YNU-HPCC at SemEval-2024 Task10: Pre-trained Language Model for Emotion Discovery and Reasoning its Flip in Conversation

Chenyi Liang , Jin Wang and **Xuejie Zhang**
School of Information Science and Engineering
Yunnan University
Kunming, China
liangchenyi@stu.ynu.edu.cn, {wangjin,xjzhang}@ynu.edu.cn

Abstract

This paper describes the application of fine-tuning pre-trained models for SemEval-2024 Task 10: Emotion Discovery and Reasoning its Flip in Conversation (EDiReF), which requires the prediction of emotions for each utterance in a conversation and the identification of sentences where an emotional flip occurs. This model is built on the DeBERTa transformer model and enhanced for emotion detection and flip reasoning in conversations. It employs specific separators for utterance processing and utilizes specific padding to handle variable-length inputs. Methods such as R-drop, back translation, and focal loss are also employed in the training of my model. The model achieved specific results on the competition's official leaderboard. The code of this paper is available at https://github.com/jiaowoobjiuhao/SemEval-2024-task10.

1 Introduction

Navigating the complexities of emotional dynamics within conversations presents a formidable challenge in natural language processing (NLP). Human interactions are characterized by rapid emotional shifts, influenced by context and subtle linguistic nuances, requiring sophisticated models for accurate capture and interpretation. Thus, understanding and precisely identifying emotions, especially within conversations marked by emotional transitions, is a significant and challenging endeavor in NLP research.

The SemEval-2024 competition introduces the Emotion Discovery and Reasoning its Flip in Conversations (EDiReF) task (Kumar et al., 2024), divided into three subtasks designed to explore the nuanced landscape of emotional dynamics within dialogues:

- Subtask 1: Identify and classify the emotional states expressed in each utterance within a conversation (Kumar et al., 2023a). As shown in Table 1, the emotion of each utterance is identified through the first two columns.
- Subtask 2: Identify specific utterances that mark an emotional transition within Hindi-English code-mixed dialogues (Kumar et al., 2022, 2023b). As shown in Table 1, the triggers of emotions are identified through the first three columns.
- Subtask 3: Identify specific utterances that mark an emotional transition within English conversations (Kumar et al., 2022, 2023b). The first three columns in Table 1 identify emotional reversal triggers.

In the previous sentiment analysis work, various hand-crafted features and sentiment lexicons were utilized to construct solution systems. These systems were developed by integrating traditional methods such as Naive Bayes, Support Vector Machines (SVM) (Mohammad et al., 2013), and Decision Trees (Blake, 2007). Following the advent of deep learning, Convolutional Neural Networks (CNNs) (Kim, 2014), based on Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997)architectures, were employed for sentence classification tasks. Additionally, GloVe (Pennington et al., 2014) was utilized for learning sentence features, and Bidirectional Long Short-Term Memory (Bi-LSTM) (Kong et al., 2020; Zhang et al., 2018) models were applied to sentence classification to enhance performance. However, these methods encountered challenges in effectively capturing the contextual information of longer texts. With the progression toward larger models, BERT-based large-scale pre-training models marked a significant breakthrough in sentiment analysis (Zheng et al., 2022)

This study proposes a deep learning system for Task 10 in SemEval-2024. We use a

Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024), pages 777–784
June 20-21, 2024 ©2024 Association for Computational Linguistics

Speaker	Utterance	Emotion	Trigger
Sp1	I had an awful day today!	Sad	0
Sp2	Oh no! What happened?	Sad	0
Sp1	Somebody ate my sandwich!	Sad	0
Sp2	I can make you a new one right now!	Joy	1
Sp1	That would be great! Thanks!	Joy	0

Table 1: Examples of EDiReF

decoding-enhanced bert with disentangled attention(DeBERTa) (He et al., 2020)sequence classification model as the base model. Our enhancement to the DeBERTa model introduces a pivotal integration of specialized mechanisms for processing [SEP] tokens and handling label padding with -1, along with the innovative incorporation of a KL divergence (Wu et al., 2021) loss function, known as R-drop. This strategic amalgamation ensures that each utterance within a conversation is precisely mapped to its corresponding emotional state, facilitating a more accurate representation of emotional dynamics. Introducing R-drop is critical in preventing overfitting by enforcing consistency between the model's outputs for various data sub-samples, thus enhancing the model's generalization ability across different conversational contexts. The contributions of this study are as follows.

- We introduce a foundational model utilizing a pre-trained Deberta sequence classification model for the label sequence classification issue.
- Incorporation of KL Divergence Loss (R-drop) for Overfitting Prevention and adoption of focal loss to address data imbalance issues.
- The model employs [SEP] tokens and -1 padding to align utterances with their corresponding labels and grasp the context within conversations.

The remainder of this paper is organized as follows: Section 2 provides an overview of our proposed model and system. Section 3 conducted the experiments to analyze the effectiveness of the proposed method. The paper concludes with a summary and reflections in Section 4.

2 System Description

This section delves into the architecture of the proposed model, detailing its essential components: the tokenizer, the pre-trained Deberta model, and the implementation of Regularized Dropout and Focal loss for Neural Networks. Specifically, the model tailored for Task 1, which addresses the multi-label sequence classification problem, is illustrated in Figure 1. Meanwhile, the models designed for Tasks 2 and 3, focusing on binary sequence classification issues, are depicted in Figure 2.

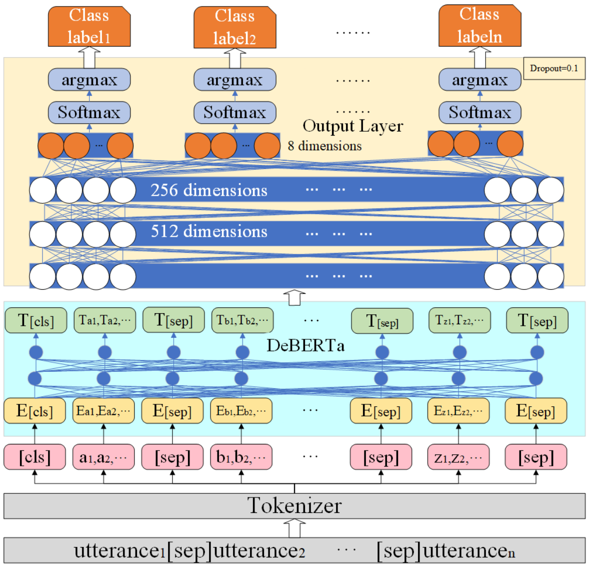

Figure 1: Multi-emotion label sequence classification model

2.1 Tokenizer

Transforming raw text into a machine-readable format is a preliminary step for many NLP tasks. To achieve this, a tokenizer is utilized, segmenting the text into discrete elements and encoding them uniquely. In our model, the DeBERTa tokenizer, mainly designed for handling long texts in sequence classification challenges, is employed to process the text for NLP tasks. Input texts are segmented to accommodate the extensive length of dialogues in subtasks 1 and 2 using a 2048 token cutoff, ensuring comprehensive coverage of conversa-

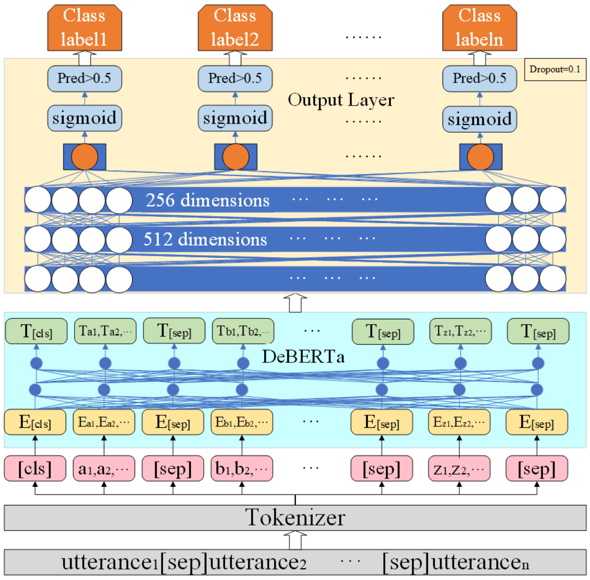

Figure 2: Binary label sequence classification model

tions without truncating critical emotional context in later utterances. For subtask 3, a 1024 token limit is applied, optimizing for shorter textual inputs. The final output X of the tokenizer is denoted as:

$$X = [CLS]\, a_1 \cdots a_n\, [SEP]\, b_1 \cdots b_m\, [SEP] \cdots z_1 \cdots z_p\, [SEP] \quad (1)$$

where n, m, and p denote the lengths of distinct utterances within the dialogue. With [CLS] marking the start and [SEP] serving as a delimiter between utterances, it ensures the model recognizes dialogue flow. For subtasks 1 and 2, sequences shorter than 2048 tokens are padded with zeros, and longer ones are truncated to maintain this limit, optimizing for more extended dialogues. Conversely, subtask 3 employs a 1024 token threshold, adjusting for its specific data structure and requirements.

2.2 DeBERTa Model

DeBERTa enhances BERT's (Devlin et al., 2019) architecture by introducing disentangled attention and an enhanced mask decoder, making it highly suitable for complex dialogue tasks requiring a detailed understanding of context and word positions. Like BERT, DeBERTa comprises two core components: an Embedding block for initial word vector representation and a Transformer Encoder block for deep contextual processing. Additionally, DeBERTa introduces a third major component, the Enhanced Mask Decoder (EMD).

Following the tokenizer's segmentation of input texts and incorporation of special tokens ([CLS] and [SEP]), these tokens are embedded, capturing the nuances of words as vectors that signify their meanings and relationships. The Transformer Encoder further processes these vectors by employing disentangled attention to analyze dialogues' contextual relationships and depth intricately. The EMD, leveraging content and positional information, refines the model's ability to predict and understand masked language elements, thoroughly comprehending dialogue intricacies. Consequently, the final layer's hidden state representation, denoted as H_L, is passed to the output layer, where L represents the number of layers in the Transformer.

2.3 Output Layer

Subtask1. This subtask involves multi-sequence sentiment classification, with the model designed to recognize [sep] and label padding of -1. This setup allows for processing the DeBERTa model's sequence output through a custom classifier, generating logits for each utterance to predict labels, detailed in section 3. The layer initially maps the data dimensions from L to 512 dimensions, then applies the ReLU activation function and dropout to refine and classify the data further, followed by mapping from 512 to 256 dimensions, adding ReLU and Dropout again, and finally mapping to the label dimension (8 dimensions) to obtain logits. After obtaining the classification probability distribution P , calculate the loss with the real classification label y and learn the model weight. The calculation formula of the probability distribution is as follows.

$$P = softmax\,(W_0 H_0 + b_0) \quad (2)$$

where W_0 is the weight matrix of the final linear layer, with dimensions of $R^{8\times256}$, H_0 is the feature vector input to this linear layer, with a dimensionality of 256; and b_0 is the bias term, with a dimensionality of 8.

Subtask2&subtask3. These two subtasks involve binary sequence classification, where the main difference in the output layer from subtask 1 is the transformation of the model's output logits into probabilistic classifications through a sigmoid layer. The calculation formula of the probability distribution is as follows.

$$P = sigmoid\,(W_1 H_1 + b_1) \quad (3)$$

where W_1 is the weight matrix of the final linear layer, with dimensions of $R^{1\times256}$, H_1 is the feature vector input to this linear layer, with a dimensionality of 256, and b_1 is the bias term, with a dimensionality of 8.

2.4 Methods

Regularized Dropout. Due to the existence of dropout, the same model with identical inputs will produce two distinct distributions, effectively treating them as two different network models. Denoted as $P_\theta\left(y|x\right)$ and $P_\theta^{'}\left(y|x\right)$, these distributions represent the output probabilities of the model under dropout conditions. The primary objective of R-Drop is to minimize the KL Divergence between these two distributions throughout the training process. Given the asymmetry of KL divergence, a globally symmetric version is indirectly employed by interchanging the positions of these distributions, a concept known as bidirectional KL divergence. Furthermore, the model is trained on both distributions' negative log-likelihood (NLL) loss terms. Given (x_i, y_i) as training set input ,The final loss is as follows:

$$\begin{aligned} L_{KL}^i &= \alpha\left[D_{KL}\left(P_\theta(y_i|x_i)||P_\theta^{'}(y_i|x_i)\right)\right.\\ &\quad\left.+D_{KL}\left(P_\theta^{'}(y_i|x_i)||P_\theta(y_i|x_i)\right)\right]\\ L_{NLL}^i &= -\log P_\theta(y_i|x_i) - \log P_\theta^{'}(y_i|x_i)\\ L_{R-drop}^i &= L_{KL} + L_{NLL} \end{aligned} \tag{4}$$

Focal Loss. Focal Loss (Lin et al., 2017) is utilized in our model as the primary loss function, specifically designed to mitigate the impact of class imbalance by dynamically adjusting the importance of each class and the difficulty of each sample. Two parameters α and γ are introduced to modulate each class's loss contribution and focus more on challenging, misclassified samples rather than those easily classified. P_t represents the probability of class t output by softmax or sigmoid function and α_t is a training parameter. The formula is listed as follows.

$$FL\left(P_t\right) = -\alpha_t(1-P_t)^\gamma \log\left(P_t\right) \tag{5}$$

3 Experimental Results

3.1 Datasets

The training sets for these three subtasks are derived from dialogues in various scenarios within TV dramas. In subtask 1, 343 training sets are provided, including four columns: episode, speakers, emotions, and utterances, with eight types of emotions contained within the emotions column. For subtasks 2 and 3, 4893 and 4000 training sets are provided, each with an additional column named triggers compared to subtask 1.

3.2 Evaluation Metrics

The evaluation tools employed for these three subtasks are Precision, Recall, and Micro-F1, with their formulas categorized as follows:

$$\begin{aligned} Precision &= \frac{TP}{TP+FP}\\ Recall &= \frac{TP}{TP+FN}\\ F1 &= \frac{2\times Precision\times Recall}{Precision+Recall} \end{aligned} \tag{6}$$

3.3 Implementation Details

Training Set Preprocessing. To align each utterance with its label throughout an entire dialogue and to learn the relationships within the dialogue, each dialogue is separated by [sep]. The label data are filled with -1 to match the maximum number of utterances in the training and validation sets, and a mask is incorporated into the model. This approach ensures that labels marked as -1 are excluded from loss calculation, allowing the model to handle dialogues of varying lengths. Specifically, the maximum number of utterances for subtasks 1 and 2 is 106, while for subtask 3, it is 24. The labels for subtask 1 are emotions, with eight types: anger, contempt, disgust, fear, joy, neutral, sadness, and surprise. These are mapped to data values 0-7, facilitating correct processing by the model. For subtasks 2 and 3, initially in string format as 0 and 1, the label data are converted to floating-point numbers 0.0 and 1.0. Given the limited training dataset for subtask 1, data cleaning and normalization are first performed using ekphrasis, which improves the model's learning from dialogues. Text augmentation is then conducted through back-translation (Edunov et al., 2018) and synonym replacement; Hindi dialogues are translated into English and then back, while the process is reversed for English dialogues. Synonym replacement involves exchanging words with the same meaning for different expressions. Finally, subtask 1 is expanded to 1029 training sets.

Imbalanced Data Handling. Due to the predominant proportion of neutral and joy labels in subtask

contempt	542
disgust	127
fear	514
joy	1596
neutral	3909
sadness	558
surprise	441

Table 2: Occurrences of Emotional Labels in Subtask

1, as illustrated by the quantities in Table 2, as well as the prevalence of the 0.0 label in subtasks 2 and 3, focal loss and oversampling methods (Chawla et al., 2002) have been utilized. This approach enables the model to learn more effectively from samples that appear less frequently, thereby enhancing the model's performance.

Prediction Challenges. When tokenizing text inputs, lengths of 2048 were selected for truncation in subtasks 1 and 2, while 1024 was chosen for subtask 3. However, during the prediction phase for subtask 2, the number of labels predicted fell short of the expected count. This shortfall could be attributed to dialogues in the test set that exceed the maximum length of 2048. The constraints posed by GPU capabilities also resulted in our model's inability to fully perform the prediction task for subtask 2. We hope to try using Longformer (Beltagy et al., 2020) to address the issue of long dialogues in the future.

Model Comparison. For all tasks, `bert-base-cased`, `bert-large-cased`, `deberta-base`, and `debertav2-xlarge` were compared. When employing the `debertav2-xlarge` model, the AdaLoRA (Zhang et al., 2023) model was used for fine-tuning to prevent exceeding the GPU memory limits.

Optimizer and Loss Parameter Configuration. The AdamW (Loshchilov and Hutter, 2017) optimization was employed to train the model across all subtasks, with a batch size of 1. To achieve the expected results, we experimented with different learning rates and epochs to observe their impact on the F1 score for Subtask 1 using the deberta-base model. Figures 3 and 4 are presented below. For subtask 1, the learning rate for AdamW was set at 5e-6, while for subtasks 2 and 3, it was established at 5e-5. Focal loss parameters for subtask 1 were defined as alpha=0.1 and gamma=0.3, while for subtasks 2 and 3, the parameters were set to

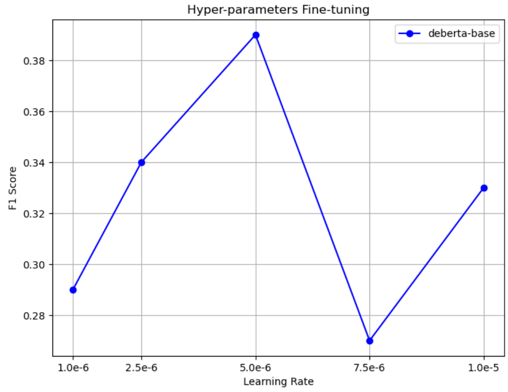

Figure 3: The impact of different learning rates on the F1 score for Subtask 1

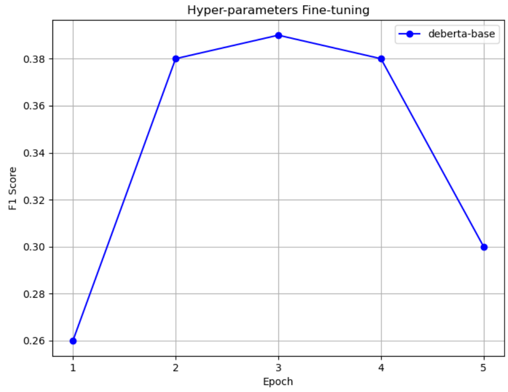

Figure 4: The impact of different learning rates on the F1 score for Subtask 1

alpha=1 and gamma=5.

3.4 Results and Analysis

Subtask1. Validation set results for different models for the multi-label sequence classification task are presented in Table 4. Performance increases from the bert to the DeBERTa phase, yet a significant decline occurs at the `debertav2-xlarge` model phase. This decline may be attributed to the large parameter size of the `debertav2-xlarge` model and the small dataset size, making it challenging for the model to learn features from a small dataset. The overall low scores for the model could be due to the approach of predicting the entire dialogue segment and calculating loss against actual values rather than calculating loss for each utterance individually. This approach might have contributed to the suboptimal performance of our model. Another potential reason could be the selection of 2048 as the trunca-

Speaker	Utterance	Predicted label	True label
maya	indu tumne vah mere earplugs dekhe hain	neutral	neutral
indravardhan	earplugs kyon	anger	surprise
maya	<time>baj rahe hain na madhubhai ki bhatiji ka sone ka time ho gaya hai	neutral	neutral
indravardhan	are baap re yyane announcement shuru ho jayegi	anger	fear
dvd player	train sound	anger	neutral
maya	a <elongated>	anger	fear

Table 3: Model's prediction results on the test set for Subtask 1

Model	Dev set		
	P	R	F1
DeBERTa-base	0.39	0.39	0.39
DeBERTaV2-xlarge	0.18	0.18	0.18
Bert-base	0.28	0.28	0.28
Bert-large	0.31	0.31	0.31

Table 4: Validation set results for different models for Subtask 1

Model	Dev set		
	P	R	F1
DeBERTa	0.25	0.25	0.25
DeBERTa-focalloss	0.36	0.36	0.36
DeBERTa-rdrop	0.35	0.35	0.35
DeBERTa-focalloss-rdrop	0.39	0.39	0.39

Table 5: Validation set results for different methods for Subtask 1

Model	Dev set		
	P	R	F1
DeBERTaV2-xlarge	0.90	0.90	0.90

Table 6: Validation Set Results for Subtask 2

tion value. Although this ensures that a few longer dialogue texts are fully captured, it may hinder the model's ability to learn information from long-distance texts for most shorter dialogues, leading to poor learning outcomes. There is a keen interest in attempting to segment longer texts in the future to mitigate the adverse effects on learning caused by long texts.

As indicated, the model `deberta-base` outperforms others on the validation set. Subsequent experiments will explore the impact of different methods on the model's performance based on `deberta-base`. The results are presented in Table 5, which reveals that the baseline model, not utilizing focal loss or r-drop, and instead using CrossEntropyLoss as the loss function, achieves an F1 score of only 0.25. Introducing either focal loss or r-drop results in improved scores, reaching 0.36 and 0.35, respectively. Combining these two methods and applying them to the `deberta-base` model on the validation set increases the F1 score to 0.39, outperforming the previous three configurations. The experiments demonstrate that both rdrop and focal loss contribute to enhancements in model performance.

The model `deberta-base-focalloss-rdrop` was employed to make predictions on the test set, with the results presented in Table 7, which indicates that the predictions for shorter sentences are not very accurate, which may be due to the model's insufficient learning of brief phrases. Another reason could be that the pre-trained model, `deberta-base`, was primarily trained in English, resulting in inadequate learning for languages like Hindi. Applying a multilingual model might yield better results, and further experiments are hoped to be conducted.

Subtask2. Validation Set Results for the Binary Label Sequence Task are shown in Table 6.When the `debertav2-xlarge` model was attempted for prediction, 2048 was selected as the truncation length for tokenizing the test set dialogues. It was found that the number of predicted labels did not meet the expected count, possibly due to dialogues exceeding the length of 2048, leading to this shortfall. Given the GPU constraints, our model could not effectively predict the test set.

Subtask3. Validation Set Results are presented in Table 8.It was observed that the values of precision and recall are identical across all tasks, which

Speaker	Utterance	Emotion	Predicted trigger	True trigger
Mark	why do all your coffee mugs have numbers on the bottom	surprise	0.0	0.0
Rachel	oh. that is so Monica can keep track. That way if one of them is missing, she can be like, where is number <number>?! <repeated>	anger	0.0	0.0
Rachel	y ’ know what ?	neutral	0.0	0.0

Table 7: Model’s prediction results on the test set for Subtask 3

Model	Dev set		
	P	R	F1
DeBERTa-base	0.82	0.82	0.82
DeBERTaV2-xlarge	0.82	0.82	0.82
Bert-base	0.82	0.82	0.82
Bert-large	0.82	0.82	0.82

Table 8: Validation set results for different models for Subtask 3

may be attributed to using micro-F1 as the evaluation metric and calculating loss based on entire dialogue segments rather than extracting individual utterances. This approach resulted in identical calculated values. The prevalence of 0.0 labels in every dialogue segment possibly made it challenging for the model to learn and perform well on the test set. The aspiration is to learn more practical models in the future to address this issue. The results obtained from predicting the test set using the `debertav2-xlarge` model are shown in Table 7, which shows that the model performs well in identifying non-emotional triggers. Based on my overall prediction results, the model’s ability to predict triggers is unsatisfactory, which is an area I should aim to improve in the future.

4 Conclusions

This paper proposes a deep learning model for sentence sequence classification tasks, utilizing the DeBERTa sentence sequence classification model as the foundation. Achievements have been made in the final submission for SemEval-2024 Task10. However, there remains significant room for improvement in both the model and its parameters. Therefore, in future studies, enhancements will be made to the model to achieve better results.

Acknowledgements

The authors would like to thank the anonymous reviewers for their constructive comments. This work was supported by the National Natural Science Foundation of China (NSFC) under Grant Nos.61966038 and 62266051.

References

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Catherine Blake. 2007. The role of sentence structure in recognizing textual entailment. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 101–106.

Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. 2002. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018. Understanding back-translation at scale. *arXiv preprint arXiv:1808.09381*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.

S Hochreiter and J Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014*

Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL, pages 1746–1751. ACL.

Jun Kong, Jin Wang, and Xuejie Zhang. 2020. Hpcc-ynu at semeval-2020 task 9: A bilingual vector gating mechanism for sentiment analysis of code-mixed text. *arXiv preprint arXiv:2010.04935*.

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, Md Shad Akhtar, Tanmoy Chakraborty, et al. 2023a. From multilingual complexity to emotional clarity: Leveraging commonsense to unveil emotions in code-mixed dialogues. *arXiv preprint arXiv:2310.13080*.

Shivani Kumar, Shubham Dudeja, Md Shad Akhtar, and Tanmoy Chakraborty. 2023b. Emotion flip reasoning in multiparty conversations. *IEEE Transactions on Artificial Intelligence*.

Shivani Kumar, Anubhav Shrimal, Md Shad Akhtar, and Tanmoy Chakraborty. 2022. Discovering emotion and reasoning its flip in multi-party conversations using masked memory network and transformer. *Knowledge-Based Systems*, 240:108112.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Saif M Mohammad, Svetlana Kiritchenko, and Xiaodan Zhu. 2013. Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets. *arXiv preprint arXiv:1308.6242*.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Lijun Wu, Juntao Li, Yue Wang, Qi Meng, Tao Qin, Wei Chen, Min Zhang, Tie-Yan Liu, et al. 2021. R-drop: Regularized dropout for neural networks. *Advances in Neural Information Processing Systems*, 34:10890–10905.

Qingru Zhang, Minshuo Chen, Alexander Bukharin, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. 2023. Adaptive budget allocation for parameter-efficient fine-tuning. *arXiv preprint arXiv:2303.10512*.

You Zhang, Jin Wang, and Xuejie Zhang. 2018. Ynu-hpcc at semeval-2018 task 1: Bilstm with attention based sentiment analysis for affect in tweets. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 273–278.

Guangmin Zheng, Jin Wang, and Xuejie Zhang. 2022. Ynu-hpcc at semeval-2022 task 6: Transformer-based model for intended sarcasm detection in english and arabic. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 956–961.