https://github.com/ParsaMK/BERT-MELD-NLP

Final Project

Mirco Lescart, Freddy Fernandes, Parsa Mastouri Kashani
Arina Sadeghi Khiabanian
Master's Degree in Artificial Intelligence, University of Bologna
{ mirco.lescart, freddy.fernandes, parsa.mastouri, arina.sadeghi }@studio.unibo.it

February 14, 2024

Abstract

The goal of this report is to explain and report our findings about solving the problems of figuring out the feeling of each side of a conversation and also determining their triggers which is the part of the conversation which has invoked that feeling. In this project, we implemented two approaches to tackle this problem. Our first approach comprised of a Bert model and it produces emotions and triggers for one utterance each time it gets run given the whole conversation. Our second approach also uses a Bert model specialized for figuring out the emotion in one utterance without its surrounding context (surrounding utterances) and another Bert model for figuring out the triggers for each utterance given the whole conversation .

1 Introduction

In order to ensure optimal model performance, we implemented a preprocessing pipeline for our data analysis. We cleaned and structured the conversation datasets to extract information, including speakers, utterances, emotions, and triggers. We applied tokenization, a crucial step in natural language processing, to represent textual data in a format suitable for machine learning models.
For the emotion detection task, we designed an EmotionDataset class to encapsulate the training, validation, and testing sets. Tokenization utilized a tokenizer compatible with the BERT models, ensuring consistency and coherence.
Similarly, we crafted the TriggerDataset class to facilitate the preprocessing of data for trigger detection. The format of speakers, utterances, emotions, and triggers was adapted to suit the requirements of the specialized BERT models used in this context.
We use a BERT model from Transformers, as the primary methodology. This model was fine-tuned to extract emotions and triggers for individual utterances within the larger context of the entire conversation. Our customized BERT model, incorporated a dropout layer and dense

layers to transform the model's output into the final emotional predictions. We computed class weights to handle imbalances in the emotion distribution, and the model performed training using a combination of CrossEntropyLoss and a linear scheduler.

The second approach involved the utilization of two distinct BERT models for trigger detection. The first model focused on discerning emotions within isolated utterances, independent of surrounding context. The second model, designed specifically for trigger identification, operated on the entire conversation context. The BERTClass architecture was adapted for trigger detection, employing a dropout layer and dense layers to process the BERT output. We chose Binary CrossEntropyLoss with class weights as the criterion for training.

Both emotion and trigger detection models underwent training using DataLoader instances configured for the respective datasets. The training process incorporated optimization through the Adam optimizer, with a linear learning rate scheduler to enhance convergence. We evaluated model performance during validation, saving checkpoints based on minimum validation losses.

2 Background

Understanding the intricate ways in which individuals express emotions during conversations poses a significant challenge in the realm of natural language processing. Within the scope of this project, we have embarked on a journey aimed at achieving two primary objectives: emotion recognition and the identification of triggers, both crucial for attaining a deeper understanding of human emotional experiences.

The task of deciphering emotions within conversational contexts necessitates a comprehensive analysis of not only the lexical content but also the subtle contextual nuances inherent in human communication. Teaching a computer to navigate through the diverse array of emotional expressions, especially within the dynamic landscape of multi-party interactions, represents a formidable endeavor in computational linguistics.

Triggers, analogous to subtle cues that serve as precursors to emotional shifts, encompass a myriad of linguistic elements and interaction patterns. Unraveling these triggers holds the key to unraveling the complex dynamics underlying emotional transitions within conversations, offering invaluable insights into human behavior and psychology.

To tackle these intricate tasks, we leverage the power of BERT models. BERT, renowned for its prowess in contextual understanding and language representation, emerges as a fitting choice for unraveling the intricacies of human discourse and emotional expression.

In the subsequent sections of this discourse, we delve into the intricacies of our data preprocessing methodologies, elucidating the steps taken to prepare the raw conversational data for analysis. Furthermore, we expound upon the meticulous process of fine-tuning BERT mod-

els, tailoring them to effectively capture and interpret the underlying emotional dynamics and triggers embedded within the fabric of conversational exchanges.
Through this concerted effort, we aspire to shed light on the intricate interplay of emotions and triggers within human conversations, paving the way for a deeper understanding of the human psyche and facilitating advancements in natural language processing and computational linguistics.

3 System Description

Our system was built using Google Colab as the primary platform for coding and collaboration. The architecture and coding efforts were a collaborative endeavour, with each team member contributing to different aspects of the project.
The design of the system involved the implementation of two distinct methodologies for emotion and trigger detection in conversations.
For emotion detection, a BERT model, named BERTClass, was designed. This model featured a dropout layer and dense layers to process the BERT output, transforming it into final emotional predictions. The architecture was designed to capture the nuances of emotional shifts within conversations.
The trigger detection methodology employed two specialized BERT models. The first model focused on discerning emotions within isolated utterances, regardless of the surrounding context. The second model, designed specifically for trigger identification, operated on the entire conversation context. Both models utilized the BERTClass architecture with adaptations to suit the unique requirements of trigger detection.
A data preprocessing pipeline was implemented, involving cleaning and structuring conversation datasets. Tokenization, a critical step in natural language processing, was applied to represent textual data in a format suitable for machine learning models. Custom dataset classes, EmotionDataset, and TriggerDataset were designed to encapsulate the preprocessing steps for emotion and trigger detection tasks.
The training process involved DataLoader instances configured for emotion and trigger datasets. Optimization was performed using the Adam optimizer, and a linear learning rate scheduler was applied. Validation evaluation helps the training process, with checkpoints saved based on minimum validation losses.

4 Data

We utilized the MELD dataset, comprising 13,708 utterances spoken by multiple speakers across 1,433 dialogues from the popular sitcom FRIENDS. Each utterance is associated with an emotion label representing one of Ekman's seven basic emotions: anger, fear, disgust, sadness, joy, surprise, and neutral. We extended the dataset for our trigger task by splitting it each time an emotion flip occurs. A dialogue example is illustrated in Figure 1. Here, the dialogue is split into two dialogues at the fourth and fifth utterances, with triggers $[0, 0, 1, 0]$ and $[0, 0, 0, 1, 0]$ respectively.

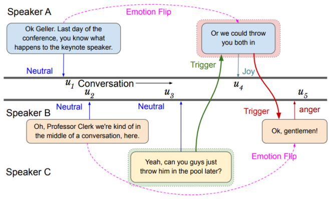
Speaker A
Emotion Flip
Ok Geller. Last day of the conference, you know what happens to the keynote speaker.
Or we could throw you both in
Neutral
Trigger
Joy
u_1 Conversation
u_2
u_3
u_4
u_5
Speaker B
Neutral
Neutral
Trigger
anger
Oh, Professor Clerk we're kind of in the middle of a conversation, here.
Ok, gentlemen!
Emotion Flip
Speaker C
Yeah, can you guys just throw him in the pool later?

Figure 1: Dialog Example.

For the emotion model, we drop the dialogue level and trigger to consider all utterances independently, i.e., without context. The input text is each utterance, and emotions are formatted in a one-hot format. To handle duplicate dialogs due to emotion flips, we drop duplicates, resulting in a dataset of 8,331 observations.

For the trigger model, we need context. We iteratively split the dataset for each trigger in the triggers vector, ensuring one trigger as the output to predict and avoid different output dimensions. The input text is all dialogue utterances concatenated with emotions and trigger values from previous utterances. After dropping duplicates, we obtained 11,610 observations. For the trigger model, contextual information is crucial. The input text is constructed by concatenating all the dialogue's utterances and the corresponding emotions and trigger values from preceding utterances. To illustrate, consider the input text format for the third trigger split:

$[Speaker1]$utterance 1.
$[Speaker2]$utterance 2.
$[Speaker1]$utterance 3.
emotions: [emotion 1, emotion 2]
triggers: [trigger 1, trigger 2]

In this manner, the input text encapsulates the dialogue's context, allowing the model to consider the unfolding conversation history along with associated emotions and triggers, facilitating a comprehensive understanding of the sequence.

Both datasets are tokenized using the BERT tokenizer, and we split the data into train-validation-test sets with an 80-10-10 ratio.

Next, a descriptive analysis of the original dataset is conducted. Figure 2 illustrates the frequency of each emotion in different datasets. While the emotions have the same distribution, the classes are unbalanced, e.g., 3,733 neutral occurrences compared to 247 fear occurrences.

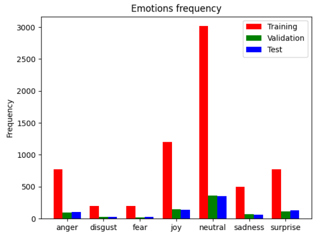

Figure 2: Frequency of each emotion in the different datasets.

Triggers exhibit the same distribution in the different sets, with approximately 84.1%, 83.9%, and 84.2% zeros. Figure 3 shows the distance between emotion flips and active triggers. The trigger probability decreases exponentially with the distance from the emotion flip, with a maximum at distance 1 (possibly following a log-normal distribution).

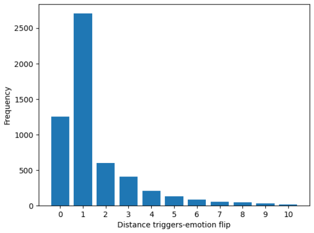

Figure 3: Distance between emotion flip and active triggers.

5 Experimental setup and results

5.1 Experimental Setup

In our NLP project, the primary goal was to improve the detection of emotions and triggers within conversational data. To achieve this, we leveraged BERT-based models renowned for their cutting-edge performance in comprehending the context and semantics of language.

5.2 Architectures

Baseline Models: For baseline comparison and initial evaluation, we incorporated the DummyClassifier from the scikit-learn library, employing two distinct strategies:

Uniform Strategy: This strategy randomly predicts labels, providing a baseline for comparison against models making random guesses.

Most Frequent Strategy: This strategy predicts the most common label observed in the training set. It serves as a simple benchmark against which to compare the performance of more sophisticated models.

These strategies allow us to establish a baseline performance level and assess the effectiveness of our BERT-based models in comparison to simpler, rule-based approaches.).

Advanced Models: BERT-based models for deep contextualized representation learning.

5.3 Configuration and Hyperparameters

Max Length: The maximum sequence length was set to 50 for emotions (`MAX_LEN_EMO`) and 512 for triggers (`MAX_LEN_TRI`).

Batch Size: A batch size of 1 for model processing.

Epochs: The model was trained for 12 epochs.

Learning Rate: We used a learning rate of 3e-05.

Optimization: Adam optimizes with a linear learning rate scheduler.

5.4 Metrics

Evaluation Metrics: We used accuracy, precision, recall, F1-score, and validation loss as our key performance indicators.

5.5 Numerical Results

In the three given tables, we present our findings in both the emotion and trigger models.In the first table,we provided the results for emotion model after 5 epochs.In the second table,we provided results for our full emotion model after 10 epochs.In the third table,results for the trigger model is provided.In the last table,we provided sequence for trigger full model.

Model	Weighted Avg(F1 score)	Macro Avg(F1 score)	Micro Avg(F1 score)
Freezed Emotion Model	42%	24%	51%
Full Emotion Model	48%	31%	52%

Model	Weighted Avg(F1 score)	Macro Avg(F1 score)	Micro Avg(F1 score)
full Emotion Model	52%	39%	53%

Model	Weighted Avg(F1 score)	Macro Avg(F1 score)	Accuracy
Freezed Trigger Model	76%	57%	76%
Full Trigger Model	81%	71%	82%

Sequence	Micro Avg(F1 score)	Macro Avg(F1 score)	Accuracy
Full Trigger Model	78%	67%	78%

6 Emotion and Trigger Detection

Frozen Model (Limited Learning): Demonstrated reasonable performance and was particularly effective at recognizing the majority class.
Full Model (Full Learning): Exhibited superior performance across all metrics, with the validation loss graph showing consistent improvement, indicating effective learning and optimization.

7 Discussion

7.1 Discussion of Quantitative Results

In our project, we tested our emotion and trigger detection models against simple dummy classifiers as a starting point. For both tasks, the uniform dummy classifier, which guesses outcomes randomly, had low accuracy: 15% for emotions and 50% for triggers. This showed that random guessing is ineffective in these complex tasks. The most frequent dummy classifier, always predicting the common class, did better with 45% accuracy for emotions and 84% for triggers, but this was misleading because it mostly guessed the predominant class without truly understanding the data.

Our advanced models showed more promise. Both the frozen and full models were more accurate than the dummy classifiers. The frozen models were slightly better than the dummy classifiers because they could identify common classes well, with 44% accuracy for emotions and 50% for triggers. The full models, which could learn from all data, had the best results with 57% accuracy for the emotion model and 55% for the trigger model. They were better at understanding the nuances of the data, which was also shown in the lower validation losses they achieved.

The validation loss graph further supported the superiority of our full model. While the frozen model maintained a relatively constant validation loss of around 1.65-1.6, the full model consistently achieved a lower loss, stabilizing at approximately 1.35 from the first epoch onwards. This suggests that the full model learned more effectively, reaching a better optimization point compared to the frozen model.

The chart for accuracy against the distance from the target showed that our models were most accurate when predictions were close to the actual labels. However, the number of mistakes increased as the predictions got further from the target. This highlighted areas for potential improvement in the models.

Overall, the full models using BERT with fine-tuning were the most effective, suggesting that allowing the model to learn from the entire dataset is beneficial. However, the frozen model's reasonable results indicate that in some cases, freezing layers to prevent overfitting and reduce training time can be a practical approach for both emotion and trigger detection tasks.

7.2 Error Analysis

During the course of our project, several challenges and obstacles arose, which we will outline in this report. Foremost among these challenges was the limitation posed by the dataset itself. One prominent issue was the lack of sufficient data representing emotions such as anger and disgust. Despite our efforts to balance the weights within the dataset, the absence of ample samples for these emotions significantly impacted the model's ability to accurately recognize and classify them.

Another significant challenge we encountered was the issue of insufficient data within our dataset, leading to overfitting problems. The limited volume of data available for training resulted in the model becoming overly specialized to the nuances of the training set, thereby hindering its ability to generalize well to unseen data. This overfitting phenomenon posed a substantial obstacle to the effectiveness and reliability of our NLP model.

While acknowledging the potential richness of the MELD dataset, it's important to note that for the scope of our project, we worked with a reduced subset of data. While the MELD dataset likely contains a wealth of examples, our project's constraints necessitated the use of a smaller subset. Regrettably, this reduction in data size proved to be a limiting factor, as it constrained the diversity and quantity of samples available for model training and validation. As a result, we encountered challenges related to insufficient data, which affected the robustness and generalization capabilities of our NLP model.

Furthermore, a notable aspect of the MELD dataset is its inclusion of audio and video components alongside textual data. However, for the purposes of our project, we focused solely on textual data, thus excluding the audio and video modalities. This decision to remove audio and video elements presented a distinctive challenge, as humans typically rely on multiple modalities, including tone of voice and facial expressions, to accurately discern emotions. Consequently, restricting our analysis to text alone posed difficulties for both our model and human observers in accurately recognizing and interpreting emotions. This limitation underscores the inherent complexity of emotion recognition tasks and may have contributed to suboptimal performance in our model's ability to discern emotions based solely on textual cues.

8 Conclusion

In this project, we worked on the difficult task of enhancing emotion and trigger detection in conversational speech through the utilization of advanced natural language processing techniques, specifically using BERT-based models. The results were insightful, showcasing a clear hierarchy in model performance. The full model, with its ability to capture intricate contextual nuances, outshone both uniform and most frequent dummy classifiers, as well as a frozen model with restricted learning. The validation loss graph further proved the full model's superiority, demonstrating consistent improvement in optimization over epochs. However, surprisingly, the frozen model performed remarkably well while not surpassing the full model, suggesting that strategic weight freezing could be a viable approach in scenarios with limitations on fine-tuning. The limitations of our solution include the dependence on labelled data with imbalanced classes, as the models' effectiveness hinges on the availability of diverse and representative training samples. Furthermore, the complexity of human emotions introduces several challenges, and the model's performance may vary across different conversational contexts.

References

- Dialog Act Classification with BERT Models

- Dialogue Act Classification with

Context-Aware Self-Attention

- HunEmBERT: A Fine-Tuned BERT-Model for Classifying Sentiment and Emotion in Political Communication

- Fine-Tuning DistilBERT for Emotion Classification

- Knowledge-based BERT word embedding fine-tuning for emotion recognition

- BERT 101-State Of The Art NLP Model Explained

- Why AdamW matters