{NOT} + False_{NOT}} \tag{5}$$

$$R_{HOF} = \frac{True_{HOF}}{True_{HOF} + False_{NOT}} \tag{6}$$

$$F1_{NOT} = 2 * \frac{P_{NOT} * R_{NOT}}{P_{NOT} + R_{NOT}} \tag{7}$$

$$F1_{HOF} = 2 * \frac{P_{HOF} * R_{HOF}}{P_{HOF} + R_{HOF}} \tag{8}$$

$$W_P = \frac{P_{NOT} * T_{NOT} + P_{HOF} * T_{HOF}}{T_{NOT} + T_{HOF}} \tag{9}$$

$$W_R = \frac{R_{NOT} * T_{NOT} + R_{HOF} * T_{HOF}}{T_{NOT} + T_{HOF}} \tag{10}$$

$$W_{F1} = \frac{F1_{NOT} * T_{NOT} + F1_{HOF} * T_{HOF}}{T_{NOT} + T_{HOF}} \tag{11}$$

$$Accuracy = \frac{True_{NOT} + True_{HOF}}{T_{NOT} + T_{HOF}} \tag{12}$$

https://doi.org/10.1017/nlp.2024.28 Published online by Cambridge University Press

where $True_{NOT}$ = True-negative (model predicted the texts as *NOT*, and the actual value of the same is also *NOT*), $True_{HOF}$ = True-positive (model predicted the texts as *HOF*, and the actual value of the same is also *HOF*), $False_{NOT}$ = False-negative (model predicted the texts as NOT, but the true value of the same is *HOF*), $False_{HOF}$ = False-positive (model predicted the texts as *HOF*, but the true value of the same is *NOT*), P_{NOT} = Precision of *NOT* class, P_{HOF} = Precision of *HOF* class, R_{NOT} = Recall of *NOT* class, R_{HOF} = Recall of *HOF* class, $F1_{NOT}$ = F1 score of *NOT* class, $F1_{HOF}$ = F1 score of *HOF* class, T_{NOT} = The total number of *NOT* class text present in test set, T_{HOF} = The total number of *HOF* class text present in test set.

5.2 Best TLMs per dataset

For the Hindi dataset, the weighted F1 score of four models MuRIL-BERT, Maha-RoBERTa, NeuralSpaceHi-BERT, and XLM-RoBERTa are very close. Maha-BERT, Maha-RoBERTa, m-BERT, and Roberta-Hindi scored at the top of the Marathi dataset. MuRIL-BERT, Bangla-BERT, XLM-RoBERTa, and m-BERT models are the scoring models for the Bangla dataset. For the Assamese dataset, the Assamese-BERT monolingual model scores are at the top. The m-BERT model performs best on the Bodo dataset, and MuRIL-BERT scores well, too. Figure 5 shows the confusion matrix of the best models on five datasets separately.

5.3 Multilingual models vs monolingual models

On the Hindi dataset, multilingual models like MuRIL-BERT and XLM-RoBERTa perform better, but the monolingual model NeuralSpaceHi-BERT also gives tough competition. We can conclude that multilingual models perform well, but the difference in performance between monolingual and multilingual models is negligible. Maha-BERT and Maha-RoBERTa models provide the highest weighted F1 score for the Marathi dataset, and m-BERT also performs well, whereas MuRIL-BERT scores a little less. We use two monolingual pre-trained models for Bangla, i.e., Bangla-BERT and Assamese-BERT; it performs very well, but the MuRIL-BERT wins marginally. Indic-BERT and m-DistilBERT models' performance is significantly less on all datasets than in other models. Therefore, developing better resources for the Hindi and Bangla languages is necessary, as language-specific fine-tuning does not guarantee the best performance. For the Assamese dataset, we observe that Assamese-BERT gives the top result. We fine-tune NeuralSpaceHi-BERT and Maha-RoBERTa with the Bodo dataset, and both of the models provide a little less result than the multilingual models. For the Bodo dataset, there are no existing monolingual models available, so we tried only two experiments, i.e., fine-tuning with multilingual models and cross-lingual experiments.

5.4 Cross-lingual experiments

The purpose of this experiment is to open a door for researchers who are dealing with low-resourced languages. During the cross-lingual experiments, we consider the Marathi models on the Hindi dataset and vice versa, as both languages share the Devanagari script. Maha-RoBERTa performs well on the Hindi dataset, and Maha-BERT, RoBERTa-Base-Mr, and Maha-AlBERT also score sufficiently. NeuralSpaceHi-BERT and Roberta-Hindi perform well on the Marathi dataset. Still, surprisingly, DistilBERTHi performs poorly on the Hindi dataset rather well on the Marathi dataset, though performance also depends on the amount and diversity of data. We also perform cross-lingual experiments on the Bangla and Assamese datasets, as they both share the same script. Surprisingly, Bangla-BERT on Assamese data and Assamese-BERT on the Bangla dataset perform well. In the Bodo dataset, we tried to fine-tune NeuralSpaceHi-BERT and Maha-RoBERTa, which is one kind of cross-lingual experiment. This kind of transfer learning also works well in cross-lingual experiments.

https://doi.org/10.1017/nlp.2024.28 Published online by Cambridge University Press

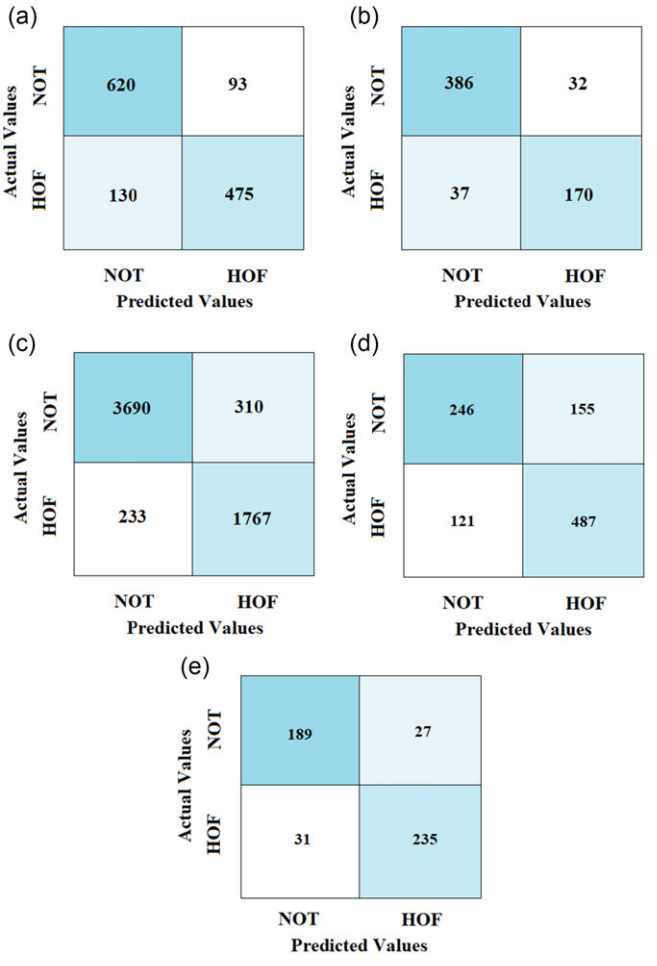

Figure 5. Confusion matrix of best models such as MuRIL-BERT for HASOC-Hindi (2019) (a), Maha-BERT for HASOC-Marathi (2021) (b), MuRIL-BERT for HS-Bangla (2021) (c), Assamese-BERT for HS-Assamese (d) and m-BERT for HS-Bodo (e).

5.5 How models gather knowledge!

All our experiments, like monolingual, multilingual, and cross-lingual, are based on transfer learning. We are using pre-trained TLMs and fine-tuning those models with our tagged hate dataset. In Section 4.3, we explain that pre-trained TLMs are already trained on huge amounts of multilingual raw data or only monolingual data in a self-supervised manner, and then we fine-tuned those existing pre-trained models with our small tagged datasets for a particular task. Here, TLMs need vast amounts of data for scratch training where low-resource languages suffer; then, we have to use existing pre-trained models. So, from a pre-trained model to a fine-tuning model, we are actually transferring the language-related knowledge. According to Rogers *et al.* (2020), TLMs gather syntactic knowledge, semantic knowledge, and world knowledge during the training procedure. Syntactic knowledge (Berwick 1985) is about understanding the rules and structures that help to arrange the words and phrases to build grammatically correct sentences in a particular language. Semantic knowledge (Patterson, Nestor, and Rogers 2007) is all about understanding the meaning of words, phrases, and sentences as well as the relationships among them. It also gathers information on similar or opposite words, contextual understanding, different senses of words,

https://doi.org/10.1017/nlp.2024.28 Published online by Cambridge University Press

etc. World knowledge (Hagoort *et al.* 2004) includes information about the history, country, society, common sense, culture, scientific principles, geographical, geopolitical, etc. TLMs gather all that information in some vector components via embedding for each word/ token. TLMs perform word embedding using vectors of length 768, which are contextual representations of words. It not only retrieves the important features but also captures the context information in a bidirectional manner of a word in a sentence. Hence, some components of the embedding vector store the features, and some combined components preserve the context information.

If pre-trained multilingual or monolingual models are used for fine-tuning with the same language data, then it is obvious that the model is learning the same language patterns, features, and structure because of the same vocabulary and syntax. Now, the existing pre-trained model is in one language and fine-tuned in another language, i.e., our cross-lingual experiment. In such cases, we are considering either the same language family (Hindi—Marathi and Assamese—Bangla) or the same script (Hindi—Marathi—Bodo and Assamese—Bangla). Each pair for the same language family shares linguistic features, patterns, and structures. The hateful language might share a similar kind of linguistic feature, and it is captured during the fine-tuning model. If it is so, then it will behave almost similarly to the other language of the same language family and hence perform well.

6. Conclusion

In major languages like English, significant work is done. A little work has been done on the Indian languages, like Hindi, Bangla, Marathi, Tamil, Malayalam, etc. In short, we conclude this work: (i) We explore variants of language models based on transformers in Indic languages. (ii) Two hate speech datasets have been created in the Assamese and Bodo languages named HS-Assamese and HS-Bodo. (iii) A comparison has been drawn on monolingual and multilingual language model which uses transformers for hate speech detection data like HASOC-Hindi, HASOC-Marathi, HS-Bangla, HS-Assamese, and HS-Bodo datasets. (iv) Cross-lingual experiments have been done successfully on the mentioned language pairs like Hindi-Marathi, Hindi-Bodo, Marathi-Bodo, and Bangla-Assamese. We can witness that monolingual training only sometimes ensures superior performance only if raw data are sufficient while scratch training is done. Multilingual models performed best on the Hindi, Bangla, and Bodo datasets, whereas monolingual models were superior on the Marathi and Assamese datasets. We also observe that the "0" class precision, recall, and F1 score is slightly higher than the "1" class, indicating the data imbalance. So, we can apply SMOTE (Bowyer *et al.* 2011), ADASYN (He *et al.* 2008), or data augmentation (techniques to increase the amount of data) (Nozza 2022) to handle data imbalance in the future. Research on hate speech affects both technical and socio-linguistic concerns, such as freedom of speech and legislation on both the national and international levels. We are preparing more new datasets and conducting experiments on the dataset, representing the first attempt at detecting hate speech in Northeast languages. We present a dataset of annotated anti-Bodo discourse. In the future, we intend to add more data to the dataset and train the entire pre-trained model on our dataset. Hopefully, researchers will find this work and dataset helpful and may imply a cross-lingual effect on every area of NLP.

References

Bashar Md.A. and Nayak R. (2020). Qutnocturnal@hasoc'19: CNN for hate speech and offensive content identification in hindi language. *CoRR*, abs/2008.12448.

Berwick R. C. (1985). *The Acquisition of Syntactic Knowledge*, vol. **16**. MIT Press. ISBN 9780262022262.

Bhardwaj M., **Akhtar M. S.**, **Ekbal A.**, **Das A. and Chakraborty T.** (2020). Hostility detection dataset in hindi, *arXiv preprint arXiv:2011.03588*.

Bhatia M., **Bhotia T. S.**, **Agarwal A.**, **Ramesh P.**, **Gupta S.**, **Shridhar K.**, **Laumann F. and Dash A.** (2021). One to rule them all: Towards joint indic language hate speech detection. *CoRR*, abs/2109.13711.

https://doi.org/10.1017/nlp.2024.28 Published online by Cambridge University Press

Bhattacharyya P. (2010). Indowordnet. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*.

Bowyer K. W., Chawla N. V., Hall L. O. and Kegelmeyer W. P. (2011). SMOTE: Synthetic minority over-sampling technique. *CoRR*, abs/1106.1813.

Census (2011a). Abstract of speakers' strength of languages and mother tongues. *Census*.

Census (2011b). Comparative speakers' strength of languages and mother tongues - 1971, 1981, 1991, 2001 and 2011*Census of India 2011*.

Conneau A., Khandelwal K., Goyal N., Chaudhary V., Wenzek G., Guzmán F., Grave E., Ott M., Zettlemoyer L. and Stoyanov V. (2019). Unsupervised cross-lingual representation learning at scale. *CoRR*, abs/1911.02116.

Das M. and Senapati A. (2023). Development of the co-reference resolution tagged data set in assamese @ a semi-automated approach. In *2023 IEEE Guwahati Subsection Conference (GCON)*, pp. 1–4. https://doi.org/10.1109/GCON58516.2023.10183580.

Devlin J., Chang M.-W., Lee K. and Toutanova K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Feng F., Yang Y., Cer D., Arivazhagan N. and Wang W. (2020). Language-agnostic bert sentence embedding, *arXiv preprint arXiv:2007.01852*.

Freund Y. and Schapire R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences* **55**(1), 119–139.

Gaikwad S., Ranasinghe T., Zampieri M. and Homan C. M. (2021). Cross-lingual offensive language identification for low resource languages: The case of Marathi. *CoRR*, abs/2109.03552.

Ghosh K. and Senapati A. (2022). Hate speech detection: A comparison of mono and multilingual transformer model with cross-language evaluation. In *Proceedings of the 36th Pacific Asia Conference on Language, Information and Computation*, Manila, Philippines: Association for Computational Linguistics, pp. 853–865.

Ghosh K., Senapati A., Narzary M. and Brahma M. (2023a). Hate speech detection in low-resource bodo and assamese texts with ML-DL and Bert models. In *Scalable Computing: Practice and Experience*, vol.**24**, pp. 941–955.

Ghosh K., Sonowal D., Basumatary A., Gogoi B. and Senapati A. (2023b). Transformer-based hate speech detection in assamese. In *IEEE Guwahati Subsection Conference (GCON)*, pp. 1–5. https://doi.org/10.1109/GCON58516.2023.10183497.

Ghosh S., Priyankar A., Ekbal A. and Bhattacharyya P. (2023c). A transformer-based multi-task framework for joint detection of aggression and hate on social media data. *Natural Language Engineering* **29**(6), 1495–1515. https://doi.org/10.1017/S1351324923000104.

Subhankar Ghosh P. K. B., Das S. and Chaudhuri B. B. (2012). Development of an assamese OCR using Bangla OCR, Association for Computing Machinery. In *Proceeding of the Workshop on Document Analysis and Recognition, DAR '12*. New York, NY, USA, pp. 68–73. https://doi.org/10.1145/2432553.2432566, ISBN 9781450317979.

Glazkova A., Kadantsev M. and Glazkov M. (2021). Fine-tuning of pre-trained transformers for hate, offensive, and profane content detection in English and Marathi, arXiv preprint arXiv:2110.12687.

Hagoort P., Hald L., Bastiaansen M. and Petersson K. M. (2004). Integration of word meaning and world knowledge in language comprehension. *Science* **304**(5669), 438–441.

He H., Bai Y., Garcia E. A. and Li S. (2008). Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, pp.1322–1328. https://doi.org/10.1109/IJCNN.2008.4633969.

Jain K., Deshpande A., Shridhar K., Laumann F. and Dash A. (2020). Indic-transformers: An analysis of transformer language models for Indian languages.

Joshi R. (2022a). L3cube-mahacorpus and mahabert: Marathi monolingual corpus, Marathi BERT language models, and resources. *CoRR*, abs/2202.01159.

Joshi R. (2022b). L3cube-Mahacorpus and Mahabert: Marathi monolingual corpus, Marathi bert language models, and resources. In *Proceedings of The WILDRE-6 Workshop within the 13th Language Resources and Evaluation Conference*, Marseille, France: European Language Resources Association, pp. 97–101.

Joshi R. (2023). L3cube-Hindbert and Devbert: Pre-trained bert transformer models for devanagari based Hindi and Marathi languages.

Kakwani D., Kunchukuttan A., Golla S., Gokul N. C., Bhattacharyya A., Khapra M. M. and Kumar P. (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*. Online. Association for Computational Linguistics, pp. 4948–4961https://doi.org/10.18653/v1/2020.findings-emnlp.445.

Karim M. R., Chakravarthi B. R., McCrae J. P. and Cochez M. (2020). Classification benchmarks for under-resourced Bengali language based on multichannel convolutional-LSTM network. In *2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA)*, IEEE, pp. 390–399.

Karim M. R., Dey S. K., Islam T., Sarker S., Menon M. H., Hossain K., Hossain M. A. and Decker S. (2021). Deephateexplainer: Explainable hate speech detection in under-resourced bengali language. In *2021 IEEE 8th International Conference on Data Science and Advanced Analytics (DSAA)*, pp. 1–10. https://doi.org/10.1109/DSAA53316.2021.9564230.

https://doi.org/10.1017/nlp.2024.28 Published online by Cambridge University Press

Khanuja S., **Bansal D.**, **Mehtani S.**, **Khosla S.**, **Dey A.**, **Gopalan B.**, **Margam D. K.**, **Aggarwal P.**, **Nagipogu R. T.**, **Dave S.**, **Gupta S.**, **Gali S. C. B.**, **Subramanian V. and Talukdar P. P.** (2021). Muril: Multilingual representations for indian languages. *CoRR*, abs/2103.10730.

Kudo T. and Richardson J. (2018). Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *CoRR*, abs/1808.06226.

Kumari S. (2020). Nohate at hasoc2020: Multilingual hate speech detection. In *Forum for Information Retrieval Evaluation*. FIRE.

Lan Z., **Chen M.**, **Goodman S.**, **Gimpel K.**, **Sharma P. and Soricut R.** (2019). ALBERT: A lite BERT for self-supervised learning of language representations. *CoRR*, abs/1909.11942.

Laskar S. R., **Gogoi A.**, **Dutta S.**, **Adhikary P. K.**, **Nath P.**, **Pakray P. and Bandyopadhyay S.** (2022). Investigation of negation effect for English–assamese machine translation. *Sādhanā* **47**(4), 238.

Laskar S. R., **Paul B.**, **Dadure P.**, **Manna R.**, **Pakray P. and Bandyopadhyay S.** (2023). English–assamese neural machine translation using prior alignment and pre-trained language model. *Computer Speech & Language* **82**, 101524.

Laub Z. (2019). Hate speech on social media: Global comparisons. *Council on Foreign Relations* **7**.

Litake O., **Sabane M.**, **Patil P.**, **Ranade A. and Joshi R.** (2022). Mono vs multilingual bert: A case study in hindi and marathi named entity recognition.

Liu Y., **Ott M.**, **Goyal N.**, **Du J.**, **Joshi M.**, **Chen D.**, **Levy O.**, **Lewis M.**, **Zettlemoyer L. and Stoyanov V.** (2019). Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Luoma J. and Pyysalo S. (2020). Exploring cross-sentence contexts for named entity recognition with bert. *arXiv preprint arXiv:2006.01563*.

Mandal P., **Senapati A. and Nag A.** (2022). Hate-speech detection in news articles: in the context of west bengal assembly election 2021, springer nature Singapore. In Gupta D., Goswami R. S., Subhasish Banerjee M. T. and Pachori R. B., (eds), *Pattern Recognition and Data Analysis with Applications*. Singapore, pp. 247–256. ISBN 978-981-19-1520-8.

Mandl T., **Modha S.**, **Majumder P.**, **Patel D.**, **Dave M.**, **Mandlia C. and Patel A.** (2019). Overview of the hasoc track at fire 2019: Hate speech and offensive content identification in Indo-European languages, Association for Computing Machinery. In *Proceedings of the 11th Forum for Information Retrieval Evaluation, FIRE '19*, New York, NY, USA, pp. 14–17, https://doi.org/10.1145/3368567.3368584. ISBN 9781450377508.

Mandl T., **Modha S.**, **Anand Kumar M. and Chakravarthi B. R.** (2020). Overview of the hasoc track at fire 2020: Hate speech and offensive language identification in Tamil, Malayalam, Hindi, English and German, Association for computing machinery. In *Forum for Information Retrieval Evaluation, FIRE 2020*. New York, NY, USA, pp. 29–32. https://doi.org/10.1145/3441501.3441517, ISBN 9781450389785.

McCarley J. S., **Chakravarti R. and Sil A.** Structured pruning of a bert-based question answering model, arXiv preprint arXiv:1910.06360, 2019.

Mensonides J.-C., **Jean P.-A.**, **Tchechmedjiev A. and Harispe S.** (2019). IMT mines ales at hasoc 2019: Automatic hate speech detection. In *FIRE 2019-11th Forum for Information Retrieval Evaluation*, vol. **2517**, p. 279.

Mishra S. and Mishra S. (2019). 3idiots at hasoc 2019: Fine-tuning transformer neural networks for hate speech identification in Indo-European languages. In *FIRE (Working Notes)*, pp. 208–213.

Modha S., **Mandl T.**, **Shahi G. K.**, **Madhu H.**, **Satapara S.**, **Ranasinghe T. and Zampieri M.** (2021). Overview of the hasoc subtrack at fire 2021: Hate speech and offensive content identification in English and Indo-Aryan languages and conversational hate speech. In *Forum for Information Retrieval Evaluation, FIRE 2021*. New York, NY, USA: Association for Computing Machinery, pp. 1–3. https://doi.org/10.1145/3503162.3503176, ISBN 9781450395960.

Mujadia V., **Mishra P. and Sharma D. M.** (2019). IIIT-Hyderabad at Hasoc 2019: Hate speech detection. In *FIRE (Working Notes)*, pp.271–278.

Narzary S., **Brahma M.**, **Singha B.**, **Brahma R.**, **Dibragede B.**, **Barman S.**, **Nandi S. and Som B.** (2019). Attention based English-Bodo neural machine translation system for tourism domain. In *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*, pp. 335–343. https://doi.org/10.1109/ICCMC.2019.8819699.

Narzary S., **Brahma M.**, **Narzary M.**, **Muchahary G.**, **Singh P. K.**, **Senapati A.**, **Nandi S. and Som B.** (2022). Generating monolingual dataset for low resource language Bodo from old books using Google keep. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, Marseille, France: European Language Resources Association, pp. 6563–6570.

Nath A., **Mannan S. and Krishnaswamy N.** (2023). Axomiyaberta: A phonologically-aware transformer model for assamese.

Nene M., **North K.**, **Ranasinghe T. and Zampieri M.** (2021). Transformer models for offensive language identification in Marathi. In *FIRE*.

Nozza D. (2022). Nozza@LT-EDI-ACL2022: Ensemble modeling for homophobia and transphobia detection. In *Proceedings of the Second Workshop on Language Technology for Equality, Diversity and Inclusion*, Dublin, Ireland: Association for Computational Linguistics, pp.258–264. https://doi.org/10.18653/v1/2022.ltedi-1.37.

Patterson K., **Nestor P. J. and Rogers T. T.** (2007). Where do you know what you know? The representation of semantic knowledge in the human brain. *Nature Reviews Neuroscience* **8**(12), 976–987.

https://doi.org/10.1017/nlp.2024.28 Published online by Cambridge University Press

Raffel C., Shazeer N., Roberts A., Lee K., Narang S., Matena M., Zhou Y., Li W. and Liu P. J. (2019). Exploring the limits of transfer learning with a unified text-to-text transformer.

Raj R., Srivastava S. and Saumya S. (2020). NSIT & IIITDWD @ HASOC 2020: Deep learning model for hate-speech identification in Indo-European languages. In *FIRE*.

Rajrani K. and Ashok K. D. (2019). Exploring linguistic diversity in India: A spatial analysis. In *Handbook of the Changing World Language Map*.

Ramanathan A. and Rao D. (2003). A lightweight stemmer for Hindi.

Rogers A., Kovaleva O. and Rumshisky A. (2020). A primer in bertology: What we know about how bert works.

Romim N., Ahmed M., Talukder H., Saiful I., *et al.* (2021). Hate speech detection in the Bengali language: A dataset and its baseline evaluation. In *Proceedings of International Joint Conference on Advances in Computational Intelligence*, Springer, pp. 457–468.

Romim N., Ahmed M., Islam M. S., Sharma A. S., Talukder H. and Amin M. R. (2022). BD-SHS: A benchmark dataset for learning to detect online Bangla hate speech in different social contexts. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, Marseille, France: European Language Resources Association, pp. 5153–5162.

Sanh V., Debut L., Chaumond J. and Wolf T. (2019a). Distilbert, a distilled version of BERT: Smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108.

Sanh V., Debut L., Chaumond J. and Wolf T. (2019b). Distilbert, a distilled version of bert: Smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108.

Sarker S. (2020). Banglabert: Bengali mask language model for Bengali language understading.

Shibata Y., Kida T., Fukamachi S., Takeda M., Shinohara A., Shinohara T. and Arikawa S. (1999). Byte pair encoding: A text compression scheme that accelerates pattern matching.

Sun C., Qiu X., Xu Y. and Huang X. (2019). How to fine-tune bert for text classification? In *China National Conference on Chinese Computational Linguistics*, Springer, pp. 194–206.

Ulčar M. and Robnik-Šikonja M. (2020). Finest bert and crosloengual bert. In *International Conference on Text, Speech, and Dialogue*, Springer, pp. 104–111.

Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A. N., Kaiser L. and Polosukhin I. (2017). Attention is all you need. *CoRR*, abs/1706.03762.

Velankar A., Patil H., Gore A., Salunke S. and Joshi R. (2022). L3cube-mahahate: A Tweet-based Marathi hate speech detection dataset and BERT models. *CoRR*, abs/2203.13778, https://doi.org/10.48550/arXiv.2203.13778.

Wu Y., Schuster M., Chen Z., Le Q. V., Norouzi M., Macherey W., Krikun M., Cao Y., Gao Q., Macherey K., Klingner J., Shah A., Johnson M., Liu X., Kaiser Ł., Gouws S., Kato Y., Kudo T., Kazawa H., Stevens K., Kurian G., Patil N., Wang W., Young C., Smith J., Riesa J., Rudnick A., Vinyals O., Corrado G., Hughes M. and Dean J. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation.

Zampieri M., Malmasi S., Nakov P., Rosenthal S., Farra N. and Kumar R. (2019). SemEval-2019 task 6: Identifying and categorizing offensive language in social media (OffensEval). In *Proceedings of the 13th International Workshop on Semantic Evaluation*, Minneapolis, Minnesota, USA: Association for Computational Linguistics, pp. 75–86, https://doi.org/10.18653/v1/S19-2010.

Cite this article: Ghosh K and Senapati A (2025). Hate speech detection in low-resourced Indian languages: An analysis of transformer-based monolingual and multilingual models with cross-lingual experiments. *Natural Language Processing* **31**, 393–414. https://doi.org/10.1017/nlp.2024.28