BiERU: Bidirectional Emotional Recurrent Unit for Conversational Sentiment Analysis

Wei Li, Wei Shao, Shaoxiong Ji, and Erik Cambria, Senior Member, IEEE

Abstract—Sentiment analysis in conversations has gained increasing attention in recent years for the growing amount of applications it can serve, e.g., sentiment analysis, recommender systems, and human-robot interaction. The main difference between conversational sentiment analysis and single sentence sentiment analysis is the existence of context information which may influence the sentiment of an utterance in a dialogue. How to effectively encode contextual information in dialogues, however, remains a challenge. Existing approaches employ complicated deep learning structures to distinguish different parties in a conversation and then model the context information. In this paper, we propose a fast, compact and parameter-efficient party-ignorant framework named bidirectional emotional recurrent unit for conversational sentiment analysis. In our system, a generalized neural tensor block followed by a two-channel classifier is designed to perform context compositionality and sentiment classification, respectively. Extensive experiments on three standard datasets demonstrate that our model outperforms the state of the art in most cases.

Index Terms—Conversational sentiment analysis, emotional recurrent unit, contextual encoding, dialogue systems

I. INTRODUCTION

SENTIMENT analysis is of vital importance in dialogue systems and has recently gained increasing attention [1]. It can be applied to a lot of scenarios such as mining the opinions of speakers in conversations, improving the feedback of robot agents, and so on. Moreover, sentiment analysis in live conversations can be used in generating talks with certain sentiments to improve human-machine interaction. Existing approaches to conversational sentiment analysis can be divided into party-dependent approaches, like DialogueRNN [2], and party-ignorant approaches, such as AGHMN [3]. Party-dependent methods distinguish different parties in a conversation while party-ignorant methods do not. In this paper, we propose a fast, compact and parameter-efficient party-ignorant framework based on emotional recurrent unit (ERU), a recurrent neural network that contains a generalized neural tensor block (GNTB) and an emotion feature extractor (EFE) to tackle conversational sentiment analysis.

Context information is the main difference between dialogue sentiment analysis and single sentence sentiment analysis tasks. It sometimes enhances, weakens, or reverses the raw sentiment of an utterance (Fig. 1). There are three main steps for sentiment analysis in a conversation: obtaining the context information; capturing the influence of the context information for an utterance; and extracting emotional features for classification. Existing dialogue sentiment analysis methods like c-LSTM [4], CMN [5], DialogueRNN [2], and DialogueGCN [6] make use of complicated deep neural network structures to capture context information and describe the influence of context information for an utterance.

We redefine the formulation of conversational sentiment analysis and provide a compact structure to better encode the context information, capture the influence of context information for an utterance, and extract emotional features. To this end, we design GNTB to perform context compositionality which obtains context information and incorporates the context into utterance representation simultaneously, then employ EFE to extract emotional features. In this case, we convert the previous three-step task into a two-step task. Meanwhile, the compact structure reduces computational cost. To the best of our knowledge, our proposed model is the first to perform context compositionality in conversational sentiment analysis.

The GNTB takes the context and current utterance as inputs, and is capable of modeling conversations with arbitrary turns. It outputs a new representation of current utterance with context information incorporated (named as ‘contextual utterance vector’ in this paper). Then, the contextual utterance vector is further fed into EFE to extract emotional features. Here, we employ a simple two-channel model for emotion feature extraction.

The long short-term memory (LSTM) unit [7] and one-dimensional convolutional neural network (CNN) [8] are utilized for extracting features from the contextual utterance vector. Extensive experiments on three standard datasets demonstrate that our model outperforms state-of-the-art methods with less parameters. To summarize, the main contributions of this paper are as follows:

- We propose a fast, compact and parameter-efficient party-ignorant framework based on emotional recurrent unit.
- We design generalized neural tensor block which is suitable for different structures, to perform context compositionality.
- Experiments on three standard benchmarks indicate that our model outperforms the state of the art with less parameters.

The remainder of the paper is organized as follows: related work is introduced in Section II; the mechanism of our model is explained in Section III; results the experiments are discussed in Section IV; finally, concluding remarks are provided in Section V.
III. Method

A. Problem Definition

Given a multiple turns conversation \( C \), the task is to predict the sentiment labels or sentiment intensities of the constituent utterances \( U_1, U_2, ..., U_N \). Taking the interactive emotional database IEMOCAP [27] as an example, emotion labels include frustrated, excited, angry, neutral, sad and happy.

In general, the task is formulated as a multi-class classification problem over sequential utterances; while in some scenarios, it is regarded as a regression problem given continuous sentiment intensity. In this paper, utterances are pre-processed and represented as \( u_t \) using feature extractors described below.

B. Textual Feature Extraction

Following the tradition of DialogueRNN [2], utterances are first embedded into vector space and then fed into CNNs [8] for feature extraction. N-gram features are obtained from each utterance by applying three different convolution filters of sizes 3, 4 and 5, respectively. Each filter has 50 features-maps. [2] then use max-pooling followed by rectified linear unit (ReLU) activation [28] to process the outputs of convolution operation.

These activation values are concatenated and fed to a 100 dimensional fully connected layer whose outputs serve as the textual utterance representation. This CNN-based feature extraction network is trained at utterance level supervised by the sentiment labels.

C. Our Model

Our ERU is illustrated in Note 1 of Fig. 2, which consists of two components GNTB and EFE. As mentioned in the introduction, there are three main steps for conversational sentiment analysis, namely obtaining the context representation; incorporating the influence of the context information into an utterance; and extracting emotional features for classification.

In this paper, the ERU is employed in a bidirectional manner (BiERU) to conduct the above sentiment analysis task, reducing some expensive computations and converting the previous three-step task into a two-step task as shown in Fig. 2.

Similar to bidirectional LSTM (BiLSTM) [29], two ERUs are utilized for forward and backward passing the input utterances. Outputs from the forward and backward ERUs are concatenated for sentiment classification or regression. More concretely, the GNTB is applied to encoding the context information and incorporating it into an utterance simultaneously; while EFE takes the output of GNTB as input and is used to obtain emotional features for classification or regression.

1) Generalized Neural Tensor Block: The utterance vector \( u_t \in \mathbb{R}^d \) with the context information incorporated is named as contextual utterance vector \( p_t \in \mathbb{R}^d \) in this paper, where \( d \) is the dimension of \( u_t \) and \( p_t \). At time \( t \), GNTB (Fig. 3: (a)) takes \( u_t \) and \( p_{t-1} \) as inputs and then outputs \( p_t \), a contextual utterance vector. In this process, GNTB first extracts the context information from \( p_{t-1} \); then it incorporates the context information into \( u_t \); finally contextual utterance vector \( p_t \) is obtained. The first step is to capture the context information and the second step is to integrate the context information.
into current utterance. The combination of these two steps is regarded as context compositionality in this paper. To the best of our knowledge, this is the first work to perform context compositionality in conversational sentiment analysis. GNTB is the core part that achieves the context compositionality. The formulation of GNTB is described below:

$$p_t = f(m_t^{T^{[1:k]}}, m_t + Wm_t).$$

$$m_t = p_{t-1} \oplus u_t$$

where $m_t \in \mathbb{R}^{2d}$ is the concatenation of $p_{t-1}$ and $u_t$; $f$ is an activation function, such as $tanh$, $sigmoid$ and so on; the tensor $T^{[1:k]} \in \mathbb{R}^{2d \times 2d \times k}$ and the matrix $W \in \mathbb{R}^{k \times 2d}$ are the parameters used to calculate $p_t$. Each slice $T^{[i]} \in \mathbb{R}^{2d \times 2d}$ can be interpreted as capturing a specific type of context compositionality. Each slice $W^{[i]} \in \mathbb{R}^{1 \times 2d}$ maps contextual utterance vector $p_t$ and utterance vector $u_t$ into the context compositionality space. Here we have $k$ different context compositionality types, which constitutes $k$-dimensional context compositionality space. The main advantage over the previous neural tensor networks (NTN) [24], which is a special case of the GNTB when $k$ is set to $d$, is that GNTB is suitable for different structures rather than only the recursive structure and the space complexity of GNTB is $O(kd^2)$ compared with $O(d^3)$ in NTN. In order to further reduce the number of parameters, we employ the following low-rank matrix approximation for each slice $T^{[i]}$:

$$T^{[i]} = UV + diag(e)$$

where $U \in \mathbb{R}^{2d \times r}$, $V \in \mathbb{R}^r \times 2d$, $e \in \mathbb{R}^{2d}$ and $r \ll d$.

2) Emotion Feature Extractor: We utilize EFE to refine the emotion features from contextual vector $p_t$. As shown in Fig. 3: (b), the EFE is a two-channel model, including a LSTM cell [7] branch and a one-dimensional CNN [8] branch. The two branches receive the same contextual utterance vector $p_t$ and produce outputs independently.

At time $t$, the LSTM cell takes hidden state $h_{t-1}$, cell state $c_{t-1}$ and the contextual utterance vector $p_t$ as inputs, where $h_{t-1}$ and $c_{t-1}$ are obtained from the last time step $t-1$. The outputs of the LSTM cell are updated hidden state $h_t$ and cell state $c_t$. The hidden state $h_t$ is regarded as the emotion feature vector. The CNN receives $p_t$ as input and outputs the emotion feature vector $l_t$. Finally, the outputs of LSTM cell branch $h_t$ and CNN branch $l_t$ are concatenated into an emotion feature vector $e_t$ which is also the output of EERU. The formulas of EFE are as follows:

$$h_t, c_t = LSTMCell(p_t, (h_{t-1}, c_{t-1}))$$

$$l_t = CNN(p_t)$$

$$e_t = h_t \oplus l_t$$

3) Sentiment Classification & Regression: Taking emotion feature $e_t$ as input, we use a linear neural network $W_c \in \mathbb{R}^{d \times n_{class}}$ followed by a softmax layer to predict the sentiment labels, where $n_{class}$ is the number of sentiment labels.

Then, we obtain the probability distribution $S_t$ of the sentiment labels. Finally, we take the most possible sentiment class as the sentiment label of the utterance $u_t$:

$$S_t = Softmax(W_c^Te_t)$$

Fig. 2: (a) Architecture of BiERU with global context. (b) Architecture of BiERU with local context. Here $p_f^i$, $EFE_f$, and $GNTB_f$ are forward contextual utterance vector, EFE, and GNTB, respectively. $p_b^i$ and ERU_b stand for backward contextual utterance vector and ERU, respectively. $\hat{y}_i$ is the predicted possibility vector of sentiment labels. A, T, V are audio, textual, and visual modalities, respectively. In our model, we only focus on textual modality. The detailed structures of GNTB and EFE are shown in Fig. 3.
\[
\hat{y}_t = \arg \max_i S_i[i]
\]  
(8)

For sentiment regression task, we use a linear neural network \( W_r \in R^{D_{r} \times 1} \) to predict the sentiment intensity. Then, we obtain the predicted sentiment intensity \( q_t \):

\[
q_t = W_r^T e_t
\]

(9)

where \( W \in R^{D_{r} \times n_{\text{class}}} \), \( e_t \in R^{D_{r}} \), \( S_t \in R^{n_{\text{class}}} \), \( q_t \) is a scalar and \( \hat{y}_t \) is the predicted sentiment label for utterance \( u_t \).

4) Training: For classification task, we choose cross-entropy as the measure of loss, and use L2-regularization to relieve overfitting. The loss function is:

\[
L = -\frac{1}{N} \sum_{s=1}^{N} c(s) \sum_{i=1}^{c(i)} \log S_{i,j} [y_{i,j}] + \lambda \| \theta \|_2
\]

(10)

For regression task, we choose mean square error (MSE) to measure loss, and L2-regularization to relieve overfitting. The loss function is:

\[
L = \frac{1}{N} \sum_{s=1}^{N} c(s) \sum_{i=1}^{c(i)} (q_{i,j} - z_{i,j})^2 + \lambda \| \theta \|_2
\]

(11)

where \( N \) is the number of samples/conversations, \( S_{i,j} \) is the probability distribution of sentiment labels for utterance \( j \) of conversation \( i \), \( y_{i,j} \) is the expected class label of utterance \( j \) of conversation \( i \), \( q_{i,j} \) is the predicted sentiment intensity of utterance \( j \) of conversation \( i \), \( z_{i,j} \) is the expected sentiment intensity of utterance \( j \) of conversation \( i \), \( c(i) \) is the number of utterances in sample \( i \), \( \lambda \) is the L2-regularization weight, and \( \theta \) is the set of trainable parameters. We employ stochastic gradient descent based Adam [30] optimizer to train our network.

D. Bidirectional Emotion Recurrent Unit Variants

Our model has two different forms according to the source of context information, namely bidirectional emotion recurrent unit with global context (BiERU-gc) and bidirectional emotion recurrent unit with local context (BiERU-lc).

1) BiERU-gc: According to equation (1), GNTB extracts the context information from \( p_{t-1} \), integrates the context information into \( u_t \), and thus obtains the contextual utterance vector \( p_t \). Based on the definition of contextual utterance vector, \( p_{t-1} \) is the utterance vector that contains information of \( u_{t-1} \) and \( u_{t-2} \). In this case, the contextual utterance vector \( p_t \) holds the context information from all the preceding utterances \( u_1, u_2, \ldots, u_{t-1} \) in a recurrent manner. As shown in Fig. 2 (b), bidirectional ERUs enable \( p_t \) to capture not only the context information from preceding utterances, but also the context information from the future utterances \( u_{t+1}, u_{t+2}, \ldots, u_N \). The BiERU in Fig. 2 (a) is named as BiERU-gc.

2) BiERU-lc: Following equation (1), GNTB extracts the context information from the contextual utterance vector \( p_{t-1} \), and \( p_{t-1} \) contains the context information of all the preceding utterances \( u_1, u_2, \ldots, u_{t-2} \) as mentioned above. If replacing \( p_{t-1} \) with \( u_{t-1} \) in equation (1) and (2), \( p_t \) contains the information of \( u_{t-1} \) and \( u_t \). In other words, \( u_{t-1} \) is not only an utterance vector, but also works as the context of \( u_t \). As shown in Fig. 2 (b), bidirectional ERU makes \( p_t \) obtain the future information \( u_{t+1} \). In this case, GNTB extracts the context information from \( u_{t-1} \) and \( u_{t+1} \), which are the adjacent utterances of \( u_t \). We name this model as BiERU-lc.

IV. EXPERIMENTS

In this section, we conduct a series of comparative experiments to evaluate the performance of our proposed model (Codes will be available on our GitHub1) and perform a thorough analysis.

A. Datasets

We use three datasets for experiments, i.e., AVEC [31], IEMOCAP [27] and MELD [32], which are also used by some representative models such as DialogueRNN [2] and DialogueGCN [6]. We conduct the standard data partition rate (details in Table I).

Originally, these three datasets are multimodal datasets. Here, we focus on the task of textual conversational sentiment analysis, and only use the textual modality to conduct our experiments.

a) IEMOCAP: The IEMOCAP [27] is a dataset of two-way conversations involving with ten distinct participators. It is recorded as videos where every video clip contains a single dyadic dialogue, and each dialogue is further segmented

1https://github.com/senticnet
into utterances. Each utterance is labeled as one sentiment label from six sentiment labels, i.e., happy, sad, neutral, angry, excited and frustrated. The dataset includes three modalities: audio, textual and visual. Here we only use textual modality data in experiments.

b) AVEC: The AVEC dataset [31] is a modified version of the SEMAINE database [33] that contains interactions between human speakers and robots. Unlike IEMOCAP, each utterance in the AVEC dataset is given an annotation every 0.2 second with one of four real valued attributes, i.e, valence ([−1, 1]), arousal ([−1, 1]), expectancy ([−1, 1]), and power ([0, ∞]). Our experiments use the processed utterance-level annotation [2], and treat four affective attributes as four subsets for evaluation.

c) MELD: The MELD [32] is a multimodal and multiparty sentiment analysis/classification database. It contains textual, acoustic, and visual information for more than 13000 utterances from Friends TV series. The sentiment label of each utterance in a dialogue lies within one of the following seven sentiment classes: fear, neutral, anger, surprise, sadness, joy and disgust.

B. Baselines and Settings

To evaluate performance of our model, we choose the following models as strong baselines including the state-of-the-art methods.

a) c-LSTM [4]: The c-LSTM uses bidirectional LSTM [7] to learn contextual representation from the surrounding utterances. When combined with the attention mechanism, it becomes the c-LSTM+Att.

b) CMN [5]: This model utilizes memory network and two different GRUs [34] for two speakers for representation learning of utterance context from dialogue history.

c) DialogueRNN [2]: It distinguishes different parties in a conversation interactively, with three GRUs representing the speaker states, context, and emotion. It has several variants including DialogueRNN+Att with attention mechanism and bidirectional BiDialogueRNN.

d) DialogueGCN [6]: This model employs graph neural network based approach through which context propagation issue can be addressed, to detect sentiment in conversations.

e) AGHMN [3]: It utilizes hierarchical memory networks with BiGRUs for utterance reader and fusion, and attention mechanism for memory summarizing.

f) Settings: All the experiments are performed using CNN extracted features as described in Method section. For fair comparison with the state-of-the-art DialogueRNN model, we use their utterance representation directly.

To alleviate over-fitting, we employ Dropout [35] over the outputs of GNTB and EFE. For the nonlinear activation function, we choose the sigmoid function for sentiment classification and the relu function for sentiment regression. Our model is optimized by an Adam optimizer [30]. Hyper-parameters are tuned manually. Batch size is set as 1. We set the rank for all the experiments to \( r = 10 \). Our model is implemented using PyTorch [36].

C. Results

We compare our model with baselines on textual modality using three standard benchmarks. Overall, our model outperforms all the baseline methods including state-of-the-art models like DialogueRNN, DialogueGCN and AGHMN on these datasets, and markedly exceeds in some indicators as the results show in Table II.

For the IEMOCAP dataset as a classification problem, we use accuracy for each class, and weighted average of accuracy and f1-score for measuring the overall performance. As to the AVEC dataset, standard metrics for regression task including Pearson correlation coefficient (r) are used for evaluation. We use weighted average of accuracy as the measure of performance on MELD dataset.

1) Comparison with the State of the Art: We firstly compare our proposed BiERU with state-of-the-art methods DialogueGCN, DialogueRNN and AGHMN on IEMOCAP, AVEC and MELD, respectively.

a) IEMOCAP: As shown in Table II, our proposed BiERU-gc model exceeds the best model DialogueGCN by 0.12% and 0.02% in terms of weighted average accuracy and f1-score, respectively. And the BiERU-lc model pushes up state-of-the-art results by 0.86% and 0.47% for weighted average accuracy and f1-score, respectively. For all 14 indicators on IEMOCAP dataset, our models outperform at 7 indicators and has more balanced performances over these six classes. In particular, accuracy of “happy” of our proposed BiERU-gc is higher than the result of DialogueGCN by 14.22%. The experimental results indicate that BiERU model can effectively capture the contextual information and extract rich emotion features to boost the overall performance and achieve relatively balanced results.

b) AVEC: Among these four attributes, our model outperforms DialogueRNN for “valence”, “arousal” and “expectancy” attributes and gets the same results on “power” attribute. The Pearson correlation coefficient \( r \) of BiERU-gc is 0.04 higher than its counterpart in terms of “arousal” (Table III). As for the BiERU-lc model, it is 0.05 higher in \( r \). For the attributes “expectancy” and “valence”, the BiERU-lc model is 0.01 higher in \( r \). As for the attribute “power”, although our best model does not outperform the state-of-the-art method, it surpasses most of the other baseline methods including CMN and c-LSTM. Overall, BiERU-lc model works well on all the

\[\text{TABLE I: Statistical information and data partition of datasets used in this paper.}\]

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Partition</th>
<th>Utterance Count</th>
<th>Dialogue Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEMOCAP</td>
<td>train + val</td>
<td>5810</td>
<td>120</td>
</tr>
<tr>
<td></td>
<td>test</td>
<td>1625</td>
<td>31</td>
</tr>
<tr>
<td>AVEC</td>
<td>train + val</td>
<td>4368</td>
<td>63</td>
</tr>
<tr>
<td></td>
<td>test</td>
<td>1430</td>
<td>32</td>
</tr>
<tr>
<td>MELD</td>
<td>train + val</td>
<td>11098</td>
<td>1153</td>
</tr>
<tr>
<td></td>
<td>test</td>
<td>2610</td>
<td>280</td>
</tr>
</tbody>
</table>

\[\text{2 Extracted features of two datasets are available at https://github.com/senticnet/conv-emotion.}\]
TABLE II: Comparison with baselines on IEMOCAP and MELD datasets using textual modality. Average score of accuracy and f1-score are weighted. "-" represents no results reported in original paper.

<table>
<thead>
<tr>
<th>Methods</th>
<th>IEMOCAP</th>
<th>MELD</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Happy</td>
<td>Sad</td>
</tr>
<tr>
<td></td>
<td>Acc. F1</td>
<td>Acc. F1</td>
</tr>
<tr>
<td>c-LSTM</td>
<td>30.56</td>
<td>35.63</td>
</tr>
<tr>
<td>CMN</td>
<td>25.00</td>
<td>30.38</td>
</tr>
<tr>
<td>DialogueRNN</td>
<td>25.69</td>
<td>33.18</td>
</tr>
<tr>
<td>DialogueGCN</td>
<td>40.62</td>
<td>42.75</td>
</tr>
<tr>
<td>AGHMN</td>
<td>48.30</td>
<td>52.1</td>
</tr>
<tr>
<td>BiERU-gc</td>
<td>54.84</td>
<td>33.01</td>
</tr>
<tr>
<td>BiERU-lc</td>
<td>54.24</td>
<td>31.53</td>
</tr>
</tbody>
</table>

TABLE III: Comparison with baselines on AVEC dataset using textual modality. r stands for Pearson correlation coefficient.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Valence</th>
<th>Arousal</th>
<th>Expectancy</th>
<th>Power</th>
</tr>
</thead>
<tbody>
<tr>
<td>c-LSTM</td>
<td>0.16</td>
<td>0.25</td>
<td>0.24</td>
<td>0.10</td>
</tr>
<tr>
<td>CMN</td>
<td>0.23</td>
<td>0.29</td>
<td>0.26</td>
<td>-0.02</td>
</tr>
<tr>
<td>DialogueRNN</td>
<td>0.35</td>
<td>0.59</td>
<td>0.37</td>
<td>0.37</td>
</tr>
<tr>
<td>BiERU-gc</td>
<td>0.30</td>
<td>0.63</td>
<td>0.36</td>
<td>0.36</td>
</tr>
<tr>
<td>BiERU-lc</td>
<td>0.36</td>
<td>0.64</td>
<td>0.38</td>
<td>0.37</td>
</tr>
</tbody>
</table>

One possible explanation is that context information of a contextual utterance vector in BiERU-gc comes from all utterances in the current conversation. However, in BiERU-lc, the context information comes from neighborhood utterances. In this case, context information of BiERU-gc contains redundant information and thus has a negative impact on emotion feature extraction.

D. Case Study

Fig. 5 illustrates a conversation snippet classified by BiERU-lc method. In this snippet, person A is initially at a frustrated state while person B acts as a listener in the beginning. Then, person A changes his/her focus and questions person B on his/her job state. Person B tries to use his/her own experience to help person A get rid of the frustrating state. This snippet reveals that the sentiment of a speaker is relatively steady and the interaction between speakers may change the sentiment of a speaker. Our BiERU-lc method shows good ability in capturing the speaker’s sentiment (turns 9, 11, 12, 14) and the interaction between speakers (turn 10). The sentiment in turn 13 is very subtly. Turn 13 contains a little bit of frustration since he/she is not satisfied with his/her job state. However, considering that person B attempts to help person A, turn 13 is more likely to be in a neutral stand.
E. Visualization

We use visualization to provide some insights of the proposed model. Firstly, we visualize the confusion matrix in the form of a heat map to describe the performance of our BiERU-lc model. The heat maps of BiERU-lc on the IEMOCAP dataset are shown in Fig. 4. Our model has a balanced performance over all the sentiment classes.

Secondly, we perform deeper analysis of our proposed model and DialogueRNN by visualizing the learned emotion feature representations on IEMOCAP as shown in Fig. 6a and Fig. 6b. Vectors fed into the last dense layer followed by softmax for classification are regarded as emotion feature representations of utterances. We use principal component analysis [37] to reduce the dimension of emotion representations from our model (BiERU-lc) and DialogueRNN. The emotion representation is reduced to 3-dimensional. In Fig. 6a and Fig. 6b, each color represents a predicted sentiment label and the same color means the same sentiment label. The figures show that our model outperforms on extracting emotion features of utterances labeled with “happy”, which is consistent with the results in Table 2.

F. Efficiency Analysis

Our proposed model has advantages over DialogueRNN, which is the only one competitive method with public source code, in terms of convergence capacity, the number of trainable parameters, and training time. From the training curve in Fig. 7a, our model shows comparable convergence speed with its counterpart, but DialogueRNN turns to be easier to overfitting. Furthermore, BiERU with low-rank matrix approximation has fewer trainable parameters. For 100D feature input in IEMOCAP dataset, it has about 0.5M parameters, while DialogueRNN requires around 1M. For 600D MELD dataset, DialogueRNN has 2.9M parameters, and our model only has 0.6M. With much fewer parameters, our model consequently trains faster than its counterpart as shown in Fig. 7b, where training time is logged in a single NVIDIA GeForce GTX 965M. Our model is more parameter-efficient and less time-consuming for training.

G. Ablation Study

To further explore our proposed BiERU model, we perform ablation study on its two main components, i.e., GNTB and EFE. We conduct experiments on the IEMOCAP dataset with individual GNTB and EFE module separately, and their combination, i.e., the complete BiERU. Experimental results on the IEMOCAP dataset are illustrated in Table IV.

The performance of sole GNTB or EFE is low in terms of accuracy and f1-score. The reason is that outputs of GNTB mainly contain context information and outputs of EFE lack context information. However, when these two modules are combined together as the BiERU model, the accuracy and f1-score increase dramatically, which proves the effectiveness of our BiERU model. More importantly, the GNTB and EFE modules couples significantly well to enhance the performance.

<table>
<thead>
<tr>
<th>GNTB</th>
<th>EFE</th>
<th>Accuracy</th>
<th>F1-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>+</td>
<td>55.45</td>
<td>55.17</td>
</tr>
<tr>
<td>+</td>
<td>-</td>
<td>49.85</td>
<td>49.42</td>
</tr>
<tr>
<td>+</td>
<td>+</td>
<td>65.93</td>
<td>64.63</td>
</tr>
</tbody>
</table>

TABLE IV: Results of ablated BiERU on the IEMOCAP dataset. Accuracy and F1-score are weighted average.
Fig. 7: Training curve and time consumption. dRNN is the abbreviation of DialogueRNN.

V. CONCLUSION

In this paper, we proposed a fast, compact and parameter-efficient party-ignorant framework bidirectional emotional recurrent unit (BiERU) for sentiment analysis in conversations. Our proposed generalized neural tensor block (GNNTB), skilled at context compositionality, reduced the number of parameters and was suitable for different structures. Additionally, our EFE is capable of extracting high-quality emotion features for sentiment analysis. We proved that it is feasible to both simplify the model structure and improve performance simultaneously.

Our model outperforms current state-of-the-art models on three standard datasets in most cases. In addition, our method has the ability to model conversations with arbitrary turns and speakers, which plan to study further in the future.

ACKNOWLEDGEMENTS

This research is supported by the Agency for Science, Technology and Research (A*STAR) under its AME Programmatic Funding Scheme (Project #A18A2b0046).

REFERENCES