Zero-Shot and Multitask Learning Synergy for Robust Hate Speech Detection across English and Bangla

Kavya G[1,*], Asha Hegde[2], Sonith D[3], Subrahmanya[4] and H L Shashirekha[5]

Department of Computer Science, Mangalore University, Mangalore, Karnataka, India

Abstract

Hate speech content is the textual content that disparages or targets someone because of their race, religion, or gender, whereas offensive content includes any material that may cause discomfort or insult but may not necessarily be hateful. Detecting such content on social media is crucial to prevent harm, ensure a safe online environment and uphold community standards. The ever-evolving and diversified characteristics of online media language, the need for context-aware analysis, and the delicate balance between moderation and free speech are the main obstacles to identify Hate Speech and Offensive Content (HASOC). In this direction, "HASOC - Hate Speech and Offensive Content Detection" - a shared task organized at Forum for Information Retrieval Evaluation (FIRE) 2024, invites the research community to address the challenges of HASOC on social media in English and Bangla language. This task consists of two subtasks: Task 1: Binary Classification in English: focused on HASOC identification in Hinglish, and Task 2: Code-mixed classification in Bangla. To explore the strategies for HASOC detection on social media platforms, in this paper, we - team MUCS, proposed Zero_CS_KW+LD - a Zero-Shot Learning (ZSL) approach for Task 1, and the challenges of Task 2 are addressed by implementing Multi Task Learning (MTL) using Transfer Learning (TL) approach with two transformer models (Bidirectional Encoder Representations from Transformers (BERT) and Distilled version of Multilingual BERT (DistilMBERT)). Among the submitted models, Zero_CS_KW+LD model obtained macro F1 score of 0.5653 securing 7th rank in Task 1 and the proposed MTL model using DistilMBERT obtained macro F1 scores of 0.6761 and 0.3975 securing 4th and 1st ranks for Offensive_gold task and Target_gold task, respectively in Task 2.

Keywords

Hate and Offensive content, Zero-Shot Learning, Cosine Similarity, Multi-task Learning, Label Description

1. Introduction

People's interactions and communication have drastically changed as a result of social media platforms like Facebook and Twitter, which allow users to freely express their opinions about anything. Some users are exploiting this unprecedented level of openness and the anonymity of users on social media platforms to spread harmful content in the form of hate speech, and offensive and abusive language [1, 2, 3]. HASOC targeting people or group based on characteristics like race, gender, religion, or nationality, are among the most troubling forms of content. While hate speech refers to statements that encourage violence or prejudice against the targeted groups [4], offensive content includes hurtful or disparaging remarks that foster a hostile atmosphere which may not be violent. The spread of HASOC on social media may harm mental health of people leading to depression or in severe cases, suicidal thoughts [5, 6]. Hence, to maintain a secure social media, it is essential to identify and handle such content. Detecting HASOC manually on social media platforms presents several challenges due to the vast volume of user-generated content necessitating efficient and accurate automated systems.

India is a diverse multilingual country with each state having its own official language and most of the Indians particularly who are actively engaged in social media are comfortable in using at least two languages including English. As social media platforms do not impose any standards for language the users' use for their communication, most of the content generated by the users will be code-mixed which includes a blend of sentences, words or sub-words in more than one language and more than one script, out of which one language will be prominently English [7]. This linguistic complexity adds

Forum for Information Retrieval Evaluation, December 12-15, 2024, India

*Corresponding author.

kavyamujk@gmail.com (K. G); hegdekasha@gmail.com (A. Hegde); sonithksd@gmail.com (S. D); subrahmanyapoojary789@gmail.com (Subrahmanya); hlsrekha@mangaloreuniversity.ac.in (H. L. Shashirekha)

a layer of challenges for conventional models to accurately identify HASOC in user-generated text. Further, many of the existing HASOC detection systems are designed primarily for major languages like English, Spanish leaving gaps in support for less-represented languages like Bangla, Punjabi, Telugu, Tamil, Malayalam, Urdu and so on. These languages being under-resourced lack annotated datasets and computational tools. This highlights the need for resources and robust computational tools to identify HASOC on social media.

"HASOC 2024"[1] shared task organized at FIRE[2] 2024, invites researchers to develop models to address the challenges of detecting HASOC in code-mixed Hinglish and Bangla text. The shared task consists of two subtasks: i) Task 1 - Binary Classification in English: a task focused on HASOC identification offered for Hinglish is a coarse-grained binary classification to classify tweets into one of two classes: hate and offensive (HOF) and non- hate and offensive (NOT) and ii) Task 2 - Code-mixed classification in Bangla: this task utilizes Offensive Language Identification Dataset's (OLID's)[3] taxonomy to classify code-mixed Bangla tweets in romanized script into one of two categories: Offensive (O) - includes any form of unacceptable language or targeted offenses, or Non-offensive (N) - contains no offensive language, in level A. Additionally, the given tweet also has to be classified as targeted at an Individual (I), a Group (G), or Untargeted (U), in levels B and C which are integrated. In this paper, we describe the proposed learning models to address challenges of the shared task. As no task-specific training data is provided by the organizers, and participants were allowed to use any external resources and datasets to train the models for Task 1, this task is modeled as ZSL problem. ZSL is a Machine Learning (ML) approach that leverages the general knowledge and relationships between known and unknown categories and enables the models to classify the given unlabeled samples into categories/classes they haven't been explicitly trained on [8]. Task 2 involves classifying the given Bangla tweet at two levels: level A and levels B and C (integrated as one level) and the training set for both the levels remain the same. Hence, this task is addressed as two sub-tasks with the same training data and modeled as a MTL problem, allowing the model to learn from multiple related tasks simultaneously. MTL is a ML approach where a single model is trained to perform multiple tasks simultaneously, leveraging shared information across tasks to improve learning efficiency and performance. This approach allows the model to generalize better by learning common representations and features that are beneficial for the two sub-tasks involved [9]. ZSL and MTL techniques allow for more robust and adaptable systems to address the diverse and evolving nature of harmful content across different languages and contexts.

The rest of paper is organized as follows: Section 2 describes the recent literature on HASOC detection and Section 3 focuses on the description of the proposed models followed by the experiments and results in Section 4. The paper concludes with future works in Section 5.

2. Related Work

Identifying HASOC in social media involves recognizing the content that threatens individuals based on race, disability, or socioeconomic status. This task is challenging due to the complexity of such content, which can manifest in many forms, including indirect phrases and suggestive language, and can differ significantly across various cultural contexts. Researchers have explored various approaches including ZSL and MTL to identify HASOC. Some notable works are described below:

2.1. Zero-Shot Learning

Goldzycher and Schneider [10] have explored Natural Language Inference (NLI) models for zero-shot text classification by combining multiple hypotheses to improve NLI-based zero-shot hate speech detection in English. They fine-tuned BERT-based hate speech detection models on HateCheck and ETHOS datasets and obtained accuracies of 79.4% and 69.6% respectively. The zero-shot multilingual NLI model

[1] https://hasocfire.github.io/hasoc/2024/
[2] http://fire.irsi.res.in/fire/2024/home
[3] https://github.com/LanguageTechnologyLab/TB-OLID

(Cross-lingual Language Model - Robustly optimized BERT approach (XLM-RoBERTa), Multilingual Decoding-enhanced BERT with attention v3 (mDeBERTa-v3), Multilingual BERT (mBERT), Multilingual Text-to-Text Transfer Transformer (MT5), mDistillBERT) using Siamese network-based contrastive training on multilingual data (English, Hindi, Spanish etc.) to achieve universal zero-shot NLI proposed by Kowsher et al. [11], effectively captures meaningful semantic relationships. Among the proposed multilingual zero-shot models, NLI XLM-RoBERTa model outperformed other models with macro F1 score of 0.83. Kumar and Albuquerque [12] implemented cross-lingual XLM-RoBERTa classification model by fine-tuning English language sentiment analysis Twitter dataset and subsequently used zero-shot TL to evaluate the classification model on two Hindi sentence-level sentiment analysis (IITP-Movie and IITP-Product review) datasets. Their proposed model achieved an accuracy of 60.93 on both datasets. Yadav et al. [13] proposed zero-shot classification using the Bidirectional Auto-Regressive Transformers (BART) large model, one-shot and few-shot prompting using Generative Pre-trained Transformer-3 (ChatGPT-3) for hate speech detection in code-mixed Hinglish language. Among their proposed models, Zero-shot with BART model exhibited comparably good macro F1 score of 0.5245.

2.2. Multi-Task Learning

By jointly modeling hate and offensive content detection with related concepts like sentiment and emotion analysis, researchers have achieved significant performance improvements. Dai et al. [14] presented BERT-Based MTL models to tackle offensive language detection in English demonstrating improvements in handling the task through hierarchical model architecture. A novel MTL formulation for identifying four types of hate speech - religion, race, disability, and sexual orientation, through a fuzzy ensemble approach proposed by Liu et al. [15] utilizes single-labeled data for semi-supervised multi-label learning. Also two new metrics - detection rate and irrelevance rate, were proposed to measure the performance of this kind of learning tasks more effectively. The authors' experimental study on identifying four types of hate speech demonstrated that fuzzy ensemble approach (based on the mixed fuzzy rule formation algorithm) significantly outperformed popular probabilistic (Support Vector Machines (SVM) and multiple Deep Neural Network DNNs) methods with an overall detection rate of 0.93. Kapil and Ekbal [16] presented Convolution Neural Network (CNN) based MTL models (Random word vectors-CNN, Word-CNN, Char-CNN, Hybrid-CNN, CNN-Word-Attention, Word-CNN-Fully Shared MTL, Soft Sharing CNN-Word-MTL) for hate speech detection in English twitter dataset. Their proposed Word-CNN-Fully shared MTL model obtained comparably good macro F1 score of 0.8737. Mishra et al. [17] trained a MTL model with separate task heads using back-translation and multi-lingual approaches, for HASOC identification in Indo-European languages. The authors compared the performances of their models with models of the participants in HASOC 2019 task, and showed improvements in performances of their proposed MTL models by obtaining macro F1 scores of 0.765, 0.814, 0.612 for English, Hindi, German datasets respectively.

The above studies highlight a range of techniques for hate speech detection, with some approaches yielding lower macro F1 scores, indicating that there is still potential for further improvements. Further, the creativity of users in generating code-mixed content underscores the need for continued research and development to further refine these techniques.

3. Methodology

Task 1 shared task is modeled as ZSL and Task 2 as MTL and the proposed methodologies are described in the following subsections:

3.1. Task 1 - Binary Classification in English: Zero-Shot Learning Approach

ZSL is the task of predicting a class label that was not seen by the model during training. This method which leverages a pre-trained language model can be thought of as an instance of TL which generally refers to using a model trained for one task in a different application than what it was originally trained

for. This is particularly useful for situations where the amount of labeled data is small or not available. As there is no specific training data for Task 1, the proposed ZSL models are built on label descriptions for 'Hate and Offensive' (HOF) and 'Non-hate and Offensive' (NOT), the predefined classes in Task 1. Label descriptions are semantic representations of categories used in the classification tasks to help models understand the meaning of each labels. In ZSL, label description is beneficial as it allows the model to infer and categorize new, unseen data based on predefined labels, even without explicit examples during training [18]. Providing well-defined label descriptions helps the model generalize better and make accurate predictions on novel data by leveraging its understanding of the semantic differences between the classes.

In this study, we used the dataset of Subtask 2: Identification of Conversational Hate-Speech in Code-Mixed Languages (ICHCL) of HASOC 2021 shared task [19], to create label descriptions for the classes of Task 1. Subtask 2 dataset consisting of Hindi-English (Hinglish) code-mixed text samples categorized into two classes: 'Hate and Offensive' (HOF) and 'Non- hate and offensive' (NOT), aligns perfectly with the predefined labels of Task 1. Building a ZSL model typically involves creating label description for each class in the given task and obtaining the semantic representations for the label descriptions. Semantic representations can be obtained through pre-trained models like GloVe, Word2Vec, or transformer models like BERT or RoBERTa.

To facilitate ZSL, effective pre-processing of the data is essential. Pre-processing is the process of cleaning and transforming raw data into a format that is suitable for analysis. In this study, emojis are converted to their textual descriptions, and emoji characters that fall outside the specific Unicode ranges are removed. Further, user mentions and URLs are eliminated in addition to removing retweet indicators, newline characters, and excessive whitespaces. By following these pre-processing steps, the text is refined to a uniform format suitable for subsequent analysis.

The Subtask 2 dataset undergoes pre-processing and is randomly split into: i) Train set - consisting of 100 samples each for the two classes to create label descriptions for the classes and ii) Validation set - consisting of 250 samples each for the two classes to evaluate the model. The crux of ZSL lies in creating label descriptions for the classes and three methods proposed to create label descriptions are given below:

- **Zero_CS_LD** - To create label descriptions for each class, the samples belonging to each class are grouped into ten clusters using k-means algorithm and then the samples belonging to each cluster are merged to get ten label descriptions. HingBERT-Mixed[4] model is used to represent these label descriptions and the mean of these representations are computed to get the label embeddings for each class. HingBERT-Mixed model available on Hugging Face has been trained on a bilingual corpus, making it well-suited for extracting meaningful features from Hinglish text [20].
- **Zero_CS_KW+LD** - The steps involved in creating label description for each class is as follows:
  - All samples belonging to the class are merged and Hindi and English stop words are removed
  - Keywords are extracted automatically based on term frequency
  - A list of ten keywords describing the class as shown in Table 1 is manually curated
  - Both the keywords are merged together to get label description
  - HingBERT-Mixed model is used to obtain the semantic representation for the label description

  Keywords ensure the inclusion of essential terms related to each label enriching the texts semantic representation. Further, it enhances the models ability to generalize and accurately classify new data based on the provided definitions.
- **Zero_NLI_KW+LD** - The procedure used to obtain label descriptions and representation for the label descriptions is the same as in Zero_CS_KW+LD. However, it differs in label prediction for the given input.

[4]https://huggingface.co/l3cube-pune/hing-mbert-mixed

Table 1
Manually created list of keywords describing 'HOF' and 'NOT' class

Class Label	Keywords
HOF	'hate', 'offensive', 'disgust', 'anger', 'violation', 'नफरत', 'अस्वीकृति', 'घृणा', 'क्रोध', 'अपराध'
NOT	'great', 'positive', 'happy', 'excellent', 'wonderful', 'अच्छा', 'सकारात्मक', 'खुश', 'उत्तम', 'शानदार'

The given Test set is preprocessed as mentioned earlier and HingBERT-Mixed model is used to get the semantic representation of the preprocessed Test set. During inference using Zero_CS_LD and Zero_CS_KW+LD models, cosine similarity measure of the semantic representation of the test instance and that of the label descriptions of classes is computed and name of class with the highest similarity score is assigned to the test instance. This effectively allows the model to classify the instance into categories it has never explicitly trained on. Further, this approach enables the model to generalize from seen classes to unseen ones by leveraging the shared semantic space of the embeddings [18]. For inference using Zero_NLI_KW+LD model, sentence transformer based on distilbert-base-multilingual-cased[5] - a variant of DistilBERT fine-tuned for NLI in a multilingual setting, is used to classify the test instances. The Test instance is given as input into a fine-tuned NLI model that outputs scores for hypotheses based on the class labels "HOF" and "NOT", enabling the selection of the label with the highest score as the predicted classification. Thus, by leveraging the label description for each class in the given task, the ability of the model is enhanced to classify unseen instances by effectively utilizing the semantic relationships between the classes and the unseen instances.

3.2. Task 2 - Code-mixed Classification in Bangla: Multi-task Learning Approach

This task focuses on identifying offensive comments in code-mixed Bangla tweet in the first level and then classifying type of target of the offensive comment as 'Individual', 'Group', or 'Untargeted' in the next level. Classification of the given tweet at two levels is addressed as MTL problem with a single model performing both the tasks simultaneously [21]. MTL is a ML technique where a model is trained to perform multiple related tasks simultaneously by sharing certain network layers and parameters. This shared learning approach enhances model performance, particularly when individual tasks lack sufficient data for training. MTL architecture leverages shared lower-level features across tasks while learning higher-level features that are specific to each task. This enables the model to generalize better and efficiently utilize limited data for related tasks. By learning both tasks in a unified framework, the model benefits from shared representations that enhance its ability to detect offensive content and accurately classify the target type. Figure 1 depicts the framework of the proposed MTL model and the steps involved in implementing this model is given below

3.2.1. Pre-processing

The given dataset contains noisy and unstructured text in the form of irregularities and inconsistencies in the words, frequent use of user mentions, hashtags, and informal language. As user mentions, hashtags, urls, digits and punctuation does not contribute to classification task, they are removed from the given dataset. Further, English stopwords available in Natural Language Tool Kit (NLTK)[6] are used as references to remove English stopwords from the given dataset to retain only the content words. This step helps to improve the model's ability to learn from the data by reducing noise and capturing relevant linguistic patterns in code-mixed Bangla text.

3.2.2. Text Representation

Text representation directly impacts the model's performance on tasks like classification [22], sentiment analysis [23, 24], and language understanding [25]. Effective text representation methods, such as

[5]https://huggingface.co/pritamdeka/distilbert-base-multilingual-cased-indicxnli-random-negatives-v1
[6]https://www.nltk.org/

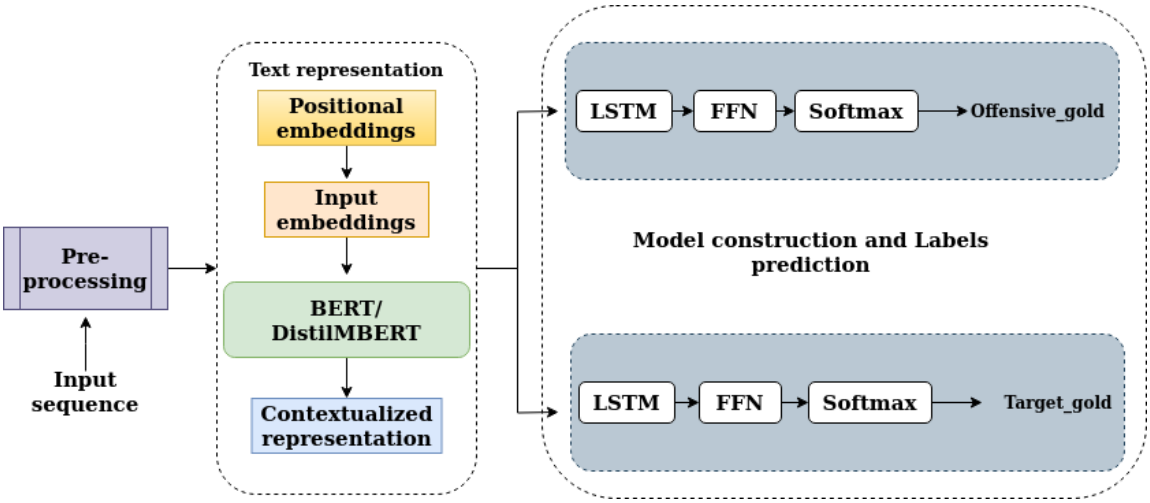


Figure 1: Framework of the proposed MTL model

word embeddings or transformer-based models, aim to capture the contextual and semantic nuances of the text. By converting text into meaningful feature vectors, these representations provide valuable information that helps learning models to understand and generalize patterns in the given data. This work makes use of BERT and DistilMBERT models for text representation and description of these models is given below:

- **BERT**[7] - utilizes 'BertTokenizer' and 'TFBertModel' from Hugging Face's Transformers library for tokenization and loading the pre-trained BERT language model for English text respectively. 'BertTokenizer' is trained on a vast corpus of English text and uses WordPiece tokenization to split words into sub-word units. 'TFBertModel' class loads the pre-trained BERT model, which captures contextual information from both the left and right sides of a given text input. This allows the model to effectively predict the next word in a sentence by understanding the full context.

- **DistilMBERT**[8] - is a compact and faster mBERT variant, created using knowledge distillation and trained on Wikipedia content in 104 languages including Bangla. It is designed to be more efficient, offering roughly twice the speed of mBERT-base while maintaining strong multilingual performance. This makes DistilMBERT an effective choice for resource-constrained environments.

Although BERT model is pre-trained on English text, it effectively captures the relevant features to train the learning models as the given dataset contains romanized Bangla text and more content words in English.

3.2.3. Model Building

MTL approach is particularly effective when labels are hierarchical and designed to be inclusive from top to bottom [26]. This hierarchical structure allows the model to leverage shared information across labels, improving learning efficiency and performance. As illustrated in Figure 1, the proposed MTL architecture leverages contextualized representations through a shared model and then divides into two distinct modules for individual subtasks. The architecture employs BERT/DistilMBERT to generate contextualized embeddings from the input sequence, which are subsequently processed by separate Recurrent Neural Network (RNN) modules equipped with Long Short-Term Memory (LSTM) cells for each subtask. Each module uses these embeddings to produce probability distributions for its respective target labels. The overall loss L is computed as $L = \sum_{i=1}^{I} w_i L_i$, where I denotes the number of labels

[7] google-bert/bert-base-uncased
[8] distilbert/distilbert-base-multilingual-cased

Table 2
Hyperparameters and their values used to train the proposed MTL classifier

Hyperparameters	Values
Epoch	30
Batch size	16
Learning rate	5.00E-05
Optimizer	AdamW
Loss function	CrossEntropyLoss

Table 3
Task 1: Performance of the proposed models on Validation set

Model	Precision	Recall	Macro F1 score	Weighted F1 score	Accuracy
Zero_CS_LD	0.67	0.66	0.66	0.66	0.66
Zero_CS_KW+LD	0.72	0.71	0.70	0.70	0.70
Zero_NLI_KW+LD	0.58	0.58	0.58	0.58	0.58

Table 4
Task 1: Performance of the proposed models on Test set

Model	Macro F1 score
Zero_CS_LD	0.2671
Zero_CS_KW+LD	**0.5653**
Zero_NLI_KW+LD	0.3752

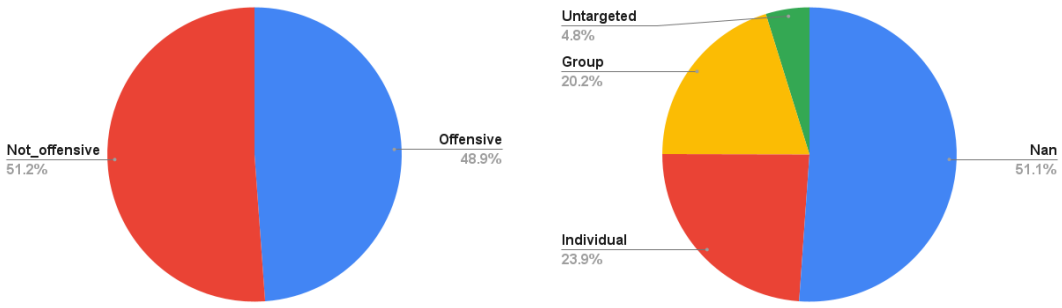
and w_i represents the loss weights for each subtask. This design allows for effective learning across multiple tasks by leveraging shared feature vectors. The hyperparameters and their values used to train the MTL classifiers are shown in Table 2.

4. Experiments and Results

Various learning models were implemented with ZSL and MTL for Task 1 and Task 2 respectively. The Test set for Task 1 provided by the organizers consists of 888 samples and performances of the proposed models on Validation (taken from Subtask 2 dataset) and Test sets are shown in Tables 3 and 4 respectively. Among all the submitted models, Zero_CS_KW+LD model (using keywords for label description and cosine similarity for comparison) outperformed other models with macro F1 score of 0.5653 securing 7^{th} rank.

The shared task organizers provided 4,000, 1,000, and 500 code-mixed Bangla text[9] samples for Training, Development, and Testing, respectively for Task 2 and the class-wise distribution of the dataset is shown in Figure 2. Several experiments were conducted using different combinations of features and classifiers and the models that demonstrated best performances on the Development sets were subsequently applied to predict labels for the Test sets. Performances of the models submitted by the participants for Task 2 were evaluated by the organizers in terms of macro F1 scores and performances of our proposed MTL models on Development and Test sets are shown in Table 5. The proposed model using DistilMBERT obtained macro F1 scores of 0.6761 for Offensive_gold task securing 4^{th} rank and 0.3975 for Target_gold task securing 1^{st} rank, in Task 2. Figures 3 and 4 give a comparison of macro F1 scores of all the participating teams in Task 1 and Task 2 respectively.

[9]https://github.com/LanguageTechnologyLab/TB-OLID



(a) Offensive_gold (b) Target_gold

Figure 2: Classwise distribution of code-mixed Bangla text for Task 2

Table 5
Performances of the proposed MTL models

Model name	Development set		Test set	
	Offensive_gold	Target_gold	Offensive_gold	Target_gold
MTL+BERT	0.6936	0.4510	0.6731	0.3856
MTL+DistilMBERT	0.6801	0.4193	**0.6761**	**0.3975**

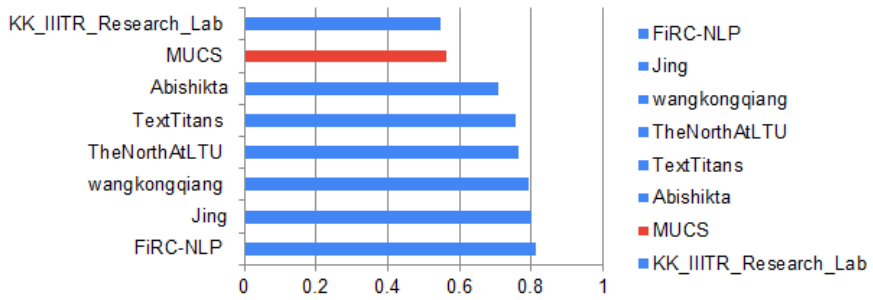


Figure 3: Comparison of performances of the participating teams in Task 1

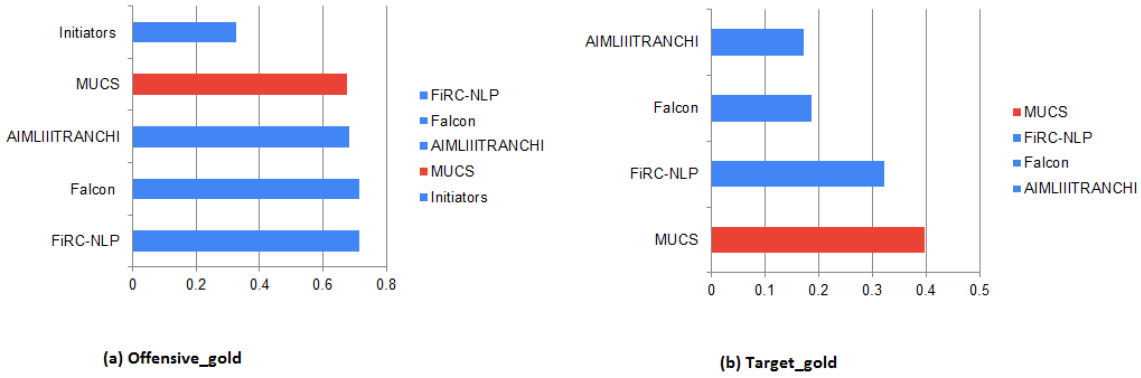


Figure 4: Comparison of performances of the participating teams in Task 2

5. Conclusion and Future Work

In this paper, we - team MUCS, describe the models submitted to "HASOC - Hate Speech and Offensive Content Detection" - a shared task at "FIRE 2024", to distinguish between the categories of HASOC English and Bangla comments. The shared task consists of two subtasks Task 1 and Task 2. As no

training data is given for Task 1, this task is addressed as ZSL with label descriptions and the models: i) Zero_CS_LD - label descriptions for the class labels obtained from the HASOC 2021 Subtask 2 dataset, representing them using HingBERT-Mixed model, and assigning the class labels to the test set based on the cosine similarity between the semantic representations of the test sample and the label descriptions of the predefined classes, ii) Zero_CS_KW+LD - label descriptions for the class labels obtained through keywords from the HASOC 2021 Subtask 2 dataset and manually curated keywords, representing them using HingBERT-Mixed model, and assigning the class labels to the test set based on the cosine similarity between the semantic representations of the test sample and the label descriptions of the predefined classes, and iii) Zero_NLI_KW+LD - using the same procedure as in Zero_CS_KW+LD for label descriptions and their representations, it uses a sentence transformer based on distilbert-base-multilingual-cased NLI model, generating scores for hypotheses based on the class labels 'HOF' and 'NOT' and selecting the label with the highest score as the predicted classification. Among all the submitted models, Zero_CS_KW+LD model obtained macro F1 score of of 0.5653 securing 7^{th} rank in Task 1. The challenges of Task 2 are addressed by implementing MTL models utilizing TL approach with two transformer models (BERT and DistilMBERT) for identifying HASOC in romanized code-mixed Bangla text. The proposed MTL model using DistilMBERT obtained macro F1 scores of 0.6761 for Offensive_gold task securing 4^{th} rank and 0.3975 for Target_gold task securing 1^{st} rank, outperforming the other model. Investigating diverse label description methods will be explored further to enhance the performance of zero-shot learning models for detecting HASOC in code-mixed Hindi text. Additionally, various loss functions will be explored for MTL models to identify HASOC in Bangla text and other languages.

Declaration on Generative AI

During the preparation of this work, the author(s) used ChatGPT in order to: Grammar and spelling check. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

[1] N. Raihan, K. Ghosh, S. Modha, S. Satapara, T. Gaur, Y. Dave, M. Zampieri, S. Jaki, T. Mandl, Overview of the HASOC Track at FIRE 2024: Hate-Speech Identification in English and Bengali, in: K. Ghosh, T. Mandl, P. Majumder, D. Ganguly (Eds.), Forum for Information Retrieval Evaluation (Working Notes) (FIRE 2024) December 9-13, Gandhinagar, India, CEUR-WS.org, 2024.

[2] K. Ghosh, N. Raihan, S. Modha, S. Satapara, T. Gaur, Y. Dave, M. Zampieri, S. Jaki, T. Mandl, Overview of the HASOC Track at FIRE 2024: Hate-Speech Identification in English and Bengali, in: FIRE '24: Proceedings of the 16th Annual Meeting of the Forum for Information Retrieval Evaluation. December 9-13, Gandhinagar, India, Association for Computing Machinery (ACM), New York, NY, USA, 2024.

[3] S. Satapara, S. Modha, T. Mandl, H. Madhu, P. Majumder, Overview of the HASOC Subtrack at FIRE 2021: Conversational Hate Speech Detection in Code-mixed language., in: FIRE (Working Notes), 2021, pp. 20–31.

[4] S. Modha, T. Mandl, G. K. Shahi, H. Madhu, S. Satapara, T. Ranasinghe, M. Zampieri, Overview of the HASOC subtrack at fire 2021: Hate Speech and Offensive Content Identification in English and Indo-Aryan Languages and Conversational Hate Speech, in: Proceedings of the 13th Annual Meeting of the Forum for Information Retrieval Evaluation, 2021, pp. 1–3.

[5] V. Dikshitha Vani, B. Bharathi, Hate Speech and Offensive Content Identification in Multiple Languages using Machine Learning Algorithms, 2022.

[6] M. Anusha, H. Shashirekha, An Ensemble Model for Hate Speech and Offensive Content Identification in Indo-European Languages., in: FIRE (Working Notes), 2020, pp. 253–259.

[7] V. Pathak, M. Joshi, P. Joshi, M. Mundada, T. Joshi, Kbcnmujal@ HASOC-Dravidian-Codemix-fire2020: Using Machine Learning for Detection of Hate Speech and Offensive Code-mixed Social Media Text, 2021.
[8] P. K. Pushp, M. M. Srivastava, Train Once, Test Anywhere: Zero-shot Learning for Text Classification, in: arXiv preprint arXiv:1712.05972, 2017.
[9] R. Caruana, Multitask Learning, in: Machine learning, volume 28, Springer, 1997, pp. 41–75.
[10] J. Goldzycher, G. Schneider, Hypothesis Engineering for Zero-shot Hate Speech Detection, in: arXiv preprint arXiv:2210.00910, 2022.
[11] M. Kowsher, M. S. I. Sobuj, N. J. Prottasha, M. S. Arefin, Y. Morimoto, Contrastive Learning for Universal Zero-Shot NLI with Cross-Lingual Sentence Embeddings, in: Proceedings of the 3rd Workshop on Multi-lingual Representation Learning (MRL), 2023, pp. 239–252.
[12] A. Kumar, V. H. C. Albuquerque, Sentiment Analysis using XLM-R Transformer and Zero-shot Transfer Learning on Resource-poor Indian Language, in: Transactions on Asian and Low-Resource Language Information Processing, volume 20, ACM New York, NY, 2021, pp. 1–13.
[13] S. Yadav, A. Kaushik, K. McDaid, Leveraging Weakly Annotated Data for Hate Speech Detection in Code-Mixed Hinglish: A Feasibility-Driven Transfer Learning Approach with Large Language Models, in: arXiv preprint arXiv:2403.02121, 2024.
[14] W. Dai, T. Yu, Z. Liu, P. Fung, Kungfupanda at SemEval-2020 task 12: Bert-based Multi-task Learning for Offensive Language Detection, in: arXiv preprint arXiv:2004.13432, 2020.
[15] H. Liu, P. Burnap, W. Alorainy, M. L. Williams, Fuzzy Multi-task Learning for Hate Speech Type Identification, in: The world wide web conference, 2019, pp. 3006–3012.
[16] P. Kapil, A. Ekbal, Leveraging Multi-domain, Heterogeneous Data using Deep Multitask Learning for Hate Speech Detection, in: arXiv preprint arXiv:2103.12412, 2021.
[17] S. Mishra, S. Prasad, S. Mishra, Exploring Multi-task Multi-lingual Learning of Transformer Models for Hate Speech and Offensive Speech Identification in Social Media, in: SN Computer Science, volume 2, Springer, 2021, pp. 1–19.
[18] Y. Xian, C. H. Lampert, B. Schiele, Z. Akata, Zero-shot Learning—a Comprehensive Evaluation of the Good, the Bad and the Ugly, in: IEEE transactions on pattern analysis and machine intelligence, volume 41, IEEE, 2018, pp. 2251–2265.
[19] T. Mandl, S. Modha, G. K. Shahi, H. Madhu, S. Satapara, P. Majumder, J. Schäfer, T. Ranasinghe, M. Zampieri, D. Nandini, et al., Overview of the HASOC Subtrack at fire 2021: Hate Speech and Offensive Content Identification in English and Indo-Aryan Languages, in: arXiv preprint arXiv:2112.09301, 2021.
[20] R. Nayak, R. Joshi, L3Cube-HingCorpus and HingBERT: A Code Mixed Hindi-English Dataset and BERT Language Models, in: Proceedings of the WILDRE-6 Workshop within the 13th Language Resources and Evaluation Conference, European Language Resources Association, Marseille, France, 2022, pp. 7–12. URL: https://aclanthology.org/2022.wildre-1.2.
[21] Y. Zhang, Q. Yang, A Survey on Multi-task Learning, in: IEEE transactions on knowledge and data engineering, volume 34, IEEE, 2021, pp. 5586–5609.
[22] A. Hegde, F. Balouchzahi, K. G, H. L. Shashirekha, Trigger Detection in Social Media Text, in: CLEF 2023 – Conference and Labs of the Evaluation Forum, 18-21 September 2023, Thessaloniki - Greece, 2023.
[23] B. Prathvi, K. Manavi, K. Subrahmanyapoojary, A. Hegde, G. Kavya, H. Shashirekha, MUCS@ DravidianLangTech-2024: A Grid Search Approach to Explore Sentiment Analysis in Code-mixed Tamil and Tulu, in: Proceedings of the Fourth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages, 2024, pp. 257–261.
[24] S. Coelho, A. Hegde, P. Lamani, G. Kavya, H. L. Shashirekha, MUCSD@ DravidianLangTech2023: Predicting Sentiment in Social Media Text using Machine Learning Techniques, in: Proceedings of the Third Workshop on Speech and Language Technologies for Dravidian Languages, 2023, pp. 282–287.
[25] K. Girish, A. Hegde, F. Balouchzahi, H. L. Shashirekha, Profiling Cryptocurrency Influencers with Sentence Transformers., in: CLEF (Working Notes), 2023, pp. 2599–2607.

[26] M. Zampieri, S. Malmasi, P. Nakov, S. Rosenthal, N. Farra, R. Kumar, SemEval-2019 task 6: Identifying and Categorizing Offensive Language in Social Media (offenseval), in: arXiv preprint arXiv:1903.08983, 2019.