Crossing Borders: Multilingual Hate Speech Detection

Supriya Chanda[1], Abhishek Dhaka[2] and Sukomal Pal[1]

[1]Department of Computer Science and Engineering, Indian Institute of Technology (BHU), Varanasi, INDIA, 221005
[2]Department of Computer Science and Engineering, B.K. Birla Institute of Engineering&Technology, Pilani, INDIA,333031

Abstract

With the relentless growth of technology usage, particularly among younger generations, the alarming prevalence of hate speech on the internet has become an urgent global concern. This research paper addresses this critical need by presenting an extensive investigation encompassing three distinct hate speech detection tasks across a diverse linguistic landscape. The first task involves hate and offensive speech classification in Gujarati and Sinhala, assessing sentence-level hatefulness. The second task extends to fine-grained BIO tagging, enabling precise identification of hate speech within sentences. Finally, the third task expands the scope to hate speech classification in Bengali, Bodo, and Assamese using social media data, categorizing content as hateful or not. Employing state-of-the-art deep learning techniques tailored to each language's characteristics, this research contributes significantly to the development of robust and culturally sensitive hate speech detection systems, imperative for nurturing safer online spaces and fostering cross-cultural understanding.

Warning: The content of this paper may contain offensive material, reader discretion is advised.

Keywords

Hate Speech, Social Media, Gujrati, Sinhala, Assamese, Bengali, Bodo, Multilingual BERT,

1. Introduction

In light of the burgeoning utilization of technology, accompanied by a substantial rise in users, particularly among the younger demographic, the presence of hate speech on the internet has emerged as a pressing concern. While the internet was initially envisioned as a platform for individuals to express their thoughts freely, it is equally imperative that this unbridled expression does not encroach upon the dignity and beliefs of others. Safeguarding these principles is pivotal to sustaining the unfettered expression of individuals' thoughts.

Hate speech refers to " Any form of communication, whether spoken or expressed non-verbally, that displays hostility towards specific social groups. These groups are typically targeted based on factors like race and ethnicity (which encompasses racism, xenophobia, anti-Semitism, etc.), gender (including sexism and misogyny), sexual orientation (involving homophobia and transphobia), age (ageism), and disability (known as ableism)".

In the landscape of digital connectivity, India exhibited remarkable statistics in early 2023. With a population of 1.10 billion, cellular mobile connections thrived, reaching an impressive 77.0% of the total population. Simultaneously, the internet had made significant inroads, with

Forum for Information Retrieval Evaluation, December 15-18, 2023, Goa, India

supriyachanda.rs.cse18@itbhu.ac.in (S. Chanda); abhishek.dhaka7340@gmail.com (A. Dhaka); spal.cse@itbhu.ac.in (S. Pal)
0000-0002-6344-8772 (S. Chanda); 0000-0002-9421-8566 (S. Pal)

692.0 million users in India, representing 48.7% of the populace. In the realm of social media, India stood out with 467.0 million users in January 2023, accounting for 32.8% of the total population. Additionally, data from top social media platforms' ad planning tools revealed that 398.0 million users aged 18 and above were actively engaged in social media usage, forming a substantial 40.2% of the adult population at the beginning of 2023. These statistics collectively underscore the pervasive presence and impact of digital technology and social media within India's diverse and expansive demographic landscape.

India, renowned for its linguistic diversity, is home to a population of 1.4 billion, comprising individuals who speak a myriad of languages and hold diverse beliefs. Among these, there are 121 officially recognized languages, each with over 10,000 speakers. Hindi boasts the largest number of speakers, at 436 million, followed by Bengali with 83 million, Assamese with 12.6 million, Gujrati with 62 million, and Bodo with 1.4 million (according to the 2011 census). Sinhala spoken by the Sinhalese people of Sri Lanka, who make up the largest ethnic group on the island, numbering about 16 million. While considerable research has been conducted on hate speech identification in Hindi, it is equally vital to address this issue in other under-resourced languages such as Bodo, Assamese, Gujrati, Sinhala and Bengali because many individuals prefer to communicate in their native languages, as it fosters a sense of connection and cultural grounding. Hence, it is crucial to identify hate speech in these languages to uphold cultural respect.

In this study, we approach all four tasks of hate speech detection as a text classification problem and delve into various deep learning methodologies for its resolution. The datasets for all four task and all languages like Assamese, Bengali, Gujrati, Sinhala, Bodo, and Hindi-English code mixed data utilized are obtained from the FIRE 2023 (Forum for Information Retrieval Evaluation) Hate Speech and Offensive Content Identification in English and Indo-Aryan Languages (HASOC). All the task descriptions are mentioned below.

1.1. HASOC Tasks

The goal of HASOC 2023 was to establish a testbed for the automated detection of hate speech and objectionable material in social media posts. HASOC 2023 included four tasks, and our team actively participated in tasks 1, 3 and 4. The tasks in this study are distributed as follows:

Task 1: Offensive Language Identification in Gujrati, and Sinhala.

- **Offensive(OFF)**- Contain offensive language.
- **Non Hate-Offensive(NOT)-** No offense or profanity is present.

Task 3: Hateful Span Detection in Texts

Task 3 focuses on identifying hateful parts within a sentence that is already considered hateful. A hateful span is a continuous set of words that together express explicit hate in a sentence.

So in the above example, the input texts are in English. In this sequence labeling problem each token in the sequence is manually tagged with the start and end of a hateful span using BIO notation. 'B' represents the start of a hate span, 'I' continues it, and 'O' indicates non-hate. "You all niggers are cancers" → "O O B I I." It's important to note that 'I' cannot stand alone and must follow either 'I' or 'B'. Additionally, a 'B' can be followed immediately by an 'O' for single-word spans.

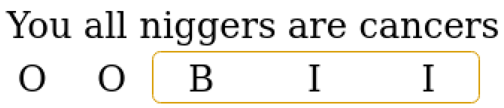

Figure 1: Example of task 3

Task 4: Offensive Language Identification in Assamese, Bengali and Bodo.

- **Hate and Offensive(HOF)**- Contains Offensive language
- **Non Hate-Offensive(NOT)**- No offense or profanity is present.

Table 1 provides examples of the various posts and associated labels.

Table 1
Example tweets from the HASOC2023 dataset for all classes

Language	Sample tweet from the class	Task 4
Bodo	बुरबकलै साला नों बायदि मानसिनि जेबो अबदान गैयालै जाथिआव	HOF
Bodo	Btr आव bjp आ साव लाबोबाय ।बे सावाव बर leader फ्रा लोगो लाफाबाय।	NOT

The remaining sections of the paper are structured as follows. In Section 2, a brief outline of some previous attempts is provided. The dataset description is presented in Section 3. Computational methods, model descriptions, and the evaluation methodology are discussed in Section 4. Results and discussions are presented in Section 5, and the conclusion is provided in Section 6.

2. Related Work

The identification of hate speech and offensive content has garnered significant attention in both academic and business contexts. While a substantial body of research has concentrated primarily on English due to its global prevalence, there exists a pressing need for relevant corpora in other languages to comprehensively address this issue. Several studies have delved into the varied aspects of offensive content, such as *abusive language* [1, 2], *cyber-aggression* [3], *cyber-bullying* [4, 5], and *toxic comments or hate speech* [6, 7, 8]. A brief overview of some notable works in these areas is provided.

- *Hate Speech Identification :* Hate speech, a pervasive challenge, has been systematically categorized into various types based on the nature of its textual content. Diverse datasets have been curated to cater to these distinct categories of hate speech. Notably, a common dataset [9] has served as a foundation for identifying hate speech and profanity, with recent work by Davidson et al. [7] making use of a dataset comprising nearly 24,000 labeled tweets.

- *Offensive Content and Cyberbullying :* The broader domain of offensive content encompasses abusive language [3], cyber-aggression, cyber-bullying, and toxic comments. Previous investigations have employed techniques such as sentiment analysis, topic modeling [4], and user-related features [5] to tackle this multifaceted problem.

Efforts have extended beyond English, with endeavors in languages including German [10, 11], Spanish, Arabic [2, 12], Greek [13], Slovene [14] and Chinese [15]. Mubarak et al. [2] introduced a collection of profane terms, known as SeedWords (SW), and applied the Log Odds Ratio (LOR) to individual word unigrams and bigrams. Saroj et al. [16] adopted a Support Vector Machine (SVM) approach alongside TF-IDF features, targeting hate speech and offensive language in Arabic and Greek.

In recent years, initiatives like HASOC [10] and GermEval [17] have spotlighted the importance of addressing hate speech detection in various languages and contexts. Dravidian LangTech [18], for example, focused on detecting offensive language in a code-mixed dataset comprising Tamil–English, Malayalam–English, and Kannada–English. The application of multilingual models, including BERT variants and IndicBERT, has shown promise in this regard. Transfer learning has shown potential in enhancing offensive language recognition, particularly in code-mixed contexts. Researchers have leveraged transfer learning from English datasets to improve offensive language recognition in code-mixed Kannada [19], Malayalam [20], and Tamil [21].

Detecting hate speech in conversational Hindi-English code-mixed data presents additional complexities due to the conversational nature of such content. The hierarchical structure of posts, comments, and replies necessitates a nuanced approach, with techniques ranging from unified text treatment to novel hierarchical neural network architectures. Multiple comments can be associated with each post, and each comment may have several replies. In the case of English-Hindi data, each component of the tuple can exhibit code-mixing between Hindi and English, be exclusively in English, exclusively in Hindi, in the form of romanized Hindi, or a combination thereof. As a result, complex input patterns emerge. The labels assigned to replies or comments are significantly influenced by the contextual information provided by the parent text. To address this, Chanda et al. [22] treated all the post, comments, and replies as a single unified text and applied a pre-trained multilingual BERT model. To maintain the context of post to comments and reply, Chanda et al., [23] concatenate. Bagora et al., [24] proposed a novel hierarchical neural network architecture, while Madhu et al., [25] employed a pipeline consisting of an LSTM classifier followed by a fine-tuned SentBERT model.

3. Dataset

In this study, we utilized the HASOC 2023 datasets, generously provided by the organizers of the FIRE 2023. The organizers furnished the training data for all four tasks and, for the final evaluation, made available the test data, for which participants were required to submit prediction files for each data sample.

For tasks 1 [26] and 4, the data files were formatted in a simple CSV structure, with one column dedicated to the text and another to the corresponding label. Task 3, however, presented a distinct dataset structure, comprising four columns in the training data: 'id,' 'sentence,' 'span,' and ’bio.’ The ’span’ column denoted the word indices at which hate content commenced and concluded, while the ’bio’ column utilized these span indices to represent the respective words as ’B’ (beginning), ’I’ (intermediate), or ’O’ (outside hate content).

The corpus collection and class distribution is shown in Table 2.

Table 2
Statistical overview of the Training Data and Test Data

Task-1				
Data	Language	#Sentence	NOT	HOF
Train	Gujrati	200	100	100
Test	Gujrati	1196	-	-
Train	Sinhala	7500	4324	3176
Test	Sinhala	2500	-	-
Task-3				
Data	Language	# Sentences		
Train	English	2421		
Test	English	606		
Task-4				
Data	Language	# Sentences	NONE	HOF
Train	Assamese	4036	1689	2347
Test	Assamese	1009	-	-
Train	Bengali	1281	766	515
Test	Bengali	320	-	-
Train	Bodo	1679	681	998
Test	Bodo	420	-	-

4. Proposed Methodology

HASOC2023 includes four distinct tasks, and in the subsequent subsections, the methodology employed for task 1,3 and 4 will be outlined individually. It is essential to underscore that preprocessing, a pivotal facet in addressing various text-related downstream tasks, will be discussed initially before delving into the specifics of each task’s methodology. The code for all proposed methods can be found on GitHub.[1]

[1] GitHub repository: https://github.com/abhishekdhakaab/FIRE-2023

4.1. Preprocessing

Social media data exhibits a high degree of structural informality and is susceptible to noise due to the colloquial nature of Twitter conversations . This inherent characteristic poses a potential challenge to the accuracy of processing techniques. Consequently, it has been deemed imperative to subject all data to preprocessing procedures aimed at mitigating the impact of less informative textual components.

Notably, for tasks 1 and 4, the following preprocessing was done. Conversely, for task 3, no preprocessing measures were deemed necessary. Below, we provide a comprehensive enumeration of the preprocessing steps that were applied.

- Perform cleaning by removing usernames, punctuation and URLs, mentions and hashtags.
- Use ekphrasis which is a text processing tool, geared towards text from social networks, such as Twitter or Facebook. ekphrasis performs tokenization, word normalization, word segmentation (for splitting hashtags) and spell correction.
- Normalizing hashtags (for example, “#BlackLivesMatters” is segmented into “Black”, “Lives”,and “Matters”).

For the binary classification task, the ‘HOF’ labels have been converted to integer ‘1’, representing instances of harmful or offensive content, while the ‘NOT’ labels have been converted to integer ‘0’, indicating non-harmful content.

The preprocessing steps were little bit different for task 3. To maintain consistent sentence lengths for word-level sequence classification using BIO tagging, padding was applied to both the input data and the corresponding true labels. Specifically, the input data was padded to a maximum length of 128 tokens, and the true labels were augmented with ‘0’ values, ensuring that the padded sections consistently predicted ‘O’ label.

4.2. Methodology for Task 1

The selection of embedding techniques was contingent upon the size of the vocabulary, and as such, various transformer-based embeddings were explored, including mBERT and many more. The empirical evaluation revealed that fasttext exhibited a larger presence of commonly used words while minimizing

1. **Sinhala :** For the Sinhala language, four distinct submission strategies were employed, each utilizing a specific model methodology. Herein, a comprehensive overview of these methodologies is provided:

   A. *Fasttext-CNN :* In this approach, a 300-dimensional Fasttext embedding was utilized. The training dataset comprised 90% of the available data, with the remaining 10% designated for validation. The maximum sequence length was set to 128 tokens. Notably, the dataset encompassed 39,793 word tokens, while the Fasttext embedding contained 30,277 words of these 39,793 words. The model architecture included two convolutional layers, both with 300 filters and kernel sizes of 3 and 2, respectively. These convolutional layers were concatenated and fed into a subsequent CNN layer with 500 filters, followed by a dropout layer with a rate of 0.3. Subsequently, another

CNN layer with 300 filters was applied, and Global Max pooling was employed. A dense layer with 50 units and a ReLU activation function preceded a final sigmoid activation layer. Hyperparameters encompassed a learning rate of 5e-5, a batch size of 32, AdamW optimization, and a loss function of Binary Cross entropy. Early stopping criteria were also employed, and this configuration consistently yielded the best performance scores within the range of 0.1 to 5e-7 for learning rate.

B. *Full-Data Fasttext-CNN for Sinhala :* This submission retained the same model architecture and hyperparameters as Submission 1, with the exception that all available data was used for training, and no validation data was set aside.

C. *BiLSTM-Attention for Sinhala :* For this strategy, a 300-dimensional embedding was employed, followed by a bidirectional LSTM layer with 300 units, incorporating an attention mechanism. A subsequent dense layer with 50 units with a ReLU activation function, along with a dropout layer (rate = 0.3). Finally, a dense layer with a single unit and sigmoid activation function concluded the model.

D. *BiLSTM-Attention for Sinhala (2) :* Submission 4 utilized a 300-dimensional embedding followed by a bidirectional LSTM layer with 128 units and a dropout layer (rate = 0.3). An attention layer was applied to the output of the bidirectional LSTM, followed by Global Max pooling and flattening. A dropout layer with a rate of 0.3 preceded a dense layer with 64 units, followed by another dropout layer with the same rate. The model culminated with a final dense layer featuring a single unit and a sigmoid activation function.

2. **Gujrati :** For the Gujarati language, two distinct submission strategies were employed, each utilizing a specific model methodology. Here, a detailed overview of these methodologies is provided:

   A. *FastText-CNN :* This approach utilized a 300-dimensional FastText embedding. The training dataset comprised 90% of the available data, with the remaining 10% reserved for validation.Padding was applied to each sentence to maintain a consistent length of 128 tokens. Notably, the dataset contained 4,412 word tokens, while the FastText embedding encompassed 3,931 words out of 4,412 words. The same DNN classifier was used as mentioned in Sinhala A.

   B. *BiLSTM-Attention Approach for Gujarati :* In this strategy, a 300-dimensional embedding was employed, followed by a bidirectional LSTM layer comprising 128 units and a dropout layer (rate = 0.3). An attention layer was applied to the output of the bidirectional LSTM, followed by Global Max pooling and flattening. Subsequently, a dropout layer with a rate of 0.3 preceded a dense layer featuring 1024 units and an additional dropout layer with a rate of 0.3. The model further incorporated a dense layer with 256 units, followed by another dense layer with 32 units. The final layer consisted of a dense unit with a single node, activated by a sigmoid activation function.

The rationale behind opting for a Deep Neural Network (DNN) over a Transformer-based model is rooted in a critical observation. It has been noted that when dealing with a relatively limited training dataset consisting of only 200 examples, a DNN tends to outperform a Transformer architecture. This preference for a DNN stems from the fact that Transformer models typically require a larger volume of training data to achieve their optimal performance. In situations where data scarcity is a significant concern, as evidenced by the small training dataset in this context, the DNN's ability to generalize and learn effectively from limited examples becomes a compelling choice.

4.3. Methodology for Task 3

For this task, the initial step involved mapping the true labels as follows: 'O' to 0, 'B' to 1, and 'I' to 2. No preprocessing of the data was required. The dataset was then divided into 20% for validation and 80% for training purposes. To enhance word embeddings, Glove embeddings trained on the Twitter 27b token dataset [2] were employed. The model architecture is inspired from [27].

The model architecture comprised several key components. Initially, input tokens were embedded using Glove embeddings, followed by a 64-unit attention layer. Subsequently, the output of the attention layer was passed through two BiLSTM layers, each consisting of 512 units and a dropout rate of 0.2. The outputs from both BiLSTM layers were added. This was followed by a time-distributed dense layer with 50 units. Additionally, the output of each time-distributed dense layer (resulting in a shape of (batch_size, 128, 50)) was further processed through a simple dense layer with 3 units, resulting in a shape of (batch_size, 128, 3) and then it was passed to CRF, with number of crf tag set to 3. The training of this model was carried out using the Adam optimizer with a learning rate of 0.001 for a total of 5 epochs.

4.4. Methodology for Task 4

The selected models for submission were meticulously chosen following extensive experimentation involving various learning rates ranging from 0.1 to 1e-7. Furthermore, different combinations of LSTM layers, ranging from a single LSTM layer to up to 4 LSTM layers, were evaluated to discern the impact of the number of LSTM layers. Additionally, variations in the number and sizes of dense layers were explored. Almost for all experiment, approximately 90% of the dataset was allocated to training, while the remaining 10% was set aside for validation.

1. **Assamese :**
   A. *Multilingual BERT for Assamese :* In this approach, the utilization of mBERT (bert-base-multilingual-cased) with BertForSequenceClassification, a state-of-the-art transformer model, was aimed at assessing its efficacy in grasping the nuances of the Assamese language. Given the constraints of Assamese data available for model training, a separate test dataset was not utilized. A learning rate of 2e-5 was chosen for effective training, alongside a batch size of 32. The training process encompassed 50 epochs.

   B. *Fine-Tuned Assamese BERT :* This approach entailed the fine-tuning of a BERT variant, tailor-made for the Assamese language, using a dedicated monolingual dataset. The maximum sequence length was restricted to 128 tokens to accommodate the inherent characteristics of Assamese text. Fine-tuning concentrated on optimizing the last three layers of the BERT model. The subsequent architectural flow incorporated the transformation of BERT embeddings via a Bi-LSTM (128) layer, followed by an LSTM (128) layer and an additional LSTM (64) layer. This intricate representation

[2] https://github.com/stanfordnlp/GloVe

was then directed through a dense layer, composed of 32 neurons, invoking ReLU activation, followed by dropout with a rate of 0.3, an additional linear layer, and a sigmoid activation function. Model training, conducted over 200 epochs, utilized a batch size of 16, with the best model selection contingent on validation accuracy. The learning rate was optimized to 1e-6, with the AdamW optimizer in play.

   C. *Fine-Tuned Assamese BERT with Variant Learning Rate :* This variant closely adhered to the fine-tuned Assamese BERT approach described in (B). However, it introduced a different learning rate, specifically 5e-8, during training to explore its impact on model performance.

2. **Bengali :**
   A. *XLM-RoBERTa for Bengali :* Implemented XLM Roberta in conjunction with BERTForSequenceClassification. To accommodate the lengthier nature of Bengali sentences, the maximum sequence length was set to 256 tokens. Training involved fine-tuning XLM Roberta's weights with a learning rate of 2e-5 and a batch size of 16 over 10 epochs.

   B. *Multilingual BERT for Bengali :* Used mBERT with the same hyperparameters as XLM Roberta and employed the BERTForSequenceClassification framework to assess its performance in Bengali text classification.

   C. *BengaliBERT :* Leveraged BengaliBERT, a model pre-trained on monolingual Bengali data. Notably, no further fine-tuning was performed on this BERT for classification tasks. The architecture incorporated a combination of LSTM and dense layers. Data was propagated through a Bi-LSTM (128) layer, followed by an LSTM (128) layer, and subsequently to a Dense layer comprising 32 neurons, enhanced with ReLU activation. To mitigate overfitting, a dropout rate of 0.3 was applied, followed by another linear layer and a sigmoid activation function. Training spanned 200 epochs with a batch size of 16, and model selection was based on validation accuracy. The learning rate was set to 1e-5, and the AdamW optimizer was utilized.

3. **Bodo :**
   A. *XLM-RoBERTA for Bodo :* Employed BertForSequenceClassification for Bodo language text classification. The maximum sequence length was set at 256 tokens to accommodate the language's characteristics. The learning rate was configured to 2e-5, with a batch size of 16 for 10 training epochs.

   B. *HBERT for Bodo :* Utilized L3Cube's Hindi Bert v2 due to the similarities between Bodo and Hindi scripts. However, recognizing the distinctions between the languages, the last three layers of the Bert model were fine-tuned. The same DNN classifier was used as mentioned in Assamese B. Two different learning rates, lr=1e-6 and lr=1e-5, were tested, with results saved as HBERT.csv and HBERT_2.csv, respectively. The AdamW optimizer was employed for training.

   C. *BodoBERT :* Employed BodoBERT, a model specifically tailored for the Bodo language. The same DNN classifier was used as mentioned in Assamese B. The learning rate was set to 1e-6, and the AdamW optimizer was utilized for training.

   D. *Ensemble Method :* Introduced an ensemble approach that amalgamates the outcomes of Method 2 (HBERT for Bodo), Method 3 (BodoBERT), aninto a cohesive predictive framework. This ensemble method explores the synergy between different models to enhance the overall accuracy and robustness of hate speech classification in the Bodo language dataset.

Given the substantial computational demands of transformers, all models based on transformers were trained on Colab's T4 GPU, while non-transformer based models were trained on Colab's CPU, boasting 12.7GB of RAM. The conversion of model predictions back to binary labels (HOF and NOT) was executed using a threshold value of 0.5, a common practice in binary classification tasks.

5. Results and Discussion

The model was validated on the training and development sets due to the limited amount of data available for training. Subsequently, the prediction file was submitted on the test data to obtain the final results [28, 29, 30, 31].

In Task 1, focused on Sinhala and Gujrati text classification, our team achieved competitive scores. Table 3 shows the best performing team and our official performances on the test data as shared by the organizers. In the Sinhala category, we earned a respectable Macro F1 score of 0.78, while in the Gujrati category, our score was 0.68. For reference, the top-performing team, ”FiRC-NLP,” secured scores of 0.83 and 0.84 in Sinhala and Gujrati, respectively.

5.1. Results for Task 1 :

Table 3
Evaluation results for Task 1 on test data

Language	Team Name	Macro F_1 score	precision	recall
Sinhala	FiRC-NLP	0.83	0.83	0.83
	IRLab@IITBHU (**Fasttext-CNN**)	0.78	0.78	0.78
Gujrati	FiRC-NLP	0.84	0.83	0.86
	IRLab@IITBHU (**Fasttext-CNN**)	0.68	0.69	0.72

5.2. Results for Task 3 :

In Task 3, which involved offensive span detection, our team faced more significant challenges. Table 4 shows the best performing team and our official performances on the test data as shared by the organizers. On the public leaderboard, our team stood at rank 3. In both the public and private datasets, our team achieved scores of 0.45 and 0.51, respectively.

5.3. Results for Task 4 :

Task 4 encompassed text classification in Assamese, Bengali, and Bodo languages. Our team, ”IRLab@IITBHU,” achieved competitive scores in all three categories. Table 5 shows the best

Table 4
Evaluation results for Task 3 on test data

Leaderboard	Team Name	score
Public	FiRC-NLP	0.53
	IRLab@IITBHU	0.45
Private	FiRC-NLP	0.57
	IRLab@IITBHU	0.51

performing team and our official performances on the test data as shared by the organizers. For Assamese, our score was 0.70, while ”InclusiveTechies” led with a score of 0.80. In Bengali, we scored 0.65, whereas ”Sanvadita” achieved a score of 0.77. In the Bodo category, our team secured a score of 0.74, closely following ”SATLab,” which led with a score of 0.86.

Table 5
Evaluation results for Task 4 on test data

Language	Team Name	score
Assamese	InclusiveTechies	0.80
	IRLab@IITBHU (**Fine-Tuned Assamese BERT**)	0.70
Bengali	Sanvadita	0.77
	IRLab@IITBHU (**BengaliBERT**)	0.65
Bodo	SATLab	0.86
	IRLab@IITBHU (**HBERT for Bodo**)	0.74

5.4. Discussion

During the analysis of Sinhala and Gujrati, it was observed that the training data for Gujarati was insufficient to train a model effectively. In the investigation of hate speech classification across Bengali, Assamese, and Bodo languages, a noteworthy revelation emerged: validation accuracy alone does not necessarily encapsulate a model’s true ability. Instead, we found that validation loss holds paramount importance. A model with marginally lower validation accuracy but a considerably lower validation loss often outperforms a model with higher accuracy but slightly greater loss. This discrepancy underscores the significance of validation loss in gauging a model’s confidence in its predictions. In practical scenarios, it is often more prudent to err on the side of caution, minimizing the risk of false positives, where benign content is mistakenly flagged as hate speech.

In all most all tasks, it was observed that employing approximately 2 LSTM layers proved sufficient, as marginal improvements were discerned beyond 2 to 3 LSTM layers. However, such enhancements came at the cost of increased computational complexity.

While numerous embedding techniques are available for deep learning models, our experimentation revealed that FastText embeddings exhibited the most extensive vocabulary coverage for our dataset. This finding underscores the value of selecting embeddings tailored to the specific language and task at hand.

Given the challenges posed by low-resource languages and limited embedding resources, an alternative approach emerged: initial randomization of word embeddings, followed by training them may hold potential for optimizing model performance under resource constraints.

In our quest for the optimal optimizer, our experimentation indicated that Stochastic Gradient Descent (SGD) with a slightly higher learning rate converges more rapidly. Conversely, the AdamW optimizer with a higher learning rate exhibited a zig-zag convergence pattern. Notably, AdamW performed optimally with a lower learning rate, typically around 1e-5. However, it is important to recognize that SGD with a marginally higher learning rate can be a pragmatic choice for quick model testing, particularly in the context of Transformer-based models. This approach provides insights into a model's convergence tendencies before committing to more computationally intensive optimization methods.

Upon the completion of preprocessing the raw data, the rationale behind the utilization of a transformers-based model was to employ BERT for word-level embedding. This choice was made because BERT leverages the contextual information of each word to enhance the quality of word embeddings. Additionally, the motivation behind incorporating a bidirectional LSTM (bi-LSTM) layer into the model was to ensure that each word's embedding would encompass a comprehensive contextual context, spanning both preceding and subsequent words. Following the bi-LSTM layer, an additional LSTM layer can be applied to further process the bidirectional output of the preceding LSTM layer. Subsequently, after approximately 2 to 3 LSTM layers, the hidden state of the last time step of the final LSTM layer is passed to a dense layer for subsequent binary classification into classes 0 and 1. It is noteworthy that in all approaches, the final layer consists of a dense layer with a single neuron and a sigmoid activation function.

6. Conclusion

In the course of our investigation encompassing four diverse hate speech detection tasks, the following insights emerged:

Task 1: In the context of low-resourced languages, such as those examined in this study, the amalgamation of Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) models yielded the most efficacious results. The intricacies of these languages, coupled with the paucity of data, necessitated a tailored approach.

Task 3: Conditional Random Fields (CRF) emerged as the preeminent choice for Task 3, demonstrating superior performance in offensive span detection. Its efficacy surpassed that of alternative methods, underscoring its relevance and utility in this context.

Task 4: Task 4 underscored the value of models fine-tuned on language-specific monolingual data for the classification of text in low-resourced languages. These meticulously tailored models exhibited enhanced performance in text classification, emphasizing the significance of linguistic specificity in classification endeavors.

7. Acknowledgements

We would want to express our gratitude to the HASOC organisers for organising this interesting shared task and for immediately responding to all of our questions. Our investigation benefited from well-structured data with minimal spelling errors, rendering the classification task more straightforward.

References

[1] C. Nobata, J. Tetreault, A. Thomas, Y. Mehdad, Y. Chang, Abusive language detection in online user content, in: Proceedings of the 25th International Conference on World Wide Web, WWW '16, International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 2016, p. 145–153. URL: https://doi.org/10.1145/2872427.2883062. doi:10.1145/2872427.2883062.

[2] H. Mubarak, K. Darwish, W. Magdy, Abusive language detection on Arabic social media, in: Proceedings of the First Workshop on Abusive Language Online, Association for Computational Linguistics, Vancouver, BC, Canada, 2017, pp. 52–56. URL: https://www.aclweb.org/anthology/W17-3008. doi:10.18653/v1/W17-3008.

[3] R. Kumar, A. K. Ojha, S. Malmasi, M. Zampieri, Benchmarking aggression identification in social media, in: Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018), Association for Computational Linguistics, Santa Fe, New Mexico, USA, 2018, pp. 1–11. URL: https://www.aclweb.org/anthology/W18-4401.

[4] J.-M. Xu, K.-S. Jun, X. Zhu, A. Bellmore, Learning from bullying traces in social media, in: Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL HLT '12, Association for Computational Linguistics, USA, 2012, p. 656–666.

[5] M. Dadvar, D. Trieschnigg, R. Ordelman, F. de Jong, Improving cyberbullying detection with user context, in: P. Serdyukov, P. Braslavski, S. O. Kuznetsov, J. Kamps, S. Rüger, E. Agichtein, I. Segalovich, E. Yilmaz (Eds.), Advances in Information Retrieval, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, pp. 693–696.

[6] N. Djuric, J. Zhou, R. Morris, M. Grbovic, V. Radosavljevic, N. Bhamidipati, Hate speech detection with comment embeddings, in: Proceedings of the 24th International Conference on World Wide Web, WWW '15 Companion, Association for Computing Machinery, New York, NY, USA, 2015, p. 29–30. URL: https://doi.org/10.1145/2740908.2742760. doi:10.1145/2740908.2742760.

[7] T. Davidson, D. Warmsley, M. W. Macy, I. Weber, Automated hate speech detection and the problem of offensive language, CoRR abs/1703.04009 (2017). URL: http://arxiv.org/abs/1703.04009. arXiv:1703.04009.

[8] I. Kwok, Y. Wang, Locate the hate: Detecting tweets against blacks, in: Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence, AAAI'13, AAAI Press, 2013, p. 1621–1622.

[9] S. Malmasi, M. Zampieri, Detecting hate speech in social media, CoRR abs/1712.06427 (2017). URL: http://arxiv.org/abs/1712.06427. arXiv:1712.06427.

[10] T. Mandl, S. Modha, P. Majumder, D. Patel, M. Dave, C. Mandlia, A. Patel, Overview of the hasoc track at fire 2019: Hate speech and offensive content identification in indo-european languages, in: Proceedings of the 11th Annual Meeting of the Forum for Information Retrieval Evaluation, FIRE '19, Association for Computing Machinery, New York, NY, USA, 2019, p. 14–17. URL: https://doi.org/10.1145/3368567.3368584. doi:10.1145/3368567.3368584.

[11] B. Ross, M. Rist, G. Carbonell, B. Cabrera, N. Kurowsky, M. Wojatzki, Measuring the reliability of hate speech annotations: The case of the european refugee crisis, CoRR abs/1701.08118 (2017). URL: http://arxiv.org/abs/1701.08118. arXiv:1701.08118.

[12] H. Mubarak, A. Rashed, K. Darwish, Y. Samih, A. Abdelali, Arabic offensive language on twitter: Analysis and experiments, arXiv preprint arXiv:2004.02192 (2020).

[13] Z. Pitenis, M. Zampieri, T. Ranasinghe, Offensive Language Identification in Greek, in: Proceedings of the 12th Language Resources and Evaluation Conference, ELRA, 2020.

[14] D. Fišer, T. Erjavec, N. Ljubešić, Legal framework, dataset and annotation schema for socially unacceptable online discourse practices in Slovene, in: Proceedings of the First Workshop on Abusive Language Online, Association for Computational Linguistics, Vancouver, BC, Canada, 2017, pp. 46–51. URL: https://www.aclweb.org/anthology/W17-3007. doi:10.18653/v1/W17-3007.

[15] H.-P. Su, Z.-J. Huang, H.-T. Chang, C.-J. Lin, Rephrasing profanity in Chinese text, in: Proceedings of the First Workshop on Abusive Language Online, Association for Computational Linguistics, Vancouver, BC, Canada, 2017, pp. 18–24. URL: https://www.aclweb.org/anthology/W17-3003. doi:10.18653/v1/W17-3003.

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

[17] M. Wiegand, M. Siegel, J. Ruppenhofer, Overview of the germeval 2018 shared task on the identification of offensive language, Proceedings of GermEval 2018, 14th Conference on Natural Language Processing (KONVENS 2018), Vienna, Austria – September 21, 2018, Austrian Academy of Sciences, Vienna, Austria, 2018, pp. 1 – 10. URL: http://nbn-resolving.de/urn:nbn:de:bsz:mh39-84935.

[18] B. R. Chakravarthi, R. Priyadharshini, N. Jose, A. Kumar M, T. Mandl, P. K. Kumaresan, R. Ponnusamy, H. R L, J. P. McCrae, E. Sherly, Findings of the shared task on offensive language identification in Tamil, Malayalam, and Kannada, in: Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages, Association for Computational Linguistics, Kyiv, 2021, pp. 133–145. URL: https://aclanthology.org/2021.dravidianlangtech-1.17.

[19] S. Sai, Y. Sharma, Towards offensive language identification for Dravidian languages, in: Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages, Association for Computational Linguistics, Kyiv, 2021, pp. 18–27. URL: https://aclanthology.org/2021.dravidianlangtech-1.3.

[20] T. Ranasinghe, S. Gupte, M. Zampieri, I. Nwogu, WLV-RIT at hasoc-dravidian-codemix-fire2020: Offensive language identification in code-switched youtube comments, CoRR

abs/2011.00559 (2020). URL: https://arxiv.org/abs/2011.00559. arXiv:2011.00559.

[21] M. Zampieri, S. Malmasi, P. Nakov, S. Rosenthal, N. Farra, R. Kumar, Predicting the type and target of offensive posts in social media, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 1415–1420. URL: https://aclanthology.org/N19-1144. doi:10.18653/v1/N19-1144.

[22] 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, 2021.

[23] S. Chanda, S. Sheth, S. Pal, Coarse and fine-grained conversational hate speech and offensive content identification in code-mixed languages using fine-tuned multilingual embedding, 2022.

[24] A. Bagora, K. Shrestha, K. Maurya, M. S. Desarkar, Hostility detection in online hindi-english code-mixed conversations, in: 14th ACM Web Science Conference 2022, 2022, pp. 390–400.

[25] H. Madhu, S. Satapara, S. Modha, T. Mandl, P. Majumder, Detecting offensive speech in conversational code-mixed dialogue on social media: A contextual dataset and benchmark experiments, Expert Systems with Applications 215 (2023) 119342. URL: https://www.sciencedirect.com/science/article/pii/S0957417422023600. doi:https://doi.org/10.1016/j.eswa.2022.119342.

[26] T. Ranasinghe, I. Anuradha, D. Premasiri, K. Silva, H. Hettiarachchi, L. Uyangodage, M. Zampieri, Sold: Sinhala offensive language dataset, arXiv preprint arXiv:2212.00851 (2022).

[27] S. Masud, M. Bedi, M. A. Khan, M. S. Akhtar, T. Chakraborty, Proactively reducing the hate intensity of online posts via hate speech normalization, 2022. arXiv:2206.04007.

[28] S. Satapara, H. Madhu, T. Ranasinghe, A. E. Dmonte, M. Zampieri, P. Pandya, N. Shah, M. Sandip, P. Majumder, T. Mandl, Overview of the hasoc subtrack at fire 2023: Hate-speech identification in sinhala and gujarati, in: K. Ghosh, T. Mandl, P. Majumder, M. Mitra (Eds.), Working Notes of FIRE 2023 - Forum for Information Retrieval Evaluation, Goa, India. December 15-18, 2023, CEUR Workshop Proceedings, CEUR-WS.org, 2023.

[29] S. Masud, M. A. Khan, M. S. Akhtar, T. Chakraborty, Overview of the HASOC Subtrack at FIRE 2023: Identification of Tokens Contributing to Explicit Hate in English by Span Detection, in: Working Notes of FIRE 2023 - Forum for Information Retrieval Evaluation, CEUR, 2023.

[30] K. Ghosh, A. Senapati, A. S. Pal, Annihilate Hates (Task 4, HASOC 2023): Hate Speech Detection in Assamese, Bengali, and Bodo languages, in: Working Notes of FIRE 2023 - Forum for Information Retrieval Evaluation, CEUR, 2023.

[31] T. Ranasinghe, K. Ghosh, A. S. Pal, A. Senapati, A. E. Dmonte, M. Zampieri, S. Modha, S. Satapara, Overview of the HASOC subtracks at FIRE 2023: Hate speech and offensive content identification in assamese, bengali, bodo, gujarati and sinhala, in: Proceedings of the 15th Annual Meeting of the Forum for Information Retrieval Evaluation, FIRE 2023, Goa, India. December 15-18, 2023, ACM, 2023.