Project#8: Emotion Discovery and Reasoning its Flip in Conversation

Sachin Bhadang[1], **Yashwant Mahajan**[2], **Ritam Acharya**[3]
[1]200831, [2]201156, [3]210859
[1]ME, [2]EE, [3]CHE
{sachinb20, yashwantm20, ritam21}@iitk.ac.in

Abstract

Efficiently discerning a speaker's emotional states within a multi-party conversation holds significant importance in the design of conversational agents that emulate human-like interactions. Throughout a conversation, a speaker's cognitive state frequently undergoes changes influenced by prior utterances, potentially resulting in shifts in their emotional state. Consequently, it becomes crucial to uncover the catalysts or triggers responsible for these emotional shifts during a conversation, as this understanding can provide explanations for the emotional labels assigned to individual utterances. In this report, we address the task of emotion recognition in conversations (ERC) and Emotion-Flip Reasoning (EFR). EFR is designed to identify the specific past utterances that have led to a speaker's emotional state changing at a particular moment in the conversation.

1 Introduction

A comprehensive examination of the Emotion Recognition in Conversations (ERC) task reveals several promising avenues for research that go beyond conventional emotion recognition. One such avenue pertains to understanding and explaining the emotional dynamics of speakers during a conversation. We all experience emotional shifts during conversations, typically due to two reasons: implicit (or external) and explicit (or internal). The implicit factor is tough because it's hard to identify what exactly caused the emotional change. For instance, an individual's emotions can change without any discernible verbal cues. In contrast, the explicit factor invariably involves identifiable triggers, such as visual or verbal signals from another speaker, which instigate emotional shifts in an individual. The definitions for the tasks of Emotion-Recognition in Conversations (ERC) and Emotion-Flip Reasoning (EFR) are as follows. Given a dialogue, ERC aims to assign an emotion to each utterance from a predefined set of possible emotions while EFR aims to identify the trigger utterance(s) for an emotion-flip in a multi-party conversation dialogue. While Emotion Recognition has been explored quite a lot EFR remains a quite unexplored field.

While Emotion Recognition (ER) has received substantial attention, Emotion-Flip Reasoning (EFR) remains relatively underexplored. In our literature review, we came across two notable references, (Kumar et al., 2021) and (Kumar et al., 2023), discussing EFR. Notably, cite1 aligns closely with our research task as it employs triggers to ascertain the reasons behind emotional shifts. On the other hand, (Kumar et al., 2023) focuses on identifying the instigator responsible for a speaker's emotional change within a conversation.

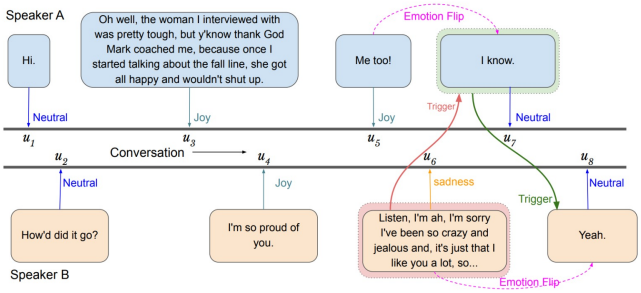

Figure 1: Among the five emotion flips, we present only two for clarity ($u_5 \rightarrow u_7$ and $u_6 \rightarrow u_8$), while the others are $u_1 \rightarrow u_3$, $u_2 \rightarrow u_4$, and $u_4 \rightarrow u_6$, with corresponding triggers being u_3, u_3, and u_6, respectively.

Structure The rest of the paper is organized as follows: Section 2 defines the task formally; Section 3 presents Related Works; Section 4 gives a detailed description and data analysis of the corpus; Section 5 describes the current work done and proposes future work. Finally, Sections 6, 7 and 8 contains the contributions of each member, Concluding remarks, and Presentation Feedback.

2 Project Github Link

https://github.com/ritam-cs779-t8/EDiReF
(Link to our GitHub repository. The Readme doc contains the training commands to run on Google Colab)

3 Problem Definition

Formally, we can define the Emotion-Flip Reasoning (EFR) task as follows: Consider a dialogue conversation D consisting of n utterances, with m distinct speakers denoted as $S = \{s_1, s_2, \ldots, s_m\}$. As the conversation progresses, speakers express their emotions or viewpoints in response to previous utterances. Each utterance u_{sji} in this dialogue is associated with an emotion E_{sji} and it's defined as $E_{sji} = f_E(u_{sji})$. Consequently, if the emotional tone of utterance u_{sji} changes compared to the previous utterance by speaker s_j, there could be a set of trigger-utterances u_k, where $1 \leq k \leq i$, responsible for this emotional shift. In other words, we can express this set of trigger-utterances as $[...u_k...] = f_T(u_{sji})$. In instances where there is no change in emotion, we label the current utterance as 'non-trigger'.

The EDiReF shared task at SemEval 2024 is an amalgamation of three subtasks tasks:

1. Emotion Recognition in Conversation (ERC) in Hindi-English code-mixed conversations,
2. Emotion Flip Reasoning (EFR) in Hindi-English code-mixed conversations
3. EFR in English conversations.

4 Related Work

Many recent studies, including some state-of-the-art methods, emphasize the utilization of multimodal data. In contrast, our research exclusively concentrates on text-based data. We will categorize the related literature into two subsections: ERC and EFR.

4.1 Emotion Recognition in Conversation

(Lei et al., 2023) propose a new approach called InstructERC. Instead of using a discriminative method, they use Large Language Models (LLMs) to create a generative framework. InstructERC offers two important contributions: enhances performance by incorporating multigranularity dialogue supervision and implicit models of dialogue roles and emotional tendencies. This method is the current SOTA on the MELD Dataset.

In their work, (Kumar et al., 2021) utilize a Masked Memory based Network (MMN). This choice is motivated by the recognition that certain utterances carry varying levels of importance within a conversation, with some being less significant and others having a lasting influence. Crucially, pivotal utterances are more likely to contribute to predicting emotions for multiple utterances in the dialogue.

(Ghosal et al., 2019) employ a Graph Convolutional Neural Network (GCN) to capture emotional dynamics within a conversation. Utilizing a GCN offers the advantage of modeling dependencies not only among speakers but also within individual speakers' utterances. Furthermore, it takes into consideration the anticipation of future emotions in the conversation.

4.2 Emotion Flip Recognition

In (Kumar et al., 2021), the authors explore trigger-based reasoning for emotional shifts in conversations and frame the emotion-flip reasoning task as an instance-based classification problem. On the other hand, (Kumar et al., 2023) focuses on Instigator-Based Emotion Flip Reasoning (EFR), aiming to identify the catalyst behind a speaker's emotional change within a conversation. They model this task as a multi-label instance classification problem, acknowledging that there can be multiple instigators for each trigger.

5 Corpus/Data Description

The MaSaC dataset is sourced from the TV show "Sarabhai vs Sarabhai" and comprises Hindi-English mixed text annotated with eight emotions: Disgust, Contempt, Anger, Neutral, Joy, Sadness, Fear, and Surprise for MaSaC datasets. It's worth noting that the dataset is not balanced, with the majority of the emotions being Neutral.
The MELD dataset is extracted from the "Friends". It has also similar emotions and triggers, but this data is purely in English not Hindi English code mix. Like the previous one it is also imbalanced with bias towards Neutral emotion.

5.1 Task 1: MaSaC ERC

Training data encompasses a total of 343 episodes and 8,506 sentences, and validation data contains total 46 episodes and 1354 utterances making it a

valuable resource for examining sentiment in multilingual content. Figure 2 the train data distribution for Task 1.

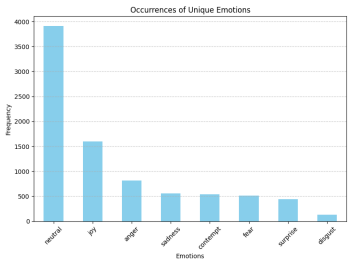

Figure 2: Emotion Distribution of Task 1 train dataset

5.2 Task 2: MaSaC EFR

In terms of size, it is substantial, comprising 4,893 instances and 98,777 sentences and validation data has a total of 389 episodes and 7,462 utterances. Notably, some episodes feature multiple triggers. Training data has 6,539 triggers and validation data has 431 triggers. Figure 3 shows the train data emotion distribution for Task 2.

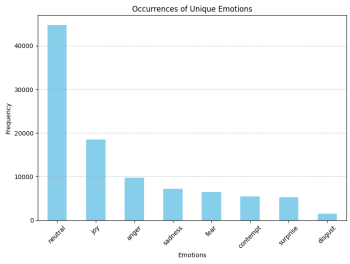

Figure 3: Emotion Distribution of Task 2 Training dataset

5.3 Task 3: MELD EFR

The training dataset encompasses a substantial 821 episodes, containing a total of 34,860 sentences and in validation dataset has 94 episodes and 3,512 utterances. The training data has a total of 5,516 triggers and the validation data has 489 triggers. Figure 4 shows the train data emotion distribution for Task 3.

6 Proposed Approach

We go through multiple approaches for both ERC and EFR tasks. A description of each approach is given below.

6.1 CNN-based model for ERC

Input is a batch of tokenized sentences(with padding) of max length 20 and Output is the probability of each class.

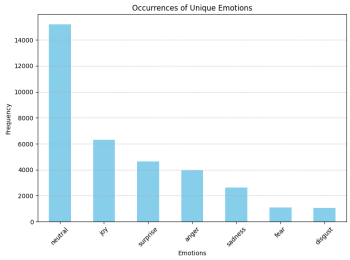

Figure 4: Emotion Distribution of Task 3 Training dataset

Computed the average embedding vector for the sentence's tokens and then fed it through two convolutional layers, followed by two fully connected layers. Max pooling and ReLU activations were applied between these layers.

We utilized the Cross-Entropy loss function and employed the SGD optimizer with an initial learning rate of 1.0. To dynamically adjust the learning rate throughout training, we incorporated a learning rate scheduler. The model underwent training for a total of 20 epochs.

6.2 ERC-MMN and EFR-TX

ERC-MMN and ERC-TX, developed by Kumar et al(Kumar et al., 2021) (2021), serve as the baseline models for Emotion Recognition (ERC) and Emotion Flip Reasoning (EFR), respectively. We evaluate the MaSaC and MELD datasets using both of these baseline models.

For MaSaC ERC and EFR tasks, we utilize HingBert (Nayak and Joshi, 2022), a pre-trained BERT model trained on the Code-Mixed Hindi-English Corpus, to generate word embeddings. Using these word embeddings, we generate sentence embeddings by averaging the individual word embeddings in a sentence.

For MELD EFR tasks, we directly obtain the sentence embeddings using a pretrained SBERT model.

Due to the dataset's inherent bias, we have observed that the performance of ERC-MMN falls below our expectations. During the training phase, the model's performance on the validation set exhibited significant fluctuations. We attempted to mitigate this issue by reducing the learning rate; however, regrettably, it did not lead to any improvements in the training results.

Additionally, we set the weights for the Cross-

Entropy Loss inversely proportional to the support of the dataset for each emotion.

6.2.1 Hypothesis and Data Augmentation

To mitigate data imbalance in EFR, which is biased toward '0' as depicted in Figure 5, we propose a hypothesis: triggers are more likely to occur within the utterances between the emotion flip of the target speaker S* as shown in Figure . We thus augment

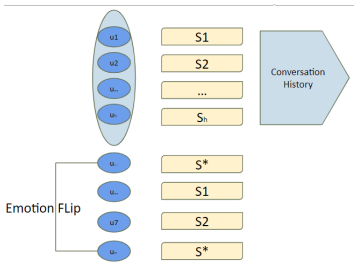

Figure 5: Hypothesis

our data into two sections: Conversation History and Probable Trigger Zone (PTZ). The table below shows the changed statistics focusing on the augmented data. We observed a 10 percent reduction in triggers after augmentation, confirming the validity of our hypothesis.

Dataset (EFR)	Triggers	Non-Triggers	Average length
MaSaC	6970	99252	19.68
MaSaC (Hypo)	6304	16284	4.25
MELD	6005	32357	8.5
MELD (Hypo)	5330	9735	3.32

Table 1: Overview of Dataset after Hypothesis

6.3 Contrastive Learning Based Approach

Previous methods encountered difficulties stemming from data imbalance within the label classes. To tackle this issue, we take inspiration from the approach outlined in the paper by Song et al. (2022)(Song et al., 2022) as shown in Figure 7. We explored various strategies for both ERC and EFR tasks.

6.3.1 Representation Extraction

SimCSE We built a prompt-based context encoder upon SimCSE(Gao et al., 2021) to get speaker and context-aware emotion representations. The architecture of the context encoder is illustrated in Figure 14. We feed the training data into SimCSE and then get the last hidden state and then we use the embedding of the special token <mask> as our representation.

HingBERT To get representations of the MaSaC Hinglish dataset we use the pre-trained HingBERT model to generate utterance representations corresponding to each emotion. This is also done by averaging the individual word embeddings in a sentence.

HingBERT with GRU Obtaining sentence representations by averaging the word embeddings did not yield satisfactory results. To address this issue, we developed a GRU-based representation, where we pass individual word embeddings through a GRU to generate sentence representations for each emotion.

6.3.2 Context Modeling

We propose a prompt for EFR akin to ERC. A

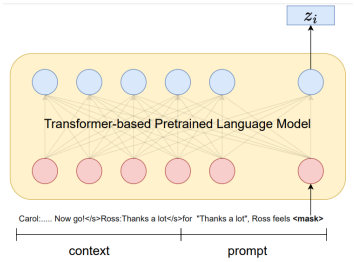

Figure 6: Context modeling in the SPCL approach

conversation is built from (s_t,u_t) which is speaker of t-th turn , utterance in the t-th turn.

Ct = [s_{t-k}, u_{t-k}, s_{t-k+1}, ..., s_t, u_t]

To calculate representation for u_t, we use the most recent k turns of utterances and speakers. We then constructed a prompt for the t-th turn as follows as shown in Figure 6.

Prompt for ERC

P_t = for u_t, s_t feels <mask>

Proposed Prompt for EFR

P_t = for u_t, s_t feels e_t is possibly <mask>

e_t being the emotion of speaker s_t

The full input of the encoder is C_t concatenated with P_t. In addition we created extra training points too, which have same C_t but different P_t which helps the model to pay more attention to the target sentence and generate reasonable representations.

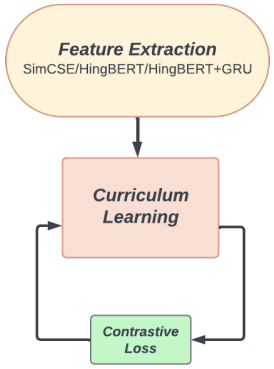

Figure 7: Contrastive Loss based approach

6.4 Prototypical Contrastive Learning

In our approach, we introduce a Supervised Prototypical Contrastive Learning (SPCL) loss function, which incorporates prototype vectors for each emotion category into the $L_{suploss}$. To implement this, we maintain a fixed-size representation queue for each emotion category. Specifically, a representation queue for the i-th emotion, denoted as Q_i and of size M, is represented as $Q_i = [z_i^1, z_i^2, ..., z_i^M]$.

Here's how the process works:

1. When a new representation z_i for the i-th emotion is generated, we first remove the oldest element from Q_i if $|Q_i|$ reaches its maximum capacity (M). We then detach the gradient of z_i and add it to Q_i.

2. Second, to compute the prototype vector for the i-th category, we randomly select K samples from Q_i to form the support set S_K. The prototype vector T_i is calculated as the mean of the support set S_K.

This approach allows us to integrate prototype vectors into the loss function, enhancing the performance of the model in emotion categorization.

After obtaining the prototype vectors, we treat each of them as an example of the corresponding category, so the sum of negative scores of z_i can be calculated as follows

$$\mathrm{N}_{SPCL}(i) = N_{SUP}(i) + \sum_{k \in E \setminus y_i} F(z_i, T_k)$$

$$\mathrm{P}_{SPCL}(i) = P_{SUP}(i) + F(z_i, T_{y_i})$$

$$\mathrm{L}_{SPCL}(i) = -\log\left(\frac{1}{P(i)+1} \cdot \frac{P_{SPCL}(i)}{N_{SPCL}(i)}\right)$$

In summary, by introducing the prototype vectors, the SPCL loss ensures that there are at least one positive pair and |E|-1 negative pairs for each sample in a batch. In addition, random prompts and labels were also input into the encoder with the same context.

6.5 Curriculum Learning

When building a text-only model, some utterances are not informative enough to judge the labels. Training the model with these extreme samples will lead to performance degradation. We use curriculum learning to alleviate this issue. Finding the center of each label class as the average of the z_i of the that class. Finding difficulty of each data point by finding distance of z_i from it's label class's center. After sorting the entire training set, instead of directly splitting the training set, we design a sampling-based approach to construct a series of subsets ranging from easy to hard.

7 Experiments and Results

We evaluate our models based on both accuracy and F1-scores. F1-scores are considered more reliable, particularly in the presence of inherent data bias.

Model	F1 Score	Accuracy	Precision	Recall	F1 Macro Average
SPCL Sim-CSE	0.45	0.47	0.45	0.47	0.33
ERC-MMN-Hinglish	0.33	0.46	0.32	0.46	0.11
SPCL-Hing-BERT	0.29	0.42	0.23	0.42	0.09
SPCL-Hing-BERT-GRU	0.30	0.47	0.22	0.47	0.08

Table 2: Results for ERC

All the hyperparameters,experimental conditions are given on the github page.Github

8 Error Analysis

We provide a comprehensive analysis of all experiments conducted on our github page. Github

We encountered challenges, particularly in the generation of sentence representations using Hing-BERT, as averaging all word embeddings to create

Model	F1 Score	Accuracy	Precision	Recall	F1 Macro Average
EFR-TX MASAC	0.36	0.58	0.69	0.59	0.52
EFR-TX MASAC Hypo	0.41	0.63	0.64	0.63	0.57

Table 3: Results for EFR MaSaC

Model	F1 Score	Accuracy	Precision	Recall	F1 Macro Average
EFR-TX MELD	0.43	0.56	0.68	0.56	0.53
EFR-TX MELD Hypo	0.49	0.51	0.58	0.51	0.51
SPCL MELD	0.80	0.86	0.79	0.82	0.56

Table 4: Results for EFR MELD

sentence embeddings did not yield satisfactory results. In an attempt to address this issue, we experimented with generating sentence representations using a GRU model on individual word embeddings. However, this approach only resulted in minor improvements.

9 Future Directions

1. We have the flexibility to apply SPCL to EFR by leveraging our hypothesis and tailoring the prompts accordingly.

2. Exploring a graph-based network approach was considered due to the inherent data characteristics. However, during experimentation, memory-related issues were encountered.

3. Considering that we implement prompting in the SPCL model, it's worth exploring the use of large language models as an alternative to SimCSE.

4. Given that we trained the models from the ground up, transfer learning is also a viable option to consider.

5. The use of ensembling is a potential strategy since we employed multiple models.

10 Individual Contributions

1. Sachin: Literature Review, CNN based Model, SPCL, Hypothesis Model

2. Yashwant: Literature Review, CNN based Model, SPCL, Hypothesis Model

3. Ritam: Literature Review, Data Analysis, Data preprocessing, Benchmarking on ERC MMN and EFR TX

11 Conclusion

In conclusion we conducted a comprehensive analysis of the dataset. We developed a naive CNN-based model without taking conversation history into account We then benchmarked MaSaC ERC data on ERC-MMN and MaSaC EFR ,MELD EFR on EFR-TX model,we used a pre trained BERT model trained on Code-Mixed Hindi-English Corpus, to generate word embeddings for MaSaC ERC and MaSac EFR in the above models.We then worked on the hypothesis that triggers are more likely to occur within the utterances between the emotion flip of the target speaker S^* and then used this dataset on EFR-TX model. We then introduced a model that can work on imbalanced dataset ,namely SPCL.We then did variations on SPCL to make it work on Code-Mixed Hindi-English by introducing a pre trained BERT model and a gru to convert word embeddings into sentence embeddings. We then proposed a solution to make SPCL work for the EFR problem.

References

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Deepanway Ghosal, Navonil Majumder, Soujanya Poria, Niyati Chhaya, and Alexander Gelbukh. 2019. Dialoguegcn: A graph convolutional neural network for emotion recognition in conversation.

Shivani Kumar, Shubham Dudeja, Md Shad Akhtar, and Tanmoy Chakraborty. 2023. Emotion flip reasoning in multiparty conversations. *IEEE Transactions on Artificial Intelligence*, pages 1–10.

Shivani Kumar, Anubhav Shrimal, Md Shad Akhtar, and Tanmoy Chakraborty. 2021. Discovering emotion and reasoning its flip in multi-party conversations using masked memory network and transformer.

Shanglin Lei, Guanting Dong, Xiaoping Wang, Keheng Wang, and Sirui Wang. 2023. Instructerc: Reforming emotion recognition in conversation with a retrieval multi-task llms framework.

Ravindra Nayak and Raviraj Joshi. 2022. L3cube-hingcorpus and hingbert: A code mixed hindi-english dataset and bert language models.

Xiaohui Song, Longtao Huang, Hui Xue, and Songlin Hu. 2022. Supervised prototypical contrastive learning for emotion recognition in conversation.