A Graph Neural Network with Context Filtering and Feature Correction for Conversational Emotion Recognition

Chenquan Gan[a,b], Jiahao Zheng[a], Qingyi Zhu[b], Deepak Kumar Jain[c], Vitomir Štruc[d,*]

[a]*School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China*
[b]*School of Cyber Security and Information Law, Chongqing University of Posts and Telecommunications, Chongqing 400065, China*
[c]*Key Laboratory of Intelligent Control and Optimization for Industrial Equipment of Ministry of Education, Dalian University of Technology, Dalian, 116024, China*
[d]*Faculty of Electrical Engineering, University of Ljubljana, Trzaska cesta 25, SI-1000 Ljubljana*

Abstract

Conversational emotion recognition represents an important machine-learning problem with a wide variety of deployment possibilities. The key challenge in this area is how to properly capture the key conversational aspects that facilitate reliable emotion recognition, including utterance semantics, temporal order, informative contextual cues, speaker interactions as well as other relevant factors. In this paper, we present a novel Graph Neural Network approach for conversational emotion recognition at the utterance level. Our method addresses the outlined challenges and represents conversations in the form of graph structures that naturally encode temporal order, speaker dependencies, and even long-distance context. To efficiently capture the semantic content of the conversations, we leverage the zero-shot feature-extraction capabilities of pre-trained large-scale language models and then integrate two key contributions into the graph neural network to ensure competitive recognition results. The first is a novel *context filter* that establishes meaningful utterance dependencies for the graph construction procedure and removes low-relevance and uninformative utterances from being used as a source of contextual information for the recognition task. The second contribution is a *feature-correction* procedure that adjusts the information content in the generated feature representations through a gating mechanism to improve their discriminative power and reduce emotion-prediction errors. We conduct extensive experiments on four commonly used conversational datasets, i.e., IEMOCAP, MELD, Dailydialog, and EmoryNLP, to demonstrate the capabilities of the developed graph neural network with context filtering and error-correction capabilities. The results of the experiments point to highly promising performance, especially when compared to state-of-the-art competitors from the literature.

Keywords: Conversational emotion recognition, context filter, feature correction, graph network

*Corresponding author
Email addresses: gcq2010cqu@163.com (Chenquan Gan), s210101201@stu.cqupt.edu.cn (Jiahao Zheng), zhuqy@cqupt.edu.cn (Qingyi Zhu), deepak@cqupt.edu.cn (Deepak Kumar Jain), vitomir.struc@fe.uni-lj.si (Vitomir Štruc)

1. Introduction

Conversational emotion recognition is the process of recognizing the expressed emotion in the utterances of a conversation and represents a highly active research area within the machine-learning and natural-language processing communities. Techniques for conversational emotion recognition can be applied to real-time conversation systems to assist machines in analyzing the affective state of speakers and in a wide range of applications in healthcare systems [1], automatic driving [2] among others [3, 4].

Conversational emotion recognition is a complex task that differs from traditional text-based emotion recognition in that it is influenced by a variety of factors. For example, the same utterance can convey different emotions, depending on the context under which it was uttered and the speaker(s) involved in the conversation. Additionally, research in psychology [5] suggests that there are two critical factors that induce emotional changes in speakers during a conversation: *self-dependence* and *inter-speaker dependence*. Self-dependence refers to the speakers themselves affecting their emotions, while inter-speaker dependence refers to the emotions of the different parties in a conversation impacting each other. These factors make the emotion conveyed in the dialogue uncertain. At the same time, the need to effectively integrate various sources information and the requirement of real-time applications makes emotion recognition in dialogues a challenging task [6, 7, 8].

To examine the impact of context and speakers on emotion recognition, a considerable body of work has employed recurrent neural networks (RNNs) due to their ability to capture the temporal order of the utterances within conversations, retain historical context, and account for distinct speakers [9, 10]. However, because of the inherent limitations of RNNs, RNN-based techniques often struggle in modeling long-distance contextual information, even if the temporal order of the conversation and the impact of short-term context are both taken into account. The technique based on the self-attention mechanism and position encoding can effectively solve the problem of capturing remote context clues. Although some studies [11, 12] attempted to infuse commonsense knowledge into the conversation-modeling procedure to improve the language understanding ability of the recognition techniques, these methods complicate conversation modeling.

The emergence of graph neural networks (GNNs) and their variants [13, 14] alleviated the problem of the long-term dependencies to some extent. Due to the powerful ability of GNNs to process associative data, an increasing amount of research effort is being directed toward GNNs for conversational emotion recognition. Recent studies [15, 16], for example, have achieved highly competitive results by combining GNNs and pre-trained language models with context modeling to better understand the semantic and syntactic information in the given conversations.

GNN-based methods represent conversations in the form of graphs, with nodes representing utterances and edges representing utterance relationships. Psychological concepts such as self-dependence and inter-speaker dependence can naturally be captured through the speaker-specific utterance relationships (edges) and the semantic similarity between utterances may be utilized to initialize the edge weights in the graph with the goal of modeling conversational context. However, establishing dependencies between all utterances in a conversation (through a fully connected graph) typically leads to associations with weakly relevant or even irrelevant contextual information that adversely affect performance [17, 18], whereas considering too few dependencies results in improper context modeling and consequently poorly performing recognition models. However, these GNN-based methods still ignore another potentially critical factor for conversation modeling, i.e., *informativeness* [19]. Intuitively, if an utterance has a high enough information content to significantly change the cognition of others, causing emotional changes, the utterance should be considered appropriate for emotion recognition. Conversely, there are often utterances present in a conversation that lack a clear emotional tendency and make little contribution to the perception of emotions [20]. Considering such utterances as a source of context not only wastes computational resources, but also introduces noise into the inference process.

In addition to the potential for introducing noise into the model when establishing utterance dependencies, GNN-based models are also susceptible to noise during the learning process and error propagation from the early preprocessing steps applied to the given conversation. These issues are eventually reflected in the computed feature representations and their discriminative power for the emotion recognition task. To address this problem, Lian *et al.* [21] used graph convolutional neural networks to capture interlocutor interactions to correct some feature errors and adopted bidirectional GRUs and multi-head attention mechanisms to correct some errors due to contextual understanding. To the best of our knowledge, at present, no other work has attempted to design error-correction mechanisms for conversational emotion recognition.

Based on the above discussions, we propose in this paper a novel graph neural network for conversational emotion recognition with context-denoising and error-correction capabilities. To reduce the noise introduced into the graph-construction procedure by the context-modeling process, a *context filter* is designed to establish meaningful dependencies between the utterances of a conversation. Specifically, semantic correlations and context informativeness are considered during filtering so that only the most relevant and mutually informative utterances are connected in the graph structure of the GNN. This process not only avoids the loss of long-distance contextual information but also reduces the impact of noisy (i.e., irrelevant and uninformative) contextual cues on the model's performance. Additionally, we also propose a novel *feature-correction* procedure to further improve results. The feature-correction procedure first integrates the semantic features, extracted from the conversation by a pre-trained language model, and the emotion features, calculated from a graph neural network, into a fused representation and then corrects the information content in the fused representation through a dedicated gating mechanism. To demonstrate the capabilities of the proposed model, comprehensive experiments are performed on four commonly used conversational datasets, i.e., IEMOCAP, MELD, Dailydialog, and EmoryNLP. The experimental results show, that our model achieves highly promising results and compares favorably to the state-of-the-art.

In summary, the main contributions of this paper are as follows:

- We propose a novel (state-of-the-art) graph neural network for conversational emotion recognition that is capable of considering select contextual information and integrates a feature-correction mechanism that improves the computed feature representations and, in turn, reduces prediction errors. To facilitate reproducibility, we make the source code of our model publicly available.
- We design a context filter that focuses on semantic relevance as well as informativeness when establishing dependencies between the utterances of a conversation. The filter, thus, removes context that is not relevant or uninformative from the emotion inference task, leading to better overall performance.
- We introduce a feature-correction mechanism to further reduce prediction errors in conversational emotion recognition. The feature correction is learned end-to-end in our model and is shown to be beneficial for the emotion recognition task.

2. Related work

In recent years, there has been considerable interest in the problem of recognizing emotions in conversations, leading to a significant amount of research in this field [22, 8]. While impressive progress has been made, challenges associated with modeling semantic information, context, and speaker dependencies still require further research.

Recurrent Neural Networks (RNNs) have emerged as a promising research direction to address these challenges. For instance, Hazarika *et al.* [9] employed gate recurrent units to memorize historical information for each speaker separately, thus facilitating emotion recognition. Majumder *et al.* [10] proposed DialogueRNN, which utilizes speaker memory units and multilevel RNNs to model speakers and simulate the flow of emotions between them. Gan *et al.* [23] described a hierarchical feature interactive fusion network that integrates fine-grained emotion and act/intent information into utterance features while retaining temporal and contextual information. Zhang *et al.* [24] added “confidence gates” in front of each LSTM hidden cell to determine the trustworthiness of the previous speaker, simulating the emotional impact of the previous speaker. However, RNNs are known to suffer from information loss during the propagation of data representations, resulting in an incomplete understanding of contextual information by the emotion recognition model.

To gain a comprehensive understanding of utterance emotions, techniques for incorporating external knowledge into the recognition models were also explored in the literature. Ghosal *et al.* [11], for example, proposed an external knowledge base to comprehend the commonsense information present in the utterances, including psychological, event, and causal relationships, and to learn dependencies between speakers. More recently, the Transformer architecture has gained traction in dialogic emotion analysis due to its ability to effectively model long sequences and efficient parallel computing. BERT [25], a pre-trained transformer-based language model, has shown great efficacy in encoding semantic and grammatical information from diverse conversations. This model, trained on large language corpora, exhibits impressive zero-shot feature extraction capabilities that can be further enhanced through task-specific fine-tuning for various downstream tasks. Li et al. [26] proposed an emotion capsule structure based on the Transformer for multimodal dialogue emotion analysis, referred to as Emoformer. This structure integrates emotion vectors from three modalities and has achieved state-of-the-art results in multimodal dialogue emotion analysis. Liang et al. [27] combined the Transformer and graph neural network to introduce the position-aware Graph Neural Network (GNN). They designed a two-stream conversation converter to extract the contextual features of each interlocutor separately and then constructed a graph structure based on chronological order.

In recent studies, Zhang et al. [28, 29] employed multi-task learning frameworks to model the contextual dependencies and interactions among multiple modalities simultaneously. They leveraged the shared knowledge across tasks and captured task correlations through a multi-task co-learning approach. Yang *et al.* [30] and Song *et al.* [31] introduced curriculum learning into conversational emotion recognition to show that the order of training data affects model performance. Song *et al.* [31] also designed an adversarial contrast learning method to learn more contextual features and improve the robustness of the model. Researchers have also focused on dynamically modeling emotional changes during conversations. Song et al. [15] utilized a BERT-like model to encode utterances for conversational emotion recognition. They employed a question-answering framework to incorporate modalities and capture emotion changes using a conditional random field (CRF), achieving competitive performance. Furthermore, when a speaker possesses a pronounced personal speaking style, the emotional categories identified by the model may display inherent biases. Wang *et al.* [32] introduced the SIMR framework to attenuate such effects, whereas Liang *et al.* [27] directly integrated personal style attributes into the discourse features during the modeling process.

Graph neural networks (GNNs) are capable of associative data processing, which makes them useful for modeling conversational emotions since conversations are a collection of associative utterances. Several models have used GNNs to identify conversational emotions. Schlichtkru *et al.* [33] proposed

RGCN, the first work to apply GNN to model associative data. Ghosal *et al.* [17] introduced DialogueGCN, which entailed the construction of fully connected graphs corresponding to conversations. The approach considered distinct speakers and the temporal order of utterances, effectively addressing the challenge of propagating contextual information over long distances. However, because the number of graph nodes depends on the number of utterances in a conversation, longer conversations lead to an increase of graph nodes, and more importantly an exponential increase in graph edges, which can result in excessive memory usage and overfitting. To solve this problem, Ishiwatari *et al.* [18] extended the attention mechanism to the relational graph so that the weights between nodes can be dynamically adjusted. Shen *et al.* [34] considered the effect of past utterances and proposed a method for information transfer between different layers, in which nodes can access both node information of the previous layer and the current neighboring node information. Shou *et al.* [35] combined speaker relationships and dependent syntactic structures to model conversation based on GNNs, which improved the ability to acquire semantic information and understand utterance syntax.

Unfortunately, the GNN-based methods discussed above establish utterance dependencies under fixed windows (i.e., under fixed utterance vicinity) and are therefore still susceptible to considering the context that is only weakly related or even irrelevant. Additionally, these methods ignore the informativeness of the utterances when establishing contextual dependencies in conversations. Finch *et al.* [19] evaluated the quality of a conversation across eight dimensions, and found both relevance and informativeness to be crucial dimensions for a comprehensive understanding of various aspects of a conversation. According to recent psychological insights [36], informativeness is also closely related to emotional response, indicating that research on how context affects emotion should take informativeness into account as well. In conversations, there are often utterances with low informativeness and no apparent emotional tendency, and including such contextual information in the recognition task is expected to increase the computation effort as well as introduce noise.

To address the above-mentioned issues, we propose in this paper a novel GNN-based approach for conversational emotion recognition that explicitly considers informativeness when defining contextual dependencies between the utterances of a conversation. To the best of our knowledge, our work is the first to incorporate this key aspect into the conversation-modeling procedure.

3. The proposed method

The main contribution of this work is a novel graph neural network with context denoising and feature-correction capabilities, designed for the task of utterance-level emotion recognition in conversations. As can be seen from the high-level overview in Figure 1, the proposed model consists of five main components that aim at: (*i*) preprocessing, (*ii*) context filtering, (*iii*) graph processing, (*iv*)

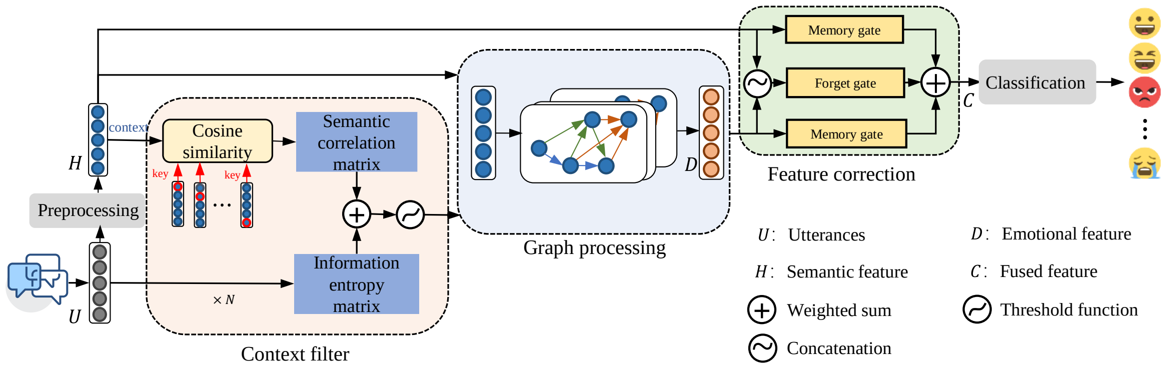


Figure 1: Overall structure of the proposed method.

feature correction, and (v) classification.

Let a conversation be represented by a collection of N utterances $U = \{u_1, u_2, \cdots, u_N\}$. During the *preprocessing stage*, the model first extracts a set of semantic features $H = \{H_1, H_2, \cdots, H_N\}$ from the utterance collection U. These features encode the semantic content expressed in the utterances and form the basis for the later stages of the model. Next, a (novel) *context filter* is utilized to identify utterances that are the most *relevant* and *informative* for the emotion recognition task. The filter relies on semantic correlations and information-theoretic principles to establish useful dependencies between the utterances of the given conversation. The semantic features and identified utterance dependencies are then supplied to a graph convolutional neural network that captures the structure of the conversation, incorporates contextual cues and speaker information, and outputs *graph-processed* features D that encode various aspects of the conversation critical for conversational emotion recognition. Finally, a *feature correction* mechanism is employed to improve the discriminability of the initial semantic H and emotion features D and generate the final fused representation for emotion *classification*.

3.1. Preprocessing

The goal of the preprocessing stage is to extract information-rich semantic features from the utterances of the given conversation. Inspired by the success of recent techniques for conversational emotion recognition that use (large-scale) pre-trained language models for this task, e.g., [10], we adopt the RoBERTa-Large [37] model to preprocess the set of utterances in U and extract their semantic features H. The RoBERTa-Large model is chosen for our work due to its excellent zero-shot feature extraction capabilities, but also the fact that it can easily be fine-tuned and adapted towards the characteristics of the selected conversational dataset. To extract features with RoBERTa-Large, the given utterance u_i is first transformed into a sequence X_i by the model's tokenizer:

$$X_i = \{[T_1], [T_2], \cdots, [T_M]\}, \tag{1}$$

where $[T_M]$ is the M-th token representation. Next, a special type of token $[CLS]$ is added in front of the sequence, and the corresponding output vector of this token is used as the representation of the utterance u_i. The input for the preprocessing stage X_i is represented as:

$$X_i = \{[CLS], [T_1], [T_2], \cdots, [T_M]\}. \tag{2}$$

Finally, the features corresponding to the added token $[CLS]$ in the last hidden layer of the model are utilized as the semantic features H_i of the utterance u_i, i.e.:

$$H_i = RoBERTa(X_i).last\_hidden\_layer[0]. \tag{3}$$

We note that because RoBERTa-Large was pre-trained on a large and diverse (language) dataset, it is able to efficiently encode the semantic content of the input utterances and extract descriptive semantic features that serve as the basis for the later stages of the proposed emotion recognition model.

3.2. Context filter

Existing techniques for conversational emotion recognition commonly model the relationships between speakers and consider temporal order to establish dependencies between the utterances of a conversation. Additionally, the semantic relevance of the utterances is analyzed to identify relevant conversational contexts. While such an approach has been shown to work well in practice, it can steer the recognition models towards focusing primarily on utterances with a high degree of semantic correlation, while also considering *noisy* contextual information with little relevance and low informativeness.

To mitigate the influence of low-relevance and uninformative contextual cues on conversational emotion recognition, we propose a novel *context filter* to remove (denoise) noisy information from the process of building dependencies between utterances. The filter considers (i) the *semantic relevance* of the utterances in a conversation by measuring the similarity of the semantic embeddings produced by the pre-trained language model, and (ii) the *informativeness* of the utterances providing context by using information-theory principles. To quantify relevance and informativeness, the filter first calculates semantic-relevance and information-entropy matrices and then combines the two into (what we refer to as) the *comprehensive-score matrix* that is ultimately analyzed and filtered to discard contextual utterances with low comprehensive scores, that are indicative of low relevance and uninformative conversation content. A formal description of the context filter is given below.

Given a set of semantic features H, extracted from the conversation U using the pre-trained language model, the context filter first evaluates the semantic relevance of each utterance u_i with respect to all other utterances in U. The semantic feature of each utterance H_i will act as a key to ask context features H about their similarity by computing the cosine similarity between H_i and H.

$$s_1^i = \frac{H_i H}{\|H_i\|_2 \|H\|_2} \in \mathbb{R}^{1\times N}, \tag{4}$$

where s_1^i denotes the $1 \times N$ (contextual) semantic relevance vector corresponding to u_i. The vector encodes the semantic correlations between u_i and U, and thus produces high scores for utterances that share similar (and, therefore, relevant) semantic content. The complete semantic relevance matrix s_1 is obtained by stacking the semantic relevance vectors of each utterance:

$$s_1 = [(s_1^1)^T, (s_1^2)^T, \cdots, (s_1^N)^T]^T \in \mathbb{R}^{N\times N}. \tag{5}$$

Next, information (Shannon) entropy is used to measure the informativeness of the utterances in U that provide context for the emotion recognition task. Here, the context filter calculates the information entropy of each utterance by aggregating the entropies of all words/tokens of the given utterance, i.e.:

$$s_2^i = -\sum_{j=1}^{M} p(T_j) \log_2 p(T_j), \tag{6}$$

where $p(T_j)$ denotes the frequency of the j-th token in the utterance u_i, and s_2^i stands for the corresponding entropy. After evaluating the above equations on all N utterances of the conversation U, the information entropy matrix $\widehat{s}_2$ is computed as follows:

$$\widehat{s}_2 = [s_2^1, s_2^2, \cdots, s_2^N] \in \mathbb{R}^{1\times N}. \tag{7}$$

To ensure that the dimensions of the information-entropy matrix match those of the semantic-relevance matrix, we stack N copies of $\widehat{s}_2$ to construct the final matrix s_2 as:

$$s_2 = diag(\widehat{s}_2) \cdot 1_N \in \mathbb{R}^{N\times N}, \tag{8}$$

where $diag(\cdot)$ is an operator that generates a diagonal matrix and 1_N is an $N \times N$ matrix of all ones.

In order to consider both semantic relevance and informativeness when evaluating context for the emotion recognition task, the semantic-relevance matrix and information-entropy matrix are weighted and summed to obtain the comprehensive-score matrix:

$$s = (1-\alpha)s_1 + \alpha s_2, \tag{9}$$

where α is a weight hyperparameter that balances the contribution of the two components. By taking the weighted sum of these two matrices, we obtain a comprehensive influence matrix, where each aggregated element reflects the combined influence of semantic relevance and information value of the contexts on the target utterance. In the proposed emotion recognition model, the comprehensive-score matrix serves as the basis for defining the adjacency matrix $A = \{a_{ij}\} \in \mathbb{R}^{N \times N}$ that is needed for the graph construction procedure. Specifically, we first apply the context filter on the comprehensive-score matrix by truncating all elements below the threshold γ to zero. Additionally, to avoid self-connections in the graph, all diagonal elements of the adjacency matrix are also set to 0. If we denote an entry in the $N \times N$ adjacency matrix as a_{ij}, then the context filtering procedure can formally be described as:

$$a_{ij} = \begin{cases} 0, & s^{ij} < \gamma \ or \ i = j, \\ 1, & s^{ij} \geqslant \gamma, \end{cases} \tag{10}$$

where γ is a hyperparameter that represents the threshold, s^{ij} denotes the comprehensive score of the context utterance u_j with respect to the target utterance u_i, and a value of 1 implies that a connection should be present in the constructed graph between the two utterances, while a value of 0 suggests the opposite.

3.3. Graph processing

In order to capture the dependencies between the utterances of a conversation and their context, a relational graph of the following form $G = \{V, E, R\}$ is constructed for our emotion recognition model, where V and E represent the set of nodes and edges, respectively, and R denotes the edge type. It is important to highlight that the proposed graph neural network represents a unidirectional graph, indicating the presence of a causal relationship as context passes through a context filter. We provide a comprehensive description of the graph construction process in the graph processing module. In the context of conversations, a causal relationship refers to the transmission of information from the preceding context to the subsequent one, while the emotional state of the previous discourse remains unaffected by subsequent words. Specifically, when examining emotional transmission in conversations, it becomes evident that the emotional state of a speaker during previous conversations remains unaffected by the emotions expressed in subsequent interactions. To simulate this unidirectional emotion transfer, we construct a directed graph denoted as $G = \{V, E, R\}$ to depict the flow of emotions in a conversation. Within this graph, each utterance is represented as a node, and the directed edges indicate the flow of information from one utterance to the subsequent one. From the perspective of emotional flow, the directed edges in the graph ensure that the emotions of subsequent utterances do not impact the emotional states of preceding utterances. With this graph formulation, each utterance is represented as a node $v_i \in V$, and each node is represented by the semantic features H_i extracted during the preprocessing stage. The nodes v_i and v_j are in general connected by the corresponding edge $e_{ij} \in E$ and the presence of the edge is dependent on the respective output of the context filter, i.e., a_{ij}. Additionally, each edge e_{ij} can correspond to one of two edge types $r_{ij} = \{0, 1\} \in R$, where a value of 1 indicates that the nodes v_i and v_j are from the same speaker, and a value of 0 indicates that they are from different speakers. We use an L-layer relational Graph Convolutional Network (GCN) as the basis for our model.

To initialize the edge weights between an utterance node v_i and a context node v_j in the l^{th} layer of the graph and, thus, encode the degree of influence of u_j on u_i, a similarity-based attention mechanism is first utilized, i.e.:

$$\alpha_{ij}^l = softmax_{j \in \lfloor A_i \rfloor}(W_\alpha^l concat(H_j^l, H_i^l)), i \in [0, N], \tag{11}$$

where $A_i \in \mathbb{R}^{1 \times N}$ represents the adjacency matrix of node u_i (i.e., a row from A), the operator $\lfloor x \rfloor$ returns the indices of the non-zero elements of x, and W_α^l stands for the parameters that need to be learned during training.

Next, to model information propagation across the graph, we follow [33], and compute the semantic features H_i^{l+1} of the $(l+1)^{th}$ layer by aggregating information across the neighboring nodes of u_i in the l^{th} layer. This process partially maintains the sentiment information of the i-th utterance from the l^{th} layer, but infuses additional information into the features by incorporating additional contextual cues from neighboring (relevant and informative) utterances:

$$H_i^{l+1} = \sum_{r \in R} \sum_{j \in \lfloor A_i \rfloor} \frac{\alpha_{ij}}{\sum \alpha_{ij}} W_{ij}^l H_j^l + \alpha_{ii} W_i^l H_i^l, \tag{12}$$

where W_{ij}^l, W_i^l are trainable parameters and α_{ii} is the edge weight of the i-th node connecting to itself between different layers. This weight can be interpreted as the semantic self-similarity of the utterance, which defaults to 1, so the formula in Eq. (12) can be rewritten as:

$$H_i^{l+1} = \sum_{j \in \lfloor A_i \rfloor} \alpha_{ij}^* W_{ij}^l H_j^l + W_i^l H_i^l, \tag{13}$$

where α_{ij}^* is the normalized version of α_{ij}. At each of the L layers of the GCN, a set of semantic features is, thus, computed. Here, the set of semantic (node) features H^l for the entire conversation at the l^{th} layer can be written as:

$$H^l = [H_1^l, H_2^l, \cdots, H_N^l]. \tag{14}$$

To obtain context-embedded emotional features G for the entire conversation U from the graph structure, the node features H^l from all layers (0 to L) are concatenated, such that:

$$G = concat(H^l), \text{ where } \ l = \{1, 2, \dots, L\}. \tag{15}$$

Ultimately, the final emotion features D of the utterances are generated by the fully connected layer of the model and its respective activation function, i.e.:

$$D = PReLU(W^d G + b^d), \tag{16}$$

where W^d and b^d are the trainable weights and the bias of the fully connected layer, respectively. The PReLU activation is used with our model to help with over-fitting problems.

3.4. Feature correction

After processing the given conversation U through the GCN, emotional features D are generated, which encode context information and account for the relations and dependencies between the utterances. To facilitate emotion recognition, a common strategy from the literature is to combine the semantic features H and the emotion features D and produce an aggregated representation $C = concat(D, H)$ with higher discriminative power. However, such a naive strategy may be sub-optimal and propagate potential errors from the previous stages of the model into the naively fused features. Lian *et al.* [21] employed graph convolutional neural networks to capture interactions and address certain errors, while bidirectional GRUs and multi-head attention mechanisms were utilized to correct errors stemming from contextual understanding. In our graph processing module, we consider the interaction between speakers, which helps mitigate errors resulting from inadequate interaction and limited contextual understanding to a certain extent. To avoid such issues and make full use of the computed feature representations, we propose a novel feature correction mechanism. The mechanism is inspired by the enhanced LSTM network from [38] and aims at reducing model prediction errors. While speaker dependence and contextual information contribute to the understanding of the emotion of the target utterance, excessive connections can sometimes lead to incorrect predictions during model training. To address this issue, our feature correction module focuses on rectifying erroneous predictions that arise from excessive reliance on speaker relationships and contextual connections within the graph processing modules. The problem of over-connection is mitigated by incorporating a gating mechanism that selectively discards emotional features from the graph processing module.

As illustrated in Figure 2, the feature correction process utilizes a gating mechanism to control the semantic features H, the context-infused emotion features D, and their fused combination, so that the recognition model may pay attention to the semantics of the given utterance, while also taking the informativeness and relevance of the utterances providing context into account. Because the graph

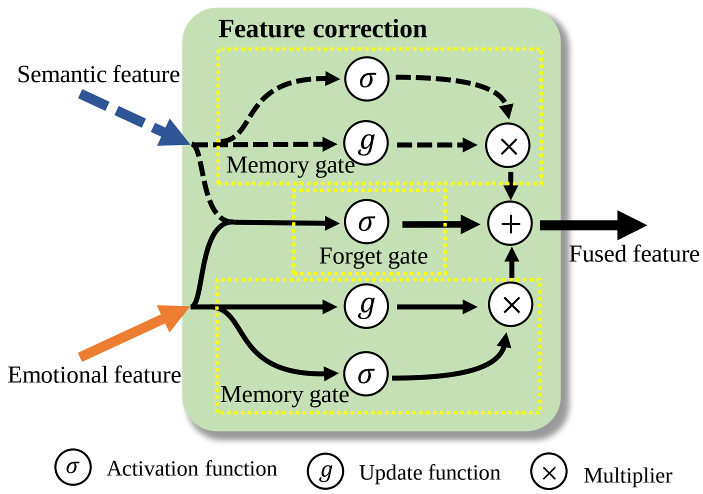


Figure 2: The structure of the feature correction module.

network already captures and propagates long-distance relationships between utterances, the proposed feature correction module does not utilize any chain structure to incorporate such information.

The feature correction module is divided into three branches and relies on two distinct inputs. The input to the upper branch comes from the preprocessing stage and represents the semantic features H, extracted by the pre-trained language model. The input to the lower branch comes from the graph processing and represents the emotion features D. Additionally, the two inputs are combined to generate fused features, which are then passed through the third (fusion) branch. To achieve efficient feature correction, a *forgetting gate* is first utilized to forget part of the semantic information in H as well as part of the emotion cues in D and obtain (information-deprived) fused features f, i.e.,

$$f = \sigma(W^f H + Q^f D + b^f), \tag{17}$$

where W^f and Q^f are the trainable weights corresponding to the semantic and emotional features, respectively. σ is the activation function and b^f is the trainable bias. Next, the calculated (initially fused) features f are updated through the outputs of the upper and lower module branches that process the semantic features H and emotion features D through two separate memory gates, constructed by combining two activation functions and a multiplier, i.e.:

$$\begin{aligned} z &= \sigma(W^z H + b^z), \\ c &= tanh(W^c H + b^c), \\ \widetilde{H} &= z \otimes c. \end{aligned} \tag{18}$$

$$
\begin{aligned}
m &= \sigma(W^m D + b^m), \\
s &= tanh(W^s D + b^s), \\
\widetilde{D} &= m \otimes s,
\end{aligned} \tag{19}
$$

where $W^{(\cdot)}$, $Q^{(\cdot)}$, and $b^{(\cdot)}$ are trainable parameters and the different superscripts imply that the weights are not shared, and $tanh$ denotes the activation function (marked g in Figure 2). In general, a memory gate consists of a *forgetting gate* and a *learning gate* and can be used to modify the information content of the processed features. With the semantic features H, for example, the forgetting gate z decides what information is less relevant and needs to be updated in H. The learning gate c, on the other hand, learns to incorporate new information into the features that help to make them more descriptive and improve their discriminative power. Based on these two gates, the initial semantic features H are then updated through the multiplier to $\widetilde{H}$. In the same way, the emotion features D are updated to $\widetilde{D}$. Finally, to compensate for the forgotten part of the information in the initially fused features f, we add the updated semantic $\widetilde{H}$ and emotion features $\widetilde{D}$ to f and calculate the final fused (and corrected) features C for the classification task, as follows:

$$
C = f + \widetilde{H} + \widetilde{D}. \tag{20}
$$

3.5. Classification

The output of the feature-correction mechanism C is used in the proposed model as the final feature representation of the given utterance. For classification purposes, a fully connected network is adopted and utilized to obtain the probability P_i of each of the considered emotion categories. The category corresponding to the highest probability is taken as the final emotion label:

$$
P_i = softmax(W^p C_i + b^p), \tag{21}
$$

$$
\widehat{y}_i = \arg\max_k P_i(k), \tag{22}
$$

where C_i and $\widehat{y}_i$ are the (corrected) fused features and predicted emotion of the utterance u_i, respectively. W^p and b^p are the trainable weights and the bias of the fully connected layer. The proposed model is trained end-to-end using the cross-entropy as the loss function, which can be expressed as:

$$
\mathcal{L}_i(\theta) = (y_i)\log(\widehat{y}_i) + (1 - y_i)(1 - \log(\widehat{y}_i)), \tag{23}
$$

where θ is the set of all parameters that need to be learned for the model, $\widehat{y}_i$ is the highest prediction probability of the i–th emotion label, and y_i is the one-hot encoded ground truth emotion label for utterance u_i.

4. Experiments

To demonstrate the performance and merits of the proposed model, we conduct comprehensive experiments on *four* different datasets and compare our approach to *eight* competing state-of-the-art (SOTA) techniques. In this section, we first describe the experimental setup (i.e., the selected datasets, implementation details, SOTA baselines, and evaluation metrics) and then discuss the results and their implications.

4.1. Datasets

Four standard datasets are adopted for the experiments. The selected datasets represent a diverse cross-section of data commonly used to evaluate the performance of techniques for conversational emotion recognition. Details on the datasets are given below.

- **IEMOCAP** [39] is a multimodal dataset, consisting of 151 conversations recorded from 5 speaker pairs. The dataset contains annotations for nine emotional categories, i.e.: angry, excited, fear, sad, surprised, frustrated, happy, disappointed, and neutral. To facilitate comparisons with prior work, we used six primary emotions for the experiments, i.e.: neutral, happy, sad, angry, frustrated, and excited. The remaining three categories appear less frequently in the dataset and were not included in the comparative assessments.

- **MELD** [40] is a multimodal dataset containing 1400 conversation pairs and 13,000 utterances. The dataset was constructed from recordings of the Friends TV show and, therefore, features a rich set of emotional conversations. The MELD dataset is annotated with the name of speakers, and emotion labels spanning seven distinct categories: anger, disgust, sadness, joy, neutrality, surprise, and fear, alongside additional meta-information.

- **Dailydialog** [41] is a conversational dataset with 13118 conversations and 102979 utterances, each annotated with one of six emotion labels: anger, disgust, fear, happiness, sadness, surprise. The dataset contains human-written text on diverse topics, follows a multi-turn dialog flow that resembles human communications and is designed specifically for the task of conversational emotion recognition.

- **EmoryNLP** [42] is a plain text dataset, containing 12,606 utterance annotations from one of six emotional labels: sad, mad, scared, powerful, peaceful, and joy. The dataset consists of multi-party dialogues created from transcripts of a popular TV show and hence features a rich set of (emotional) dialogues in various settings and circumstances.

A high-level overview (and comparison) of the datasets is given in Table 1. Here, information is provided on the number of conversations in each dataset, the number of utterances, the average conversation length (in utterances), and the average length of each utterance in the dataset (in words). For the experiments, we partition the datasets into three non-overlapping sets for training (train), development (dev), and testing (test) in accordance with the official splits (or as used with the methods selected for comparison if an official split is not available). We use the training set to learn the proposed model, the development set to monitor convergence, and the test set for the final performance reporting.

Table 1: High-level comparison of the four experimental datasets

Dataset	**Number of conversations**			**Number of utterances**			**Average conversation length**			**Average utterance length**		
	train	dev	test	train	dev	test	train	dev	test	train	dev	test
IEMOCAP [39]	100	20	31	6490	1404	2196	64.9	70.05	70.84	14.93	15.9	15.72
MELD [40]	1038	114	280	9989	1109	2610	9.62	9.73	9.32	11.41	11.32	11.71
Dailydialog [41]	11118	1000	1000	87170	8069	7740	7.84	8.07	7.74	15.49	15.38	15.68
EmoryNLP [42]	713	99	85	9934	1344	1328	13.93	13.58	15.62	15.03	14.09	14.51

4.2. Implementation details

The proposed model was implemented on a Desktop PC with an eight-core CPU and a Tesla T4 16G GPU. All experiments were conducted within the Ubuntu 18.04 operating system using Python 3.7, Pytorch 1.10, CUDA 10.2, and AdamW, as the optimizer for the model-learning procedure. To accommodate different dataset characteristics and ensure reasonable convergence, different training parameters were used for the optimization process, as summarized in Table 2. Additionally, details are available in the publicly released source code[1].

Table 2: Training parameters used to learn the proposed model on each dataset

Parameters	**IEMCOAP**	**MELD**	**Dailydialog**	**EmoryNLP**
Optimizier	AdamW			
Embedding size	1024			
Hidden size	300			
Dropout rate	0.1			
Learning rate	1e-6	2e-5	2e-5	2e-5
Batch size	16	32	64	32
Epoch	100	100	50	100
Weight α	0.75	0.80	0.80	0.75
Threshold γ	1.0	1.5	1.2	2.3

[1] https://github.com/Jahao26/denoiseGNN.

Using the presented hardware, parameter settings, and well-pre-trained RoBERTa-Large, the model was trained for 100 epochs on each dataset except Dailydialog. Due to the large amount of Dailydialog data, 50 epochs can be trained well. All reported results on the comparison experiments are averaged over 5 runs.

4.3. Baselines and state-of-the-art (SOTA) methods

To demonstrate the capabilities of the proposed model and provide a reference frame for the generated results, we consider multiple (conceptually distinct) baseline and state-of-the-art methods in the experiments, i.e.:

- **CMN** [9]. The Conversational Memory Network (CMN) uses a gated recurrent unit (GRU) to memorize the utterance information of each speaker from the conversion history and provide contextual information for the emotion recognition task.
- **bc-LSTM** [43]. The bi-directional contextual LSTM (bc-LSTM) model consists of two stacked LSTM models with different directions. Because of the opposing directions of the models, bc-LSTM considers contextual information from utterances occurring either before or after a given target utterance for conversational sentiment analysis.
- **DialogueRNN** [10]. DialogueRNN uses a recurrent neural network to model three aspects that are important for the emotion recognition problem, i.e.: the speaker, the context, and the emotion from the preceding utterances. These aspects are modeled through three types of GRUs that account for the global, speaker, and emotional state of the conversation.
- **DialogueGCN** [17]. DialogueGCN is a Graph Convolutional Neural Network (GCN) that uses intra- and inter-speaker dependencies to model conversations and generate graph-encoded representations to capture the structure of a conversation and the associated context information. Compared with the traditional recurrent neural networks, it alleviates the problem of the difficulty of modeling long-distance context information.
- **DialogXL** [12]. DialogXL exploits knowledge encoded in the pre-trained XLNet language model and uses enhanced memory to store the conversation history to model context. Additionally, it utilizes a dialogue-aware self-attention mechanism to model dependencies between speakers.
- **COSMIC** [11]. COSMIC represents a common-sense guided framework for conversational emotion recognition. It uses external knowledge to understand the commonsense information appearing in the utterances and to model complex interactions between speakers, emotions, events, and other related influential factors that facilitate efficient emotion recognition.

- **DAG-ERC** [34]. The DAG-ERC network represents a directed acyclic graph that captures the structure of the conversations and combines characteristics of graph-based models and recurrent neural networks. The model intuitively models the long-distance dependencies between utterances in a conversation as well as nearby contextual information.

- **DSAGCN** [35]. DSAGCN is a graph convolutional neural network (GCN) that uses speaker relations and dependency syntactic analysis (DSA) to establish utterance relations and analyze utterance sentiment. Specifically, the syntactic structure of the dialogue context used in the model allows for highly efficient emotion recognition.

4.4. Evaluation metrics

Following established evaluation methodology [35, 44], we report the accuracy (Acc) and weighted $F1$ scores to evaluate the performance of the tested methods on the IEMOCAP, MELD, and EmoryNLP datasets. Here, accuracy is defined as [45]:

$$Acc = \frac{\sum_{i=1}^{n}(TP_i + TN_i)}{\sum_{i=1}^{n}(TP_i + TN_i + FP_i + FN_i)}, \tag{24}$$

$$F1 = \frac{1}{n}\sum_{i=1}^{n}\omega_i(\frac{2TP_i}{2TP_i + FN_i + FP_i}), \tag{25}$$

where n denotes the number of classes. ω_i denotes the weight of i-th class according to the quantity difference of all classes. TP_i and TN_i denote the number of true positive and true negative predictions for the i-th class, whereas FP_i and FN_i denote the number of false positive and false negative predictions for the i-th class, respectively.

Because the Dailydialog dataset has a severe class-imbalance problem, where the "neutral" class represents 77.94% of the data, the $MacroF1$ and $MicroF1$ are utilized to report performance on this dataset, similarly to [11].

$$MacroF1 = \frac{1}{n}\sum_{i=1}^{n}(\frac{2TP_i}{2TP_i + FN_i + FP_i}), \tag{26}$$

$$MicroF1 = \frac{\sum_{i=1}^{n} 2TP_i}{\sum_{i=1}^{n}(2TP_i + FN_i + FP_i)}. \tag{27}$$

4.5. Results and discussions

We evaluate the proposed model on four datasets to demonstrate its capabilities and compare it to the SOTA competitors. However, it should be noted that not all considered baselines were experimentally validated on all four datasets, so the selection of comparative methods differs from dataset to dataset. In the following sections, we therefore analyze the results for each dataset separately.

Table 3: Comparison results on IEMOCAP

Methods	Happy		Sad		Neutral		Angry		Excited		Frustrated		*Acc* (↑)	*F*1 (↑)
	Acc	*F*1	*Acc*	*F*1	*Acc*	*F*1	*Acc*	*F*1	*Acc*	*F*1	*Acc*	*F*1		
bc-LSTM [43]	29.1	34.4	57.1	60.8	54.1	51.8	57.1	56.7	51.1	57.9	67.1	58.9	55.2	54.9
CMN [9]	25.0	30.3	55.9	62.4	52.8	52.3	61.7	59.8	55.5	60.2	71.1	60.6	56.5	56.1
DialogueRNN [10]	33.5	35.4	69.0	68.8	54.1	54.7	**67.1**	61.1	55.9	60.4	62.9	60.3	58.3	58.1
DialogueGCN [17]	45.7	47.7	**86.9**	**84.4**	41.9	48.5	61.5	62.2	**72.4**	69.3	51.5	56.6	59.0	56.1
DSAGCN [35]	**60.1**	**62.6**	84.8	82.3	44.5	47.5	63.7	59.6	69.3	71.5	54.8	62.1	63.5	61.7
DialogXL [12]	44.0	44.0	69.4	77.1	64.5	64.6	54.7	61.5	68.5	69.7	75.6	66.9	65.7	65.8
DAG-ERC [34]	43.4	45.1	82.9	80.6	69.8	68.1	65.9	**66.9**	64.9	69.2	**71.7**	**69.8**	68.6	68.4
Ours	53.1	54.9	81.6	81.9	**74.8**	**73.5**	66.0	66.4	68.7	**73.3**	65.5	68.0	**69.7**	**69.7**

4.5.1. Comparison on IEMOCAP

Table 3 shows the accuracy and weighted $F1$ for each emotion label on the IEMOCAP dataset. It can be seen that the performance of the proposed model is highly competitive, with the highest overall Accuracy (69.7%) and F1 score (69.7%) over the entire dataset among all of the evaluated methods. With the "neutral" class, the accuracy and weighted $F1$ of our method are 5.0% and 5.4% better than that of DAG-ERC [34]. This can be ascribed to the fact that DAG-ERC only models nearby contexts, while our context filtering expands the context acquisition range, thus, leading to better performance. The syntactic-dependency analysis used in DSAGCN [35] improves the ability to recognize obvious emotions (such as happy and sad), but it performs poorly in predicting the "neutral" emotion class. Similarly, DialogueGCN [17] achieves the best accuracy and weighted $F1$ score of 86.9% and 84.4%, respectively, for the "sad" class, but only yields an accuracy of 45.7% and a weighted $F1$ score of 47.7% with the "happy" class. Conversely, our method achieves competitive results in predicting both, the "happy" and "sad" classes, while all other methods, except DSAGCN, perform quite poorly with these two emotion categories. These results are a consequence of the filtering mechanism implemented with the proposed context filter that enables the removal of noisy connections during the graph construction step of our model, leading to highly competitive performance.

In Figure 3, we provide the confusion matrix of our method on IEMOCAP, which shows a more in-depth picture of the performance of the proposed model. We observe that our method exhibits the weakest performance when trying to recognize similar emotions, such as "happy" and "excited". The difference between these emotion categories is in their intensity, but our method does not capture these subtle differences well enough to be capable of efficiently discriminating between the two. A possible solution for this issue is to emotion intensity as an auxiliary label for model training and we plan to explore such extensions as part of our future work.

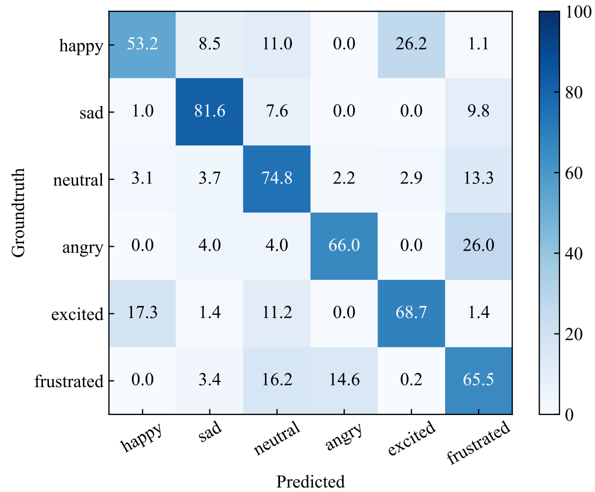


Figure 3: The confusion matrix of the proposed method on IEMOCAP.

4.5.2. Comparison on MELD

In Table 4, we show comparative results on the MELD dataset. We again observe that the proposed model achieves the highest accuracy (67.3%) and weighted $F1$ score (66.4%) overall among all considered methods. Our model fares particularly well with the "neutral" class, but similarly to all other methods, performs less convincingly with the "fear" and "disgust" classes. The reason for such a behavior is that the "fear" and "disgust" classes only account for 1.9% and 2.6% of the data in MELD and, as a result, none of the evaluated models can be sufficiently trained from the few available samples to efficiently recognize these two emotions. This class imbalance eventually leads to incorrect recognition results and predictions that favor emotion categories with a higher representation within the dataset. Nonetheless, it can be observed that the GNN-based methods perform significantly better than the remaining techniques. We conjecture that there is a significant amount of short-range dependencies between the utterances in the MELD dataset compared to IEMOCAP, which heavily impacts the recognition procedure. Mechanisms for modeling a much wider context are, therefore, needed to recognize the emotion categories accurately on this dataset, especially with the under-represented classes. The context filter (integrated into our model) allows us to better capture the long-range conversational context, as well as the utilized pre-trained language model that enables (zero-shot) extraction of descriptive semantic information from the conversations, hence, leading to significantly better performance of our model in recognizing the "fear" and "disgust" emotions when compared to the baselines. The performance is only rivaled by the DAG-ERC approach, which also features a graph structure and mechanism for modeling longer-range contextual information.

Figure 4 shows the confusion matrix of our method on MELD. It can be seen that most of the errors come from misclassifying different emotions as "neutral". This is most evident with the "fear", "disgust", and "sadness" classes, where a significant portion of the test data is assigned a "neutral" label. The reason for such behavior is that the "neutral" class accounts for 48.12% of the data in MELD, leading to a highly imbalanced recognition problem during training and testing. Furthermore, it is highly challenging to efficiently distinguish "fear", "disgust", and "sadness" from the "neutral" class given text, as the only source of information for the emotion recognition task. These limitations are reflected in the results of our model and, as discussed above, are even more problematic for most of the competing techniques.

Table 4: Comparison result on MELD

Methods	**MELD**															
	Neutral		Surprise		Fear		Sadness		Joy		Disgust		Anger		*Acc* (↑)	*F*1 (↑)
	Acc	*F*1	*Acc*	*F*1	*Acc*	*F*1	*Acc*	*F*1	*Acc*	*F*1	*Acc*	*F*1	*Acc*	*F*1		
bc-LSTM [43]	78.4	73.8	46.8	47.7	3.8	5.4	22.4	25.1	51.6	51.3	4.3	5.2	36.7	38.4	57.5	55.9
CMN [9]	76.2	74.9	43.3	45.5	4.6	3.7	18.2	21.1	46.1	49.4	8.9	8.3	35.3	34.5	54.3	55.0
DialogueRNN [10]	72.1	73.5	54.4	49.4	1.6	1.2	23.9	23.8	52.0	50.7	1.5	1.7	41.0	41.5	56.1	55.9
DialogueGCN [17]	70.3	72.1	42.4	41.7	3.0	2.8	20.9	21.8	44.7	44.2	6.5	6.7	39.0	36.5	54.9	54.7
DSAGCN [35]	76.7	74.4	48.6	45.5	5.2	4.8	24.4	22.1	52.5	49.6	7.4	8.7	**52.2**	46.9	60.9	58.7
DialogXL [12]	79.4	78.5	63.7	57.5	0.0	0.0	29.8	33.1	60.9	61.2	0.0	0.0	55.3	49.9	64.2	62.7
DAG-ERC [34]	77.4	77.2	**67.3**	57.1	**42.0**	**48.4**	30.3	35.7	**66.4**	61.7	25.0	**31.8**	42.0	48.4	63.9	63.3
Ours	**84.4**	**80.7**	63.7	**59.7**	20.0	22.2	**31.7**	**40.7**	66.4	**64.3**	**26.5**	31.3	48.7	**53.2**	**67.7**	**66.7**

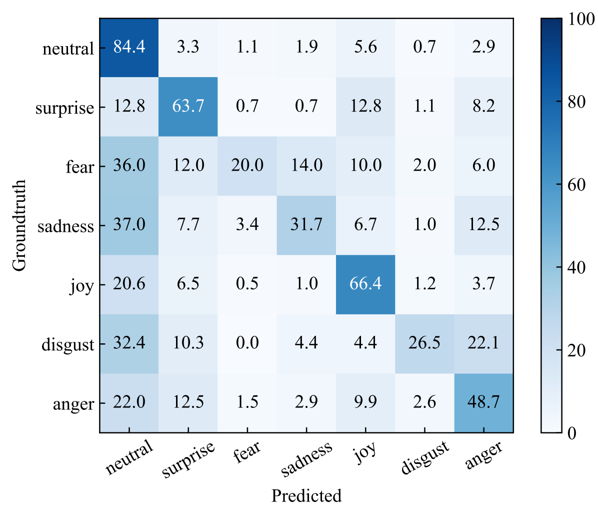


Figure 4: The confusion matrix of the proposed method on MELD.

4.5.3. Comparison on Dailydialog

On the Dailydialog dataset, our model performs better than all competing methods in terms of the $MicroF1$ score, as shown in Table 5. The $MicroF1$ accounts for class imbalances when quantifying performance and our model convincingly outperforms all considered baselines in this regard. The proposed model is the runner-up behind COSMIC [11] when the $MacroF1$ score is considered,

Table 5: Comparison result on DailyDialog

Methods	**DailyDialog**													
	Happinese		Anger		Sadness		Fear		Surprise		Disgust		$MacroF1$ (↑)	$MicroF1$ (↑)
	Acc	$F1$	Acc	$F1$	Acc	$F1$	Acc	$F1$	Acc	$F1$	Acc	$F1$		
DialogueRNN [10]	62.5	60.3	0.0	0.0	6.8	11.1	0.0	0.0	12.9	21.5	0.0	0.0	43.4	51.5
DialogXL [12]	59.5	62.8	31.3	35.2	29.4	34.6	0.0	0.0	50.0	46.6	0.0	0.0	40.3	55.6
COSMIC [11]	**82.6**	60.4	37.2	36.9	**59.8**	33.5	29.4	16.9	61.2	42.0	**40.4**	**41.7**	52.2	58.9
DAG-ERC [34]	60.9	63.4	**38.9**	43.4	32.3	38.4	11.7	20.0	53.4	52.1	21.3	28.5	**53.4**	59.1
Ours	64.1	**77.6**	38.1	**52.0**	43.1	**59.4**	**29.4**	**45.5**	**52.6**	**60.1**	23.4	33.8	48.6	**59.6**

where a few poorly performing categories typically adversely affect the overall $MacroF1$ result. With the proposed approach, the "disgust" class is not sufficiently learned due to the insufficient number of training samples available, negatively impacting its $MacroF1$ score. Nevertheless, compared to DialogueRNN [10] and DialogXL [12], our model yields a significantly higher $MacroF1$ score. The reason for this result lies in the use of the pre-trained language model and its (zero-shot) feature extraction capabilities that allow us to infer information-rich and descriptive representations from the provided utterances that result in highly competitive downstream emotion recognition capabilities.

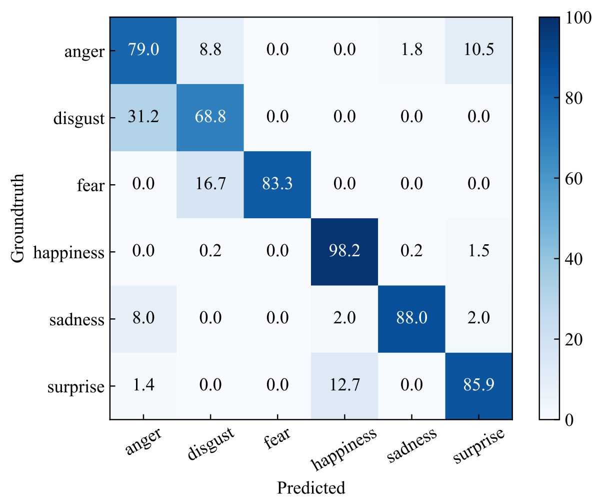


Figure 5: The confusion matrix of the proposed method on Dailydialog.

Figure 5 shows the confusion matrix of our method on Dailydialog. We observe that the model exhibits the strongest performance with the "happiness" class and the weakest with the "disgust" class. As already suggested above, the underrepresentation of "disgust" samples in this dataset leads to classification errors, where "disgust" is most often incorrectly labeled as "anger". Among other common (and somewhat consistent) substitutions, we also see "surprise" being confused with "happiness", "fear" with "disgust", and "sadness" being labeled as "anger". Such misclassification is, in a sense, expected given the nature of the emotions and is still sufficiently rare to result in competitive $MicroF1$ scores, as reported in Table 5.

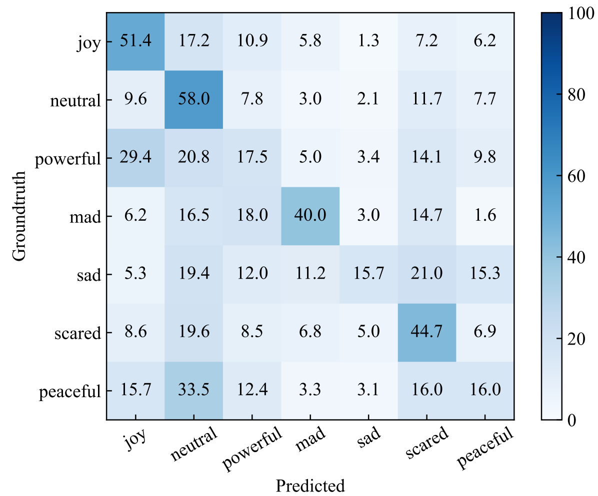


Figure 6: The confusion matrix of the proposed method on EmoryNLP.

4.5.4. Comparison on EmoryNLP

As illustrated in Table 6, the proposed model achieves the best performance among all evaluated (state-of-the-art) methods on the EmoryNLP dataset with an accuracy of 40.65% and a weighted $F1$ score of 39.71%. However, compared to the results on the three other datasets, i.e., IEMOCAP, MELD, and Dailydialog, the performance of all tested techniques is much lower overall. We ascribe this result to the definition of the emotion classes in EmoryNLP. While similarly to MELD, EmoryNLP was constructed from conversations of the *Friends* TV show, the class labels between the two datasets differ significantly. This suggests that non-standard classes, such as "powerful" or "peaceful" may not be clearly expressed in the conversations and are therefore more difficult to recognize. This can also be seen from the confusion matrix of our model in Figure 6, where conversations labeled "powerful" are easily confused with "joy", and utterances labeled "peaceful" with "neutral". The number of misclassified samples for the "powerful" and "peaceful" categories even exceeds the number of correctly predicted samples.

Table 6: Comparison result on EmoryNLP

Methods	**EmoryNLP**															
	Joy		Neutral		Powerful		Mad		Sad		Scared		Peaceful		*Acc* (↑)	*F*1 (↑)
	Acc	*F*1	*Acc*	*F*1	*Acc*	*F*1	*Acc*	*F*1	*Acc*	*F*1	*Acc*	*F*1	*Acc*	*F*1		
DialogXL [12]	55.3	50.16	61.8	50.0	6.9	8.3	46.9	35.8	15.3	21.9	32.9	37.3	0.6	1.1	38.4	34.6
COSMIC [11]	58.9	**53.0**	51.3	51.0	1.0	1.9	**51.1**	36.5	**21.4**	**26.5**	**49.1**	37.3	4.5	7.0	40.4	37.1
DAG-ERC [34]	**59.2**	52.7	**67.0**	**53.9**	0.0	0.0	47.7	37.7	17.3	21.5	34.6	34.0	6.2	10.1	**41.0**	36.0
Ours	51.3	52.0	57.9	53.5	**17.5**	**17.2**	40.0	**40.9**	15.7	21.0	44.7	**39.9**	**15.9**	**18.6**	40.7	**39.7**

Table 7: Comparision with the latest models

Model	IEMOCAP	MELD	DailyDialog	EmoryNLP
	$F1$	$F1$	$MicroF1$	$F1$
EmoCaps [26]	69.49	63.51	-	-
M2FNet [46]	66.20	66.23	-	-
SACL-LSTM [47]	69.22	66.45	-	39.65
CoMPM [48]	66.61	66.52	60.34	37.37
S+PAGE [27]	68.72	63.32	**64.07**	39.14
Ours	**69.71**	**66.70**	59.62	**39.73**

4.5.5. Comparision with the latest models

Among the latest comparison methods, models that combine transformers with other neural networks [26, 48, 27] have shown competitive results across multiple benchmark datasets. However, these models often struggle to achieve a balanced performance across all datasets. In contrast, Hu *et al.* [47] proposed a context-antagonistic strategy that enhances the learning of contextual features, resulting in a more robust model that outperforms other approaches on three experimental datasets. This learning strategy, which emphasizes model robustness, is a rarity in the field of conversational sentiment analysis, yet it demonstrates clear reliability and effectiveness.

We have identified the lack of robustness in existing models as a concern and have taken measures to address this issue. Specifically, we have introduced two hyperparameters to adapt to the variations across different datasets and enhance the model's contextual understanding. Additionally, by leveraging the powerful contextual understanding capabilities of transformers and the interactive capabilities of GNNs, our model exhibits promising performance that surpasses some recent comparison models.

4.6. Ablation study

In order to verify the importance of the proposed *context filtering* and *feature correction* components of the proposed model, we perform comprehensive ablation studies using all four experimental datasets. Specifically, we ablate the context filter by setting the corresponding threshold to zero, so the filtering operation has no effect, i.e., no context is filtered out. As a result, each given conversation is represented as a fully connected graph. For the feature correction ablation experiment, we adopt a similar approach to other GNN-based methods in the literature. We concatenate the emotion features produced by the graph processing module with the original features and use this combined input as the input to the final emotion classifier. This allows us to compare the performance of our model with and without the feature correction stage. The results of the ablation studies are presented quantitatively in Tables 8-11, and in the form of confusion matrices for the feature-correction ablations in Figure 7.

4.6.1. Ablation on IEMOCAP

After ablating the context filter on IEMOCAP, the proposed model yields an accuracy of 67.69% and a weighted $F1$ score of 67.41%, as summarized in Table 8. Compared with the complete model, the accuracy and weighted $F1$ degrade by 1.99% and 2.28%, respectively, due to the absence of the context-filtering mechanism. The lack of the filtering mechanism results in dependencies between all utterances in a given conversation, regardless of whether a given utterance is relevant and informative for the emotion recognition task, i.e., irrespective of the context of said utterance. Without the evaluation of contextual relevance, it is possible (and even likely) that distant utterances with irrelevant/weak contextual information are considered during the inference process, leading to suboptimal results. Similarly, without the measurement of informativeness, weakly correlated utterances with (potentially) high information content may not be considered to a sufficient extent by the proposed model due to the similarity-based attention mechanism used in graph processing.

After ablating the feature correction stage on IEMOCAP, the accuracy and weighted $F1$ decrease by 0.94% and 0.86%, respectively, compared to the results of the entire model. The performance degradation due to the removal of the feature correction process is slightly lower than the degradation caused by the removal of the context filter but still points to its importance for the performance of the overall model. If we compare the confusion matrices in Figure 3 and Figure 7(a), we can find that there is a considerable 4% to 5% decrease in the recognition performance for the "happy" and "excited" emotion classes if the feature correction mechanism is not used, while the accuracy is also reduced for "sad", "angry" and "neutral" categories, albeit to a lesser extent.

Table 8: Ablation results on IEMOCAP

Context filter	**Feature correction**	*Acc* (↑)	$F1$ (↑)
✗	✓	67.69	67.41
✓	✗	68.74	68.73
✓	✓	69.68	69.69

4.6.2. Ablation on MELD

Table 9 shows the results on the MELD dataset after ablating the context filter and feature correction mechanism. The results show a similar picture as the ablation experiments on IEMOCAP. The accuracy (now 66.05%) and weighted $F1$ scores (65.02%) decrease by 1.69% and 1.65%, respectively, when removing the context filter. This implies that it is unreasonable to treat all utterances (regardless of context) as influencing factors when recognizing emotions. Some of these utterances may introduce misleading contextual cues into the emotion recognition task and, consequently, adversely affect performance.

Next, we ablate the feature correction stage on MELD and observe an accuracy of 66.55% and a weighted $F1$ score of 65.43%. This corresponds to a performance decrease of 1.19% and 1.24%, respectively, compared to the complete model. If we compare the confusion matrices in Figure 4 and Figure 7(b), we find that the feature correction mechanism adversely affects all emotion categories. This is due to the stronger dependencies between utterances in the MELD dataset, and, consequently, the larger impact of contextual information on the recognition performance. If the feature correction stage is removed, the model is more susceptible to spurious contextual information that is not rectified during the feature correction stage, resulting in reduced performance on MELD.

Table 9: Ablation results on MELD

Context filter	**Feature correction**	Acc (↑)	$F1$ (↑)
✗	✓	66.05	65.02
✓	✗	66.55	65.43
✓	✓	67.74	66.67

4.6.3. Ablation on Dailydialog

The ablation-study results on Dailydialog in Table 10 show that removing the context filter results in performance degradations of 1.82% for the $MacroF1$ and 0.52% for the $MicroF1$ score compared to the scores achieved by the complete model, i.e., 46.84% and 59.10%, respectively. Compared to the IEMOCAP and MELD datasets, the performance degradations are smaller, but still suggest that the context filtering contributes to the overall performance. If we remove the feature correction stage on Dailydialog, we observe $MacroF1$ and $MicroF1$ scores of 46.83% and 59.40%, respectively, which corresponds to a decrease of 1.83% and 0.22%, when compared to the complete model. From the comparison of Figure 5 and Figure 7(c), we can see that the performance difference with and without the use of the feature correction mechanism is relatively modest. While we do see degradations for the "fear", "sadness", and "surprise" categories, these degradations are quite minute. This is because the Dailydialog dataset is about an order of magnitude larger than the other datasets (in 1), so the emotion features that are learned are able to ensure reasonable performance even without the feature correction. Therefore, the performance differences caused by the feature correction on the Dailydialog dataset are less obvious.

4.6.4. Ablation on EmoryNLP

Finally, we present ablation results for the EmoryNLP dataset in Table 11. After removing the context filter, the accuracy and weighted $F1$ scores are 39.68% and 38.95%, suggesting a decrease of 0.97% and 0.76% compared to the complete model. The accuracy and weighted $F1$ score after ablating

Table 10: Ablation results on Dailydialog

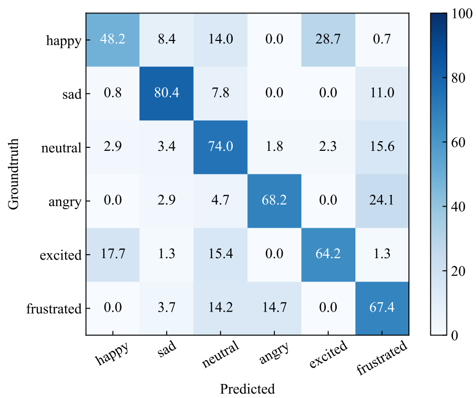
Context filter	**Feature correction**	$MacroF1$ (↑)	$MicroF1$ (↑)
✗	✓	46.84	59.10
✓	✗	46.83	59.40
✓	✓	48.66	59.62

the feature correction mechanism weigh in at 40.36% and 39.52%, respectively, which corresponds to a decrease by 0.29% and 0.19% compared to the setting where the mechanism is used. Looking at Figures 6 and 7(d), we find that on the EmoryNLP dataset, the performance degradation caused by the removal of the feature correction stage is less obvious, and has various degrees of impact on the performance across the individual emotion categories. The feature correction module demonstrates its effectiveness in correcting neutral labels by leveraging the rich feature information obtained from a large number of neutral emotion utterances. Consequently, the performance of the model improves after incorporating the feature correction module. However, it is important to acknowledge that the annotations in the EmoryNLP dataset can be subjective and controversial. There is a lack of consensus among annotators regarding emotional labels, with the lowest level of agreement observed in annotations for the "powerful" emotion, reaching only 0.8% agreement among all four annotators [42]. This subjectivity and ambiguity in emotional labels pose challenges for the feature correction module in learning emotional features specific to certain emotions and distinguishing them from other similar emotions. It is worth noting that the accuracy and weighted $F1$ scores on EmoryNLP are about 40% lower than on the other datasets. This observation (together with the ablation-study results) suggests that the feature correction stage has a limited ability to correct the information content in the feature representations if this content is too ambiguous. Furthermore, the reported results may to a certain extent also be related to the definition of the emotion categories on this dataset. Regardless of whether the feature-correction mechanism is present or not, the weighted $F1$ score of some emotional categories with less obvious emotional tendencies, such as "powerful", "sad" and "peaceful", are always lower than 20%, greatly impacting the performance of the overall model.

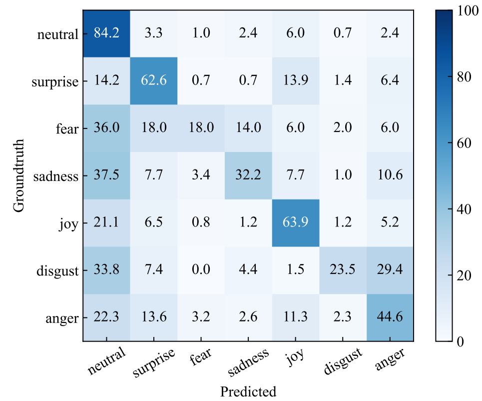
Table 11: Ablation results on EmoryNLP

Context filter	**Feature correction**	Acc (↑)	$F1$ (↑)
✗	✓	39.68	38.95
✓	✗	40.36	39.52
✓	✓	40.65	39.71

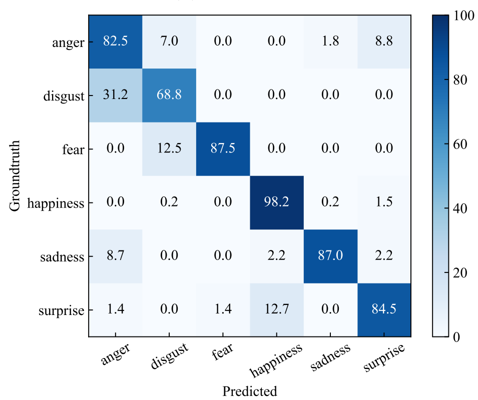
The feature correction module exhibits varying patterns of decline for different emotional categories



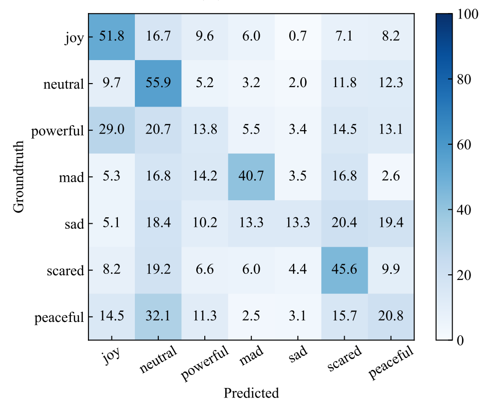
(a) IEMOCAP



(b) MELD



(c) Dailydialog



(d) EmoryNLP

Figure 7: The confusion matrices after ablating the feature correction.

in different datasets due to the characteristics of the data. Table 1 provides insights into the specific characteristics of each dataset. The differences in data volume, data characteristics, and data imbalance contribute to the varying degree of fit achieved by the feature correction module for different emotional categories across different datasets.

Interestingly, when comparing the performance before and after the ablation of the feature correction module, we observed that the module outperformed EmoryNLP in highly unbalanced datasets such as IEMOCAP, MELD, and Dailydialog. This observation was supported by the comparison of confusion matrices, which revealed the module's ability to effectively correct mispredictions in categories that have a larger proportion in unbalanced datasets. Notably, in the MELD dataset, the feature correction module demonstrated exceptional performance in correcting mispredictions related to categories such as "disgust" and "anger".

4.6.5. Parameter analysis

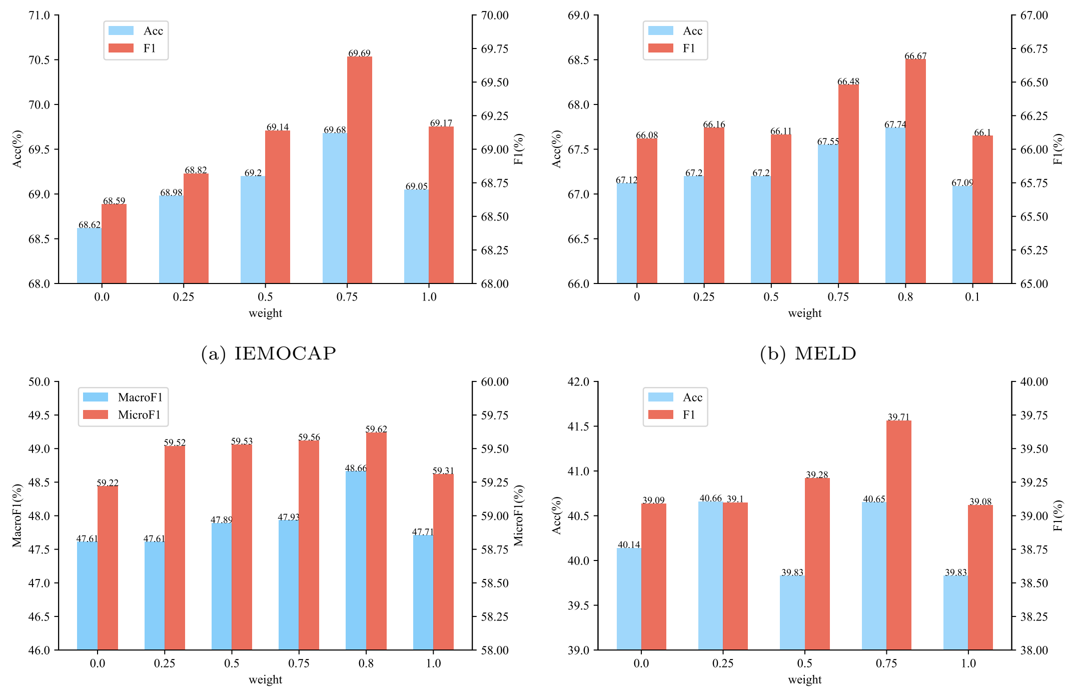
In order to study the influence of semantic relevance and informativeness on the performance of our model, we explore the impact of changing the weight parameter α (given in Eq (9)) in the context filter.

When the weight is set to 0, the comprehensive score s is equal to the semantic relevance score s_1, and the context filter is completely dependent on the semantic similarity between utterances. When the weight is 1, the comprehensive score s is equal to the informativeness score s_2, and the context filter depends on the informativeness of the contextual cues In addition to two edge cases, we also explore various weights that maximize the model's performance on each dataset. For the sake of simplicity, we report results only for a subset of weights that are the most informative for the analysis. The experimental results are shown in Figure 8.

Figure 8(a) illustrates the variation in performance as a function of the weight parameter on IEMOCAP. One can see that when the weight is 0, the accuracy and weighted $F1$ score are only 68.62% and 68.59%, respectively. An initial weight increase can bring some improvement to the performance, and the highest accuracy and weighted $F1$ are 69.68% and 69.69% when the weight is equal to 0.75. Increasing the weight beyond this value does not bring additional performance gains. When the weight is 1 and the model is completely dependent on the informativeness of utterances but ignores the semantic relevance, the performance decreases, leading to the accuracy and weighted $F1$ scores of 69.05% and 69.17%.

Figure 8(b) displays the variation in accuracy and weighted $F1$ scores due to changes in the weight parameter on MELD. When the weight is 0, the accuracy and weighted $F1$ of the model are 67.12% and 66.08%, respectively. However, different from IEMOCAP, when the weight is less than 0.5, increasing the value of the weight parameter does not significantly improve performance, and the accuracy is always around 67.2%. When the weight is set to 0.8, the performance is the highest but then decreases with further increases in the weight value. When the weight is set to 1, the accuracy and weighted $F1$ score are 67.09% and 66.1%. This is because the average length of the utterances in MELD is shorter than that in IEMOCAP (see also Table 1), while the utterances also contain noise components that impact the expressivity of the emotions. As a result, the informativeness of the utterances is still comparably low, even if the informativeness score is considered with the maximum possible weight.

Figure 8(c) demonstrates the change of the $MacroF1$ and $MicroF1$ scores with respect to the weight parameter on Dailydialog. When the weight is 0, the $MacroF1$ and $MicroF1$ scores are 47.61% and 59.22%, respectively. The scores then slowly increase and reach the optimal/highest $MacroF1$ and $MicroF1$ values at the weight of 0.8, i.e., $MacroF1 = 48.66\%$ and $MicroF1 = 59.62\%$. Figure 8(d) shows the change in accuracy and weighted $F1$ scores caused by the weight changes on the EmoryNLP dataset. We observe that the accuracy fluctuates significantly with changes in the weight parameter values and is impacted by the data imbalance of this dataset. When the weight is 0, the weighted $F1$ score is 39.09%. The score then slowly increases to the highest value of 39.71%, at which time the weight is 0.75. When the weight is 1, the recognition accuracy and $F1$ score are reduced to 39.83% and 39.08%, respectively.



(a) IEMOCAP (b) MELD

(c) Dailydialog (d) EmoryNLP

Figure 8: The influence of semantic relevance and informativeness on model performance.

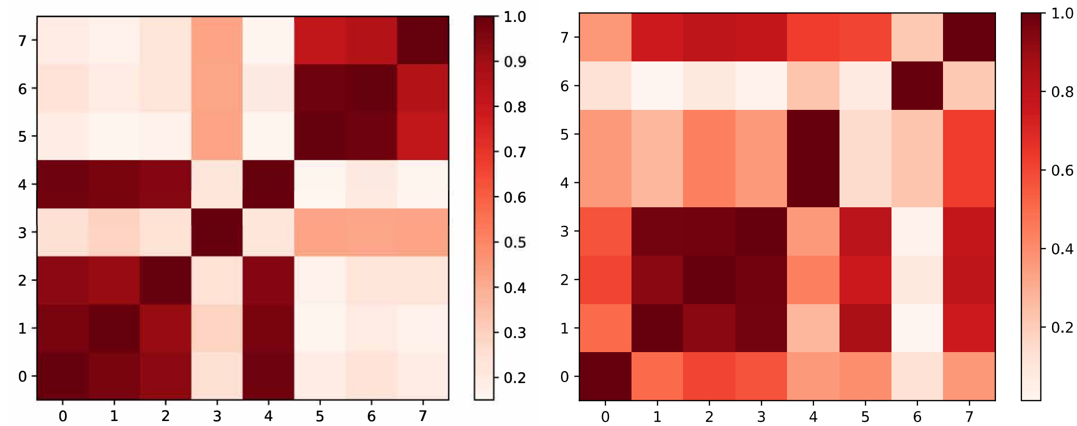
4.7. Case study

To validate the effectiveness of cosine similarity in capturing contextual relevance, we conducted a case study using two instances from the MELD dataset. We computed the specific contextual correlation between these instances to visually depict the degree of correlation. To delve deeper into the examination of the impact of context filter and gain a more profound comprehension of the errors rectified by the feature correction mechanism, we opt for a dialogue scenario extracted from the test set of IEMOCAP. We aim to visually depict the contextual evaluation process and analyze the predictive outcomes in both the presence and absence of feature correction.

Table 12: Two case conversation in MELD dataset

Index	NO.38		No.59	
	Utterance	Label	Utterance	Label
0	Oh.	neutral	Does Monica still turn on the lights in her bedroom?	anger
1	But I don't. Me, Phoebe.	neutral	It looks like a women's purse.	neutral
2	Well, I'm not I'm not at all surprised they feel that way.	neutral	No Joey, look. Trust me, all the men...	neutral
3	You're not? See, that's why you're so great!	surprise	See look,	neutral
4	Actually it's, it's quite, y'know, typical behavior...	neutral	Exactly! Unisex!	neutral
5	Y'know, this kind of co-dependant, emotionally ...	anger	Maybe you need sex.	neutral
6	Define me!	anger	No! No Joey! U-N-I-sex.	joy
7	Love me, I need love!	anger	Well, I ain't gonna say no to that.	neutral

The selected cases for the analysis are from conversations No.59 and No.38 in the MELD dataset. By comparing the labels and heatmaps, we can observe a clear pattern of high semantic similarity,



(a) Similarity matrix in No.38 conversation (b) Similarity matrix in No.59 conversation

Figure 9: The case of context-relevant in MELD dataset

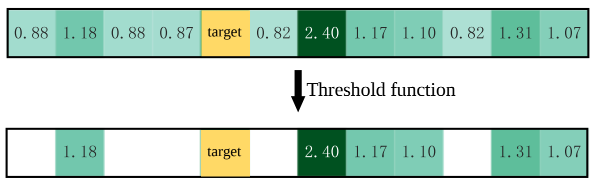


Figure 10: Heatmap of context assessment scores for the target utterance.

as measured by cosine similarity, among words belonging to the same label. For instance, in Figure 9(a), words associated with the neutral label in the first three sentences and the anger label in the last three sentences exhibit significantly higher similarity compared to words in other contexts. When expressing intense emotions, the model tends to focus more on the utterance itself rather than relying on semantic similarity. This can be observed in Figure 9(b), where the first anger utterance and the penultimate joy utterance have lower similarity to utterances in other contexts.

As shown in Figure 10, the context filter evaluates the comprehensive score of the target utterance and, in a sense, quantifies the amount of contextual information that can be obtained from the rest of the utterances in the conversion for the selected target. The threshold function then filters out utterances whose scores are lower than the predefined threshold. Table 13 shows a conversation case detailing which feature errors are corrected by the feature correction. Column 3 of the table shows the predicted labels when using a fully connected network to classify semantic features. Columns 4 and 5 show the predicted emotion labels with and without the feature correction, respectively. By comparing the predicted labels, one can find that although the feature correction can correct part of the prediction errors by fusing semantic features, the feature correction still cannot correct prediction errors that are caused by factors other than context.

Comparing Table 13 and Figure 10, one can find that most utterances are short texts, and it is

Table 13: Comparison of the results before and after feature correction for a conversation case of IEMOCAP

Utterance	Label	Prediction		
		after preprocessing	direct fusion	feature correction
With the most perfect poise.	exc	hap	hap	neu
Yes, I shall probably do a Court Curtsey.	exc	hap	hap	hap
The whole business is really rather ridiculous.	neu	hap	hap	hap
Meaning exactly that.	neu	hap	hap	*neu*
What does it all mean? That's what I asked myself in my ceaseless quest for the ultimate truth. Dear God, what does it all mean?	exc	hap	neu	neu
Who's they?	neu	hap	hap	hap
All the futile mortals who try to make life unbearable. Laugh at them. Be flippant. Laugh at everything, all their sacred shibboleths. Flippancy brings out the acid in their damned sweetness and light.	exc	*exc*	neu	neu
Certainly you must. We're figures of fun alright [LAUGHTER].	neu	*neu*	hap	*neu*
Well, what if-what happens when our love-	exc	neu	neu	neu
Who knows?	exc	neu	neu	neu
No, that fire will fade along with our passion.	neu	*neu*	hap	*neu*
It all depends on how well we played.	exc	*exc*	neu	*exc*

challenging to reliably recognize the correct emotion labels from these utterances. If we compare the predicted labels with the reference emotion labels, one can find that the model has difficulty distinguishing between similar emotions by utterance and context, such as the emotion labels "frustrated" and "sad", "frustrated" and "angry", "excited" and "happy". Similarly, it can be seen that with some samples, the model can not discriminate between different levels of intensity of the emotion. Additionally, there are also cases where "happy" and "frustrated" are predicted as "neutral". Since most conversational datasets do not contain labels that describe emotional states from multiple perspectives, such as arousal, valence, and dominance, it is challenging to distinguish utterances with different emotions in intensity only through the text modality and context. Most existing models do not perform well in discriminating similar emotions, which is one of the main open issues in conversational sentiment analysis.

5. Conclusion

In this paper, we proposed a model for recognizing emotions in conversations using a graph neural network supplemented with a novel context filter and feature correction mechanism. In order to identify utterances that are most relevant and informative for mining contextual information, a context filter was designed to consider both the semantic relevance and the information content of the utterances. The context filter was shown to be adaptable to the characteristics of different datasets by varying weights and thresholds. Additionally, the proposed feature correction mechanism was demonstrated to be able to correct the extracted feature representations that would otherwise cause incorrect predictions. By combining emotional and semantic features, the feature correction mechanism was illustrated to adapt the fused features and to rectify the potentially erroneous fused features that are employed during classification. Finally, through comprehensive and rigorous experiments on four diverse datasets, i.e., IEMOCAP, MELD, Dailydialog, and EmoryNLP, it was shown that the proposed model yields superior performance compared to the latest methods commonly used in the literature for the task of conversational emotion recognition.

Acknowledgements

The authors are grateful to the anonymous reviewers and the editor for their valuable comments and suggestions. This work was supported by the Chongqing Research Program of Basic Research and Frontier Technology (No. cstc2021jcyj-msxmX0761) and the Slovenian ARRS research program P2-0250.

References

[1] D. Zeng, R. Peng, C. Jiang, Y. Li, J. Dai, Csdm: A context-sensitive deep matching model for medical dialogue information extraction, Information Sciences 607 (2022) 727–738.

[2] M. Tauqeer, S. Rubab, M. A. Khan, R. A. Naqvi, K. Javed, A. Alqahtani, S. Alsubai, A. Binbusayyis, Driver's emotion and behavior classification system based on internet of things and deep learning for advanced driver assistance system (adas), Computer Communications 194 (2022) 258–267.

[3] B. Wang, G. Dong, Y. Zhao, R. Li, Q. Cao, K. Hu, D. Jiang, Hierarchically stacked graph convolution for emotion recognition in conversation, Knowledge-Based Systems (2023) 110285.

[4] S. Bashath, N. Perera, S. Tripathi, K. Manjang, M. Dehmer, F. E. Streib, A data-centric review of deep transfer learning with applications to text data, Information Sciences 585 (2022) 498–528.

[5] P. Kuppens, N. B. Allen, L. B. Sheeber, Emotional inertia and psychological maladjustment, Psychological science 21 (7) (2010) 984–991.

[6] S. Zou, X. Huang, X. Shen, H. Liu, Improving multimodal fusion with main modal transformer for emotion recognition in conversation, Knowledge-Based Systems 258 (2022) 109978.

[7] Y. Liu, Q. Li, B. Wang, Y. Zhang, D. Song, A survey of quantum-cognitively inspired sentiment analysis models, ACM Computing Surveys (2023).

[8] S. Liu, P. Gao, Y. Li, W. Fu, W. Ding, Multi-modal fusion network with complementarity and importance for emotion recognition, Information Sciences 619 (2023) 679–694.

[9] D. Hazarika, S. Poria, A. Zadeh, E. Cambria, L.-P. Morency, R. Zimmermann, Conversational memory network for emotion recognition in dyadic dialogue videos, in: Proceedings of the conference. Association for Computational Linguistics. North American Chapter. Meeting, Vol. 2018, NIH Public Access, 2018, p. 2122.

[10] N. Majumder, S. Poria, D. Hazarika, R. Mihalcea, A. Gelbukh, E. Cambria, Dialoguernn: An attentive rnn for emotion detection in conversations, in: Proceedings of the AAAI conference on artificial intelligence, Vol. 33, 2019, pp. 6818–6825.

[11] D. Ghosal, N. Majumder, A. Gelbukh, R. Mihalcea, S. Poria, Cosmic: Commonsense knowledge for emotion identification in conversations, in: Findings of the Association for Computational Linguistics: EMNLP 2020, 2020, pp. 2470–2481.

[12] W. Shen, J. Chen, X. Quan, Z. Xie, Dialogxl: All-in-one xlnet for multi-party conversation emotion recognition, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2021, pp. 13789–13797.

[13] M. Zhao, J. Yang, J. Zhang, S. Wang, Aggregated graph convolutional networks for aspect-based sentiment classification, Information Sciences 600 (2022) 73–93.

[14] H. T. Phan, N. T. Nguyen, D. Hwang, Convolutional attention neural network over graph structures for improving the performance of aspect-level sentiment analysis, Information Sciences 589 (2022) 416–439.

[15] X. Song, L. Zang, R. Zhang, S. Hu, L. Huang, Emotionflow: Capture the dialogue level emotion transitions, in: ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2022, pp. 8542–8546.

[16] Q. Gao, B. Cao, X. Guan, T. Gu, X. Bao, J. Wu, B. Liu, J. Cao, Emotion recognition in conversations with emotion shift detection based on multi-task learning, Knowledge-Based Systems 248 (2022) 108861.

[17] D. Ghosal, N. Majumder, S. Poria, N. Chhaya, A. Gelbukh, Dialoguegcn: A graph convolutional neural network for emotion recognition in conversation, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 2019, pp. 154–164.

[18] T. Ishiwatari, Y. Yasuda, T. Miyazaki, J. Goto, Relation-aware graph attention networks with relational position encodings for emotion recognition in conversations, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2020, pp. 7360–7370.

[19] S. E. Finch, J. D. Choi, Towards unified dialogue system evaluation: A comprehensive analysis of current evaluation protocols, in: Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue, 2020, pp. 236–245.

[20] Z. Zhan, J. Zhao, Y. Zhang, J. Gong, Q. Wang, Q. Shen, L. Zhang, Grabbing the long tail: A data normalization method for diverse and informative dialogue generation, Neurocomputing 460 (2021) 374–384.

[21] Z. Lian, B. Liu, J. Tao, Decn: Dialogical emotion correction network for conversational emotion recognition, Neurocomputing 454 (2021) 483–495.

[22] J. Deng, F. Ren, A survey of textual emotion recognition and its challenges, IEEE Transactions on Affective Computing (2021) 1–20.

[23] C. Gan, Y. Yang, Q. Zhu, D. K. Jain, V. Struc, Dhf-net: A hierarchical feature interactive fusion network for dialogue emotion recognition, Expert Systems with Applications 210 (2022) 118525.

[24] Y. Zhang, P. Tiwari, D. Song, X. Mao, P. Wang, X. Li, H. M. Pandey, Learning interaction dynamics with an interactive lstm for conversational sentiment analysis, Neural Networks 133 (2021) 40–56.

[25] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805 (2018).

[26] Z. Li, F. Tang, M. Zhao, Y. Zhu, Emocaps: Emotion capsule based model for conversational emotion recognition, in: Findings of the Association for Computational Linguistics: ACL 2022, 2022, pp. 1610–1618.

[27] C. Liang, J. Xu, Y. Lin, C. Yang, Y. Wang, S+ page: A speaker and position-aware graph neural network model for emotion recognition in conversation, in: Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing, 2022, pp. 148–157.

[28] Y. Zhang, A. Jia, B. Wang, P. Zhang, D. Zhao, P. Li, Y. Hou, X. Jin, D. Song, J. Qin, M3gat: A multi-modal multi-task interactive graph attention network for conversational sentiment analysis and emotion recognition, ACM Transactions on Information Systems (2023).

[29] Y. Zhang, J. Wang, Y. Liu, L. Rong, Q. Zheng, D. Song, P. Tiwari, J. Qin, A multitask learning model for multimodal sarcasm, sentiment and emotion recognition in conversations, Information Fusion 93 (2023) 282–301.

[30] L. Yang, Y. Shen, Y. Mao, L. Cai, Hybrid curriculum learning for emotion recognition in conversation, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 36, 2022, pp. 11595–11603.

[31] X. Song, L. Huang, H. Xue, S. Hu, Supervised prototypical contrastive learning for emotion recognition in conversation, in: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, 2022, pp. 5197–5206.

[32] J. Wang, S. Wang, M. Lin, Z. Xu, W. Guo, Learning speaker-independent multimodal representation for sentiment analysis, Information Sciences 628 (2023) 208–225.

[33] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. v. d. Berg, I. Titov, M. Welling, Modeling relational data with graph convolutional networks, in: European semantic web conference, Springer, 2018, pp. 593–607.

[34] W. Shen, S. Wu, Y. Yang, X. Quan, Directed acyclic graph network for conversational emotion recognition, arXiv preprint arXiv:2105.12907 (2021).

[35] Y. Shou, T. Meng, W. Ai, S. Yang, K. Li, Conversational emotion recognition studies based on graph convolutional neural networks and a dependent syntactic analysis, Neurocomputing 501 (2022) 629–639.

[36] S. Hareli, S. Elkabetz, U. Hess, Drawing inferences from emotion expressions: The role of situative informativeness and context., Emotion 19 (2) (2019) 200.

[37] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized bert pretraining approach, arXiv preprint arXiv:1907.11692 (2019).

[38] L. Xu, Z. Jie, W. Lu, L. Bing, Better feature integration for named entity recognition, in: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2021, pp. 3457–3469.

[39] C. Busso, M. Bulut, C.-C. Lee, A. Kazemzadeh, E. Mower, S. Kim, J. N. Chang, S. Lee, S. S. Narayanan, Iemocap: Interactive emotional dyadic motion capture database, Language resources and evaluation 42 (4) (2008) 335–359.

[40] S. Poria, D. Hazarika, N. Majumder, G. Naik, E. Cambria, R. Mihalcea, Meld: A multimodal multi-party dataset for emotion recognition in conversations, arXiv preprint arXiv:1810.02508 (2018).

[41] Y. Li, H. Su, X. Shen, W. Li, Z. Cao, S. Niu, DailyDialog: A manually labelled multi-turn dialogue dataset, in: Proceedings of the Eighth International Joint Conference on Natural Language Processing, 2017, pp. 986–995.

[42] S. M. Zahiri, J. D. Choi, Emotion detection on tv show transcripts with sequence-based convolutional neural networks, AAAI Publications, 2018.

[43] S. Poria, E. Cambria, D. Hazarika, N. Majumder, A. Zadeh, L.-P. Morency, 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), 2017, pp. 873–883.

[44] H. Ma, J. Wang, H. Lin, X. Pan, Y. Zhang, Z. Yang, A multi-view network for real-time emotion recognition in conversations, Knowledge-Based Systems 236 (2022) 107751.

[45] C. Gan, Q. Feng, Z. Zhang, Scalable multi-channel dilated cnn–bilstm model with attention mechanism for chinese textual sentiment analysis, Future Generation Computer Systems 118 (2021) 297–309.

[46] V. Chudasama, P. Kar, A. Gudmalwar, N. Shah, P. Wasnik, N. Onoe, M2fnet: Multi-modal fusion network for emotion recognition in conversation, in: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, 2022, pp. 4651–4660.

[47] D. Hu, Y. Bao, L. Wei, W. Zhou, S. Hu, Supervised adversarial contrastive learning for emotion recognition in conversations, arXiv preprint arXiv:2306.01505 (2023).

[48] J. Lee, W. Lee, Compm: Context modeling with speaker's pre-trained memory tracking for emotion recognition in conversation, in: Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2022, pp. 5669–5679.