TECHSSN1 at SemEval-2024 Task 10: Emotion Classification in Hindi-English Code-Mixed Dialogue using Transformer-based Models

Venkatasai Ojus Yenumulapalli, Pooja Premnath, Parthiban Mohankumar, Rajalakshmi Sivanaiah and Angel Deborah Suseelan
Department of Computer Science and Engineering,
Sri Sivasubramaniya Nadar College of Engineering,
Chennai - 603110, Tamil Nadu, India
{venkatasai2110272, pooja2110152, parthiban2110207}@ssn.edu.in,
{rajalakshmis, angeldeborahs}@ssn.edu.in

Abstract

The increase in the popularity of code mixed languages has resulted in the need to engineer language models for the same . Unlike pure languages, code-mixed languages lack clear grammatical structures, leading to ambiguous sentence constructions. This ambiguity presents significant challenges for natural language processing tasks, including syntactic parsing, word sense disambiguation, and language identification. This paper focuses on emotion recognition of conversations in Hinglish, a mix of Hindi and English, as part of Task 10 of SemEval 2024. The proposed approach explores the usage of standard machine learning models like SVM, MNB and RF, and also BERT-based models for Hindi-English code-mixed data- namely, HingBERT, Hing mBERT and HingRoBERTa for subtask A.

1 Introduction

Code-mixed Hindi and English, also referred to as 'Hinglish', has gained widespread usage, especially in the realm of social media. With the increasing prevalence of code-mixed languages like Hinglish, there arises a necessity to analyze and understand this linguistic material. While language models designed for individual languages like English or Hindi (Ly, 2022) are quite robust and effective, they often struggle to perform well with code-mixed languages. This difficulty stems from the colloquial nature of the conversations in code-mixed dialogue, with no formal grammar rules.

Traditional machine learning models perform well on code-mixed data only when the nature of the classification task is simple, like in the form of sentiment analysis (classification into positive, neutral, and negative emotions). Task 10 of SemEval 2024 (Kumar et al., 2024) contains emotions from the extended Ekman model(Ekman, 1992), which contain emotions that are more complex to discern and distinguish between like contempt versus anger.

This paper explores the usage of both classical machine learning models as well as Transformer-based BERT models, specifically designed for Hinglish data.

2 Related Work

Thakur et al. (2020) delve into the current landscape of Hindi-English code-mixed natural language processing and their work meticulously surveys the progress made in sentiment analysis within this domain while also dissecting the inherent issues and challenges it encounters.

Sentiment analysis in code-mixed data is done in a plethora of ways, spanning from machine translation to corpus processing based on sentence structure. Jadhav et al. (2022) introduced a framework employing a pipeline for the conversion of Hinglish to English, offering a structured approach to the task. Similarly, Sinha and Thakur (2005) present a method for translating Hinglish to both English and Hindi, leveraging Hindi and English morphological analyzers and implementing cross-morphological analysis to achieve accurate conversion. Ensemble learning for identifying emotions in contextual texts was proposed by (Angel Deborah et al., 2020). Additionally, (S et al., 2022) proposed a lexicon-based solution for recognising emotions in Tamil texts.

Das and Singh (2023) embraced a deep learning paradigm, implementing convolutional neural networks (CNN), long short-term memory (LSTM), and bi-directional long short-term memory (Bi-LSTM) for sentiment analysis. Meanwhile, Ravi and Ravi (2016) conclusively identified a combination of TF-IDF vectorizer, gain ratio-based feature selection, and a Radial Basis Function Neural Network (RBFN) as the optimal pipeline for sentiment analysis of Hinglish data. Patwa et al. (2020) utilized M-BERT and the Transformers framework,

Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024), pages 833–838
June 20-21, 2024

diverging from traditional methods. Singh (2021) employed diverse techniques for sentiment analysis of Hinglish, leveraging various embeddings such as count vectorizer and word2vec across different machine learning algorithms including SVM, KNN, and Decision Trees. A similar work by (Deborah et al., 2022) focused on recognizing emotions using Gaussian Process and decision trees.

However, the task of emotion classification poses a much greater challenge compared to the simpler task of sentiment analysis. It necessitates the utilization of specific techniques to process and balance data across a broader spectrum of classes. This paper attempts to utilize both traditional and Transformer based approaches for Hinglish emotion classification.

3 Dataset

The SemEval 2024 Task 10 dataset (Kumar et al., 2023) comprises 8056 samples, featuring fields such as ID, speaker, utterance, and emotion. The ID uniquely identifies each episode of the conversation, while the speaker field denotes the person speaking. The utterance field represents the dialogue, expressed in Hinglish, and the emotion field indicates the corresponding emotion conveyed in the utterance. Adding on, the validation dataset contains 1354 samples while the test dataset contains 1580 samples. Table 1 shows the distribution of labels in the dataset.

Emotion	Count
Anger	819
Contempt	542
Disgust	127
Fear	514
Joy	1596
Neutral	3909
Sadness	558
Surprise	441

Table 1: Distribution of emotions and their respective counts.

4 Data Preprocessing

In the domain of code-mixed emotion recognition, preprocessing the utterances is essential for effective model training. The emotion column, representing a spectrum of eight distinct emotions—'disgust', 'contempt', 'anger', 'neutral', 'sadness', 'fear', and 'surprise'—is encoded using a label encoder for standardized representation. Code-mixed data inherently presents spelling ambiguities, demanding robust normalization techniques. For example, the word 'friend' in Hindi could be spelled as 'dost', 'dhosth', 'dhost' etc. Spelling correction is done using a phonetic similarity assessment. For each word, a phonetic code is computed and identifies feasible correction candidates from a dynamically created phonetic dictionary. The Levenshtein distance metric is used to evaluate the dissimilarity between the input word and potential corrections. This procedure is applied to all the utterances, on each word. The resultant corrected words are subsequently merged to form a spell-corrected utterance. A dictionary of all the speakers is also created, and the speaker names present in the utterances are removed, along with numbers and symbols.

5 Proposed Methodology

5.1 Support Vector Machine, Multinomial Naive Bayes and Random Forest

To classify the utterances into one of the eight emotion classes, emotion labels were encoded using LabelEncoder. The CountVectorizer transformed text into numerical features. Initially, standard classification models like Support Vector Machines (SVM), Multinomial Naive Bayes (MNB), and Random Forest(RF) were utilized. These models were chosen based on their suitability for text classification tasks and their potential effectiveness in handling emotion classification within Hindi-English code-mixed data. These models were trained on the training set and evaluated on the validation set using accuracy and the weighted F1 score metrics. Table 2 and 3 shows the precision scores and other performance metrics of each of the standard machine learning models.

5.2 Long Short Term Memory (LSTM)

A Bidirectional LSTM model was then leveraged to address the challenges that could not be resolved by the SVM, MNB, and RF models. This model architecture is well-suited for sequential data processing tasks due to its inherent ability to capture long-range dependencies in text sequences. Figure 1 shows the architecture diagram of the Bidirectional LSTM model.

This bidirectional processing allows the model to effectively capture contextual information from

Emotion	SVM	MNB	RF
Anger	0.00	0.12	**0.19**
Contempt	**0.33**	0.00	0.17
Disgust	0.00	0.00	**1.00**
Fear	**0.33**	0.00	0.24
Joy	0.55	**0.58**	0.55
Neutral	0.43	0.43	**0.44**
Sadness	0.00	0.27	**0.28**
Surprise	0.22	**0.29**	0.27

Table 2: Precision scores of standard machine learning models

Metric	SVM	MNB	RF
Testing Accuracy	**0.44**	0.40	0.43
Testing Weighted F1	0.31	0.30	**0.33**

Table 3: Performance metrics of standard machine learning models

preceding and succeeding words. The model architecture is described as follows:

Embedding Layer: This layer transforms input words into dense vectors of fixed size. It facilitates the representation of words in a continuous vector space, where similar words have similar representations.

Spatial Dropout1D Layer: This layer applies dropout to the input features with a dropout rate of 0.2. It helps prevent overfitting by randomly dropping input units during training.

Bidirectional LSTM Layers: The model consists of two Bidirectional LSTM layers. Each layer comprises 64 units and processes input sequences in both forward and backward directions.

Dense Layers: Two dense layers follow the LSTM layers. The first dense layer has 64 units and uses the ReLU activation function. The final dense layer has 8 units (equal to the number of emotion classes) and uses the softmax activation function for multi-class classification.

The training parameters are as follows:

Optimizer: The model is optimized using the Adam optimizer, a popular choice for training neural networks due to its adaptive learning rate.

Loss Function: Sparse categorical cross-entropy is used as the loss function, suitable for multi-class classification tasks with integer-encoded target labels.

Early Stopping: Training includes early stopping with a patience of 3 epochs. It monitors the loss metric and restores the best weights when no improvement is observed after the specified number of epochs.

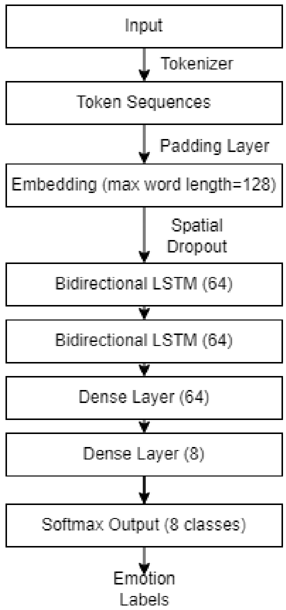

Figure 1: Architecture diagram of the Bidirectional LSTM model

Emotion	LSTM Precision Values
Anger	0.06
Contempt	0.08
Disgust	0.017
Fear	0.48
Joy	0.38
Neutral	0.12
Sadness	0.12
Surprise	0.21

Table 4: Precision scores of LSTM model

Batch Size: Training is performed with a batch size of 32.

Epochs: The model is trained for a maximum of 10 epochs.

The Bidirectional LSTM model achieves a test accuracy of **0.35** with a weighted F1 score of **0.43** on the testing set. Table 4 shows the precision scores of LSTM model.

5.3 Hindi-English Code Mixed BERT Models

The usage of BERT (Bidirectional Encoder Representations from Transformers) models tailored for Hindi-English code-mixed data can significantly enhance the accuracy and effectiveness of emotion classification tasks. These models are pre-trained on large corpora of code-mixed text and can be fine-tuned for specific classification tasks. In this section, three models from the L3Cube Pune team (Nayak and Joshi, 2022), are utilized- namely HingBERT, Hing-mBERT, and HingRoBERTa.

5.3.1 HingBERT

HingBERT, akin to its BERT counterpart, comprises a stack of transformer blocks, typically 12 in number, with self-attention mechanisms and feed-forward neural networks. The model's architecture includes special tokens such as [CLS] and [SEP] to denote sentence boundaries and separation.

5.3.2 Hing mBERT

Hing mBERT inherits the architecture of BERT but is trained across a multitude of languages, including Hindi and English. Its architecture remains consistent with BERT's stack of transformer blocks, each equipped with self-attention mechanisms for capturing contextual information.

5.3.3 Hing RoBERTa

Hing RoBERTa, an extension of the RoBERTa architecture, delves into the intricacies of Hindi-English code-mixed text by integrating advanced architectural modifications. Built upon the foundation of RoBERTa's transformer-based architecture, Hing RoBERTa leverages deeper stacks of transformer layers, intricate attention mechanisms, and optimized weight initialization strategies to handle the nuances of bilingual conversations. With augmented batch sizes and increased learning rates, Hing RoBERTa optimizes gradient descent algorithms to navigate the vast parameter space effectively(Liu et al., 2019). Figure 2 shows the architecture diagram of the Transformer-based models.

5.3.4 Implementation

The implemented framework revolves around fine-tuning the HingBERT, Hing mBERT, and Hing RoBERTa Transformer-based models.

Architecture: The architecture is characterized by the transformer's ability to capture long-range dependencies and intricate contextual nuances within text sequences. Each model comprises a series of transformer blocks, with HingBERT and Hing mBERT featuring 12 transformer layers, while HingRoBERTa encompasses a more extensive architecture with 12 or more layers, as per its pre-defined configuration. Within each transformer block, self-attention mechanisms enable the model to dynamically weigh the importance of individual tokens based on their contextual relevance, facilitating effective feature extraction and representation learning.

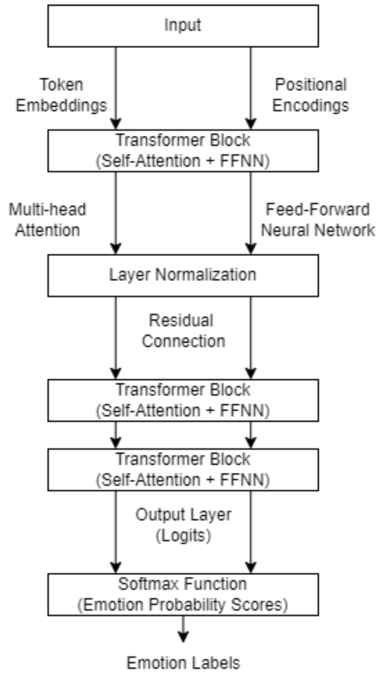

Figure 2: Architecture diagram of Transformer-based models

Multi-Head Attention Mechanism: The attention mechanism, a pivotal component of the transformer architecture, is augmented with multi-head attention, allowing the model to attend to different parts of the input sequence simultaneously.

Feed-Forward Neural Networks (FFNN): Following the self-attention mechanism, token representations are fed through feed-forward neural networks (FFNN) within each transformer block. FFNNs consist of multiple layers of linear transformations, interspersed with non-linear activation functions, such as the Rectified Linear Unit (ReLU), facilitating nonlinear transformations and feature extraction at each layer.

Gradient Clipping: Gradient clipping is employed during the backpropagation phase to alleviate the issue of exploding gradients, ensuring stable training dynamics and promoting convergence.

Embedding Layers: Token embeddings are employed to represent individual tokens within the input sequences, with dimensions determined by the pre-trained embedding matrices. Positional encodings are added to the token embeddings to convey positional information, allowing the model to differentiate between tokens based on their relative positions within the sequence.

Emotion	Hing BERT	Hing mBERT	Hing RoBERTa
Anger	0.28	0.27	**0.33**
Contempt	0.19	0.16	**0.26**
Disgust	**0.25**	0.20	0.20
Fear	0.24	0.23	**0.34**
Joy	0.45	0.49	**0.54**
Neutral	**0.52**	**0.52**	**0.52**
Sadness	0.35	0.28	**0.36**
Surprise	0.31	**0.34**	0.30

Table 5: Precision scores of BERT based models

	Hing BERT	Hing mBERT	Hing RoBERTa
Accuracy	0.45	0.44	**0.47**
Weighted F1	0.42	0.43	**0.45**

Table 6: Performance metrics of BERT based models

Activation Functions and Layer Normalization: Activation functions such as the GELU (Gaussian Error Linear Unit) are applied within the feed-forward neural networks to introduce non-linearity and enable the modeling of complex relationships within the data.

Tables 5 and 6 show the precision value across emotions and the accuracy and weighted F1-scores for the three Transformer-based models.

6 Results and Analysis

6.1 SVM, MNB and RF

The Support Vector Machine (SVM) classifier demonstrates varying performance across different emotions. Notably, it achieves relatively high precision for Contempt and Fear classes, scoring 0.33 for each. However, its precision is very low for Anger, Disgust, and Sadness, achieving 0.00 precision for these emotions. SVM's performance seems to struggle particularly with emotions characterized by intensity and subtlety. Multinomial Naive Bayes (MNB) exhibits competitive performance, particularly evident in its precision for Joy and Surprise emotions, achieving 0.58 and 0.29 respectively, which are among the highest precision values across all models.

Random Forest (RF) emerges as a robust performer across various emotions, demonstrating balanced precision values across the emotion spectrum. RF achieves perfect precision (1.00) for Disgust, indicating its capability to discern this emotion accurately within code-mixed text. Additionally, RF performs consistently well for Neutral and Sadness emotions.

While SVM and MNB show specific strengths for certain emotions, such as Fear and Joy respectively, RF emerges as a more balanced performer across the emotion spectrum, particularly excelling in capturing nuances associated with Disgust.

6.2 LSTM

The LSTM model's precision values exhibit notable variations across different emotions. While it achieves relatively high precision in classifying Fear (0.48) and Joy (0.38), its performance significantly diminishes in categorizing Disgust (0.017) and Anger (0.06). Despite its recurrent nature and ability to retain sequential information, the LSTM model appears to struggle with the contextual intricacies present in the emotion classification task. It achieves a weighted F1-score of 0.43.

6.3 Hindi-English Code-Mixed BERT Models

The BERT-based models showcase more consistent and generally higher precision values across various emotions. Specifically, Hing RoBERTa emerges as the top performer among the BERT-based models, achieving the highest precision scores in several emotional categories, including Contempt (0.26), Fear (0.34), Joy (0.54), and Sadness (0.36). Hing BERT and Hing mBERT also demonstrate competitive precision values, albeit slightly lower than Hing RoBERTa. HingRoBERTa achieves the highest weighted F1-score of 0.45. Table 5 and 6 shows the precision scores and other performance metrics of BERT-based models.

Our team, TechSSN1, placed 7th out of 39 participating teams in the shared subtask A.

7 Conclusion

The future scope of this work entails improving and enhancing the proposed models to handle a wider variety of data. The unstructured nature of Hinglish poses a challenge to the model's performance. By understanding the nuances, fine-tuning can be implemented to enhance the model's efficacy. Additionally, the work can be extended to encompass the classification of other types of emotions apart from the traditional Ekman model and refined to undertake tasks such as sarcasm or humor detection.

References

S Angel Deborah, S Rajalakshmi, S Milton Rajendram, and TT Mirnalinee. 2020. Contextual emotion detection in text using ensemble learning. In *Emerging Trends in Computing and Expert Technology*, pages 1179–1186. Springer.

Shubham Das and Tanya Singh. 2023. Sentiment recognition of hinglish code mixed data using deep learning models based approach. In *2023 13th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, pages 265–269. IEEE.

S Angel Deborah, Rajendram S Milton, TT Mirnalinee, and S Rajalakshmi. 2022. Contextual emotion detection on text using gaussian process and tree based classifiers. *Intelligent Data Analysis*, 26(1):119–132.

Paul Ekman. 1992. An argument for basic emotions. *Cognition & emotion*, 6(3-4):169–200.

Ishali Jadhav, Aditi Kanade, Vishesh Waghmare, Sahej Singh Chandok, and Ashwini Jarali. 2022. Code-mixed hinglish to english language translation framework. In *2022 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS)*, pages 684–688. IEEE.

Shivani Kumar, Md Shad Akhtar, Erik Cambria, and Tanmoy Chakraborty. 2024. Semeval 2024 – task 10: Emotion discovery and reasoning its flip in conversation (ediref). In *Proceedings of the 2024 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics.

Shivani Kumar, Ramaneswaran S, Md Akhtar, and Tanmoy Chakraborty. 2023. From multilingual complexity to emotional clarity: Leveraging commonsense to unveil emotions in code-mixed dialogues. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9638–9652, Singapore. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Cong Khanh Ly. 2022. English as a global language: An exploration of efl learners' beliefs in vietnam. *International Journal of TESOL Education*, 3:19–33.

Ravindra Nayak and Raviraj Joshi. 2022. 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*, pages 7–12, Marseille, France. European Language Resources Association.

Parth Patwa, Gustavo Aguilar, Sudipta Kar, Suraj Pandey, Srinivas Pykl, Björn Gambäck, Tanmoy Chakraborty, Thamar Solorio, and Amitava Das. 2020. Semeval-2020 task 9: Overview of sentiment analysis of code-mixed tweets. *arXiv preprint arXiv:2008.04277*.

Kumar Ravi and Vadlamani Ravi. 2016. Sentiment classification of hinglish text. In *2016 3rd International Conference on Recent Advances in Information Technology (RAIT)*, pages 641–645. IEEE.

Varsini S, Kirthanna Rajan, Angel S, Rajalakshmi Sivanaiah, Sakaya Milton Rajendram, and Mirnalinee T T. 2022. Varsini_and_Kirthanna@DravidianLangTech-ACL2022-emotional analysis in Tamil. In *Proceedings of the Second Workshop on Speech and Language Technologies for Dravidian Languages*, pages 165–169, Dublin, Ireland. Association for Computational Linguistics.

Gaurav Singh. 2021. Sentiment analysis of code-mixed social media text (hinglish). *arXiv preprint arXiv:2102.12149*.

R Mahesh K Sinha and Anil Thakur. 2005. Machine translation of bi-lingual hindi-english (hinglish) text. In *Proceedings of Machine Translation Summit X: Papers*, pages 149–156.

Varsha Thakur, Roshani Sahu, and Somya Omer. 2020. Current state of hinglish text sentiment analysis. In *Proceedings of the International Conference on Innovative Computing & Communications (ICICC)*.