Emotion Discovery and Reasoning its Flip in Conversation (EDiReF) - SemEval 2024 – Task 10 - Task C – EFR in English conversation
NLP Course Project

Giorgia Castelli, Alice Fratini, Madalina Ionela Mone and **Francesco Pigliapoco**
Master's Degree in Artificial Intelligence, University of Bologna
{ giorgia.castelli2, alice.fratini2, madalina.mone, francesco.pigliapoco }@studio.unibo.it

Abstract

This study presents an analysis of the resolution of the "Emotion Recognition in Conversations" (ERC) and the "Emotion Flip Reasoning" (EFR) tasks, with a focus solely on Task 10, subtask C which is restricted to English dialogues only, utilizing two BERT-based models. These tasks are of particular significance within the field of Natural Language Processing (NLP) as they aim to enhance emotional intelligence and context-aware analysis in conversational AI systems. The models are designed as dual-headed classifiers and are trained on various dialogue datasets to evaluate how different configurations impact their ability to interpret emotions conveyed in sentences and detect emotional shifts within conversations. Additionally, various tests are conducted to assess the generalization ability of these models to new and previously unseen conversations. This analysis aims to identify the limitations of the method and provide a comparison with two baselines models: the majority and the random classifier.

1 Introduction

The Emotion Recognition in Conversation (ERC) task is a fundamental NLP challenge focused on identifying the specific emotion for each sentence within a dialogue in order to understand the speaker's emotional state at different points. In contrast, the Emotion Flip Reasoning (EFR) task aims to understand which utterances are crucial behind a speaker's emotions flip within a conversation. In this case the model will reason about how and why these changes occur during time based on the conversation's context. The problem described in this report aims to identify both emotion, from an existing and limited set of emotions, and trigger for each sentence. Further details are described in subtask 3 of SemEval 2024 Task 10: Emotion Discovery and Reasoning its Flip in Conversation (SemEval, 2024).

Regarding ERC, various approaches have been proposed.

A first approach involves using models based on RNN and LSTM, such as "DialogueRNN: An Attentive RNN for Emotion Detection in Conversations" (Majumder et al., 2019), a system characterized by recurrent neural networks with an attention mechanism to track emotional dynamics. In this model, the RNNs are separated for each participant, and their emotional state is tracked through three main components: the global state (the overall context of the conversation), the party state (the specific emotional state of each speaker within that global context), and the emotional state (the current emotional condition).

Another approach leverages Graph Neural Networks (GNN), as seen in (Ghosal et al., 2019), to model relationships between participants. In this model, nodes represent the speakers and edges represent their interactions, enabling the system to capture complex emotional dynamics within the dialogue.

For the EFR task, an effective solution is described in SemEval 2024 – Task 10: Emotion Discovery and Reasoning its Flip in Conversation (EDiReF) (Kumar et al.). This approach uses a Transformer-based model that integrates the content of utterances with the identity of the speaker, making it speaker-centric. The utterance embeddings, generated using BERT, are combined with the emotion and the speaker's identity — both encoded as one-hot vectors — and then processed by the Transformer to account for the conversation's context. In parallel, an Emotion-GRU network computes the emotional history of the conversation. Finally, the model uses these contextualized representations to predict the final emotion, providing an accurate analysis that takes into account not only what is said, but also who is speaking and how emotions evolve within the dialogue.

The models implemented in our approach are

Transformer-based, as they leverage a pre-trained BERT model. Specifically, two versions were implemented: one with frozen BERT weights (Freezed) and one with unfrozen weights (Unfreezed). Additionally, two baseline models were used for comparison and evaluation.

Finally, two further experiments have been developed in order to enhance the performances on minority labels: the first involved data augmentation combined with class weighting, while the second integrated both data augmentation and enhanced context by providing full discourses instead of single sentences.

2 System description

The core architecture of our system is centered around the BERT Transformer, specifically the `bert-base-uncased` model, which was downloaded from Hugging Face. This model consists of 12 layers, each with 768 hidden units, and a total of 110 million parameters. It is renowned for its bidirectional context understanding, analyzing each word in relation to all others within a sentence, rather than in a strictly sequential manner. Pretrained on the English language using a masked language modeling (MLM) objective, the bert-base-uncased model treats words as case-insensitive, meaning it does not distinguish between "english" and "English." This model is designed to be fine-tuned for tasks that require whole sentence analysis — such as sequence classification, token classification, or question answering. It was pretrained on two major corpora: BookCorpus, which contains 11,038 unpublished books, and the English Wikipedia (excluding lists, tables, and headers). The texts were lowercased and tokenized using the WordPiece method, with a vocabulary size of 30,000. The model training was performed on 4 cloud TPUs in a Pod configuration (16 TPU chips total) for one million steps with a batch size of 256, allowing for efficient learning of complex language patterns. In this architecture, BERT functions as the central feature extractor, generating contextual embeddings from the input data, which are crucial for capturing the relationships between words.

Our architecture consists of the following components:

- A pre-trained BERT layer, which generates contextualized representations from input sequences and produces high-dimensional embeddings. This layer serves as the input layer, processing tokenized text where each token is represented by its corresponding input_ids, attention_mask, and token_type_ids.

- An emotion prediction branch, which follows the BERT layer. This branch transforms the embeddings into predictions for emotion classification at the utterance level. It comprises two key components:

  1. A dropout layer with a dropout rate of 0.3, randomly deactivating 30% of the neurons during training to improve the model's ability to generalize to new data and prevent overfitting.
  2. A linear layer responsible for producing a 7-dimensional output vector. The BERT output, with a dimensionality of 768, is processed by this linear layer, resulting in a 7-element array representing the predicted emotions. Of these 7 elements, only one is active, corresponding to the predicted emotion for that specific utterance. This is achieved by applying a OneHotEncoder, which ensures that the correct emotion is encoded as the active element.

- A trigger prediction branch, which mirrors the structure of the emotion prediction branch. It includes a dropout layer with the same 0.3 rate and a linear layer. However, in this case, the output is a 2-dimensional vector indicating the presence or absence of conversational trigger. A OneHotEncoder is applied to ensure that the correct trigger is accurately predicted.

3 Data

The dataset used in this study (SemEval, 2023) was sourced from the official SemEval 2023 challenge Github repository. Specifically, the training dataset was utilized, which we further divided into training, validation, and test sets in an 80%, 10%, and 10% split, respectively. The dataset consists of dialogues, each containing a varying number of utterances. To ensure consistency in the analysis, we verified that all utterances for each dialogue were included in the appropriate set, resulting in slightly adjusted ratios of 80.1% for training, 9.9% for validation, and 10% for testing.

Each utterance is annotated with both an emotion and a trigger, indicating whether an emotional shift occurs at that point in the conversation. The emotions assigned include ’anger’, ’disgust’, ’fear’, ’joy’, ’neutral’, ’sadness’ and ’surprise’, while the trigger is represented as a binary value (1 or 0). Notably, many dialogues in the dataset are extensions of previously existing ones, and in some cases, identical utterances are assigned different triggers.

The dataset also contained some missing (NaN) values, which were replaced with 0.

To better understand the distribution of emotions across the training, validation, and test sets, pie charts were generated. These visualizations clearly showed that the ’neutral’ emotion was by far the most represented in the dataset, followed by ’joy,’ ’surprise,’ and ’anger’ as the next most frequent emotions. In contrast, ’disgust’ and ’fear’ were the least represented. Similarly, a histogram of the trigger distribution revealed a significant imbalance, with trigger 0 overwhelmingly dominating the dataset.

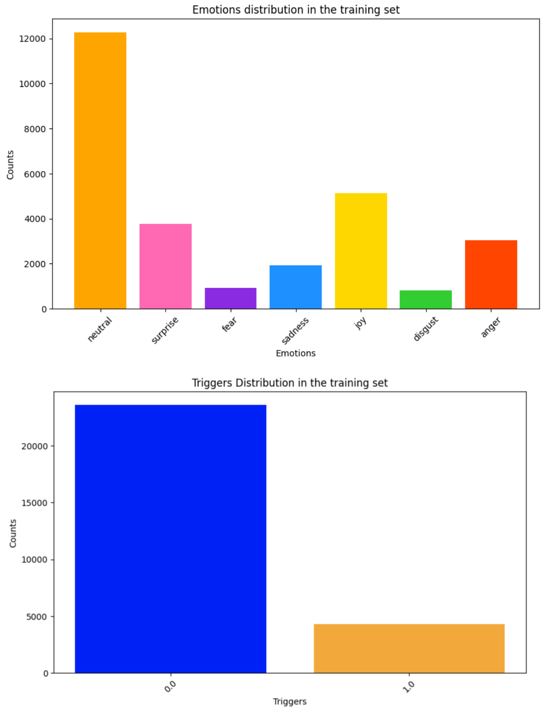

Figure 1: Imbalance of the dataset: emotions and triggers distribution

Given the imbalance in the distribution of emotions and triggers, class weights were applied to mitigate this issue.

The weight for each class emotion(j) was calculated using the following formula:

$$\text{weight(j)} = \frac{\text{total samples}}{\text{number of emotions} \times \text{count(j)}}$$

This ensures that underrepresented classes receive higher weights, while more frequent classes are assigned lower weights.

As a further experiment, a Data Augmentation approach was used to improve the models’ performances, which addresses the problem of unbalanced classes for emotions and triggers. Specifically, the ‘random swap’ method is used, which randomly modifies the order of only two words within the sentences of a specially selected dialogue. The dialogues added and modified are those containing at least one trigger at ‘1’ or the emotion with the lowest frequency, i.e. in our case ‘fear’. the final size of the dataset is slightly more than doubled.

Further statistical analyses were conducted, focusing on the number of words in each utterance and the total word count in each dialogue. The distribution of the number of words in dialogues showed an average of approximately 70 words, with the longest dialogue containing 263 words.

The most important step in the data preparation process is performed within the BERTDataset class. This class manages padding, truncation and encoding. The dataset is processed using the _create_data function, which ensures that for each row, the ‘episode,’ ‘utterance,’ ‘emotion,’ and ‘trigger’ are extracted. The dictionary used for encoding each emotion and trigger is represented as a one-hot vector, where only one position is set to 1, corresponding to the specific emotion or trigger, while the remaining values are set to 0.

The tokenizer converts all text to lowercase and removes any accents to ensure consistency in the input data. Each word is mapped to a numerical index representing its position in the model’s vocabulary. Special tokens such as [CLS] and [SEP] are also added to mark sentence boundaries and aid the model in understanding the structure of the conversation. Each utterance is converted into an array of length 256, with padding and truncation applied as necessary to maintain uniformity across the dataset.

The data is then batched and provided to the model as a dictionary containing the following elements:

- episode: a string identifier for the dialog;
- input_ids: tokenized sequences represented as integer indices;
- attention_mask: indicates which tokens should be ignored (e.g. padding);
- token_type_ids: distinguishes between different segments of the dialogue and separates utterances;
- target_emotion: the encoded true emotion for each utterance;
- target_trigger: the encoded true trigger for each utterance.

Finally, the DataLoader component was utilized to facilitate the efficient loading and management of data during model training and evaluation. Specifically, we employed PyTorch's DataLoader to ensure that the data is correctly formatted and batched, making it compatible for input into the model.

4 Experimental setup and results

To begin our experimental setup, we employed two baseline classifiers: the random classifier and the majority classifier. These classifiers act as performance benchmarks for the BERT-based models, representing the most straightforward approaches to making predictions.

The random classifier assigns emotions and triggers to sentences within a dialogue at random, without following any specific pattern. In this case there isn't a specific pattern and the assignment is random. In contrast, the majority classifier identifies the most frequently occurring emotion and trigger across the entire dataset, assigning these most common values to all sentences.

The general metric used in this task to evaluate the models is the F1 score. Specifically, both Unrolled Sequence F1 and Sequence F1 were calculated for each model. With the Sequence F1 score, the F1 score is computed for each dialogue individually, and then the average of these scores is reported. This metric emphasizes the performance of the model across dialogues as a whole.

In contrast, the Unrolled Sequence F1 score involves flattening all utterances across dialogues and calculating the F1 score for the entire flattened sequence. This metric evaluates the model's performance on all individual utterances collectively.

The Sequence F1 score gives a balanced view of performance across dialogues, while the Unrolled Sequence F1 score focuses on the model's ability to classify individual utterances correctly. As a result, the Sequence F1 score is less affected by errors in longer dialogues since it averages performance across entire dialogues, whereas the Unrolled Sequence F1 score treats all utterances uniformly, potentially highlighting performance on shorter segments.

For hyperparameter tuning, a GridSearch was conducted on two BERT-based models: one with all parameters trainable and one with frozen parameters (freezed model). The goal was to optimize the average accuracy in classifying emotions and triggers within dialogues. The GridSearch considered five random seeds ([42, 123, 200, 322, 506]), two epoch values ([3, 5]), two learning rates ([1e-5, 5e-5]), and two patience values ([2, 5]). Each combination was trained and evaluated on the validation set. The optimizer used for the training part of both BERT-based models is torch.optim.Adam improving the handling of textual data complexity. During the final training phase, the early stopping technique was implemented to improve the efficiency of the model.

The optimal performance was achieved with the following configuration:

Hyperparameter	**Optimal Value**
Seed	**42**
Epochs	**5**
Learning Rate	**5e-5**
Patience	**2**

Table 1: Optimal Hyperparameters for BERT-Based Models

This configuration shows an average accuracy equal to **0.865**.

5 Discussion

The results obtained with the basic classification models reveal their intrinsic weaknesses because they adopt extraordinarily simple approaches and are therefore unsuitable to manage complexity and variability of more diverse data distributions.

The accuracy for the random classifier of general emotion is equal to 0.14. From the classification report, metrics such as accuracy, recall and F1 score are also very low, belonging to a range between 0.10 and 0.18 on average, with a peak to be reported for precision for the 'neutral' emotion classification.

The same baseline model predict the target with a better accuracy equal to 0.49 and also with a F1 score equal to 0.49.

For the majority classifier of emotions, assigning the same label to each data item always results in an accuracy of 0.14 for emotions but with precision, recall and F1-score values close to zero for most other labels.

While the same model for the trigger prediction achieves a very high accuracy equal to 0.78 because the dataset, as mentioned in the second paragraph, is extremely unbalanced.

After obtaining the best parameters via GridSearch, the results in terms of accuracy for the full model and freezed model are **0.8505** and **0.8548** respectively.

A marked improvement can therefore be seen in the accuracy of the models. After obtaining the best parameters via GridSearch, the results in terms of accuracy for the full model and freezed model are 0.8505 and 0.8548 respectively. A marked improvement can therefore be seen in the accuracy of the models.

Considering the metric 'Averaged F1 score', the Full Bert Model scored 0.5677 for emotion classification, showing a significant improvement over baseline classifiers.

For triggers, the model also achieved a value of 0.8093, which is significantly higher than for triggers.

Regarding Mean and Standard Deviation over different seeds, the performance of the models, recorded during training phase, are shown in table 2.

Score Type	**Mean**	**Standard Deviation**
Average Full Emotions	0.5626	0.0065
Average Full Triggers	0.8093	0.0
Average Freezed Emotions	0.4422	0.0137
Average Freezed Triggers	0.8093	0.0
Unrolled Full Emotions	0.5481	0.0085
Unrolled Full Triggers	0.8548	0.0
Unrolled Freezed Emotions	0.4417	0.0109
Unrolled Freezed Triggers	0.8548	0.0

Table 2: F1 Scores for Full and Freezed Models - Emotions and Triggers

The averaged F1 score of the freezed model for emotions (0.4422) is notably lower than that of the full model, demonstrating the importance of fine-tuning for emotion classification. However, both models show identical high performance for trigger classification (0.8093), suggesting that trigger classification is less sensitive to parameter adjustments. The unrolled F1 scores are consistent with the averaged F1 scores, showing higher performance in trigger classification compared to emotion classification. The full model again outperforms the freezed model in emotion classification (0.5481 vs. 0.4417), emphasizing the benefit of fine-tuning. For trigger classification, both models achieve a high and identical unrolled F1 score of 0.8548.

Despite their efforts, these models struggled with accurately recognizing all emotions and trigger labels.

To address these challenges, we explored two additional experiments: the first, involving data augmentation combined with class weighting, showed an Average F1 Score equal to 0.9107 for Emotions and 0.83660 for full model; the second experiment, integrating both data augmentation and enhanced context by providing full discourses instead of single sentences, showed instead Average F1 Scores equal to 0.74390 and 0.32500.

In both experiments, Unrolled F1 Score performances show that the full model exhibits the capability to correctly identify true positives across all emotions and triggers, including those in minority classes. This indicates an enhanced ability to recognize less frequent emotions, underscoring the model's higher robustness in handling diverse class distributions.

These results suggest that, contrary to our initial expectations, context plays a less significant role in improving performance, while data augmentation has a stronger impact on enhancing model accuracy.

The main classification errors were concentrated in minority classes, where imbalanced data led to poorer performance, especially when class distinctions were conceptually less clear — such as the overlap between Fear and Sadness in certain contexts.

Incorporating context into the models provided some improvement, helping to better disambiguate these subtle differences. However, the most significant boost in performance came from increasing the data representation of minority classes, which helped the model better recognize and classify these less frequent emotions.

6 Conclusion

This study assessed two BERT-based models for Emotion Recognition in Conversations (ERC) and Emotion Flip Reasoning (EFR) in English dialogues, aiming to improve emotion detection and identify emotional shifts. Both models outperformed random and majority baseline classifiers, which had low accuracy. The random classifier achieved just 0.14 accuracy, while the majority classifier, assigning the most common emotion to all data, also performed poorly. In contrast, the BERT models excelled in classifying emotions and detecting conversational triggers.

In contrast, the frozen and full BERT models reached **accuracies** of **0.8505** and **0.8548**, respectively. These results suggest that the BERT models are much more capable of handling the complexity and variability found in conversations. In addition to accuracy, the F1 score was used as a metric to evaluate the performance of the models. The full BERT model achieved a **0.5677 F1 score for emotion classification** and a **0.8093 F1 score for trigger detection**. These scores reflect a significant improvement compared to the baseline models and indicate that the BERT-based approach is more effective at understanding emotional context in conversations. The frozen BERT model performed slightly better than the full BERT model in terms of accuracy, but both models were generally good with predictions. The imbalance in the dataset, where certain emotions and triggers were much more frequent than others, was addressed by applying class weights. This adjustment helped the models perform better on underrepresented emotions like "disgust" and "fear," which appeared less frequently in the dataset compared to more common emotions like "neutral" and "joy." Despite this, further improvements could still be made, especially in handling these less common emotions.

In summary, the BERT-based models showed strong performance in ERC and EFR tasks, significantly outperforming basic classifiers. Fine-tuning BERT improved emotion prediction and trigger detection. While the results are promising, future work could focus on enhancing the models for more complex emotions and conversations. Expanding the dataset to include diverse dialogue types could also boost their effectiveness.

References

SemEval. 2023. Semeval 2024 task 10: Emotion discovery and reasoning its flip in conversation (ediref).

Navonil Majumder, Soujanya Poria, Devamanyu Hazarika, Rada Mihalcea, Alexander Gelbukh, Erik Cambria. 2019. DialogueRNN: An Attentive RNN for Emotion Detection in Conversations.

Deepanway Ghosal, Navonil Majumder, Soujanya Poria, Niyati Chhaya and Alexander Gelbukh. 2019. DialogueGCN: A Graph Convolutional Neural Network for Emotion Recognition in Conversation.