



École Polytechnique

BACHELOR THESIS IN COMPUTER SCIENCE

Emotion Recognition in Conversation through Emotion Flow

Author:

Bruno Iorio, École Polytechnique

Advisor:

Gaël Guibon, LIPN - Université Sorbonne Paris Nord

Academic year 2024/2025

Abstract

Emotion Recognition in Conversation (ERC) is a very important domain, which has been gaining more attention in recent years, especially within NLP. In the scope of Emotion Recognition, identifying emotions in dialogues plays an essential role. This is because most of the emotional text data collection happens in the context of a conversation between two or more parties (e.g. customer service survey). In this paper we discuss the limitation of previous approaches to the ERC task, while also evaluating an original approach to the same problem using Causal learning, which we identify as the Emotion Flow and attention mechanisms. **(initial version)**

Contents

1	Introduction	4
1.1	Background	4
1.2	Task definition	5
1.3	Contributions	5
2	Related Work	5
3	Datasets used	6
4	Methodology	6
4.1	Text preprocessing	6
4.2	Emotion preprocessing	6
4.3	Models	7
5	Experimental Protocol	9
6	Results	9
7	Limitation of our approach	9
8	Conclusion	9
9	Perspectives	9
10	References	10
A	Appendix	11

1 Introduction

1.1 Background

Emotions can be understood as the psychological state of an individual, which can be directly influenced by external factors. In human interactions, they serve as a natural response to these external agents, affecting communication, actions, speech tone, etc. Emotion Recognition (ER) deals with the problem of detecting emotions in speech, and this comes with the challenge of studying how different factors – such as context – can shape emotions. Studying ER, therefore, provides us important information about the dynamics in human communication.

Emotion Recognition in Conversation (ERC) is a growing field within NLP which tackles the task of identifying emotions in conversation. The collection of emotional data through conversation provides a realistic, and rich source of information compared to isolated text examples. Having this information, and utilising it properly has become increasingly important in many different sectors. For instance, companies can leverage ERC to measure customer’s satisfaction, allowing a better perspective on how to improve their services.




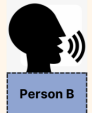
 Person A	Hello, how are you?	Neutral
 Person B	Are you going to pretend nothing happened yesterday?	Angry
 Person A	What do you mean?	Confused
 Person B	Ah... You are always like this!	Disappointed

Figure 1: Emotion Recognition example in coversation.

Restricting Emotion Recognition to text has he benefit of allowing us to maintain many important emotional agents, while keeping compact and easy-to-process data. However, it eliminates factors that are strongly connected to emotions, such as voice tone, gestures, facial expressions etc. This imposes strong limitations on emotion detection task, and limits the extend to which we can study emotions. Also, disambiguating emotions becomes severely more complex: emotions that are already very nuanced – such as joyfulness, hapiness, excitement etc – become even harder to be distinguished, when we surpress multiple emotional factors.

In order to overcome these challanges, many resources were introduced in ERC. Recent research has been strongly focused on understanding the strong and weak points of different architectures when applied to this task. DialogueRNN [5], leverages recursive and attention mechanisms, to retrieve a "party state", for each of the parties in the conversation, which is later used to classify each emotion. It compares how different recursive modules – such as LSTMs, and GRUs – in the same task. Task-Oriented Dialogue [10] have been integrated with LLMs for ERC, by providing emotion recognition tasks for fine-tunned LLMs. This work, however, overlooks the fact that LLMs may are trained on many publicly available datasets, possibly including the ones their models were tested on.

1.2 Task definition

An important factor influencing emotion is the Emotion Flow (EF), which can be defined as the graph describing the evolution of emotions throughout a conversation. It allows us to add an extra depth into the emotion recognition task, and it can address many issues, such as emotional disambiguating. Researchers believe that emotions have a certain resistance to change over time, a concept known as Emotional inertia. It implies that previous emotions have a strong impact on future emotions, which strongly motivates us to study EF. To the best of our knowledge, previous research has only studied Emotion Flow in an utterance level, by predicting emotions for each utterance. So, the question whether other treatments to emotion prediction could be effective still remains to be answered. More especially, we raise the question whether a word level approach, by predicting an emotion for each word in the conversation, could be effective.

In this Project, we aim at evaluating how a word level approach can be combined with the Emotion Flow in the ERC task. Along with this, we also evaluate how predicting tokens along with the emotions, in a word-level approach can affect the performance of the models.

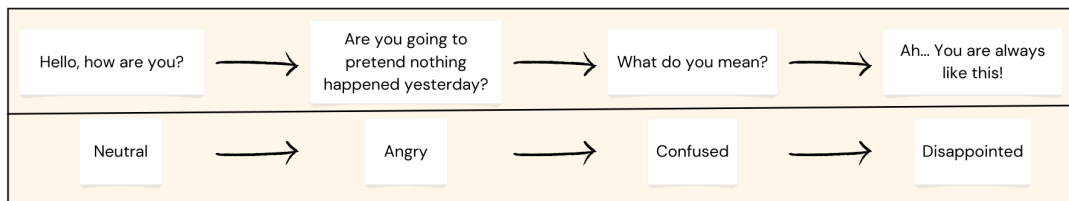


Figure 2: Utterance-level emotion flow vs. Word-level emotion flow

1.3 Contributions

complete this later All the work presented in this report can be found [here](#).

2 Related Work

Many interesting methods were previously used in ERC. In [4], it is applied a window transformer - using narrow 2D window masks - to catch short-term inter-utterance relations, used in the task of Causal Emotion Entailment(CEE), e.g. determining which parts of the text are causing each emotion. This was particularly inspired from MPEG [1], where the emotional information is embedded and later on fused with the textual information using an attention layer. This is useful because the CEE is closely related to ERC, and actually its

use can actually improve the efficiency of ERC models [4]. In [12], the model also predicts a personality profile vector for each individual participating in the conversation, by using BERT, with a multilayer perceptron. This is integrated with a fine-grained classification module, that predicts an emotion for each utterance. In [9], they achieve good performance in the ERC task by applying prompting engineering techniques, following with a fine-tuned BERT classifier. Comparatively, it is a very simple approach to this problem.

Many of the previous works provide us ways of fusing the emotion information with the textual information, while not overcomplexifying the overall architectures of the models. However, they don't address different approaches for the treatment of the Emotion Flow, specially how a word-level approach would perform.

3 Datasets used

Throughtout this project, we considered the following options of datasets:

DailyDialog: [3] Dataset containing 14,118 dialogues representing daily life dialogue situations. Each utterance is classified with one of the following emotions: anger, disgust, fear, happiness, sadness, surprise or other(no emotion).

MELD: [8] Dataset containing emotion anotation for over 1400 dialogues, and 13000 utterances, which were extract from the Friends TV show. It classifies each utterance as one of the following emotions: anger, disgust, fear, joy, neutral, sadness or surprise.

EmoryNLP: [13] 12,606 utterances extracted from the Friends TV show. It classifies each utterance as one of the following emotions: joyful, peaceful, powerful, scared, mad or sad.

After careful consideration, we decided to mainly focus on the EmoryNLP dataset. The reason for this is that EmoryNLP is considerably more balanced than the other two datasets, even though having less content than the DailyDialog dataset. This extra variety is positive, because it allows models to learn better the difference between emotions.

Insert Diagram for DD, MELD, EmoryNLP

4 Methodology

4.1 Text preprocessing

We used a classic tokenization approach. For each conversation C in the dataset, we applied the tokenizer for each utteration utt_k , getting a list of tokens $[tok_{k,1}, tok_{k,2}, \dots, tok_{k,m_k}]$. After, we proceeded to concatenate in order each of the tokenized utterances, getting $[tok_1, tok_2, \dots, tok_m]$, where m is the maximum number of words per conversation, set default to 200.

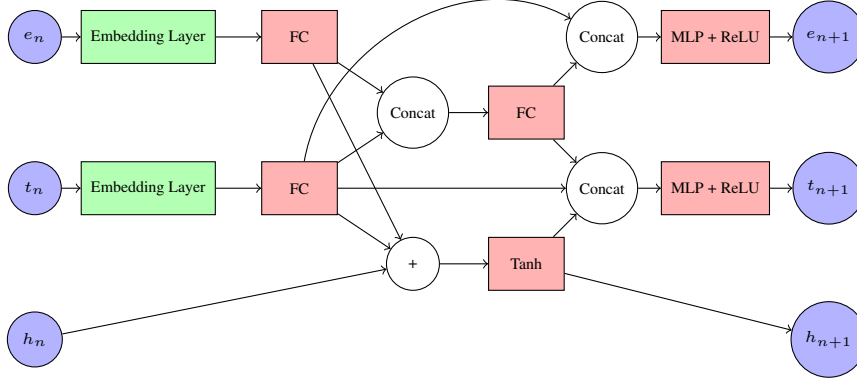
$$C = [utt_1, \dots, utt_n] \rightarrow \\ [[tok_{1,1}, tok_{1,2}, \dots, tok_{1,k_1}], \dots, [tok_{n,1}, \dots, tok_{n,k_n}]] \rightarrow \\ [tok_1, tok_2, \dots, tok_m]$$

In between different utterances we added a separation token $\langle sep \rangle$, to indicate that a sentence has ended and another one has started. Similarly, we added a padding token $\langle pad \rangle$, so that, if the length of the conversation is smaller than m tokens, we pad C until it reaches m tokens. Both these additions help us patternizing the data, and adding more information for the training process.

4.2 Emotion preprocessing

4.3 Models

Simple Recursive Model



This model integrates contextual information, by using a recursive hidden state. To fuse the textual information with the emotional information, we use a very simple concatenation followed by a linear layer.

- **Input token** (t_n)
- **Input emotion** (e_n)
- **Output token** (t_{n+1})
- **Output emotion** (e_{n+1})
- **Hidden state** (h_n)
- **Fusion layer output** (f_n)

We first encode the text using pretrained Fasttext embeddings, and we encode the emotions with trainable embeddings.

$$t_n \leftarrow \text{Embedding}_{\text{text}}(t_n)$$

$$e_n \leftarrow \text{Embedding}_{\text{emotion}}(e_n)$$

Then, we forward both in Fully Connected (FC) layers. and we use it to update the hidden state, by summing them up, and we concatenate them into a fusion layer.

$$f_n \leftarrow \text{Concat}(\text{FC}(t_n), \text{FC}(e_n))$$

$$h_{n+1} \leftarrow \text{Tanh}(\text{FC}(t_n + e_n))$$

We then finally predict the next token t_{n+1} by concatenating f_n , t_n , and h_{n+1} , and passing them through an Multi-Layer Perceptron (MLP) with ReLU. To predict the next emotion we concatenate f_n and t_n , and then we pass them through a MLP with ReLU

$$t_{n+1} \leftarrow \text{ReLU}(\text{MLP}(\text{concat}(f_n, h_{n+1}, t_n)))$$

$$e_{n+1} \leftarrow \text{ReLU}(\text{MLP}(\text{concat}(f_n, t_n)))$$

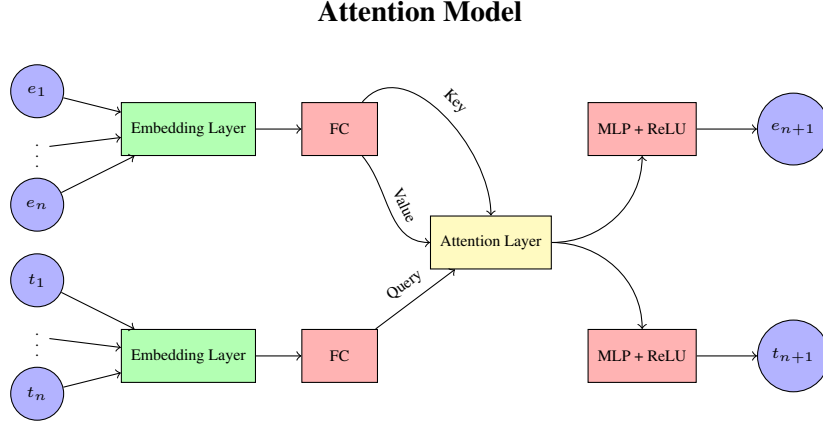


Figure 3: Attention model diagram

This model uses an multi-headed attention mechanism in the fusion layer for the emotions with the text embeddings. This idea was inspired from [4].

- **Input tokens** $((t_k)_{1 \leq k \leq n})$
- **Input emotions** $((e_k)_{1 \leq k \leq n})$
- **Output token** (t_{n+1})
- **Output emotion** (e_{n+1})
- **Fusion layer output** (f_n)

Firstly, each of the tokens t_k is sent through a pretrained embedding layer, and each emotion e_k is sent through a trainable embedding layer.

$$t_k \leftarrow \text{Embedding}_{\text{text}}(t_k)$$

$$e_k \leftarrow \text{Embedding}_{\text{emotion}}(e_k)$$

We forward them into FC layers, use the results as input for the attention layer.

$$t_k \leftarrow \text{FC}(t_k)$$

$$e_k \leftarrow \text{FC}(e_k)$$

$$f_n \leftarrow \text{Attn}((t_k)_{1 \leq k \leq n}, (e_k)_{1 \leq k \leq n}, (e_k)_{1 \leq k \leq n})$$

We finally compute the text and emotion output by simply forwarding f_n through MLP layers followed by ReLU activation.

$$t_{n+1} \leftarrow \text{ReLU}(\text{MLP}(f_n))$$

$$e_{n+1} \leftarrow \text{ReLU}(\text{MLP}(f_n))$$

5 Experimental Protocol

To run our experiments, we split the data into three: training data, test data, and validation data. The training data is used to train the model we will use. The test data set will be used to evaluate the training of each of the models (e.g. compute the metrics after each training epoch). The validation set will be used to evaluate the final performance of the model.

The metrics we used are f1-score, and mcc score.

6 Results

Model	Simple Recursive Model			Attention Model		
Metric	weighted F1	macro F1	mcc	weighted F1	macro F1	mcc
DailyDialog	-	-	-	-	-	-
MELD	-	-	-	-	-	-
EmoryNLP	-	-	-	-	-	-

Figure 4: f1-score and mcc evaluation for each of the models

7 Limitation of our approach

8 Conclusion

9 Perspectives

10 References

- [1] Tiantian Chen, Ying Shen, Xuri Chen, Lin Zhang, and Shengjie Zhao. Mpeg: A multi-perspective enhanced graph attention network for causal emotion entailment in conversations. *IEEE Transactions on Affective Computing*, 15(3):1004–1017, 2024.
- [2] Jiang Li, Xiaoping Wang, and Zhigang Zeng. A dual-stream recurrence-attention network with global–local awareness for emotion recognition in textual dialog. *Engineering Applications of Artificial Intelligence*, 128:107530, February 2024.
- [3] Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. Dailydialog: A manually labelled multi-turn dialogue dataset, 2017.
- [4] Hao Liu, Runguo Wei, Geng Tu, Jiali Lin, Dazhi Jiang, and Erik Cambria. Knowing what and why: Causal emotion entailment for emotion recognition in conversations. *Expert Systems with Applications*, 274:126924, 2025.
- [5] Navonil Majumder, Soujanya Poria, Devamanyu Hazarika, Rada Mihalcea, Alexander Gelbukh, and Erik Cambria. Dialoguernn: An attentive rnn for emotion detection in conversations, 2019.
- [6] Tao Meng, Fuchen Zhang, Yuntao Shou, Hongen Shao, Wei Ai, and Keqin Li. Masked graph learning with recurrent alignment for multimodal emotion recognition in conversation, 2024.
- [7] Patrícia Pereira, Helena Moniz, and Joao Paulo Carvalho. Deep emotion recognition in textual conversations: A survey, 2024.
- [8] Soujanya Poria, Devamanyu Hazarika, Navonil Majumder, Gautam Naik, Erik Cambria, and Rada Mihalcea. Meld: A multimodal multi-party dataset for emotion recognition in conversations, 2019.
- [9] Xiangyu Qin, Zhiyu Wu, Jinshi Cui, Tingting Zhang, Yanran Li, Jian Luan, Bin Wang, and Li Wang. Bert-erc: Fine-tuning bert is enough for emotion recognition in conversation, 2023.
- [10] Armand Stricker and Patrick Paroubek. A unified approach to emotion detection and task-oriented dialogue modeling, 2024.
- [11] Xiaohan Wang, Linchao Zhu, and Yi Yang. T2vlad: Global-local sequence alignment for text-video retrieval, 2021.
- [12] Yan Wang, Bo Wang, Yachao Zhao, Dongming Zhao, Xiaojia Jin, Jijun Zhang, Ruifang He, and Yuexian Hou. Emotion recognition in conversation via dynamic personality. In Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue, editors, *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 5711–5722, Torino, Italia, May 2024. ELRA and ICCL.
- [13] Sayyed M. Zahiri and Jinho D. Choi. Emotion detection on tv show transcripts with sequence-based convolutional neural networks, 2017.

A Appendix