

Article

SAPBERT: Speaker-Aware Pretrained BERT for Emotion Recognition in Conversation

Seunguook Lim and Jihie Kim * 

Department of Artificial Intelligence, Dongguk University Seoul, 30 Pildong-ro 1-gil, Seoul 04620, Republic of Korea

* Correspondence: jihie.kim@dgu.edu; Tel.: +82-02-2260-4973

Abstract: Emotion recognition in conversation (ERC) is receiving more and more attention, as interactions between humans and machines increase in a variety of services such as chat-bot and virtual assistants. As emotional expressions within a conversation can heavily depend on the contextual information of the participating speakers, it is important to capture self-dependency and inter-speaker dynamics. In this study, we propose a new pre-trained model, SAPBERT, that learns to identify speakers in a conversation to capture the speaker-dependent contexts and address the ERC task. SAPBERT is pre-trained with three training objectives including Speaker Classification (SC), Masked Utterance Regression (MUR), and Last Utterance Generation (LUG). We investigate whether our pre-trained speaker-aware model can be leveraged for capturing speaker-dependent contexts for ERC tasks. Experiments show that our proposed approach outperforms baseline models through demonstrating the effectiveness and validity of our method.

Keywords: natural language processing; emotion recognition in conversation; dialogue modeling; pre-training; hierarchical BERT



Citation: Lim, S.; Kim, J. SAPBERT: Speaker-Aware Pretrained BERT for Emotion Recognition in Conversation. *Algorithms* **2023**, *16*, 8. <https://doi.org/10.3390/a16010008>

Academic Editors: Melania Susi and Alwin Poullose

Received: 6 November 2022

Revised: 13 December 2022

Accepted: 14 December 2022

Published: 22 December 2022



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Advance in artificial intelligence has increased the attention to the empathetic system and emotional interaction between the human and machine, especially in a conversation system such as a chat-bot and virtual assistant. However, it remains a challenge for both the machines and humans to detect an emotion within a conversation. In addition, it is even more difficult when only the conversation text data are available.

In ERC tasks, there have been many attempts based on deep-learning to understand human emotions with only text data. For better understanding the sequential contexts in conversations, Poria et al. [1] and Majumder et al. [2] proposed the RNN-based model. To alleviate the long-term dependency problems in an RNN-based model, a graph neural network (GNN)-based model was introduced [3]. To understand the context of conversations, an external commonsense knowledge was utilized [4–6].

Due to the remarkable improvement in pre-training language models in a variety of NLP tasks [7–10], there have been attempts to use a pre-training model for ERC tasks. Hazarika et al. [11] suggested that generative conversational models can be leveraged to transfer knowledge for the ERC task, while Jiao et al. [12] suggested the ConvCom (Conversation Completion) task as an effective pre-training method for the ERC task.

One of the unique properties of conversations is the nature of the participating speakers, and the ERC task requires capturing speaker-related contextual information: self-dependency (Figure 1) and inter-speaker dependency (Figure 2). Self-dependency means the aspect of emotional influence that speakers have on themselves, while inter-speaker dependency represents the emotional interaction among speakers during a conversation. Capturing this contextual information can be a main challenge for understanding conversations and addressing the ERC task. In this study, we propose a new pre-trained model,

SAPBERT (Speaker-Aware Pretrained BERT), composed of the hierarchical BERT with utterance-level and conversation-level BERT to address this challenge.

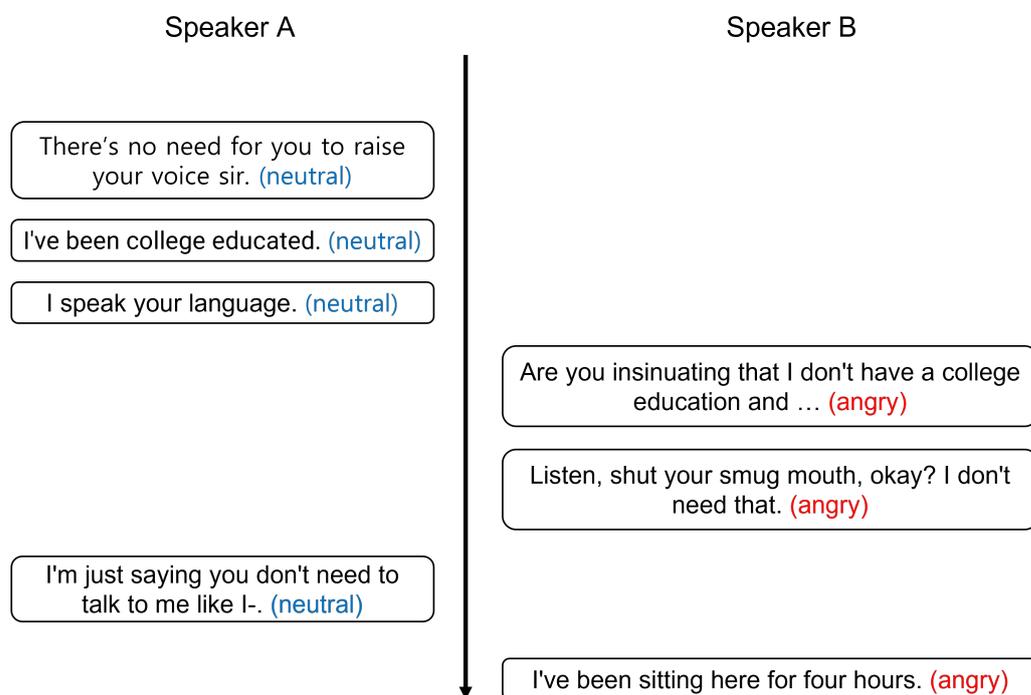


Figure 1. An example of self-dependency context. As shown in the figure, each speaker maintains a consistent emotional state regardless of the counterpart speaker’s emotional state or utterance.

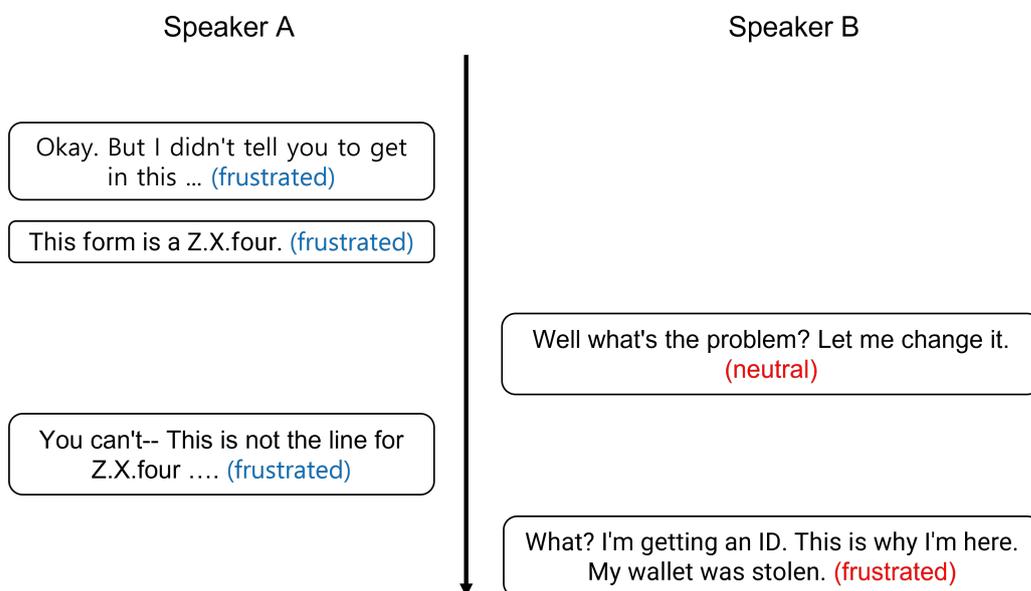


Figure 2. An example of inter-speaker dependency context. As shown in the figure, Speaker B’s emotional state shift from neutral to frustrated is triggered by Speaker A’s response.

For model’s better understanding of conversations, we aim to enhance model’s ability to capture the speaker-dependent contexts, the coherence of conversation and the whole conversation context through introducing three pre-training objectives: 1. Speaker Classification, 2. Masked Utterance Regression, and 3. Last Utterance Generation. We use the output of the SAPBERT’s pre-trained conversation-level BERT as the encoder of utterances and perform experiments on two representative ERC datasets, IEMOCAP [13]

and MELD [14]. The results show that our approach can be effective for understanding the conversation and can assist with the ERC task. In addition, we perform an ablation study on the pre-training objectives to demonstrate that the proposed approach, especially Speaker Classification, is effective.

In summary, our contributions are as follows:

- We present a new pre-training strategy for better understanding of the conversation: 1. Speaker Classification, 2. Masked Utterance Regression, and 3. Last Utterance Generation.
- We demonstrate that our pre-training strategy is effective for understanding conversations and can improve the ERC performance through experiments.

2. Related Work

2.1. Emotion Recognition in Conversation

With more accessibility to datasets such as IEMOCAP [13] and MELD [14] that have textual features of conversations, some researchers have introduced additional approaches to dealing with ERC with conversation text data. Poria et al. [1] proposed context LSTM [15] to understand contextual information of the conversation. Majumder et al. [2] introduced DialogueRNN to model the states of global, party (speaker, listener), and emotion with GRU [16] and classify the emotion by incorporating these states with the target utterance.

RNN-based models, however, show limited performances with long sequences in spite of the effectiveness of dealing with sequential data. This is called the long-term dependency problem. To mitigate this issue, a graph neural network (GNN)-based model, called DialogueGCN, was introduced [3] resulting in better performances compared to RNN-based models. Zhong et al. [4] proposed a hierarchical self-attention [17]-based model to alleviate the long-term dependency problem and applied external commonsense knowledge for enriching contextual information such as COSMIC [5] and KI-Net [6].

CESTa [18] treated the ERC task as sequence tagging through choosing the best tag sequence using CRF. DialogueCRN [19] proposed a contextual reasoning LSTM model to capture situation-level and speaker-level context and to integrate the emotional clues. DialogXL [20] employed XLNet [8] with enhanced memory and dialogue-aware self-attention.

2.2. Transfer Learning for ERC

To improve downstream conversation tasks such as ERC, pre-training with an objective such as masked-language modeling or next sentence prediction in BERT or permutation language modeling in XLNET can be used. Hazarika et al. [11] proposed that generative conversational models can be leveraged to transfer knowledge for the ERC task. Given contexts, the model's ability to capture the whole conversation context can be enhanced by training the model to generate a coherent next response utterance.

In addition, Jiao et al. [12] proposed the ConvCom (Conversation Completion) task as an effective pre-training method for the ERC task. ConvCom means selecting the correct answer from candidates to fill a masked utterance in a conversation. Such approach can help the model capture the coherent context of the conversation and also help the system use unlabeled conversation data for the training.

3. Conversation Types

Unlike regular documents or other textual data, conversations have a unique characteristic as a 'Speaker'.

3.1. Number of Speakers

Conversations can be classified according to the number of speakers: a dyadic conversation, which is a conversation between two people (Figures 3 and 4), and a multi-party conversation, which is a conversation between more than three speakers (Figure 5). In the real world, there is no limit to the number of speakers during conversations. The greater

the number of speakers in a conversation, the more difficult it could be for the model to capture speaker-dependent contexts.

A: Why does that bother you?
 B: She's been in New York three and a half years. Why all of the sudden?
 A: Maybe he just wanted to see her again?
 A: He lived next door to the girl all his life, why wouldn't he want to see her again?
 A: How do you know he is even thinking about it?

Figure 3. A one by one dyadic conversation.

A: I'm worried about something.
 B: What's that?
 A: Well, I have to drive to school for a meeting this morning, and I'm going to end up getting stuck in rush-hour traffic.
 B: That's annoying, but nothing to worry about. Just breathe deeply when you feel yourself getting upset.
 A: Ok, I'll try that.
 B: Is there anything else bothering you?
 A: Just one more thing. A school called me this morning to see if I could teach a few classes this weekend and I don't know what to do.
 B: Do you have any other plans this weekend?
 A: I'm supposed to work on a paper that'd due on Monday.
 B: Try not to take on more than you can handle.
 A: You're right. I probably should just work on my paper. Thanks!

Figure 4. A continuous dyadic conversation.

A: Ma'am, you forgot your phone.
 B: Oh, thanks, I couldn't live without this little thing.
 C: I know what you mean. It is of great significance to you. So did you enjoy your dinner?
 B: Oh yes, everything was just perfect. It's so hard to take the whole family out to eat, but your restaurant was perfect. Johnny had his own place to play in and I had time to talk with my sisters and their husbands
 A: Thanks for your compliment for the restaurant.

Figure 5. A multi-party conversation.

3.2. Continuity

Conversations can also be classified depending on the continuous utterances from the same speaker. A one by one dyadic conversation means that two speakers take turns one by one during the conversation, which makes it easy for the model to classify speakers with a certain order and capture the speaker-dependent contexts. When the speakers' turns are not fixed and some continuous utterances from the same speaker are in the conversation, it may be not an easy and simple task for model to classify speakers and capture the speaker-dependent contexts as a human.

We discovered that the composition of the speaker is different for each conversation, and we define three types of conversation depending on the composition of speakers in conversations: 1. One by one dyadic conversation, 2. Continuous dyadic conversation, and 3. Multi-party conversation.

Previous studies have tried to classify the speakers through physical methods such as speaker specific node or relation type on the graph neural network based models such as Ghosal et al. [3] or encoding utterances of each speaker separately by speaker-specific module as Majumder et al. [2], Shen et al. [20]. However, these methods can be inflexible to either the various number of speakers or the unpredictable turns of speakers in a conversation. Unlike the previous studies, we propose the self-supervised learning Speaker Classification to classify speakers only by the content of the conversation, not a fixed module. This can help our model capture speaker-dependent contexts and understand conversations better.

4. Methods and Materials

Conversations can be seen as a hierarchical structure. To model this aspect of conversation, we employ the hierarchical BERT architecture as our model. For better understanding the conversations, we assume that the model should be able to capture at least three aspects in a conversation: 1. speaker-dependent contexts, 2. coherence of the conversation, and 3. the whole conversation context. We propose three pre-training objectives to capture these aspects and enhance the ability of understanding conversations. We also adopt the multi-task learning for improving our pre-trained model.

We denote a dialogue with N utterances as $D = (u_1, \dots, u_N)$, context as $C = (u_1, \dots, u_{N-1})$ and response as $R = u_N$. Each $u_i = (w_1^i, \dots, w_M^i)$ is an utterance with M words.

4.1. Hierarchical BERT

Figure 6 shows that the two BERT models are hierarchically nested: one is utterance-level and the other is conversation-level context encoder.

We use the pre-trained RoBERTa model as an utterance-level encoder ule . The utterance-level encoder transforms an utterance into a list of vector representations and takes the first vector as a representation of the utterance, as other BERT-like models regarding the first vector as the embedding of the input sentence. The final representation of i -th utterance is c_i :

$$c_1, \dots, c_N = ule([u_1, \dots, u_N]) \quad (1)$$

The conversation-level encoder cle (BERT base model) transforms the sequence of utterance representations (c_1, c_2, \dots, c_N) into the sequence of context-encoded utterance representation $H = (h_1, h_2, \dots, h_N)$. H is the final product of the conversation-level context encoder with the hierarchical BERT:

$$h_1, \dots, h_N = cle([c_1, \dots, c_N]) \quad (2)$$

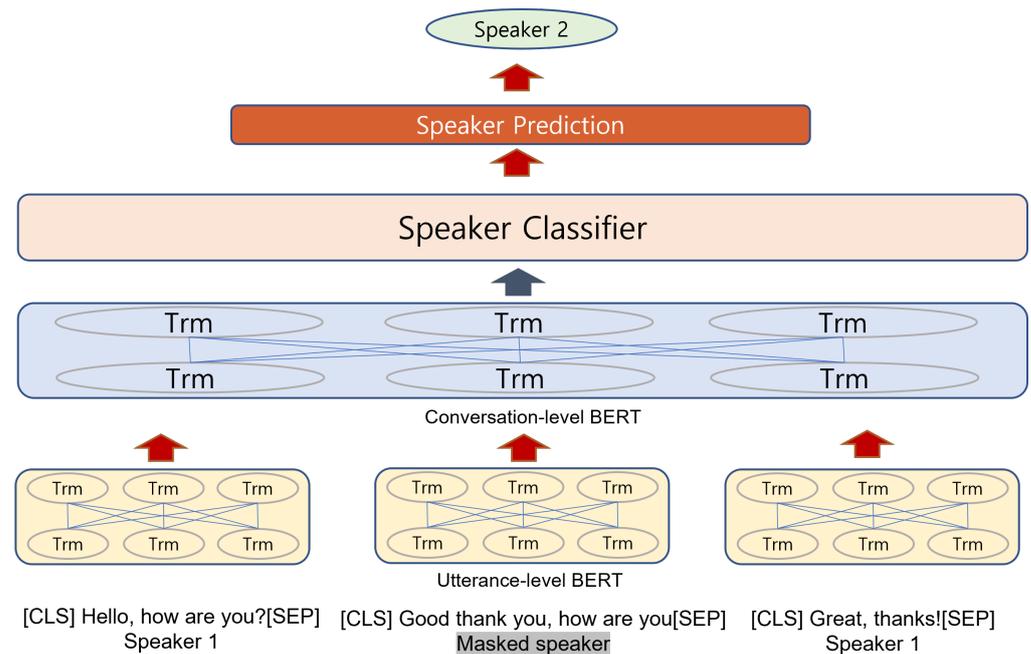


Figure 6. Overview of Speaker Classification. The model is trained to predict the true speaker on speaker-masked utterance using the context-encoded utterance and speaker classifier; TRM = Transformer Encoder.

4.2. Speaker Classification

Identifying speakers in dialogues determines the ability to capture these speaker-dependent contexts helping the system represent the dialogue contexts. Therefore, speaker-dependent contexts, such as self-dependency and inter-speaker dependency, are essential in understanding conversations.

The objective of Speaker Classification task (SC) is to classify the true speaker with only the context-encoded utterance which is randomly selected. When the number of speakers is two, the first utterance’s speaker is set as Speaker 1 and the other as Speaker 2.

The reason why only one utterance is randomly selected, not the whole utterances, is that we aim to avoid over-fitting to the utterance position. For example, if speakers take their turns just one by one, the model can classify speakers according to the utterance’s position, such as Speaker 1 at the positions of odd numbers and Speaker 2 at the positions of even numbers. In the example in Figure 6, the Speaker Classification model is trained to classify the speaker as Speaker 2 given the context-encoded representation h_2 of the utterance ‘Good thank you, how are you’:

$$\hat{s}_i = W_{SC}h_i + b_{SC} \tag{3}$$

Lastly, the Speaker Classification task aims to minimize the cross entropy loss between the speaker prediction \hat{s}_i and the true speaker s_i , where L is the total number of dialogues:

$$\mathcal{L}_{SC} = -\sum s_i \log(\hat{s}_i) \tag{4}$$

4.3. Masked Utterance Regression

In order to capture the coherence of conversation, we employ masked utterance regression task (MUR) (Figure 7), proposed by Gu et al. [21]. Like the masked LM in BERT, we expect this task to enhance the conversation-level context representation and to improve the coherence. Gu et al. [21] randomly selected one utterance in C. Then, they replaced the utterance with a mask token as [CLS, MASK, SEP] 80% of time, kept the 10%

of time unchanged, and replaced the rest of the time with a random utterance. Lastly, the representation of the masked utterance is reconstructed.

After masking a random u_{masked} in C , we obtain its contextual utterance representation H using our hierarchical BERT model. Then, restore the masked utterance representation h_{masked} back to the original utterance $c_{original}$ using the encoding converter, a fully connected neural network:

$$c_{\hat{masked}} = W_{MUR}h_{masked} + b_{MUR} \tag{5}$$

Finally, a Masked Utterance Regression task is trained for minimizing the mean squared error (MSE) between the prediction of masked utterances and their original utterance representations:

$$\mathcal{L}_{MUR} = \frac{1}{L} \sum (c_{\hat{masked}} - c_{original})^2 \tag{6}$$

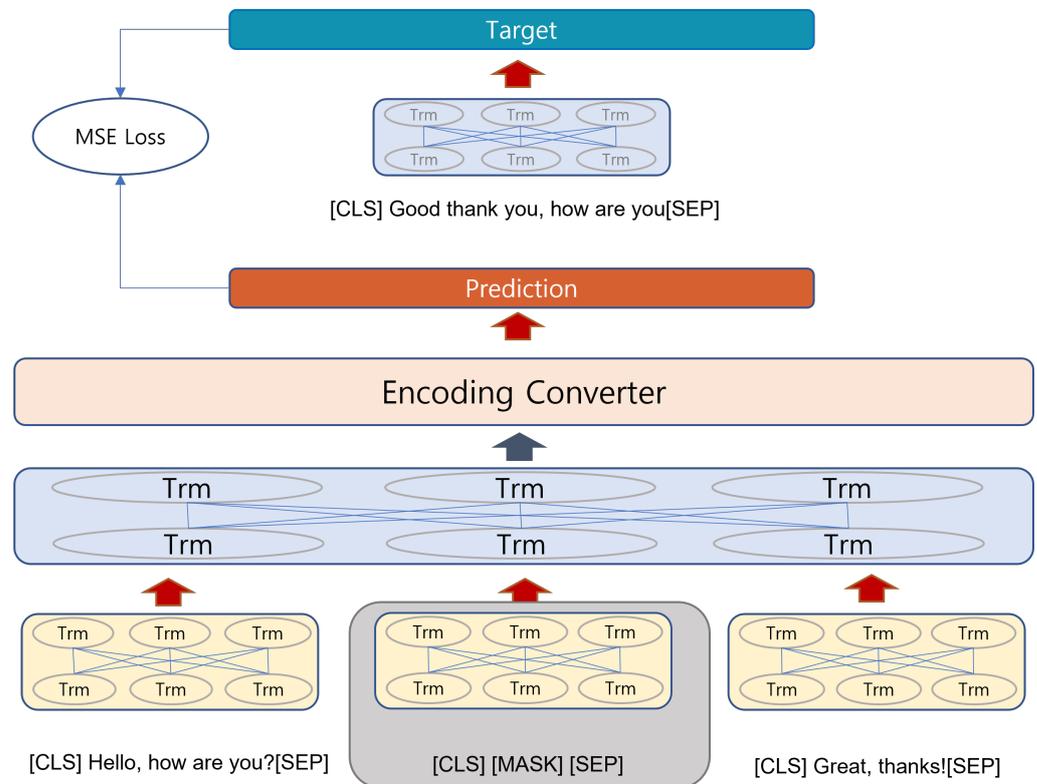


Figure 7. Overview of Masked Utterance Regression. The model estimates the masked utterance’s context-encoded representation c_{masked} and compare the c_{masked} with the real one produced by the utterance encoder.

4.4. Last Utterance Generation

In order to capture the whole dialogue context, we employ the Last Utterance Generation (LUG). We employed the generation task, inspired by Hazarika et al. [11], which proposed the effectiveness of transfer learning from the generative conversation model to the ERC task.

We expect that this task trains the model to catch the context in order from beginning to end of each dialogues. Given the dialogue context C , as shown in Figure 8, we first apply C to our model and obtain the context sensitive utterance representations H then generate response R , the last utterance in the dialogue, using a Transformer decoder. The decoder predicts each word of R , and lastly, the learning is driven by the cross entropy loss of the decoder:

$$\mathcal{L}_{LUG} = - \sum_{i=1}^N \log P_{decoder}(w_i^t | w_{<i}^t, H) \tag{7}$$

For the multi-task learning, our final objective is defined as the sum of the each loss function:

$$\mathcal{L}_{total} = \mathcal{L}_{SC} + \mathcal{L}_{MUR} + \mathcal{L}_{LUG} \tag{8}$$

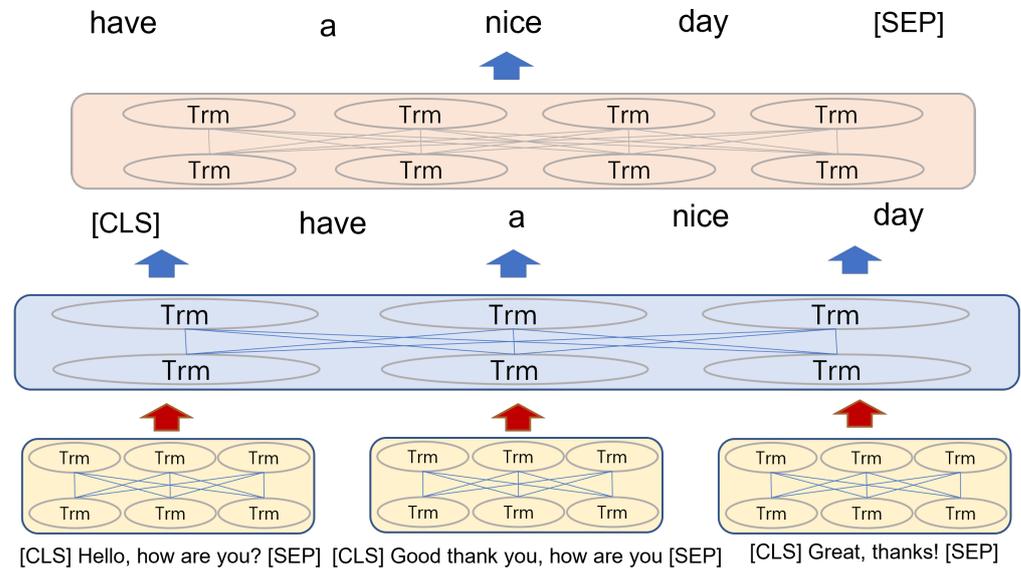


Figure 8. Overview of Last Utterance Generation. The hierarchical Transformer encoder–decoder architecture. As shown in this figure, the decoder takes the conversation-level encoded representations of contexts as an input and generates the response.

4.5. Dataset for Pre-Training

As shown in Table 1, we use two open-domain dialogue datasets in the pre-training phase: 1. Mutual [22] and 2. DailyDialog [23]. If the number of turns in the dialogue is less than 3, the data are filtered out.

Table 1. Dataset for Pre-training.

	Dialogues	Utterances
Mutual	7908	37,650
DailyDialog	13,088	102,141

Mutual (Multi-Turn dialogue Reasoning) [22] is dialogues from the Chinese student English listening comprehension exams. Mutual is composed of dialogue contexts and answer candidates. In the Mutual dataset, the answers of some dialogues are not provided; therefore, we do not include these data in our pre-training phase.

DailyDialog [23] is a widely used open-domain dialogue dataset. The dialogues in the DailyDialog dataset present daily communication and cover various topics in day to day life. DailyDialog is more formal than others constructed from conversations in social networks, which can be short and noisy.

5. Emotion Recognition in Conversation

5.1. ERC Model

Context Independent Feature Extraction We employ the RoBERTa base model, fine-tuned for emotion label classification to extract context independent utterance level feature vectors. The k -th utterance in the i -th dialogue with special tokens [CLS] and [SEP] is passed through the model, and the activation from the last layer corresponding to the [CLS] token is then used as an encoded utterance representation c :

$$c_{i,k} = RoBERTa([CLS], Utterance_{i,k}, [SEP]) \tag{9}$$

Dialogue Context Encoder We employ the conversation-level BERT from the pre-trained SAPBERT and apply the encoded utterance representations C in the i -th dialogue, resulting from Equation (9) as an input for obtaining the dialogue context encoded representations H :

$$H(h_{i,1}, h_{i,2}, \dots, h_{i,N}) = \text{SAPBERT}(c_{i,1}, c_{i,2} \dots c_{i,N}) \quad (10)$$

Emotion Classifier As shown in Figure 9, the dialogue context encoded representation h , resulting from Equation (10), is classified into emotion labels using a fully connected network for the emotion classifier (EC), and the probability is generated by a softmax layer:

$$y_{i,k}^{\hat{}} = \text{softmax}(W_{EC}h_{i,k} + b_{EC}) \quad (11)$$

Lastly, the cross entropy loss between the emotion prediction $y_{i,k}^{\hat{}}$ of the k -th utterance in the i -th dialogue and the ground-truth emotion of the utterance $y_{i,k}$ is used to train our ERC model:

$$\mathcal{L}_{EC} = -\frac{1}{\sum_{l=1}^L \tau(l)} \sum_{i=1}^L \sum_{k=1}^{\tau(i)} y_{i,k} \log(y_{i,k}^{\hat{}}) \quad (12)$$

where L is the total number of dialogues and $\tau(i)$ is the number of utterances in the i -th dialogue.

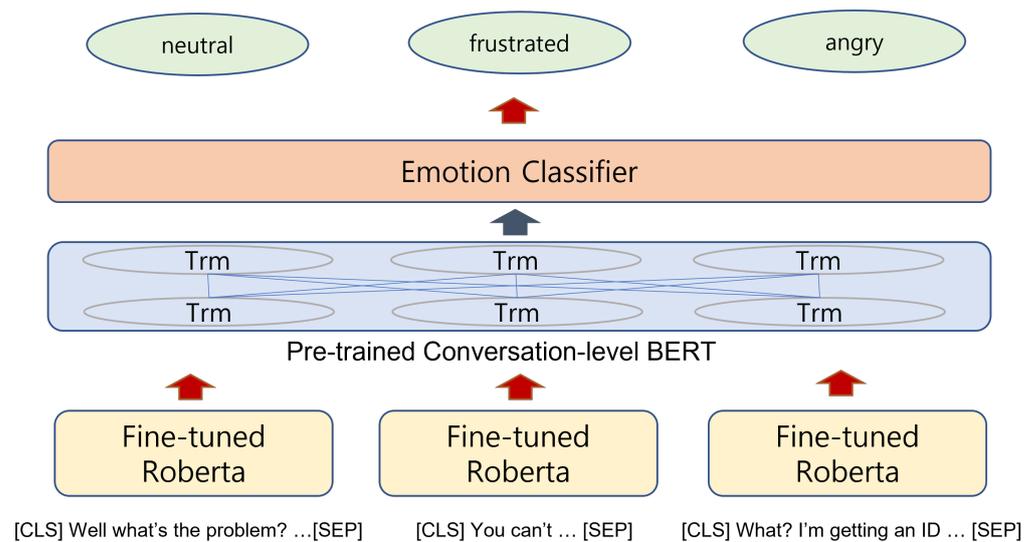


Figure 9. Overview of the ERC model.

5.2. Dataset for ERC

We evaluate our model with IEMOCAP [13] and MELD [14] datasets. The statistics of the datasets are reported in Tables 2 and 3. Both datasets are multi-modal datasets with textual, visual, and acoustic features. In this paper, we use only textual features only for ERC tasks.

Table 2. IEMOCAP Dataset Split.

IEMOCAP	Utterances	Dialogues
train + val	5810	120
test	1623	31

IEMOCAP [13] is an open-domain 2-speaker dialogue dataset for ERC tasks. The utterances are annotated with one of six emotion labels: happy, sad, neutral, angry, excited, and frustrated. We use the pre-defined train/test split provided in the IEMOCAP dataset.

We extract the validation set from the randomly shuffled training set with a ratio of 80:20 since no pre-defined validation set is provided.

MELD [14] is an open-domain multi-speaker dialogue dataset collected from a TV-series, "Friends", for an ERC task. MELD contains more than 1400 multiparty conversations and 13000 utterances. The utterances are annotated with one of seven emotion labels: anger, disgust, sadness, joy, surprise, fear and neutral. We use the pre-defined train, validation, and test set split provided by the MELD dataset.

Table 3. MELD dataset split.

MELD	Utterances	Dialogues
train	9989	1038
val	1109	114
test	1623	280

5.3. Baselines

We compare our proposed approach with the following baselines:

cLSTM [1]: Uses Bidirectional LSTM with the attention mechanism to capture the context from the surrounding utterance.

DialogueRNN [2]: Uses three GRUs to model speaker states, global contexts, and emotion context.

DialogueGCN [3]: Introduces a graph-based structure for better modeling of relations between utterances.

TL-ERC [11]: Uses the transfer learning strategy for emotion recognition during a conversation.

KET [4]: Employs a graph attention mechanism to combine commonsense knowledge into utterance representations.

AGHMN [24]: Introduces a hierarchical memory network architecture to store the dialogue context.

BiERU [25]: Introduces a party-ignorant bidirectional recurrent unit that used both sides of the conversational context for emotional predictions.

COSMIC [5]: Incorporates commonsense knowledge to learn the context of inter-speaker dependency.

DialogXL [20]: Employs XLNet with a dialog-aware self-attention to introduce the awareness of speaker-dependent contexts.

DialgCRN [19]: Introduces multi-turn reasoning modules to extract and integrate clues indicating emotion during a dialogue.

KI-Net [6]: Leverages commonsense knowledge and sentimental vocabulary in conversations to obtain more semantic information.

CESTa [18]: Employs the CRF algorithm to capture the emotions' pattern.

We employed 4-type models for comparing the results of our model: 1. pre-trained model [20], 2. knowledge model [4–6], 3. transfer learning model [11], and models that introduced a variety of methods and presented remarkable performances on Emotion Recognition in Conversation [1–3,18,19,24,25].

6. Results

Tables 4 and 5 show the results of our proposed model and other baseline models for the ERC task on the IEMOCAP [13] and MELD [14]. The overall results of our SAPBERT outperforms other models on both datasets except for COSMIC [5] on the MELD dataset.

Table 4. Categorized ERC results of the IEMOCAP and the MELD dataset; The percentage beside the category means the percentage of each category of IEMOCAP dataset, Acc. = Accuracy, F1 = F1-score, Average(w) = Weighted average, best performances are highlighted in bold.

Model	IEMOCAP												MELD			
	Happy (7.6%)		Sad (16.8%)		Neutral (23.2%)		Angry (11.5%)		Excited (18.1%)		Frustrated (22.7%)		Average (w)		Average (w)	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
c-LSTM	30.56	35.63	56.73	62.90	57.55	53.00	59.41	59.24	52.84	58.85	65.88	59.41	56.32	56.19	57.50	55.90
DialogueRNN	25.69	33.18	75.1	78.7	58.59	59.21	64.71	65.28	80.27	71.86	61.15	58.91	63.4	62.75	56.1	55.9
DialgoueGCN	40.62	42.75	89.14	84.54	61.92	63.54	67.53	64.19	65.46	63.08	64.18	66.99	65.25	64.18		58.1
AGHMN	48.3	52.1	68.3	73.3	61.6	58.5	57.5	61.9	68.1	69.7	67.1	62.3	63.5	63.5	59.5	57.5
BiERU	54.24	31.53	80.6	84.21	64.67	60.17	67.92	65.65	62.79	74.07	61.93	61.27	66.11	64.65		60.84
KI-Net	-	49.45	-	73.38	-	65.63	-	65.13	-	71.15	-	68.38	-	66.98	-	63.24
SAPBERT	57.26	52.99	76.19	80.31	66.31	65.70	58.82	61.62	73.47	72.85	64.67	63.55	67.34	67.16	64.44	64.28

Table 5. Comparisons with baselines and our method. Avg(w) refers to weighted average, Acc refers to the accuracy and F1 refers to the f1-score. Best performances are highlighted in bold.

Models	IEMOCAP		MELD
	Avg (w) Acc.	Avg (w) F1.	Avg (w) F1.
TL-ERC	-	59.30	-
KET	-	59.56	58.18
COSMIC	-	65.28	65.21
DialogXL	-	65.94	62.41
DialogueCRN	66.05	66.20	58.39
CESTa	-	67.10	58.36
Our Model	67.34	67.16	64.28

6.1. Implementation Details

We employed a base-sized BERT as the utterance-level encoder and RoBERTa model as the conversation-level model (L = 12, H = 768, A = 12). We chose a base-sized BERT and RoBERTa model since Mutual and DailyDialog datasets cannot be large enough to train a large-sized BERT and RoBERTa without over-fitting. We filtered the dialogues and less than three utterances for each pre-training method can be applied on different utterances. We used AdamW as an optimizer with a learning rate of 5×10^{-5} . We implemented all the experiments including modeling with the PyTorch library. The pre-training process took place on Ubuntu 16.04 and eight RTX-2080 Ti GPUs, and the ERC process took place on Window 10 and a single RTX-2080 Ti GPU.

6.2. Experimental Results

As shown in Table 4, our model shows the best performance in just two emotions: happy and neutral, but with the best score of both accuracy and f1-score on average. This may indicate that our proposed model is not specialized in capturing only specific emotions, but overall emotions evenly well.

Unlike other models, our model is trained to distinguish the speakers in a dialogue. Therefore, our model may be able to capture both speaker-dependent contexts and the context of emotion such as inter-speaker dependency or self-dependency better than other baselines.

TL-ERC [11] shows the efficiency of transfer learning from the generative task to the ERC task. As shown in Table 5, our model achieves approximately 8% better in F-1 score than TL-ERC with the IEMOCAP dataset. This demonstrates the effectiveness of the multi-task learning.

As shown in Tables 4 and 5, our model achieves approximately 8%, 2% and 0.2% better than each commonsense knowledge based models, KET [4], COSMIC [5], KI-Net [6], on the IEMOCAP dataset. This indicates that capturing speaker-dependent context can be important.

For the MELD dataset, COSMIC [5] shows a little better performance. This may be attributed to the structural characteristics of MELD conversation. The number of speakers in each conversation is large (up to 9) and the average length of conversations is 10 in the MELD dataset, while in IEMOCAP the average length of conversations is 50 and the number of speakers is 2. This means that MELD has very few utterances per speaker in each conversation for our model to capture speaker-dependent contexts.

Our model gains approximately 2.5% F-1 score improvement on the IEMOCAP dataset over BiERU [25], as shown in Table 4. This indicates the importance of capturing the speaker-dependent contexts. Approximately a 1% improvement in the F-1 score on the IEMOCAP dataset over DialogXL [20] and DialogueCRN [19] demonstrates that Speaker Classification can be effective for capturing the speaker-dependent contexts.

As shown in Table 5, our model achieves approximately 6% better on the MELD dataset and 0.1% better on the IEMOCAP dataset compared to the CESTa [18]. This can be attributed to the capturing of the speaker-dependent contexts during a dialogue.

As shown in Table 6, the performance of our model without the pre-training of Speaker Classification is lower than some of the baseline models such as DialogueGCN [3], AGHMN [24], BiERU [25], KI-Net [6], COSMIC [5], DialogXL [20], DialogueCRN [19] and CESTa [18]. This indicates that our model's improvement may not be due to the difference between the model structures.

7. Discussion

7.1. Error Analysis

Figures 10 and 11 show the heat map of the confusion matrix of our result on the IEMOCAP dataset and the MELD dataset, respectively. The X-coordinate indicates our model's prediction and the y-coordinate indicates the ground truth.

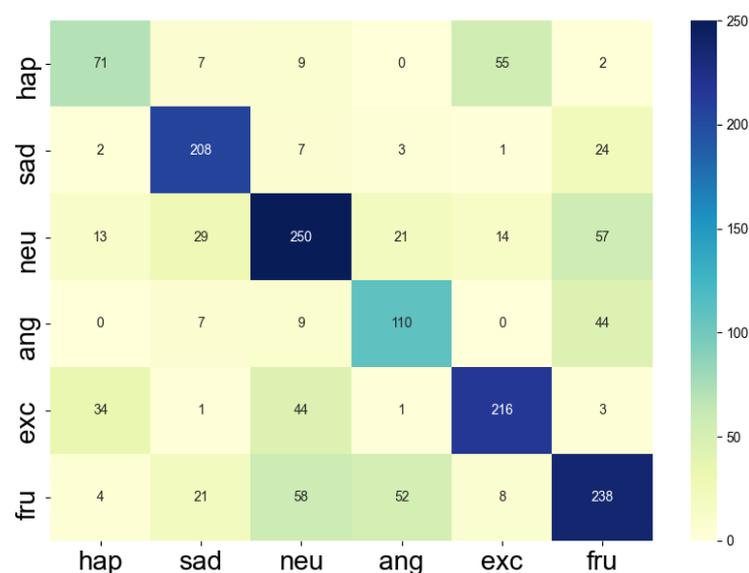


Figure 10. The heatmap of the confusion matrix on IEMOCAP dataset; hap = happy, neu = neutral, ang = angry, exc = excited, fru = frustrated. X-coordinate: model's prediction and y-coordinate: the ground truth.

With the IEMOCAP dataset, there are some misclassifications among similar emotions such as happy–excited, frustrated–angry and frustrated–sad. Our model is confused especially among sad, angry, and frustrated, which is consistent with the results shown in Table 4.

With the MELD dataset, we faced a challenge of classifying categories that have relatively small amounts of data such as sad, joy, and fear.

For the misclassification problem, the similar utterances' representation is a possible reason. It could be helpful to extract latent variables from utterances to distinguish the utterances with similar emotions, and the label smoothing technique could be one of the solutions alleviating the problem.

The classification power for each categories of the proposed model highly depends on the number of data on each respective categories. For the class imbalance problem, it can be alleviated by either data augmentation or adjusting the sample according to the number of data per emotions for the future work.

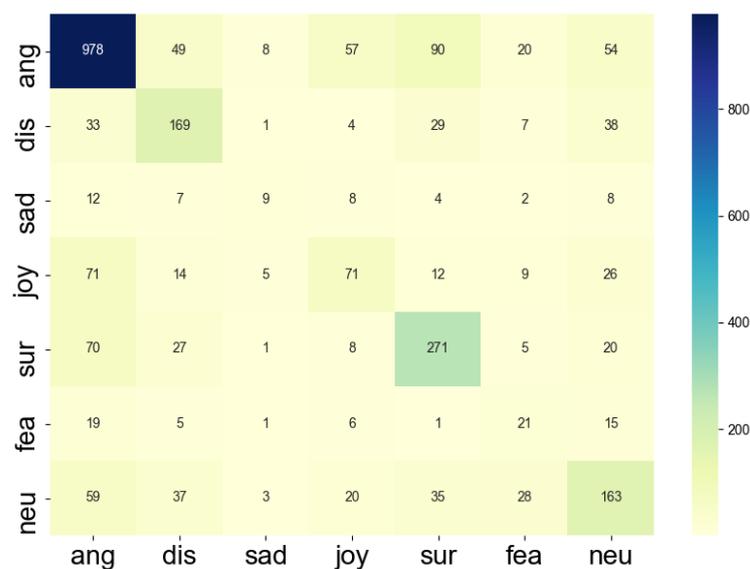


Figure 11. The heatmap of the confusion matrix on MELD dataset; ang = angry, dis = disgusting, sur = surprise, fea = fear, neu = neutral. X-coordinate: model's prediction and y-coordinate: the ground truth.

7.2. Ablation Study

In this ablation study, we analyze the effects of each pre-training objective. '-' denotes the model pre-trained without the objectives. For example, '- Speaker Classification' means the SAPBERT pre-trained with Masked Utterance Regression and Last Utterance Generation tasks only.

As shown in Table 6, the performance drops whenever any of the pre-training objective is excluded. When the Masked Utterance Regression and Last Utterance Generation objectives are excluded individually, the performance drops by 2.71%, 1.9%, (MUR) and 3.01%, 2.17% (LUG) on both datasets. This means that these objectives can be effective for capturing the both coherence and the whole context of conversations. In particular, when the Speaking Classification objective is excluded, the performance on the IEMOCAP dataset drops considerably by 4.38%.

Table 6. Results of ablation study on IEMOCAP and MELD.

Method	F-1 Score	
	IEMOCAP	MELD
SAPBERT	67.16	64.18
- Speaker Classification (SC)	62.78 (↓4.38)	63.09 (↓1.09)
- Masked Utterance Regression (MUR)	64.45 (↓2.71)	61.17 (↓3.01)
- Last Utterance Generation (LUG)	65.26 (↓1.9)	62.01 (↓2.17)

The datasets for the pre-training and for the ERC task present different patterns. All the conversations in Mutual and DailyDialog datasets for the pre-training contain two-speaker conversations with one-by-one turn taking. On the other hand, IEMOCAP contains continuous utterances of the same speaker in some cases, while MELD has multi-speaker conversations with relatively few utterances per speaker. The performance of the SC excluded model does not drop with MELD as much as with IEMOCAP. This can be attributed to the difference in the conversation pattern and the number of speakers between the pre-training datasets and MELD. The result of the ablation study for Speaker Classification indicates that the model is well trained for our purposes, not overfitting the position of utterances in the conversation.

8. Conclusions

In this paper, we introduced the SAPBERT: Speaker-Aware Pretrained BERT for Emotion Recognition in Conversation. Our objective was to enhance the model's ability to capture three conversational contexts for the model's better understanding of conversations: 1. speaker-dependent contexts, 2. the coherence of conversation, and 3. the whole conversation context. To improve the ability to capture these contexts in the conversation, we proposed the multi-task learning with three pre-training objectives: 1. Speaker Classification, 2. Masked Utterance Regression, and 3. Last Utterance Generation. Extensive experiments were conducted on two ERC benchmarks, and the results show that our model outperforms almost all other baselines on the ERC datasets (IEMOCAP and MELD). In addition, the ablation study demonstrates the effectiveness of the proposed objectives for capturing conversational contexts.

Author Contributions: Conceptualization, S.L. and J.K.; methodology, S.L. and J.K.; experiment, S.L.; validation, S.L.; formal analysis, S.L.; Writing—original draft, S.L.; Writing—review and editing, S.L. and J.K.; visualization, S.L.; supervision, J.K.; project administration, J.K.; funding acquisition, J.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the MSIT (Ministry of Science, ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2022-2020-0-01789) (50%) and under the High-Potential Individuals Global Training Program (RS-2022-00155054) (50%) supervised by the IITP (Institute for Information and Communications Technology Planning and Evaluation).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Publicly available datasets were used in this study. The datasets can be found here: (1) IEMOCAP (<https://sail.usc.edu/iemocap/>), (2) MELD (<https://affective-meld.github.io/>), (3) Mutual (<https://github.com/Nealcly/MuTual/tree/master/data>), and (4) DailyDialog (<http://yanran.li/dailydialog> the access date 1 November 2022).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Poria, S.; Cambria, E.; Hazarika, D.; Majumder, N.; Zadeh, A.; Morency, L.P. Context-Dependent Sentiment Analysis in User-Generated Videos. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Vancouver, BC, Canada, 30 July–4 August 2017; Association for Computational Linguistics: Stroudsburg, PA, USA, 2017; pp. 873–883. [CrossRef]
2. Majumder, N.; Poria, S.; Hazarika, D.; Mihalcea, R.; Gelbukh, A.; Cambria, E. Dialoguernn: An attentive rnn for emotion detection in conversations. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 6818–6825.
3. Ghosal, D.; Majumder, N.; Poria, S.; Chhaya, N.; Gelbukh, A.F. DialogueGCN: A Graph Convolutional Neural Network for Emotion Recognition in Conversation. *arXiv* **2019**, arXiv:1908.11540.
4. Zhong, P.; Wang, D.; Miao, C. Knowledge-Enriched Transformer for Emotion Detection in Textual Conversations. *arXiv* **2019**, arXiv:1909.10681.
5. Ghosal, D.; Majumder, N.; Gelbukh, A.; Mihalcea, R.; Poria, S. COSMIC: COMmonSense knowledge for eMotion Identification in Conversations. *arXiv* **2020**, arXiv:2010.02795.
6. Xie, Y.; Yang, K.; Sun, C.J.; Liu, B.; Ji, Z. Knowledge-Interactive Network with Sentiment Polarity Intensity-Aware Multi-Task Learning for Emotion Recognition in Conversations. In Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2021, Virtual Event, 7–11 November 2021; pp. 2879–2889.
7. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
8. Yang, Z.; Dai, Z.; Yang, Y.; Carbonell, J.; Salakhutdinov, R.R.; Le, Q.V. Xlnet: Generalized autoregressive pretraining for language understanding. In Proceedings of the Advances in Neural Information Processing Systems: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, Vancouver, BC, Canada, 8–14 December 2019; Volume 32.
9. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv* **2019**, arXiv:1907.11692.
10. Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I. Language models are unsupervised multitask learners. *OpenAI blog* **2019**, *1*, 9.

11. Hazarika, D.; Poria, S.; Zimmermann, R.; Mihalcea, R. Emotion recognition in conversations with transfer learning from generative conversation modeling. *arXiv* **2019**, arXiv:1910.04980.
12. Jiao, W.; Lyu, M.R.; King, I. Exploiting Unsupervised Data for Emotion Recognition in Conversations. *arXiv* **2020**, arXiv:2010.01908.
13. Busso, C.; Bulut, M.; Lee, C.C.; Kazemzadeh, A.; Mower, E.; Kim, S.; Chang, J.N.; Lee, S.; Narayanan, S.S. IEMOCAP: Interactive emotional dyadic motion capture database. *Lang. Resour. Eval.* **2008**, *42*, 335–359. [[CrossRef](#)]
14. Poria, S.; Hazarika, D.; Majumder, N.; Naik, G.; Cambria, E.; Mihalcea, R. Meld: A multimodal multi-party dataset for emotion recognition in conversations. *arXiv* **2018**, arXiv:1810.02508.
15. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [[CrossRef](#)] [[PubMed](#)]
16. Chung, J.; Gülçehre, Ç.; Cho, K.; Bengio, Y. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *arXiv* **2014**, arXiv:1412.3555.
17. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. In Proceedings of the Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, Long Beach, CA, USA, 4–9 December 2017.
18. Wang, Y.; Zhang, J.; Ma, J.; Wang, S.; Xiao, J. Contextualized Emotion Recognition in Conversation as Sequence Tagging. In Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue, 1st Virtual Meeting, 1–3 July 2020; Association for Computational Linguistics: Stroudsburg, PA, USA, 2020; pp. 186–195.
19. Hu, D.; Wei, L.; Huai, X. DialogueCRN: Contextual Reasoning Networks for Emotion Recognition in Conversations. *arXiv* **2021**, arXiv:2106.01978.
20. Shen, W.; Chen, J.; Quan, X.; Xie, Z. DialogXL: All-in-one XLNet for multi-party conversation emotion recognition. *arXiv* **2020**, arXiv:2012.08695.
21. Gu, X.; Yoo, K.M.; Ha, J.W. Dialogbert: Discourse-aware response generation via learning to recover and rank utterances. *arXiv* **2020**, arXiv:2012.01775.
22. Cui, L.; Wu, Y.; Liu, S.; Zhang, Y.; Zhou, M. MuTual: A dataset for multi-turn dialogue reasoning. *arXiv* **2020**, arXiv:2004.04494.
23. Li, Y.; Su, H.; Shen, X.; Li, W.; Cao, Z.; Niu, S. Dailydialog: A manually labelled multi-turn dialogue dataset. *arXiv* **2017**, arXiv:1710.03957.
24. Jiao, W.; Lyu, M.; King, I. Real-time emotion recognition via attention gated hierarchical memory network. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 8002–8009.
25. Li, W.; Shao, W.; Ji, S.; Cambria, E. BiERU: Bidirectional emotional recurrent unit for conversational sentiment analysis. *Neurocomputing* **2022**, *467*, 73–82. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.