NLP Project Report: Textual Emotion-Cause Pair Extraction in Conversations

Group 24 **Manav Mittal** 2021538 **Utkarsh Venaik** 2021570 **Akash Kushwaha** 2021514 **Lakshay Kumar** 2021061

Abstract

In the dynamic field of computational linguistics, understanding emotional triggers in conversations is crucial for developing empathetic AI systems. This paper introduces a novel model designed to identify the causes of emotions within conversational contexts. Leveraging interdisciplinary approaches from psychology and advanced natural language processing (NLP), our model integrates attention mechanisms, transformers, and deep learning-based convolutional neural networks (CNNs). It uniquely incorporates the Myers-Briggs Type Indicator (MBTI) to analyze speakers' personality traits and behavioral patterns. This comprehensive approach allows for precise predictions of emotional triggers, answering the fundamental question: "Why do we feel a certain way during conversations?" Our findings have significant implications for understanding and provides a approach to incorporate emotional intelligence in modern chatbots and AI-driven communication systems.

1 Introduction

Incorporating the emotional dynamics of humans alongside modern methods of computational linguistics and natural language processing is a method which promises significant advancements in how machines understand and interact with humans. Our project addresses this challenge by developing a model capable of pinpointing the exact causes of emotions during conversations. Whether emotions arise from specific statements or are influenced by inherent personal feelings, our model aims to accurately predict these causes and triggers. Our model integrates cutting-edge NLP techniques like Transformers, attention mechanisms and Convolution Deep Neural networks (CNNs) alongside psychological insights—specifically, the Myers-Briggs Type Indicator (MBTI)—our approach predicts the emotions, emotional causes and triggers based on the conversation. Please refer to background and problem statement sections below for examples of what the model is predicting. Please refer to the problem statement section below for example.

2 Background

Emotion Recognition in Conversation (ERC) aims to identify the emotion of each utterance in the dialogue. This task has been popularly explored in the NLP research community, which has wide applications in building automatic conversational agents and mining user opinions.
In Cause Pair extraction we try to find the cause utterance to emotion it triggers in the conversation. We also try to find the span of the causal utterance which triggers the emotion in this task.

3 Problem Statement

We have two tasks at out hand. First is to identify the emotion of a particular utterance in the conversation. Second, we have to find the cause of current emotion of the speaker in the conversation, we have to do this for each utterance in the conversation. There can be multiple causes. Consider the following example. We have to also identify the exact span in the causal utterance which caused the current emotion of the current speaker.
Following is the input and the expected output of the model:
Input: a conversation containing the speaker and the text of each utterance.
Output: all emotion-cause pairs, where each pair contains an emotion utterance along with its emotion category and the textual cause span in a

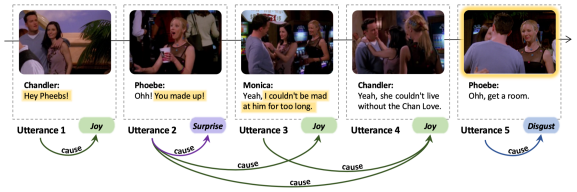

Figure 1: An example of our task and annotated dataset. Each arc points from the cause utterance to the emotion it triggers. The cause spans have been highlighted in yellow. Background: Chandler and his girlfriend Monica walked into the casino (they had a quarrel earlier but made up soon), and then started a conversation with Phoebe.

Figure 1: Example of Data Sample

specific cause utterance, e.g., (U3_Joy, U2_"You made up!").

4 Related Work

4.1 Emotion Recognition in Conversations

(Akhtar et al. 2021) in their paper have leveraged speaker specific GRU, taking all utterances of a speaker in a conversation, training such GRU for all speakers of conversation, combining it to contextual embedding of the conversation. Thus having speaker specific embeddings and overall contextual embedding which are further leveraged in architecture.

(Xiangyu et al. 2023) have taken BERT and fine-tuned it using "Suggestive Text" which serves as the indication of the contexts; that is preprocessing utterances to; Speaker_i says via emotion_j : utterance_k. This helps giving emotional context and speaker context to utterance, giving contextual embedding accordingly. They have masked the emotion of the utterance for which they wanted to predict the emotion and used BERT to predict the emotion.

4.2 Cause pair extraction

(Akhtar et al. 2021) in their paper have followed the effective utilization of the memory network for emotion recognition, adapted it to leverage the emotional relationship among several interlocutors. they employ to supplement the global and local emotional dynamics in dialogues captured through a series of recurrent layers. Moreover, handled multiple speakers instead of dyadic conversations. Their study advances on the path of explain-ability by mining the reason behind an emotion-flip of a speaker.

5 Dataset Description and EDA

We are using dataset provided by SemEval-2024 Task 3. It contains 1374 conversations. Maximum number of distinct speakers in a conversation are 9. There are a total of 7 emotions neutral, surprise, anger, sadness joy, disgust and fear. There 312 different speakers in whole conversational dataset.

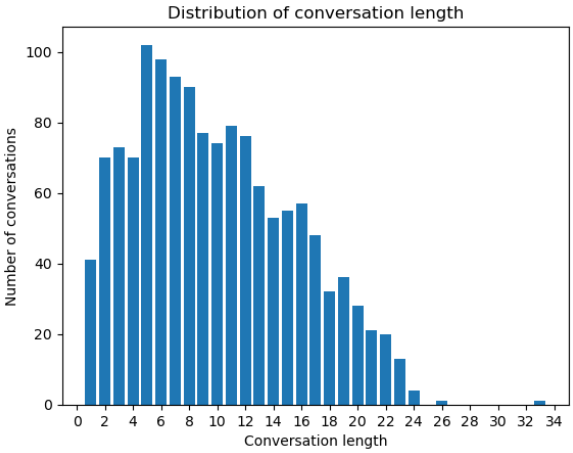

Figure 2: This is distribution of the number of conversations v/s number of utterances in the conversation

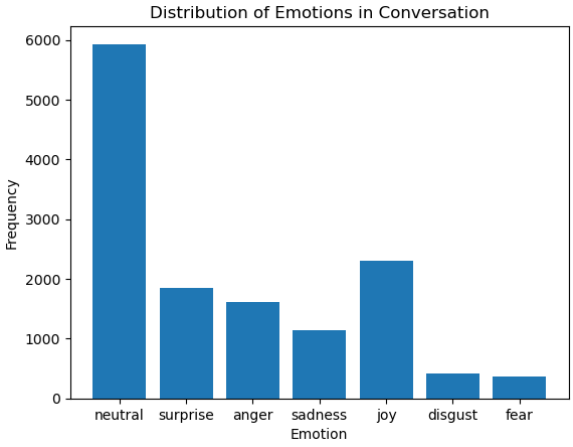

Figure 3: The distribution of number of utterances v/s emotion in the dataset

We wanted to decide the context window for the conversation in our dataset. This means that we wanted to analyse what is the span of a conversation which can effect the emotion of the current utterance. As we can clearly see that in Figure 2 there are many samples in the dataset with conversation length greater than 17 but the span which is affecting the emotion of the current utterance is only before and eight after making the total of 17 as can be seen in Figure 4.

Speakers personality is an important aspect in determining the emotion of the utterance they have

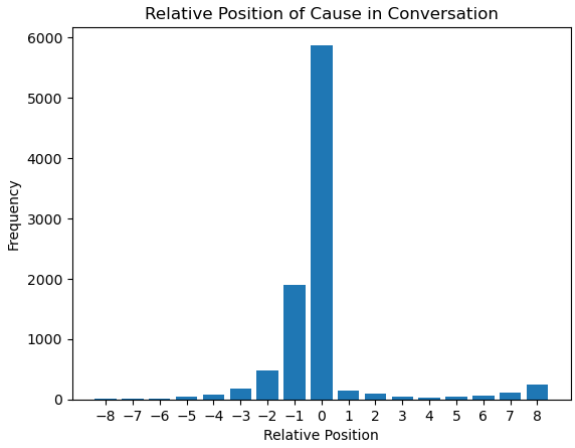

Figure 4: The distribution for the position of the cause with respect to the current utterance

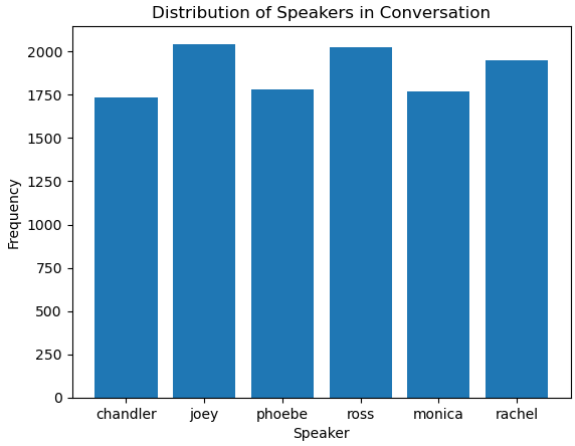

Figure 5: Distribution of number of utterance spoken by the most prominent speakers in the dataset

spoken. For example in case of 'Chandler', he usually make fun of others, the sentences he says can be joyfull but the emotion it causes to the listener can be sad. Since we have single modality in our model we try to incorporate speakers personality in the input to the model. As mentioned above we have 312 distinct speakers in out dataset there are only five speaker for which we have sufficient number of utterances to determine their personality which are shown in figure 5.

6 Experimental Setup

Our experimental setup includes Data preprocessing in which we have made data samples from the original dataset since we have to use context window in the conversation. By experimentation and EDA we concluded that context window of [-2, 0] is appropriate to train our model upon. Other than this we had one RTX 3050 and RTX 4050 to train our models upon. We have used PyTorch to program our models. We have splitted our dataset in 85:15 ratio. 85 being train and 15 being validation.

7 Proposed Methodolgy

7.1 ERC - Emotion Recognition in Conversation

We want to get good results on the conversational data. Predicting the emotion of an utterance in a conversation can be more difficult than predicting the emotion of a statement or just an English sentence since emotion in the current utterance can be very dependent upon the previous utterances. Moreover, the speaker's information is important for the same reasons.

So, to preserve these aspects of the conversation, we have converted our input into a 3- dimensional input matrix. Where 1 axis preserves the speaker information, 1 axis preserves the information of the timestamps in the conversation, and 1 axis contains the input embeddings.

We have used convolution to capture the speaker's essence and the timestamp essence in the conversation . We have treated different speakers as channels. We have considered the maximum number of speakers to be 10 (8 in the training set and 7 in the validation set). Since LSTM takes single-channel input, we have reduced the channels from 10 to 1. We want CNN only to capture speaker and timestamp data, so we have preserved the utterance embedding dimension, i.e., 768. Language understanding task we have left to LSTM.

The current dataset has enough data samples, but rather than initiating the training on this dataset with random parameters, we decided to use a similar but larger dataset, which is publicly available, to train our model to initiate the training of the current dataset with more optimal parameters rather than random, we trained our model on the MELD dataset and saved the final checkpoint. Subsequently, we fine-tuned the model's parameters on the current dataset. Finally, it was evaluated on the fine-tuned model on the current dataset.

7.2 CPI - Cause Pair Identification

Our model has four parts to perform this task. Personality Encoder, Utterance Encoder, BERT Conversation Encoder, Classification Layer. Each model will give some embeddings, their relevance

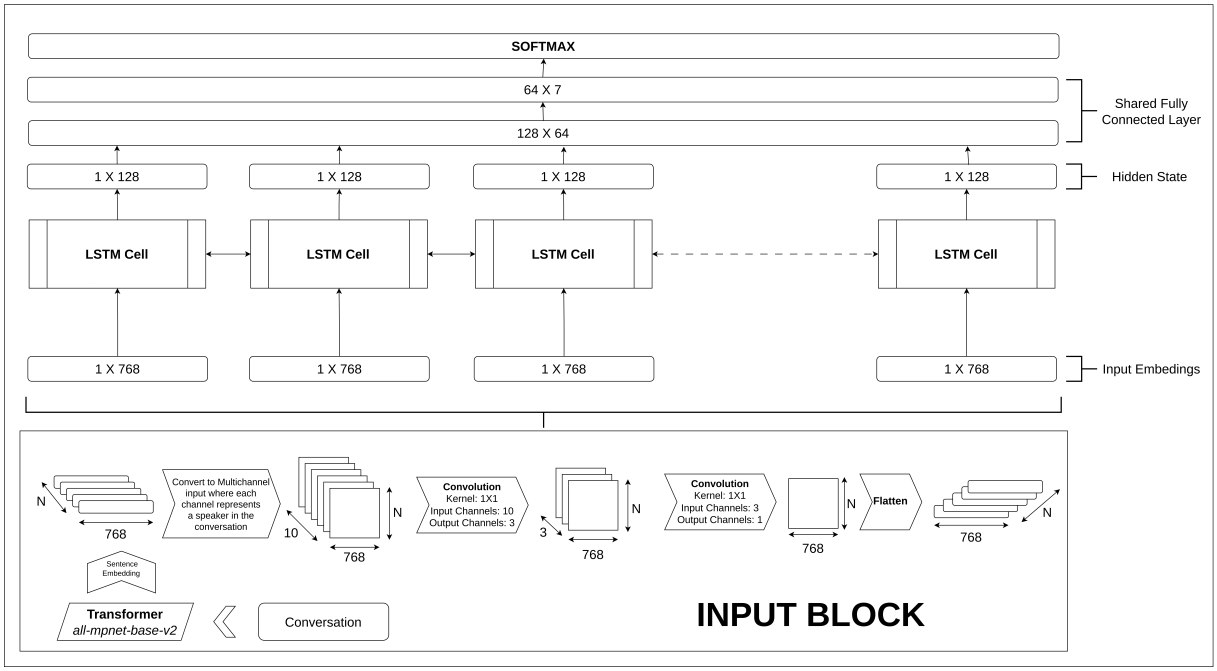

Figure 6: ERC - Model Arhitecture

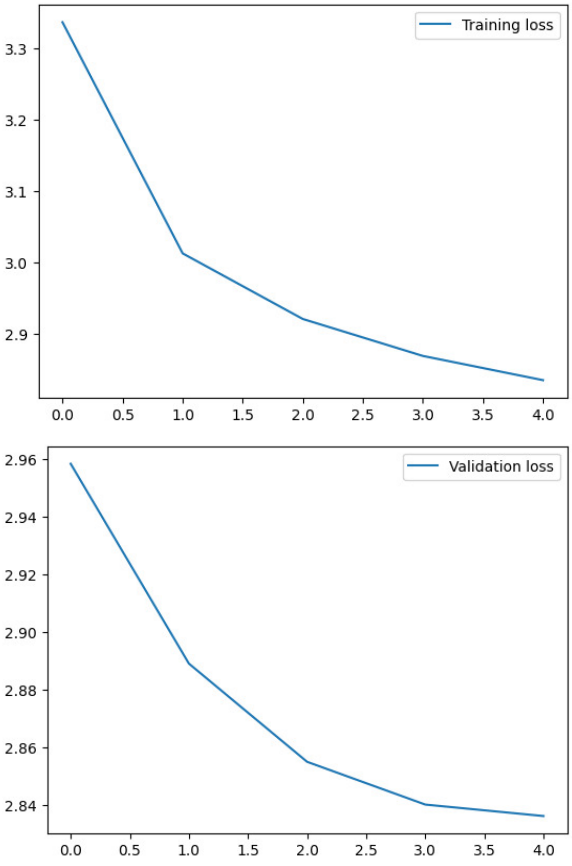

Figure 7: Loss v/s Epoch curve for ERC Model

will discusssed later in the text. These embeddings are concatenated and then are given input to an MLP to identify the causing span of the emotion in the conversation.

7.2.1 Personality Encoder

In the case of conversations, the personality of the speakers has a significant impact on the context and the emotion of the conversations. For example, Joey and Chandler will have funny conversations, while some of Monica and Chandler's conversations will also be romantic. So, the speaker's personality and the relationship between them also influence the emotion of the conversation. Moreover, we are working on a single modality dataset to get all the information about the conversation just from the text, which is difficult. To tackle this issue to some extent, we introduce personality embeddings.

We have used pre-trained MBTI transformaer which give 16 dimensional personality scores these 16 score are treated as a 16 dimensional vector by us to give to the model as input as personality embedding.

7.2.2 Utterance Encoder

This is the part where "Sentence Transformer" comes into the picture. We have encoded every utterance in a conversation. Out of these utterance

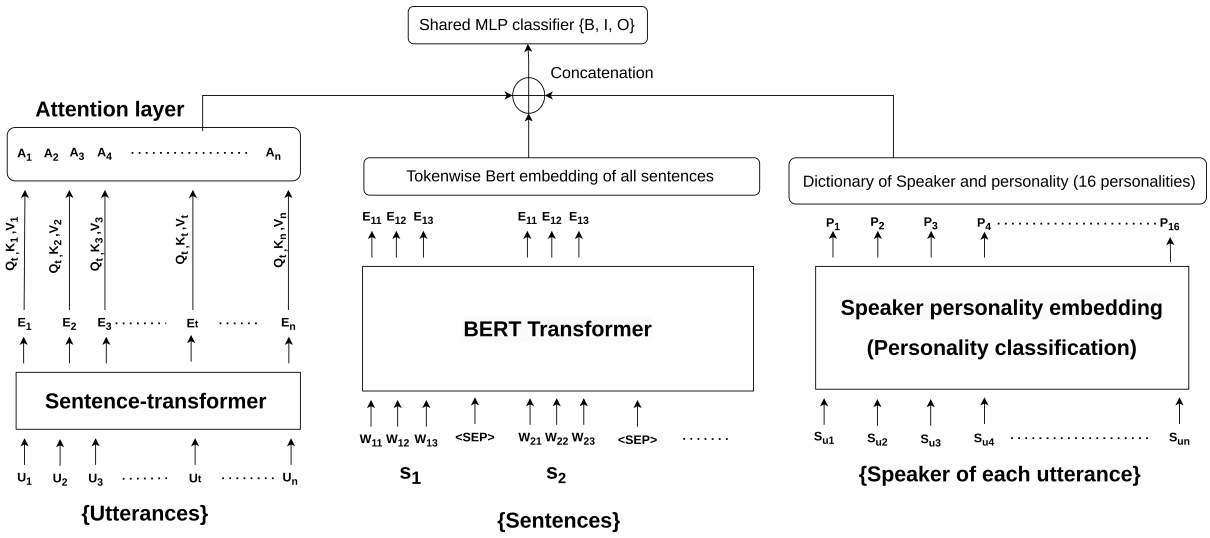

Figure 8: CPI - Model Architecture

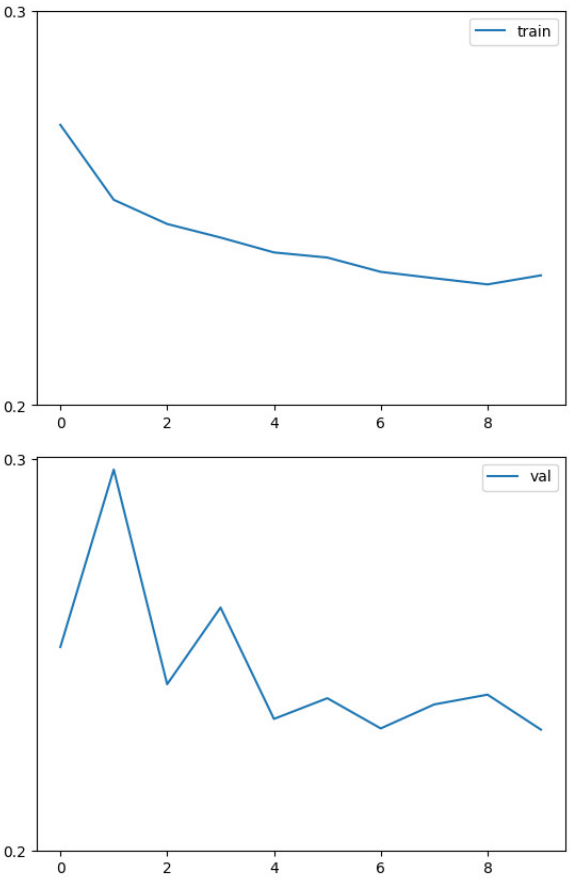

Figure 9: Loss v/s Epoch curve for CPI Model

we have to find the causal span wrt to a particular utterance. To achieve this component of the task we use attention mechanism. The utterance for which we want to find the causal span in the other utterances is used as Query for Key, Value pairs of all the utterances. This will help us in finding relevant information in all the utterances of the conversation wrt the utterance for which we want to find the causal span. After applying attention over the sentence embeddings we get attended embeddings.

7.2.3 BERT Conversation Encoder

Since we have to do token level classification, that whether a token is part of the causal span or not. Every utterance of the conversation is appended in the front with ”SPEAKER EMOTION said:” which make the final utterance to look like ”SPEAKER EMOTION said: UTTERANCE” we did so, so that BERT get some context about who the speaker is and what is their emotion at that moment. Every utterance in the converation string is seperated by [SEP] token. Then the new formatted string is given as input to BERT. BERT then provide contextual token embeddings for each word which will have the context of the whole conversation.

7.2.4 Classification Layer

For every token we get a BERT embedding this embedding is concatenated with the respective speaker personality embedding and utterance embedding we got form Personality Encoder and Utterance Encoder respectively. This concatenated embedding is given input to the the fully connected layer which classifies every token as B, I, or O. Hence giving us the causal span.

8 Results and Evaluation

8.1 Ablation Study

8.1.1 ERC - Emotion Recognition in Conversation

We tried ablating on RNN family(BiLSTM vs GRU), resulting in better

Model	Accuracy	F1 Score	Precision	Recall
CNN(k=1,1)+MLP	0.543	0.550	0.48	0.45
CNN(k=1,1)+GRU+MLP	0.577	0.554	0.56	0.42
CNN(k=1,1)+Bi-LSTM(3 layers)+MLP	0.531	0.477	0.34	0.33
CNN(k=1,1)+Bi-LSTM+MLP	**0.618**	**0.603**	**0.58**	**0.49**

Figure 10: Metrics - ERC

Model	Surprise	Neutral	Fear	Anger	Disgust	Joy	Sadness
CNN(k=1,1)+MLP	**0.437**	0.714	0.337	0.217	0.268	0.417	0.546
CNN(k=1,1)+GRU+MLP	0.400	0.737	0.151	0.250	0.158	0.476	0.578
CNN(k=1,1)+Bi-LSTM(3 layers)+MLP	0.127	0.755	0.000	0.000	0.312	0.000	**0.610**
CNN(k=1,1)+Bi-LSTM+MLP	0.319	**0.769**	**0.395**	**0.418**	**0.433**	**0.568**	0.575

Figure 11: Emotion-wise F1 Scores - ERC

8.1.2 CPI - Cause Pair Identification

We performed ablation study to find the importance of personality embedding in out model. And found out that personality embeddings are contributing significantly to the performance of the model, these metrics are shown in figure 12.

Model	B-Label F1	I-Label F1	O-Label F1	Overall F1	Precision	Recall
CPI Model without Personality Embedding	0.3064	**0.5101**	0.8871	0.8561	0.8671	0.8812
CPI Model with Personality Embedding	**0.3321**	0.5072	**0.9469**	**0.8963**	**0.8935**	**0.9021**

Figure 12: CPI - Metrics

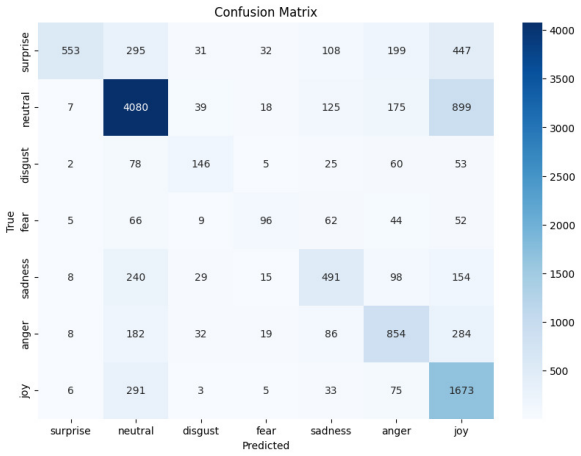

Figure 13: Train Confusion Matrix - ERC

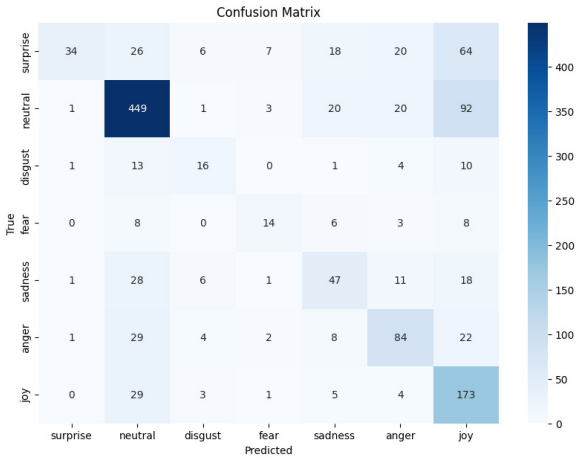

Figure 14: Validation Confusion Matrix- ERC

8.2 Confusion Matrix

Confusion matrices on the train and validation set for both the models ERC and CPI are shown in figures 13-14 and 15-16 respectively.

8.2.1 ERC - Emotion Recognition in Conversation

From the confusion matrix of train and validation dataset we can observe that model is performing well for every emotion and it is even able to learn to distinguish similar emotions fairly accurately.

8.2.2 CPI - Cause Pair Identification

In fig. 13-14, 0 denotes ”O”, 1 denotes ”B” and 2 denotes ”I”. From the confusion matrix we can see that model is predicting a lot of tokens as ”O” incorrectly. Since our dataset has a lot of tokens labelled as ”O” and the model may tending towards minimizing loss for ”O” and not learning much for ”B” and ”I”. To remedy this we reduced the size of context window so that ratio of ”B”, ”I” and ”O” tokens improve. We agree that it can affect the overall accuracy but due to resource constraints we have restrained ourselves to this choice.

8.3 Human Evaluation

From the Evaluation metrics F1 score of ERC, we can observe that our model can best classify neutral emotion, then fear, and the worst classification of surprise. For the Human Classification, we asked about the emotion of an utterance given the whole conversation. And asked to rate emotion existing in utterance for each of 7 emotions.

Conversation 1: Utterance 2 has True Emotion of Neutral
Kim: ”So , we are decided, no on plaid, yes on pink ?”,
Nancy: ”Absolutely !”,
Rachel: ”I am so on board !”,

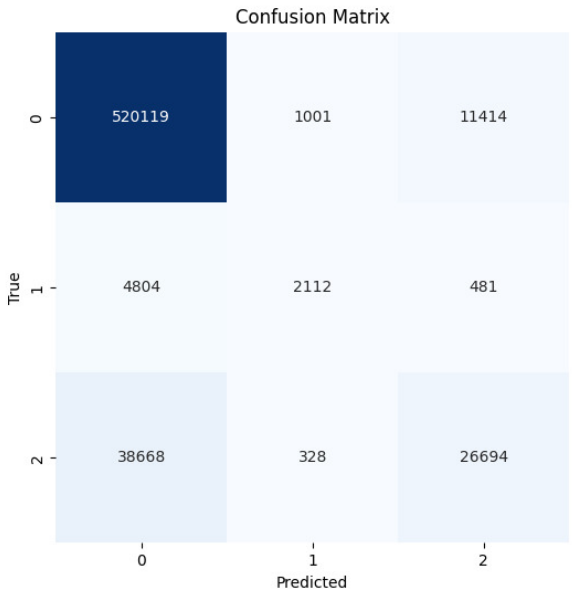

Figure 15: Train Confusion Matrix - CPI

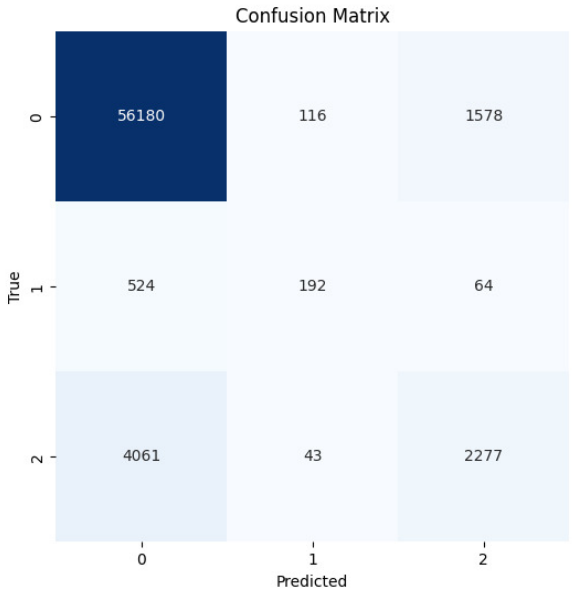

Figure 16: Validation Confusion Matrix- CPI

Kim: ”Rachel , did not you just light that ?”,
...

Conversation 2: Utterance 8 has True Emotion of Fear
Phoebe: ”So ... so you two were married huh ? What happened ? You just drift apart ?”,
Mrs. Geller: ”Here comes the bride .”,
...
Rachel: ”Hello ?”,
Joey: ”Hey ! Did Chandler show up yet ?”,
Rachel: ”Yeah , we got him back . Everything fine .”,
Joey: ”Damnit !”,

Emotion	$Conv_1$	$Conv_2$	$Conv_3$
Surprise	5.2	4.3	**9.2**
Neutral	**8.2**	**7.4**	1.2
Disgust	4.6	5.4	5.9
Fear	3.5	0.8	2.4
Sadness	0.7	1.7	4.8
Anger	0.9	4.3	8.5
Joy	7.9	4.8	0.4
Predicted	Neutral	Fear	Anger
True	Neutral	Fear	Surprise

Table 1: Human Eval.: Average Emotion Rating

Conversation 3: Utterance 6 True Emotion: Surprise
Dr. Green: ”So ? Come on ! Explain yourself Geller ! First you get my Rachel pregnant !”
Mona: ”You got Rachel pregnant ? !”
Ross: ”Who did ? !”
Dr. Green: ”You did !”
Ross: ”Yes . Yes , yes I did . But ... but it was , it was just a one night thing . It meant nothing .”
Dr. Green: ”Oh ? Really ? That is what my daughter means to you ? Nothing ?”
Ross: ”No ! No sir umm , she means a lot to me . I mean , I care ... I ... I love Rachel .”
...

8.4 Custom metric

We observed during calculating evaluation metrics that our weighted f_1 scores are coming way different than the macro f_1 scores hemce we used a metric as described below which we think may provide a more holistic assessment of the model **Macro-average F1 score** calculates the F1 score independently for each class and then takes the average. This treats all classes equally. It is particularly useful when you want to assess the model's performance on minority classes while **Weighted-F1 score** takes the F1 scores of each class and averages them, weighting them by the support of each class. This means classes with more instances have a greater impact on the overall metric. This is useful when class prevalence is reflective of real-world scenarios, and you want the metric to reflect the performance on more frequently occurring classes. By combining both these mea-

sures, you capture a balanced view of the model's performance. The macro-average ensures that the model's performance on smaller classes influences the overall assessment, while the weighted-average reflects performance on classes that are more prevalent. Using both helps in situations where one wants to ensure good predictive performance across all classes while still considering their relative importance or frequency. In contexts like fraud detection, disease screening, or any scenario where some classes are naturally rare but critically important, using this combined metric can encourage developing models that not only perform well overall but also on crucial minority classes. However, this is just our personal point of view and in the final results we are treating the normal F1's as the metric to report.

Task	Weighted F1	Macro F1	Custom F1
Emotion Recognition in Conversation	0.603	0.496	0.552
Cause Pair Identifcation	**0.896**	**0.595**	**0.76**

Figure 17: Task-wise scores on validation set

$$f_1' = \sqrt{\frac{(macro \quad f_1)^2 \quad + \quad (weighted \quad f_1)^2}{2}}$$

Thus, we wanted to capture the correctness of our model on all emotions without being too much influenced by class imbalance. We devised a metric named $Imbalance - Proof \quad f$ or f' This is centered between macro and weighted f1 score and capturing both.

9 Discussion and Future Work

We have trained the Shared MLP classifier and the attention layer on the Sentence-Transformer in the CPI model. However, the BERT Transformer and Sentence-Transformer were not fine-tuned. This can be trained on the given dataset and get fine-tuned word-wise embedding and sentence-wise embedding from the BERT Transformer and Sentence-Transformer, respectively.
Other than this we can have more data to train out model upon since the provided data had only around 1300 samples. So it is our hypothesis that increasing the number of training samples might improve the performance of the model.
Moreover, multi modality can be considered for training the model since acoustic and visual expressions can help a lot in identifying the emotions of the utterances in the conversation.

References

[1] Shivani Kumar, Anubhav Shrimal, Md Shad Akhtar, Tanmoy Chakraborty. 2021. *Discovering Emotion and Reasoning its Flip in Multi-Party Conversations using Masked Memory Network and Transformer*

[2] Xiangyu Qin1, Zhiyu Wu1, Jinshi Cui1, Tingting Zhang 2021. *BERT-ERC: Fine-tuning BERT is Enough for Emotion Recognition in Conversation*

[3] Arefa1, Mohammed Abbas Ansari 2024. *Two-step approach for multimodal ECAC using in-context learning with GPT and instruction-tuned Llama mod*

[4] Ashish Vaswani 2024. *Attention Is All You Need*