EFR and ERC in English conversations
NLP Course Project

Mauro Dore, Giacomo Gaiani, Gian Mario Marongiu and **Riccardo Murgia**
Master's Degree in Artificial Intelligence, University of Bologna
{ mauro.dore, giacomo.gaiani, gianmario.marongiu, riccardo.murgia2 }@studio.unibo.it

Abstract

In conversations, speakers often experience various emotions. Understanding how these feelings change is crucial for comprehending the emotional flow of the conversation. This study explores the performance of two BERT-based models on the Emotion Recognition in Conversation (ERC) and Emotion Flip Reasoning (EFR) tasks. These models are double-head classifiers trained on various dialogues to evaluate how different setups influence their ability to comprehend the emotions expressed in utterances and the emotional changes within dialogues. Furthermore, various tests are conducted to evaluate the generalization capability of these models on new, unseen conversations. This examination aims to uncover the method's limitations, compare it with two baselines, and conduct an error analysis to identify areas for improvement.

1 Introduction

The objective of the ERC task is to attribute an emotion from a predetermined selection of possible emotions to each statement within a dialogue. Moreover, the EFR task aims to identify the instigator behind a speaker's emotion flip within a conversation. For example, an emotion flip from joy to anger could be caused by an instigator like threat (Kumar et al., 2023). The particular version of the problem explored in this report consists in identifying both the emotions associated with every utterance of English dialogues and the sentence that triggered a significant change in the emotion of a speaker. The details of the task are outlined in the subtasks 3 of the *SemEval 2024 Task 10: Emotion Discovery and Reasoning its Flip in Conversation* (SemEval, 2023).

Existent approaches to solve this task are often based on transformers. Kumar et al. (Kumar et al., 2021), for example, proposed a combination of a masked memory network based (ERC-MMN) and Transformer-based (EFR-TX) model for ERC and EFR, respectively. In particular, ERC-Masked Memory Network (ERC-MMN) tackles emotion recognition in conversations by modeling it as a sequence-labeling problem using separate speaker-level GRUs and a dialogue-level context through a global GRU which is shared across all the speakers within a dialogue. The masked memory network is used to retain pertinent information from previous conversations, enhancing emotion prediction by considering the significance of past utterances. Due to its structure, the model emphasises emotional dynamics in conversations with a higher sensitivity to the context. EFR-TX, on the other hand, defines instances as sequence of utterances to identify triggers for the target utterance, enhancing contextual representation through emotion labeling for trigger classification. The Transformer architecture and emotion labeling allows a deepen understanding of emotions within conversations.

Two years after their initial paper, Kumar et al. (Kumar et al., 2023) introduced TGIF, a model based on a hybrid architecture that combines Transformer encoders and stacked GRU units. This hybrid architecture aims to better capture dialogue context, speaker dynamics, and emotion sequences in conversations. The model extensively models the global utterance sequence and speaker dynamics to capture the underlying dialogue semantics. Additionally, considering the significant relationship between emotions and the task at hand, TGIF integrates the encoding of emotion sequences within utterances.

The models described in this report are also based on Transformer architecture. Specifically, our approach initially defines a majority and a random classifier as our initial baseline models. The core of our testing was centered around investigating the performance impact of either freezing or not freezing the weights of the BERT layer in our models. Furthermore, as an additional test, based on our observations of the dataset, we investigated two configurations, in order to see:

- How class weights were able to address the imbalance of data for each emotion/trigger and improve the performance of the model.

- How a version of BERT with a higher input limit would perform on the given input.

Through these experiments, we conducted a deep analysis of the domain and aimed to discern the optimal configurations for improving the model's ability to understand and process conversational dynamics. Our goal was to find the optimal balance for accurate emotion and trigger detection.

2 Data

The dataset for this study was obtained from the challenge's official website (SemEval, 2023). Specifically, we utilized only the training dataset, which was further divided into training, validation, and test subsets in a ratio of 80%, 10%, and 10%, respectively. The dataset consists of dialogues, each comprising a variable number of utterances. Each utterance is associated with an emotion and a trigger indicating whether an emotion flip occurs in this part of the dialogue. The possible emotions are anger, disgust, fear, joy, neutral, sadness, surprise. The trigger value, is either 1 or 0. One of the limitations of this dataset is that it was partially created using data augmentation, resulting in some dialogues being shorter versions of others, which leads to a loss of diversity.

During the pre-processing phase, the data is analyzed. *NaN* values are replaced with 0. For a better understanding of the domain, we plotted the distribution of emotions and triggers across the entire dataset, including the number of triggers associated with each emotion. As depicted in (1), we observed an unequal distribution for both triggers and emotions. The former exhibits more occurrences of 0 (indicating that the utterance will not cause an emotion shift), while the latter shows many occurrences of the neutral emotion, and few occurrences of disgust and fear. Additionally, we noted that for an equal number of occurrences, each emotion is associated with the same number of triggers, indicating that there are no specific emotions that generate more triggers than others. Given this imbalance in the observed class values, we decided to include class weights in the study as well. In addition, we also observed that in the dataset there are only a few triggers for each dialogue and they are mainly distributed at the end of it.

To prepare the data for the models, an additional transformation is conducted during the creation phase of data loaders. Firstly, we encode the utterances using the *bert-base-uncased* (Devlin et al., 2018) tokenizer, which involves applying two special tokens: the token *[CLS]* to mark the start of input at the beginning of each dialogue, and the separation token *[SEP]* between each utterance within the same dialogue. Subsequently, padding is applied to both dialogues and utterances to ensure that all dialogues have the same number of sentences and all utterances have the same number of tokens. Following this, padding is also applied to emotions and triggers to maintain consistency with the new data structure. The resulting input data is organized in batches and fed to the models in a dictionary composed of:

- *input_ids*: the tokenized sequences converted into integer indices.
- *attention_mask*: a customized attention mask that helps the model ignore the padding tokens added.
- *token_type_ids*: a mask that helps the model recognize the beginning of every dialogue and the separation points between the utterances.

A significant limitation of this padding strategy is that the final padding is determined by the maximum length of any dialogue and the maximum length of any utterance. Consequently, while padding does not impact the model's accuracy, it does affect its performance in terms of time. This is because the presence of outliers (in terms of dialogue or utterance length) necessitates assigning very long padding to every other element of the dataset.

3 System description

After preprocessing the input data as described in Section 2, it is fed into a BERT-based model, the pipeline of which is depicted in Figure 2 and organized as follows:

- Chunking: In this initial step, the input dialogues are segmented into smaller sections to ensure they fit within the Bert's input limit of 512 tokens. The chunking occurs in such a way that sentences are not cut in half. However, dialogues may be split.
- Feature extraction: An instance of the *bert-base-uncased* transformer is used to extract textual features from the chunked input. The weights of this layer are frozen in the *Bert Freezed* model configuration and trained in the *Bert Full* configuration.
- Reshape: The output from BERT, which provides a 768-value representation for each input token, is reorganized. This step transforms the tensor so that it yields a feature set representing each sentence as a whole, rather than individual tokens. During this phase, the dialogues that were split in the chunking phase are reassembled in order to be processed by the classifiers.
- Emotions Classifier: This component classifies the emotional content of the text for each utterance of the dialogues. It consists of two linear layers with a ReLU activation function in the middle.
- Triggers Classifier: Similar to the Emotions Classifier, this classifier focuses on identifying the triggers behind emotional shifts. It consists of two linear layers with a ReLU activation function in the middle.

4 Experimental setup and results

The initial stage of our experiments involved evaluating the performance of both a random and a majority classifier. The random classifier, upon receiving a dialogue as input, arbitrarily assigns an emotion and a trigger to each sentence within the dialogue. Conversely, the majority classifier assigns the most frequently occurring emotion and trigger, as determined from the entire dataset, to each sentence. This approach allowed us

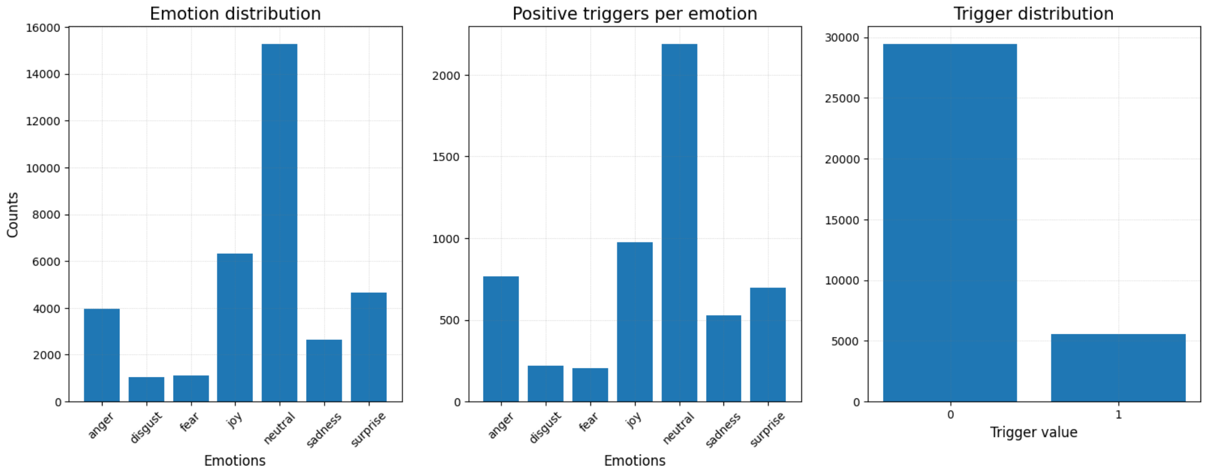

Figure 1: Data distribution

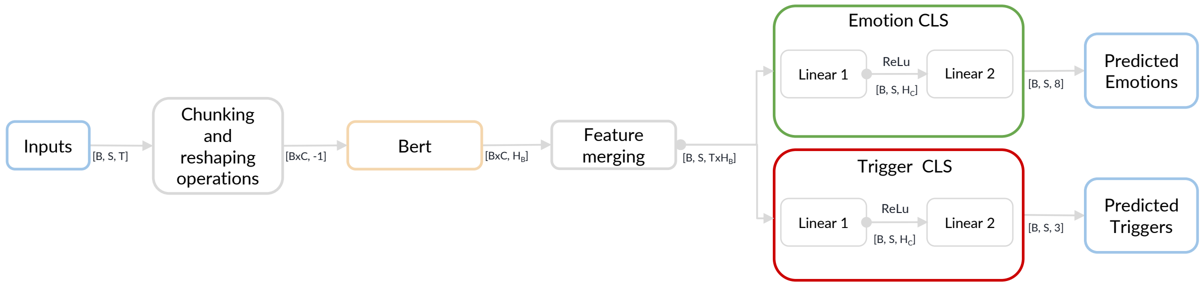

Figure 2: Architecture of the model, where B is the batch size, S is the number of utterances in each dialogue (equal for each dialogue due to padding), T is the length of each utterance (equal for each utterance due to padding), $H_b = 768$ is the hidden size of *bert-base-uncased*, $H_b = 512$

to establish baseline performances for the Bert-based models. The performances of all the models (Bert-based included) are monitored and stated by using the F1 score (as shown in table 1 and 2) in two variants:

- Unrolled sequence F1: the score is computed on list of flattened utterances and then averaged.
- Sequence F1: the score is computed for each dialogue and averaged.

The unrolled sequence F1 gives us a general estimation of the models' performances, assigning equal weight to each utterance in every dialogue. On the other hand, the sequence F1 places more emphasis on the F1 obtained in individual dialogues. Errors in short dialogues will be more heavily penalized in terms of F1, whereas in long dialogues, there are generally more opportunities for errors, even though with an equal number of errors, the F1 is less affected.

To find the best configuration of hyperparameters for the BERT-based models, we initially conducted manual tests to determine good starting values. Following this, we performed a grid search, where we only tested configurations with and without class weights, along with the 5 seeds required for a robust evaluation of performance.

Both *Bert Freezed* and *Bert Full* are trained with the *torch.optim.Adam* optimizer, using a learning rate of $1e-3$ and $2e-5$, respectively. It has been observed that in both models, the presence of class weights did not contribute to improving the performance in terms of F1 score. In fact, it was noted that the final scores decreased by a few percentage points when class weights were included. Furthermore, during the training phase, the stopping point is determined by an early stopper, which monitors the aggregate F1 unrolled sequence score between emotion and trigger in the case of *Bert Full*. For *Bert Freezed*, on the other hand, a customized version of the early stopper separately monitors the unrolled sequence F1 scores of both emotion and trigger. It then freezes the two classification heads separately when needed. This approach allows to train the two classifiers separately but cannot be applied to *Bert Full* because the weights of the BERT layer would continue to train while one of the two classification heads is frozen. This could result in a drop in performance for that head at the end of training, during the validation phase.

One of the additional tests we performed consisted of trying to obtain higher F1s by exploiting *longformer-base-4096* (Beltagy et al., 2020), a transformer based on *RoBERTa* with an input size up to 4096 tokens (instead of the 512 maximum tokens of *bert-base-uncased*).

Model	Class	Unrolled Sequence F1	Sequence F1
Random clf	Trigger	0.43227	0.41118
	Emotion	0.12880	0.09196
Majority clf	Trigger	0.45653	0.48501
	Emotion	0.08553	0.17486

Table 1: Sequence F1 and unrolled sequence F1 for the random and majority classifiers

		Unrolled Sequence F1			Sequence F1		
Model	Class	F1 val	F1 test	std deviation	F1 val	F1 test	std deviation
Bert Freezed	Trigger	0.57781	0.44621	0.01336	0.52643	0.43106	0.01603
	Emotion	0.76021	0.63317	0.01259	0.36782	0.56274	0.00962
Bert Full	Trigger	0.57002	0.51193	0.00521	0.52645	0.48246	0.01163
	Emotion	0.77570	0.64009	0.01295	0.38742	0.56666	0.00527

Table 2: Sequence F1 and unrolled sequence F1 on evaluation and test set for Bert-based models. The standard deviation is computed on the validation set averaging over 5 seeds.

5 Discussion

As shown in Table 1, the random and majority classifiers achieve poor results in terms of emotion classification, especially due to the number of possible values and their distribution, particularly evident in the case of sequence F1. However, they achieve higher results when dealing with trigger classification, with both reaching around an F1 score of 0.41-0.48. As observed from the results in Table 2, both *Bert Freezed* and *Bert Full* outperform the random and majority classifiers in every metric both on the validation and test set.

Bert Freezed, in terms of unrolled sequence F1 on triggers, did not show significant improvements compared to the same metric calculated on the results of the random classifier. However, it demonstrates significant improvements in emotions classification, both for unrolled and sequence F1.

Bert Full showed a slight improvement compared to *Bert Freezed* in almost all metrics, highlighting how training the BERT layer indeed brings benefits. However, the difference between the two models is minimal. One of the main reasons why *Bert Full* showed such little improvement compared to *Bert Freezed* is the presence of the double early stopper in the latter model. This allows it to freeze the weights of one of the classification heads before its results degrade.

The two BERT-based models perform relatively well, but there are some common errors (examples of the following observations are showed and analyzed in the *Prediction Examples and Observations* section of the notebook). All the emotions are usually correctly classified, however, we can observe occasional difficulty of the models in distinguishing between emotions that might also be confused by a human reader. For example, there are misclassifications of joy, which is sometimes interpreted as "surprise" or "anger". The simplest emotions to classify are "neutral," due to its high occurrence, "surprise," which is usually used in conjunction with exuberant punctuation (e.g., "!!!"), and anger, also used sometimes with unusual punctuation. This could mean that particular punctuation helps the model to narrow down the classification field, while emotions more closely related to body expressiveness, such as "disgust" or "sadness," are more difficult to detect. Furthermore, some words, typically associated with positive (negative) contexts, may lead to misclassification if ironically used in a negative (positive) context. The mistakes and misclassifications of emotions might be caused by several factors. We hypothesize that the spectrum of emotions we are trying to classify extends more over a range of negative emotions rather than positive ones, adding complexity to the task. For example, we need to classify disgust, fear, sadness, and anger (4 negative emotions) and only joy and surprise as positive emotions. However, it's worth noting that surprise can also be negative depending on the context.

	Class	Unrolled F1	Sequence F1
Big BertOne	Trigger	0.54995	0.52998
	Emotion	0.87445	0.81443

Table 3: Sequence F1 and unrolled sequence F1 obtained by *Big BertOne* on the test set

During trigger classification, *Bert Full* shows a much higher ability in predicting True Negative values. On the other hand, *Bert Freezed* slightly performs better with True Positive values. The struggles in detecting a positive trigger may be due to the chunking step of the model pipeline. Despite the use of special tokens $[CLS]$ and $[SEP]$, which indicate the beginning of each dialogue and the separation between utterances, the chunking process splits the dialogues into several parts, resulting in a loss of context for the model. Moreover, the inability of the models to accurately predict certain elements may be represented by the implicit triggers. As mentioned in (Kumar et al., 2024), it refers to those emotion flips prompted by external factors that are not explicitly mentioned in the dialogue.

Looking at the classification heads, they operate on one utterance at a time, potentially losing crucial information about the context. This aspect may not be a problem in emotion classification, since every emotion is strictly derived from the single utterance. However, it could be problematic for trigger classification, which depends on the overall context. One possible improvement might be to address this issue by developing classifiers capable of providing a unified classification for each utterance within the same dialogue simultaneously.

Another issue that could interfere with the model's ability to extrapolate the context of dialogues may be caused by the chunking operation, which split the model's input into multiple parts.

There are several approaches that may attenuate this problem, a first one consists in developing a model which takes as input a set of dialogues with a shorter padding. In fact, among all utterances in the dataset, only 4 have a total length exceeding 95, while every other utterance falls below 65. Consequently, removing the dialogues containing these few outliers will help the model during the chunking phase to analyze bigger portion of the dialogue in the same iteration, potentially leading to better results in less time.

A second strategy, that we directly tested, consists in trying to utilize a more complex feature extraction layer, with a higher input size, capable of processing an entire dialogue at once. *Longformer* is a tansformer based on *RoBERTa* with an input size up to 4096 token (instead of the 512 maximum token of *bert-base-uncased*). We named "Big BertOne" this new model based on it and analyzed its behavior on the same test set used for the other models. The results of this test are shown in Table 3. The model showed significant improvements in terms of F1, that are also reflected in the sequence F1 score, which was an important weakness in previous Bert-based models. In terms of trigger detection, there are no large improvements, as expected, but the model does indeed perform better in this aspect as well, surpassing *Bert Full* by 6 percentage points.

6 Conclusion

After looking at the results, the BERT-based models performed better than the baselines. While *Bert Full* didn't show a big improvement compared to *Bert Freezed*, training the BERT layer in the former resulted in slightly better outcomes as it probably helped gather more context from the dialogue. *Big BertOne*, as expected, performed much better on the emotion classification task. Whereas, it obtained only slightly higher results in detecting positive triggers.

The main limitations of the models are mainly two. The first concerns difficulties in classifying emotions that even humans would struggle with when faced with only text (e.g., anger instead of joy). Additionally, the neutral emotion is often mistaken for another due to its high occurrence. The second problem lies on the classification portion of the models, as the classification heads operate one utterance at the time, the process loses information about the context of the overall dialogue. Regarding the trigger classification, a possible solution to restrain the need of extensive amount of significant data is to use rule-based or machine learning techniques instead of an LLM-based architecture. Another solution would be to implement a more convoluted classification mechanism capable of keeping track of the global dependencies in the input sequence. Regarding the emotion classification, a potential improvement is to encode the information about the speakers into the input sequence. This could give the model supplementary information about the context, helping it to draw further conclusions. On the other hand, the additional data may lead to the generation of biases, as it introduces the risk of the model making assumptions or generalizations based on speaker name more than the utterance itself.

Future improvements in relation to the overall model performances may be: executing the training of the *Longformer* layer, similarly to the *Bert Full* model described previously, or extending the ability of the model to capture the sequential nature of dialogue and track changes in emotion over time by means of recurrent neural networks, in combination with the training of the Bert layer.

7 Links to external resources

The repository with code, data, models and results can be found at: https://github.com/GianM0027/NLP_course_project

References

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Shivani Kumar, Md Shad Akhtar, Erik Cambria, and Tanmoy Chakraborty. 2024. Semeval 2024 – task 10: Emotion discovery and reasoning its flip in conversation (ediref).

Shivani Kumar, Shubham Dudeja, Md Shad Akhtar, and Tanmoy Chakraborty. 2023. Emotion flip reasoning in multiparty conversations.

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.

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