EmoHinD: Fine-grained multilabel emotion recognition from Hindi texts with Deep learning

Debaditya Shome
School of Electronics Engineering
KIIT University,
Bhubaneshwar, India
debadityashome9@gmail.com

***Abstract*—Emotion recognition is a critical sub-field within Affective Computing and a step towards comprehensive natural language understanding. This might pave the way for a wide range of applications in natural language processing for social good, such as suicide prevention and employee mood detection as well as helping businesses provide personalized services to their users. Due to a scarcity of resources and a lack of standard labeled textual corpora for emotions in Hindi, constructing an emotion analysis system is a tough task in such a low resource language. This paper presents a novel multilabel Hindi text dataset of 58000 text samples with 28 emotion labels and a fine-tuned Multilingual BERT-based transformer model on the fine-grained dataset. The model achieves state of the art performance with an overall ROC-AUC score of 0.92 upon evaluation.**

***Index Terms*—Emotion recognition, Deep learning, Dataset, Natural Language Processing**

I. Introduction

Since last few years, deep learning, a sub-field of Artificial Intelligence, has transformed industries and made significant advances in research. It has shown super-human performance in Natural Language Processing [1], Computer Vision [2] and even in other domains like Video games [3] and telecommunications [4]. This decade has witnessed a massive increase in the amount of data generated by social media and the internet, which has captivated the interest of the worldwide Artificial Intelligence community in analysing and developing intelligent systems and models based on it. Natural language processing has become a thriving area as a result of the massive increase in internet-based data and deep learning breakthroughs [5]. Although there has been significant progress in natural language processing for English,There has been minimal recent progress in natural language processing for Indian regional languages, owing mostly to a paucity of labeled datasets and the languages complexity [6].

According to the Ethonologue 24th edition [7], the number of Hindi speakers worldwide is estimated to be over 600 million. This demonstrates the need of developing robust computational linguistic approaches for analysing Hindi textual data which can assist organisations and enterprises in gaining a better understanding of people's particular preferences and psychological characteristics. There has been progress in Affective computing, with particular focus on Emotion recognition from different modalities of data like speech, images or videos [8]. The field of emotion recognition is derived from sentiment analysis, which tries to analyse and classify human language by extracting perspectives, ideas, and thoughts and assigning polarity that are either negative, positive, or neutral. [9]. Although useful, but only classifying into two classes limits the scope of the positive impact that sentiment analysis can achieve, which was the motivation behind development of emotion recognition approaches. Among all the other modalities of data, emotion recognition from text is considered the most challenging as emotions are subjective and differ among people and the varying and unbalanced available datasets and how much does it reflects the true emotion behind the tested statement are the points of concerns [10]. In the next section, efforts in emotion recognition from texts are discussed with a major focus on works on Hindi text based emotion recognition.

II. Related Works

In [11], a dataset for sentiment analysis in Hindi text from online product reviews was presented along with a Support Vector Machine which achieved 54.1% accuracy for classifying the text into positive, negative, neutral and conflict classes. In [12], the authors used an handcrafted algorithm for classifying Hindi language tweets into positive, negative and neutral categories with an overall accuracy of approximately 74%. In [13], a procedure of extracting word embeddings from code mixed Hindi+English text was discussed. These word embeddings were used to train a classifier with 69.84% accuracy to predict positive, negative or neutral sentiments. In [14], the authors put forward a Deep learning system based on a LSTM and CNN combined architecture which was trained on a Hindi tweets dataset for binary classification of positive and negative sentiment tweets. They report an accuracy of 78% in their paper. In [15], a Hindi text based sentiment polarity detection system was proposed where they got an accuracy of 89% for the binary classification problem. In [16], a corpus of 2866 code-mixed English and Hindi tweets with six emotions (happy, sadness, anger, surprise, and sadness) is provided. Their Support Vector Classifier achieves 58.2% accuracy on the multi-class classification task. In [17], a novel dataset of 12K Hindi+English mixed text samples were gathered and a CNN-LSTM model was trained on it which achieved 83.21% classification accuracy to predict three labels: happy, sad and anger.

From the above literature review, it is evident that most of the work has been done on datasets with just few classes of emotions and less number of samples, which in turn calls for the need of fine-grained datasets on Hindi text and labeled emotions. Keeping these points in mind, this paper has the following contribution:

- EmoHinD, a multilabel fine-grained Hindi text dataset with 28 emotion labels is proposed which contains 58K samples.
- A multilingual BERT transformer is fine-tuned on the EmoHinD dataset using a robust training pipeline.
- Even on the complex dataset with so many classes, State of the art performance is achieved compared to previous Hindi text based emotion detection work in literature.

III. Data preparation

For preparation of the data, Google Translate API is used on the well known GoEmotions dataset [18] to translate the English texts to Hindi. For validating the translation, the confidence scores of the Google Translate API are checked in every iteration. If the score is lesser than 70%, then the text is passed through a word-level translation pipeline which uses the state of the art language translation model called mBART transformer [19] until it gets the confidence score higher. The procedure is visually illustrated in Figure 1. For each sample of Hindi text, there is a corresponding one-hot vector of the 28 emotion labels. The multi-processing and parallel computing capabilities were utilized to speed up the translation process.

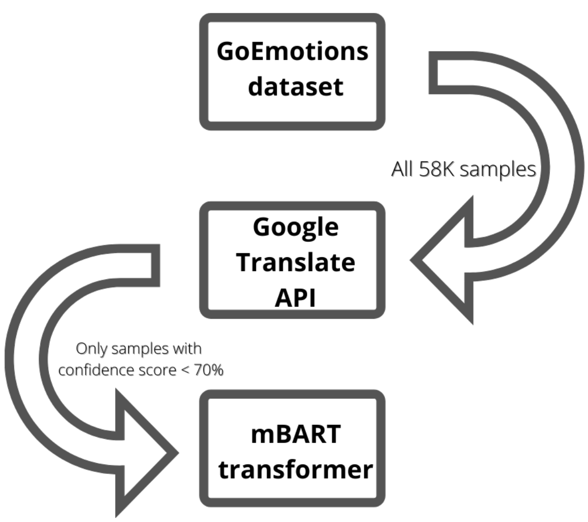

Fig. 1. Development pipeline of EmoHinD dataset

IV. Exploratory Data analysis

The dataset is split in ratio of 80:10:10 in terms of train, test and validation sets. From Figure 2 it can be clearly seen that due to the fact that the last label, ”Neutral” feeling, has a far greater number of samples, there is a class imbalance which would be handled further by the robust model and fine-tuning.

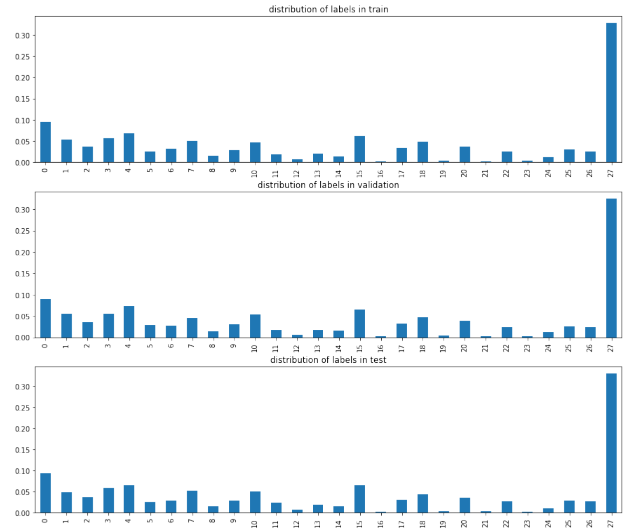

Fig. 2. Class distribution of EmoHinD dataset

V. Methodology

This section delves into the suggested methods for training the transformer on the EmoHinD dataset.

A. Tokenization

As mentioned in the original paper [20], BERT represents the full sentence by utilising the concealed state of the initial token. To do this, an additional token ”CLS” must be manually placed into the input phrase. Another token named ”SEP” is required at each point where a sentence ends and a new sentence begins for next sentence prediction-based pre-training. To begin preparing the input text data, we must first add the ”CLS” token at the beginning and the ”SEP” token at the end of each input text. Additionally, a few ”PAD” tokens need to be added in the dataset for those text samples which have less words than the maximum length chosen for the task. During tokenization, each token is assigned a unique ID. For the words not present in the pre-training phase of BERT, an ”UNK” token is assigned which refers to unknown or out of vocabulary word. As this out of vocabulary problem can lead to significant loss of information, BERT uses of a subword tokenization algorithm called Wordpiece [21]. If part of the unknown word i.e subword is already there as a word in the pre-trained vocabulary of BERT, then the subword is separated from the bigger word and represented as a new token. This makes sure of as much information as possible from the ”UNK” words. Multilingual BERT model and it’s tokenizer is already pre-trained on 104 langauges which includes Hindi, thus re-training the tokenizer on new corpus isn’t necessary for our problem [22].

B. Model Architecture and Fine-Tuning

This work uses the Multilingual BERT model [20], which is a Bi-directional transformer architecture which consists of 12 Transformer blocks / layers. In general, Transformers are built up of multiple layers (called blocks) that are stacked on top of each other. Each block consists of an attention layer followed by a non-linear function applied to each token. BERT employs 12 distinct attention mechanisms for each layer. As a result, at each tier, each token may concentrate on 12 different elements of other tokens. Because Transformers employ a large number of separate attention heads, 12*12=144 in the base BERT model, each head can focus on a different type of component combination. A dropout layer of value 0.3 has been attached internally in the architecture to reduce overfitting.

The fine-tuning procedure on our EmoHinD dataset is illustrated visually in Figure 3. The final classifer block outputs the class probabilities of all the 28 emotions labels.

The following were the features of our training procedure:

- **Optimizer:** The Adam optimizer [23] was used in the training loop. It is a first-order stochastic optimization algorithm widely used for training deep neural networks due it's efficiency in terms of time and space complexity.
- **Loss function:** Due to the fact that this is multi-label classification, Binary Cross-entropy loss is applied to the Sigmoid layer outputs, which include all class probabilities for each type of emotion.
- **Batch size:** A batch-wise dataloader was used in the training loop with a batch size of 64 samples.

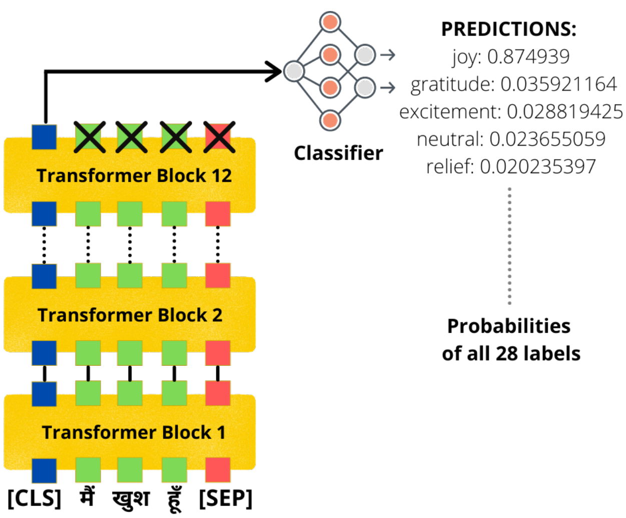

Fig. 3. Multilingual BERT architecture fine-tuned on EmoHinD dataset

VI. Results and Evaluation

For our proposed model, the training and validation perfomance were evaluated using the AUC metric which refers to the the receiver operating characteristic curve's area (ROC)". The AUC metric was preferred over others because it intuitively resembles an aggregate performance over all possible classification thresholds. As mentioned in Section IV, Since of the dataset's class imbalance, assessing on AUC during training delivers a significantly more substantial boost to performance because it separates performance across all 28 classes [24]. All the model training and evaluation was carried out on a minimalistic hardware setup which is outlined in Table I.

TABLE I
Hardware specifications of training setup

Component	Specification
Processor	Intel Core i7 @ 2.60GHz
GPU	NVIDIA GeForce GTX 1650 with Max-Q Design
GPU memory	8113 MB
CUDA cores	896
RAM	8 GB

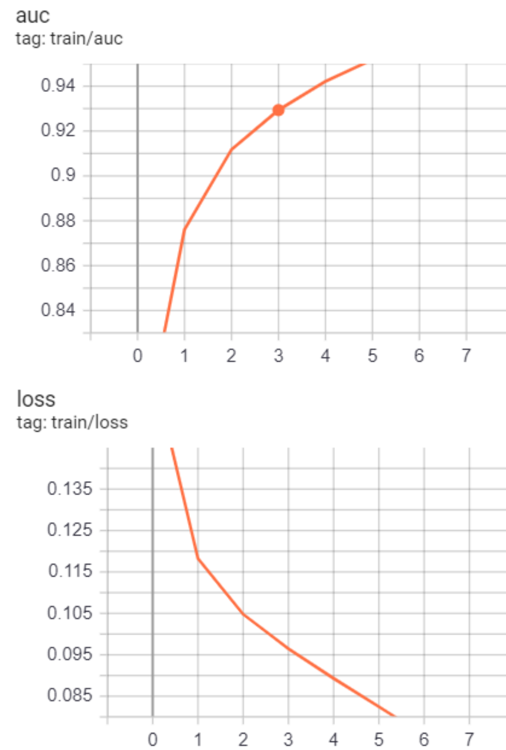

Fig. 4. Performance analysis of model on training dataset

As seen in Figure 4, the model's overall performance on the training data-set escalated to even 0.96 on the AUC scale, which intuitively indicates that on an average, the model distinguishes between all different classes with confidence scores of about 0.96. On the validation set, the AUC score reached to about 0.93 as shown in Figure 5. In both figures it is clear that training as well as validation loss progressively decreases to a global minimum which signifies that there is zero overfitting in our proposed training mechanism. The training and validation metrics were monitored using TensorBoard [25]. Upon evaluating the model on the testing phase dataset, it is observed that the model handles class imbalance very well and achieves cutting-edge performance. The detailed evaluation has been plotted as a bar graph in Figure 6. It is seen that the model performs best in analyzing in labels 15, 18 and 24 which are gratitude, love and remorse with ROC-AUC scores of 0.96, 0.94 and 0.95 respectively. However, it performs poorly on only one category called grief, with an AUC of 0.65 which is most probably due to lesser number of samples in that class as seen earlier in Figure 2.

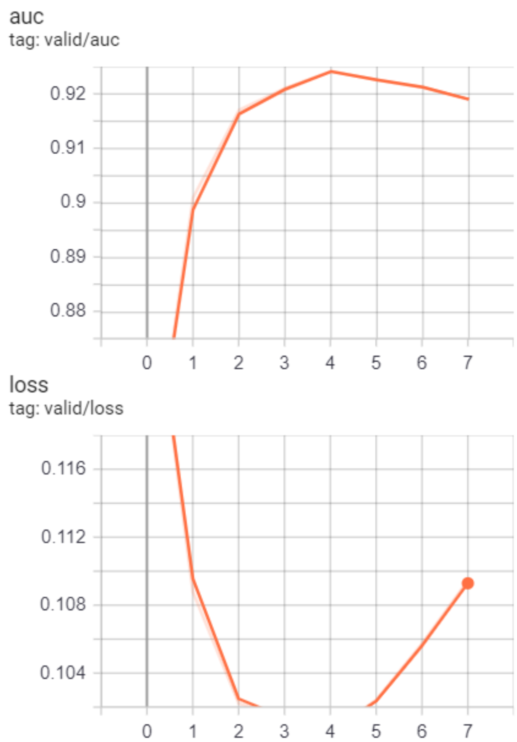

Fig. 5. Performance analysis of model on validation dataset

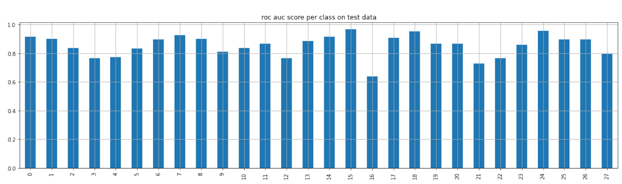

Fig. 6. ROC-AUC score of each emotion class on test set

After evaluating on the testing dataset, further evaluation was done on a few random Hindi phrases as shown in Figure 7. The output prediction probabilities are only shown for top 5 classes in this case. Almost every emotion label has been correctly assigned to the random Hindi text sample which shows that the transformer model is capable of inferencing on out-of-distribution random data and thus is ready for real world deployment purposes.

VII. Conclusion

In this work, EmoHinD, a fine-grained multilabel dataset of Hindi text having labels on a scale of 28 types of emotions is presented along with a state of the art Multilingual BERT model trained on top of EmoHind dataset. The effectiveness of our work has been outlined thoroughly in the evaluation section which shows better performance than most of the previous work on Hindi emotion recognition from text in

```
TEXT: सब ठीक हो जाएगा

PREDICTIONS:
caring 0.7438634
optimism 0.43672588
approval 0.0724037
neutral 0.030243734
gratitude 0.012664043
```

```
TEXT: यह बहुत दुखद है

PREDICTIONS:
sadness 0.8857146
disappointment 0.057790525
neutral 0.026382493
remorse 0.0163662
amusement 0.01611344
```

Fig. 7. Predictions on random Hindi phrases

literature. Although the model achieves good performance on short paragaphs or sentences, the performance decreases when the the amount of text increases. This work opens up a pathway to build up more accurate and generalizable emotion detection models from Indian langauges in the future with bigger and better Natural Language Processing models and datasets.

References

[1] C. Edwards, "The best of nlp," *Communications of the ACM*, vol. 64, no. 4, pp. 9–11, 2021.

[2] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: Scalable and efficient object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10781–10790, 2020.

[3] P. Bory, "Deep new: The shifting narratives of artificial intelligence from deep blue to alphago," *Convergence*, vol. 25, no. 4, pp. 627–642, 2019.

[4] D. Shome and A. Kudeshia, "Deep q-learning for 5g network slicing with diverse resource stipulations and dynamic data traffic," in *2021 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, pp. 134–139, IEEE, 2021.

[5] D. W. Otter, J. R. Medina, and J. K. Kalita, "A survey of the usages of deep learning for natural language processing," *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

[6] B. Harish and R. K. Rangan, "A comprehensive survey on indian regional language processing," *SN Applied Sciences*, vol. 2, no. 7, pp. 1–16, 2020.

[7] Ethnologue, "What are the top 200 most spoken languages?," 2021.

[8] E. Cambria, D. Das, S. Bandyopadhyay, and A. Feraco, "Affective computing and sentiment analysis," in *A practical guide to sentiment analysis*, pp. 1–10, Springer, 2017.

[9] F. A. Acheampong, C. Wenyu, and H. Nunoo-Mensah, "Text-based emotion detection: Advances, challenges, and opportunities," *Engineering Reports*, vol. 2, no. 7, p. e12189, 2020.

[10] S. Al-Saqqa, H. Abdel-Nabi, and A. Awajan, "A survey of textual emotion detection," in *2018 8th International Conference on Computer Science and Information Technology (CSIT)*, pp. 136–142, IEEE, 2018.

[11] M. S. Akhtar, A. Ekbal, and P. Bhattacharyya, "Aspect based sentiment analysis in hindi: resource creation and evaluation," in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pp. 2703–2709, 2016.

[12] Y. Sharma, V. Mangat, and M. Kaur, "A practical approach to sentiment analysis of hindi tweets," in *2015 1st International Conference on Next Generation Computing Technologies (NGCT)*, pp. 677–680, IEEE, 2015.

[13] S. Mukherjee, "Deep learning technique for sentiment analysis of hindi-english code-mixed text using late fusion of character and word features," in *2019 IEEE 16th India Council International Conference (INDICON)*, pp. 1–4, IEEE, 2019.

[14] C. Sindhu, S. Adak, and S. C. Tigga, "Opinionated text classification for hindi tweets using deep learning," in *2021 5th International Conference on Computing Methodologies and Communication (ICCMC)*, pp. 1217–1222, IEEE, 2021.

[15] C. Nanda, M. Dua, and G. Nanda, "Sentiment analysis of movie reviews in hindi language using machine learning," in *2018 International Conference on Communication and Signal Processing (ICCSP)*, pp. 1069–1072, IEEE, 2018.

[16] D. Vijay, A. Bohra, V. Singh, S. S. Akhtar, and M. Shrivastava, "Corpus creation and emotion prediction for hindi-english code-mixed social media text," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, pp. 128–135, 2018.
[17] T. T. Sasidhar, B. Premjith, and K. Soman, "Emotion detection in hinglish (hindi+ english) code-mixed social media text," *Procedia Computer Science*, vol. 171, pp. 1346–1352, 2020.
[18] D. Demszky, D. Movshovitz-Attias, J. Ko, A. Cowen, G. Nemade, and S. Ravi, "Goemotions: A dataset of fine-grained emotions," *arXiv preprint arXiv:2005.00547*, 2020.
[19] Y. Liu, J. Gu, N. Goyal, X. Li, S. Edunov, M. Ghazvininejad, M. Lewis, and L. Zettlemoyer, "Multilingual denoising pre-training for neural machine translation," *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 726–742, 2020.
[20] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.
[21] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, *et al.*, "Google's neural machine translation system: Bridging the gap between human and machine translation," *arXiv preprint arXiv:1609.08144*, 2016.
[22] T. Pires, E. Schlinger, and D. Garrette, "How multilingual is multilingual bert?," *arXiv preprint arXiv:1906.01502*, 2019.
[23] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
[24] J. Huang and C. Ling, "Using auc and accuracy in evaluating learning algorithms," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 3, pp. 299–310, 2005.
[25] M. T. Martinez, "An overview of google's machine intelligence software tensorflow.," 2016.