Real-Time Emotion Recognition via Attention Gated Hierarchical Memory Network

Wenxiang Jiao,[1] **Michael R. Lyu,**[1] **Irwin King**[1]
[1]Department of Computer Science and Engineering,
The Chinese University of Hong Kong, Shatin, N.T., Hong Kong, China
{wxjiao, lyu, king}@cse.cuhk.edu.hk

Abstract

Real-time emotion recognition (RTER) in conversations is significant for developing emotionally intelligent chatting machines. Without the future context in RTER, it becomes critical to build the memory bank carefully for capturing historical context and summarize the memories appropriately to retrieve relevant information. We propose an Attention Gated Hierarchical Memory Network (AGHMN) to address the problems of prior work: (1) Commonly used convolutional neural networks (CNNs) for utterance feature extraction are less compatible in the memory modules; (2) Unidirectional gated recurrent units (GRUs) only allow each historical utterance to have context before it, preventing information propagation in the opposite direction; (3) The Soft Attention for summarizing loses the positional and ordering information of memories, regardless of how the memory bank is built. Particularly, we propose a Hierarchical Memory Network (HMN) with a bidirectional GRU (BiGRU) as the utterance reader and a BiGRU fusion layer for the interaction between historical utterances. For memory summarizing, we propose an Attention GRU (AGRU) where we utilize the attention weights to update the internal state of GRU. We further promote the AGRU to a bidirectional variant (BiAGRU) to balance the contextual information from recent memories and that from distant memories. We conduct experiments on two emotion conversation datasets with extensive analysis, demonstrating the efficacy of our AGHMN models.

Introduction

Emotion recognition is a significant research topic because of the potential application in developing empathetic machines in present artificial intelligence (AI) era. We focus on the real-time scenario to detect the emotion state of speakers in an ongoing conversation at utterance-level. According to (Olson 1977), an utterance is a unit of speech bounded by breathes and pauses. We term this task as Real-Time Emotion Recognition (RTER). Inherently, emotion recognition is a multi-modal learning task that could involve text, video and audio features, but the text feature plays the most significant role (Chen et al. 1998; Poria, Cambria, and Gelbukh 2015; Poria et al. 2017; Hazarika et al. 2018b). Thus, in this paper, we tackle the RTER task in text conversations.

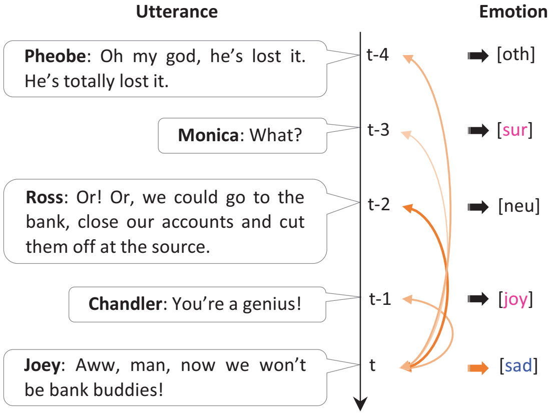
Utterance
Emotion
Pheobe: Oh my god, he's lost it. He's totally lost it.
Monica: What?
Ross: Or! Or, we could go to the bank, close our accounts and cut them off at the source.
Chandler: You're a genius!
Joey: Aww, man, now we won't be bank buddies!
t-4
t-3
t-2
t-1
t
[oth]
[sur]
[neu]
[joy]
[sad]

Figure 1: The specification of the RTER task.

Without the future context, in RTER, it becomes critical to exploit the contextual information from the historical utterances. For this purpose, one needs to take good care of two factors, i.e., the memory bank for capturing historical context, and the summarizing technique for the query to extract relevant information from the memory bank. The memory bank is usually built in a two-level fashion to simulate the hierarchical structure of conversations, i.e., words-to-utterance and utterances-to-conversation. Specifically, existing models (Hazarika et al. 2018a; 2018b; Majumder et al. 2019) obtain their memory banks by utilizing convolutional neural networks (CNNs) to learn utterance features and unidirectional gated recurrent units (GRUs) (Cho et al. 2014) to capture relationship of utterances. However, through our exploration, we find that a bidirectional GRU (BiGRU) learns better utterance features than commonly used CNNs. Moreover, the unidirectional GRU only allows each historical utterance to have context before but not after it, which may prevent information propagation in the opposite direction. As for the summarizing techniques, the commonly used Soft Attention produces a weighted sum of the memories, which can be regarded as a bag-of-memories. Just as the bag-of-words in word representation area (Mikolov et al. 2013) that lacks sensitivity to word order (Ling et al. 2015), the bag-of-memories loses the positional and ordering information of the memories, regardless of how the memory bank is built.

Incorporating these factors, in this paper, we propose an Attention Gated Hierarchical Memory Network (AGHMN) to better extract the utterance features and the contextual information for the RTER task. Specifically, we summarize our contributions as below: (1) We propose a Hierarchical Memory Network (HMN) to improve the utterance features and the memory bank for extracting contextual information. The HMN is essentially a two-level GRU encoder, including an utterance reader and a fusion layer. The utterance reader applies a BiGRU to model the word sequence of each utterance, which we demonstrate that it is more compatible with the hierarchical structure. The fusion layer adopts a BiGRU to read the historical utterances, which enables sufficient interaction between them. (2) We propose an Attention GRU (AGRU) to retain the positional and ordering information while summarizing the memories, and promote it to its bidirectional variant, i.e., BiAGRU, to capture more comprehensive context. The AGRU is formed by utilizing the attention weights of the query to the memories to update the internal state of a normal GRU. The final hidden state of AGRU serves as the contextual vector to help refine the representation of the query. The BiAGRU is dedicated to balancing the information from recent memories and that from distant memories. (3) We conduct experiments on two emotion conversation datasets with extensive analysis, demonstrating the efficacy of our proposed AGHMN models.

Related Work

Text Classification. Text-based emotion recognition is usually treated as a text classification problem. Previously proposed methods can be mainly divided into three categories: keyword-based methods (Wilson, Wiebe, and Hwa 2004), learning-based methods (Yang, Lin, and Chen 2007), and hybrid methods (Wu, Chuang, and Lin 2006). Nowadays, deep learning is dominating the text classification area due to its powerful capability in learning latent features. Representative methods include convolutional neural network (CNN) (Kim 2014), recurrent neural network (RNN) (Abdul-Mageed and Ungar 2017), and hierarchical attention network (HAN) (Tang, Qin, and Liu 2015). These work are customized for data unit without context, e.g. independent reviews or documents.

Context-Dependent Models. Recognizing emotion state of speakers in conversations requires that the query should take into account the context to carry accurate information. Existing work can be divided into two streams: the static models, and the dynamic models. The static models include sequence-based and graph-based (Zhang et al. 2019; Ghosal et al. 2019), the former of which let each utterance to have the utterances both in the history and the future as context. Among sequence-based static models, cLSTM (Poria et al. 2017) only adopts long short-term memory networks (LSTMs) (Hochreiter and Schmidhuber 1997) to capture the sequential relationship between the utterances. HiGRU (Jiao et al. 2019) employs a self-attention mechanism for context weighting and summarizing, as well as a residual connection for feature fusion. BiDialogueRNN (Majumder et al. 2019) is built on RNNs that keeps track of the individual party states throughout the conversation and uses this information for emotion recognition. These static models may adapt to the RTER task if we take their unidirectional variants. The dynamic models read the utterances in the order as they are generated so that each incoming utterance, i.e., the query, only depends on the historical utterances. These models include CMN (Hazarika et al. 2018b), DialogueRNN, and ICON (Hazarika et al. 2018a). Among them, CMN and ICON are customized for dyadic conversations, which incorporate memory networks (Sukhbaatar et al. 2015) to refine the contextual information and also consider the self- and inter-speaker emotional influence.

Our AGHMN models differ from these approaches in that we aim to produce better utterance features and memory representation by our proposed HMN and summarize the memories in a better way by our proposed AGRU and BiAGRU. We do not distinguish speakers explicitly as in DialogueRNN but find that the model itself can recognize the difference between speakers (see **Case Study**).

Task Specification

We first specify the task of Real-Time Emotion Recognition (RTER) as below:

Real-Time Emotion Recognition. Suppose that a conversation has proceeded for t turns so far with the utterance sequence $\mathcal{C}_t = \{u_1, \cdots, u_t\}$, the t-th utterance is the query utterance q, and the others are historical ones. As illustrated in Fig 1, each utterance expresses a major emotion e among a set of emotions $\mathcal{E}$, such as joy, sadness, and neutral. Our goal is to design and train a model $\mathcal{M}$ to predict the emotion expressed by q conditioned on the historical utterances.

Architecture

In this section, we will introduce the AGHMN model as illustrated in Fig 2, which consists of a Word Embedding Layer, a Hierarchical Memory Network, an Attention GRU, and a Classifier.

Word Embedding Layer

For an utterance in $\mathcal{C}_t$ composed by a sequence of words $u_{t'} = \{w_1, w_2, \cdots, w_N\}$, where $t' \in [1, t]$, N is the length of the utterance, and $w_n \in u_{t'}$ is the index of the word in the vocabulary. The utterance is fed into the word embedding layer to get a dense vector $\mathbf{x} \in \mathbb{R}^{d_w}$ for each word, where d_w is the size of the word vector. The weights of the word embedding layer are initialized by the publicly available 300-dimensional word2vec (Mikolov et al. 2013) vectors[1] trained on 100 billion words from Google News. The words not included in the word2vec vocabulary will be initialized by randomly generated vectors.

[1] https://code.google.com/archive/p/word2vec/

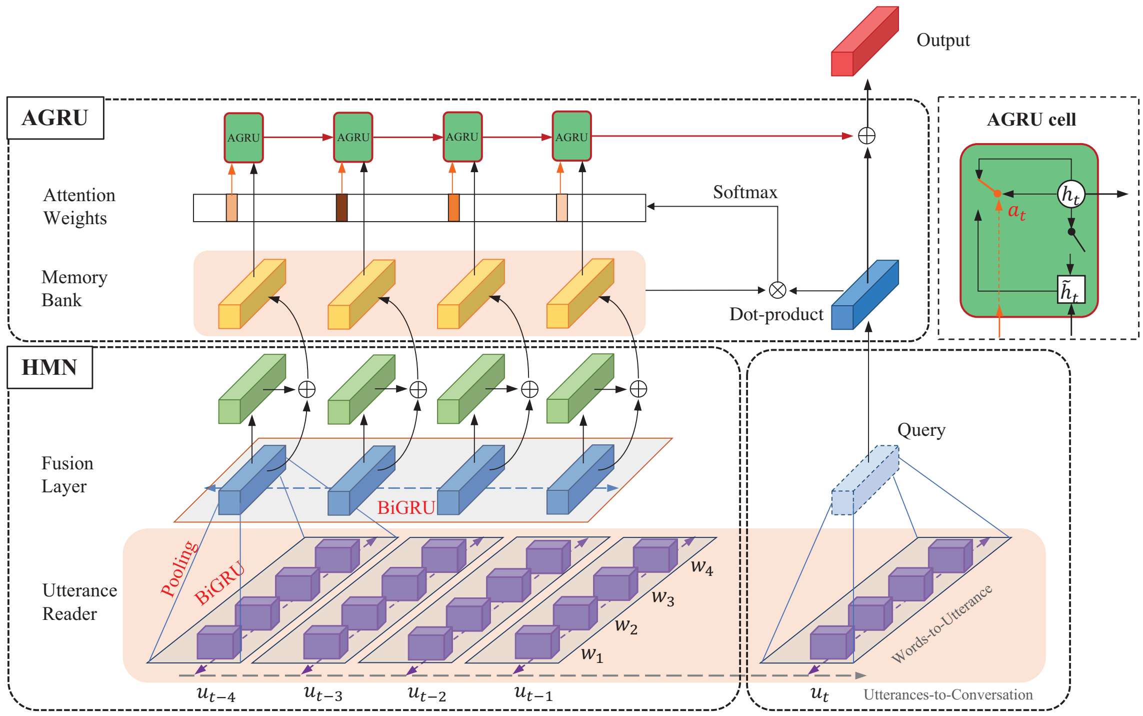

Figure 2: The architecture of our AGHMN model.

Hierarchical Memory Network

The Hierarchical Memory Network (HMN) is a two-level encoder, the lower one as an utterance reader and the upper one as the fusion layer.

Utterance Reader. Though current work (Hazarika et al. 2018a; 2018b; Majumder et al. 2019) always utilize CNN to extract utterance features, we decide to adopt a BiGRU. The BiGRU is able to model the word sequence while gather the contextual information for each word in two directions, making it better for understanding a sentence sufficiently. Actually, we find that the BiGRU performs much better than a commonly used 1-D CNN as the utterance reader (see Table 4). Specifically, the BiGRU takes an utterance represented by word vectors $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^N$ as input:

$$\overrightarrow{\mathbf{h}}_n = \overrightarrow{\mathbf{GRU}}(\mathbf{x}_n, \overrightarrow{\mathbf{h}}_{n-1}), \quad (1)$$

$$\overleftarrow{\mathbf{h}}_n = \overleftarrow{\mathbf{GRU}}(\mathbf{x}_n, \overleftarrow{\mathbf{h}}_{n+1}), \quad (2)$$

where $\overrightarrow{\mathbf{h}}_n, \overleftarrow{\mathbf{h}}_n \in \mathbb{R}^{d_1}$ are the hidden states of the forward GRU and the backward one, respectively, and d_1 is the hidden size. The hidden states of both directions are concatenated and fed to the max-over-time pooling layer. The resulted vector $\overleftrightarrow{\mathbf{h}}$ is then transformed to be the utterance embedding $\mathbf{u} \in \mathbb{R}^{d_1}$ through a tanh layer:

$$\overleftrightarrow{\mathbf{h}} = \mathrm{maxpool}(\{[\overrightarrow{\mathbf{h}}_n, \overleftarrow{\mathbf{h}}_n]\}_{n=1}^N) \quad (3)$$

$$\mathbf{u} = \tanh(\mathbf{W}_u \overleftrightarrow{\mathbf{h}} + \mathbf{b}_u), \quad (4)$$

where $\mathbf{W}_u \in \mathbb{R}^{d_1 \times 2d_1}$, and $\mathbf{b}_u \in \mathbb{R}^{d_1}$.

Fusion Layer. At t-step, the representation of the query comes from the utterance encoder: $\mathbf{q}_t = \mathbf{u}_t$. For each query $\mathbf{q}_t$, we build a memory bank $\mathbf{M}_t$ based on the most recent K historical utterances. Since the K utterances maintain a sequence, we hope to make them interact with each other so as to refine the memory representation. As shown in Fig 2, we consider two types of memory banks here:

- **Unidirectional Fusion (UniF).** Firstly, we utilize a unidirectional GRU to read these K utterances to model the sequential relationship between them. The independent utterance embeddings $\{\mathbf{u}_{t-K-1+k}\}_{k=1}^K$ are fed to the GRU, and are then connected to the output of the GRU to form the memory bank: $\mathbf{M}_t = \{\overrightarrow{\mathbf{GRU}}(\mathbf{u}_{t-K-1+k}) + \mathbf{u}_{t-K-1+k}\}_{k=1}^K$.
- **Bidirectional Fusion (BiF).** The UniF memory bank only allows each memory to have the context before it but not after it, which may prevent information propagation from the opposite direction. To address such a problem, we propose to read the K utterances through a BiGRU, and combine the output and the input to form the memory bank: $\mathbf{M}_t = \{\overleftrightarrow{\mathbf{GRU}}(\mathbf{u}_{t-K-1+k}) + \mathbf{u}_{t-K-1+k}\}_{k=1}^K$. For simplicity, we use $\overleftrightarrow{\mathbf{GRU}}$ to denote the sum of the hidden states in the two directions of the BiGRU.

Attention GRU

Generally, the query in a conversation depends on the context it follows. Thus, it is crucial to weight and summarize the context to refine the representation of the query. This process is usually realized by an attention layer (Hazarika et al. 2018a; 2018b), which allows the query to interact with the memory bank and produce a contextual vector $\mathbf{c}_t$.

Context Weighting. The attention layer should be able to retrieve relevant context from the memory bank to help predict the emotion expressed by the query. To weight the importance of each memory to the query, we adopt the dot-product attention with a $\mathrm{softmax}$ normalization. As a result, at t-step, the weight of the kth memory $\mathbf{M}_{t,k}$ will be:

$$a_k = \frac{\exp(\mathbf{q}_t^\top \mathbf{M}_{t,k})}{\sum_{k'=1}^{K} \exp(\mathbf{q}_t^\top \mathbf{M}_{t,k'})}. \tag{5}$$

Context Summarizing. Conventionally, the contextual vector $\mathbf{c}_t$ can be produced by Soft Attention as a weighted sum of the memories, i.e., $\mathbf{c}_t = \sum_{k=1}^{K} a_k \mathbf{M}_{t,k}$. This method is efficient for computing, but just as the bag-of-words in word representation area (Mikolov et al. 2013; Ling et al. 2015) it loses both the positional and ordering information of the memories. Thus, we propose an Attention GRU (AGRU) which uses the attention weight of the query to the memories to update the internal state $\tilde{\mathbf{h}}_t$ of a normal GRU. As a result, the output of the AGRU is:

$$\mathbf{h}_k = a_k \circ \tilde{\mathbf{h}}_k + (1 - a_k) \circ \mathbf{h}_{k-1}. \tag{6}$$

The GRU is advantageous for retaining the positional and ordering information of the memories and the attention weight controls the amount of information to be passed to the next step. We take the final hidden state of the AGRU as the contextual vector, i.e., $\mathbf{c}_t = \mathbf{h}_K$, then the refined query representation, i.e. the output, will be:

$$\mathbf{o}_t = \mathbf{q}_t + \mathbf{c}_t. \tag{7}$$

Furthermore, considering the tendency of RNNs to better represent recent inputs (Bahdanau, Cho, and Bengio 2015), the contextual vector from AGRU also tends to carry more information of the most recent memories. Accordingly, a backward AGRU can better represent memories distant from the query. Therefore, we promote AGRU to its bidirectional variant, i.e., BiAGRU, so as to make a balance between the information from recent memories and that from distant memories. We believe the BiAGRU is capable of capturing more comprehensive context from the memory bank, especially for long conversations. As a result, the contextual vectors produced by a BiAGRU are expressed as:

$$\mathbf{c}_t^f = \overrightarrow{\mathbf{AGRU}}(\mathbf{M}_{t,K}, a_K, \overrightarrow{\mathbf{h}}_{K-1}), \tag{8}$$

$$\mathbf{c}_t^b = \overleftarrow{\mathbf{AGRU}}(\mathbf{M}_{t,1}, a_1, \overleftarrow{\mathbf{h}}_{2}), \tag{9}$$

which are used to refine the query representation similarly.

Classifier

The refined representation of the query from the AGRU is used for prediction by a $\mathrm{softmax}$ layer:

$$\hat{\mathbf{y}}_t = \mathrm{softmax}(\mathbf{W}_o \mathbf{o}_t + \mathbf{b}_o), \tag{10}$$

where $\mathbf{W}_o \in \mathbb{R}^{d_1 \times |\mathcal{E}|}$, $\mathbf{b}_o \in \mathbb{R}^{|\mathcal{E}|}$, and $|\mathcal{E}|$ is the number of emotion classes.

Dataset	Fold	No. of Utt	No. of Conv	Avg. length of Conv
IEMOCAP	train/val	5810	120	48.4
	test	1623	31	52.4
MELD	train/val	11098	1153	9.6
	test	2610	280	9.3

Table 1: Summary of the two conversation emotion datasets: IEMOCAP, and MELD.

We train the AGHMN model by a cross-entropy loss function expressed as:

$$\mathcal{L} = \frac{1}{\sum_{l=1}^{L} T_l} \sum_{t=1}^{T_l} \sum_{e=1}^{|\mathcal{E}|} \mathbf{y}_t^e \log(\hat{\mathbf{y}}_t^e), \tag{11}$$

where T_l is the number of utterances in the lth conversation, and L is the total number of conversations in the training set. $\mathbf{y}_t$ denotes the one-hot vector of the target emotion labels, and $\mathbf{y}_t^e$ and $\hat{\mathbf{y}}_t^e$ are the elements of $\mathbf{y}_t$ and $\hat{\mathbf{y}}_t$ for the emotion class e, respectively.

Experimental Setup

In this section, we will present the details of our experimental setup, including datasets, compared methods, implementation, and training.

Datasets. We train and test our model on two conversation emotion datasets, namely, IEMOCAP (Busso et al. 2008), and MELD (Poria et al. 2019a).

- **IEMOCAP**[2]: The IEMOCAP dataset contains the acts of 10 speakers in a dyadic conversation fashion, providing text, audio, and video features. We follow the previous work (Hazarika et al. 2018a) to use the first four sessions of transcripts as the training set, and the last one as the testing set. The validation set is extracted from the randomly-shuffled training set with the ratio of 80:20. Also, we focus on recognizing six emotion classes, namely, *happy*, *sad*, *neutral*, *angry*, *excited*, and *frustrated*.
- **MELD**[3]: The MELD dataset (Poria et al. 2019a) is an extended version of the EmotionLines dataset (Hsu et al. 2018). The data comes from the Friends TV series with multiple speakers involved in the conversations. It is split into training, validation, and testing sets with 1039, 114, and 280 conversations, respectively. Each utterance has been labelled by one of the seven emotion types, namely, *anger*, *disgust*, *sadness*, *joy*, *neutral*, *surprise* and *fear*.

Compared Methods. With the different combination of memory banks and AGRUs, we consider four variants of AGHMN[4] for experiments: UniF-AGRU, UniF-BiAGRU, BiF-AGRU, and BiF-BiAGRU. These variants are compared to the following baselines:

[2] https://sail.usc.edu/iemocap/

[3] https://github.com/SenticNet/MELD

[4] https://github.com/wxjiao/AGHMN

Model	**IEMOCAP: Emotion Classes**														
	happy		*sad*		*neutral*		*angry*		*excited*		*frustrated*		**Avg.**		
	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	mF1
memnet$^{\triangle}$	25.7	33.5	55.5	61.7	58.1	52.8	59.3	55.3	51.5	58.3	67.2	59.0	55.7	55.1	53.4
CMN$^{\triangle}$	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	54.3
ICON$^{\diamond}$	-	-	-	-	-	-	-	-	-	-	-	-	58.3	57.9	-
DialogueRNN$^{\triangle}$	31.2	33.8	66.1	69.8	63.0	57.7	61.7	**62.5**	61.5	64.4	59.5	59.4	59.3	59.8	57.9
scLSTM	37.5	43.4	67.7	69.8	**64.2**	55.8	61.9	61.8	51.8	59.3	61.3	60.2	59.2	59.1	58.4
DialogueRNN	33.5	35.4	**69.0**	68.7	54.1	54.7	**67.1**	61.1	55.9	60.4	62.9	60.3	58.3	58.1	56.7
UniF-AGRU	42.7	51.1	63.4	68.0	61.3	57.4	61.9	61.8	67.5	**70.5**	64.1	60.5	61.9	61.8	61.5
UniF-BiAGRU	49.7	50.6	64.7	69.9	60.3	59.0	55.1	60.5	67.4	69.6	68.6	62.1	62.8	62.7	61.9
BiF-AGRU	48.3	52.1	68.3	**73.3**	61.6	58.4	57.5	61.9	**68.1**	69.7	67.1	**62.3**	**63.5**	**63.5**	**63.0**
BiF-BiAGRU	**62.1**	**54.5**	66.6	72.7	63.9	**59.4**	58.4	61.0	58.5	66.6	64.8	61.6	62.8	63.0	62.6

Table 2: Performance of AGHMN models on IEMOCAP.

Model	**MELD: Emotion Classes**																
	neutral		*surprise*		*fear*		*sadness*		*joy*		*disgust*		*anger*		**Avg.**		
	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	mF1
CNN†	-	74.9	-	45.5	-	3.7	-	21.1	-	49.4	-	8.2	-	34.5	-	55.0	33.9
scLSTM	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	35.3
DialogueRNN	72.1	73.5	**54.4**	49.4	1.6	1.2	**23.9**	23.8	52.0	50.7	1.5	1.7	**41.9**	**41.5**	56.1	55.9	34.5
UniF-AGRU	80.3	75.1	53.7	49.1	**9.8**	10.6	19.7	25.5	50.5	51.1	**14.0**	16.4	33.9	38.2	58.8	57.0	38.0
UniF-BiAGRU	**83.4**	**76.4**	49.1	**49.7**	9.2	**11.5**	21.6	**27.0**	52.4	**52.4**	12.2	14.0	34.9	39.4	**60.3**	**58.1**	**38.6**
BiF-AGRU	81.6	75.5	50.5	49.0	8.0	10.4	22.3	26.9	51.7	51.7	13.7	**17.2**	34.5	38.6	59.5	57.5	38.5
BiF-BiAGRU	80.7	75.4	50.7	47.9	7.2	10.0	20.9	26.3	**53.9**	52.1	10.4	12.8	33.9	38.1	59.2	57.1	37.5

Table 3: Performance of AGHMN models on MELD.

- **scLSTM** (Poria et al. 2017) is the unidirectional variant that classifies utterances using historical utterance as context realized by a LSTM.
- **CMN** (Hazarika et al. 2018b) models separate contexts for both speaker and listener to an utterance. These contexts are stored as memories to aid the prediction of an incoming utterance.
- **DialogueRNN** (Majumder et al. 2019) is the unidirectional variant with an attention layer that keeps track of the individual party states throughout the conversation and uses this information for emotion classification.
- **ICON** (Hazarika et al. 2018a) incorporates the self- and inter-speaker emotional influences into global memories to help predict the emotional orientation of utterances. It is a unidirectional model with only historical context.

For IEMOCAP, we refer to the results of **memnet** (Sukhbaatar et al. 2015), CMN, DialogueRNN from $^{\triangle}$ (Majumder et al. 2019), and that of ICON from $^{\diamond}$ (Hazarika et al. 2018a). For MELD, we refer to the results of **CNN** from † (Poria et al. 2019b). We re-run scLSTM and DialogueRNN for both datasets. CMN and ICON cannot be adapted for MELD because they are customized for dyadic conversations and may encounter scalability issue for multi-party conversation datasets (Poria et al. 2019b).

Implementation. We implement scLSTM and our proposed AGHMN models from scratch on the Pytorch[5] framework. For the extraction of textual feature, we follow (Hazarika et al. 2018a) to adopt a 1-D CNN with filters of 3, 4, and 5 each with 64 feature maps. The convolution result of each filter is fed to a max-over-time pooling layer. The pooling results are concatenated and transformed to the utterance embeddings via a relu layer. The hidden size of the LSTM is 100. As for our AGHMN models, the hidden sizes of GRUs and AGRUs are also 100. By default, the context-window size K for the memory bank is 40 for IEMOCAP and 10 for MELD, which are around the average conversation lengths of each dataset, respectively. Besides, the implementation of DialogueRNN comes from the open-source codes[6] provided by the authors of DialogueRNN.

Training. We choose Adam (Kingma and Ba 2015) optimizer with an initial learning rate $lr = 5 \times 10^{-4}$. To regulate the models, we clip the gradients of model parameters with a max norm of 5 and apply dropout with a drop rate of 0.3. We monitor the macro-averaged F1-score (mF1) of the validation sets during training and decay the learning rate by 0.95 once the mF1 stops increasing. The training process is terminated by early stopping with a patience of 10.

Results

Table 2 and Table 3 present the results on IEMOCAP and MELD testing sets, respectively. For both datasets, we report the accuracy and F1-score (Tong et al. 2017) for each emotion class and evaluate the overall classification performance using their weighted averages of all emotion classes. We also report the macro-average of F1-score (mF1) to re-

[5] https://pytorch.org/

[6] https://github.com/SenticNet/conv-emotion/tree/master/DialogueRNN

Reader	UniF-BiAGRU			BiF-AGRU		
	Acc	F1	mF1	Acc	F1	mF1
CNN	59.8	59.9	59.4	60.2	60.2	59.9
CNN$_{soft}$	58.8	58.7	58.1	59.4	59.3	58.8
BiLSTM	62.0	62.0	61.2	62.8	62.7	61.9
BiGRU	**62.8**	**62.7**	**61.9**	**63.5**	**63.5**	**63.0**

Table 4: Testing results on IEMOCAP, with CNN, BiLSTM, and BiGRU as the utterance reader, respectively.

flect the model performance on minority emotion classes, since the weighted-average is compromised by the majority classes. Each result provided by us in the tables is the average value of 10 times repeated experiments.

In Table 2, all of our AGHMN models perform better than the compared models. BiF-AGRU attains the best overall performance with significant improvement over the strongest baseline DialogueRNN$^{\triangle}$ (+4.2% Acc, +3.7% F1, +5.1% mF1). For each emotion, our ADHMN models achieve at least competitive performance as DialogueRNN$^{\triangle}$. In particular, our models attain very large improvement on *happy* (at least +11.5% Acc, +17.3% F1), which is the emotion with the least utterances. This demonstrates the capability of our models in recognizing minority emotion classes.

In Table 3, all the AGHMN models also outperform the compared methods significantly. But this time, UniF-BiAGRU becomes the best one (+2.8% Acc, +2.2% F1, +3.3% mF1). Our models perform the best on most emotion classes, especially the two minority classes *fear* and *disgust*, though this is accompanied by the performance degradation on *anger*. But referring to the mF1 value, it is safe to say that our models produce much more balanced results.

Baseline Methods. The bcLSTM model implemented by us is very strong, performing better than all the other baselines on both datasets except DialogueRNN$^{\triangle}$ on IEMOCAP. However, the results of DialogueRNN run by us are slightly worse than DialogueRNN$^{\triangle}$. This may be because we follow the default settings of the provided codes, which are customized for BiDialogueRNN. No matter what, our AGHMN models outperform them on the two datasets, suggesting the efficacy of our context-modeling scheme.

AGHMN variants. UniF-AGRU performs the worst among all the four variants on both datasets. Consistently, UniF-BiAGRU and BiF-AGRU outperform UniF-AGRU, which demonstrates the superiority of BiAGRU over AGRU and BiF over UniF, respectively. However, BiF-BiAGRU does not attain the best performance, which we speculate that the model becomes too deep to learn from the two datasets. In fact, the performance difference between the variants on MELD is limited. This is mainly because the conversations in MELD contain much fewer turns than that in IEMOCAP, making it less sensitive to different modules.

Mem-Attn	IEMOCAP			MELD		
	Acc	F1	mF1	Acc	F1	mF1
UniF-Soft	60.7	60.5	59.5	58.0	56.5	37.7
BiF-Soft	62.3	62.2	61.8	58.6	56.9	37.5
UniF-AGRU	61.9	61.8	61.5	58.8	57.0	38.0
UniF-BiAGRU	**62.8**	**62.7**	**61.9**	**60.3**	**58.1**	**38.6**

Table 5: Testing results on IEMOCAP and MELD, with different choices of attention and memory bank.

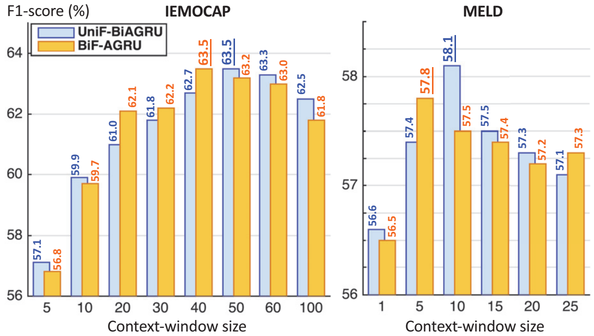

Figure 3: Trends in the performance of UniF-BiAGRU and BiF-AGRU with varying context-window size K.

Model Analysis

Utterance Readers. We argue that the BiGRU is a better reader for utterances. Here, we test UniF-BiAGRU and BiF-AGRU on IEMOCAP with the utterance reader replaced by a 1-D CNN and a bidirectional LSTM (BiLSTM), respectively. The results are reported in Table 4, where we can easily conclude that the RNN variants (BiLSTM, and BiGRU) surpass 1-D CNN with significant margins. For BiLSTM and BiGRU, the latter one performs better because GRUs are usually more powerful than LSTMs on small datasets. These results indicate that RNNs are more compatible in this Hierarchical Memory Network. Moreover, as BiGRU, BiLSTM attains better performance in BiF-AGRU than in UniF-BiAGRU, suggesting the advantage of BiF. In addition, we also conduct experiments for the CNN reader with Soft Attention, named as CNN$_{soft}$, which encounters the degradation of performance. It demonstrates that the improvement of our models is not achieved by only the BiGRU reader.

Attention Choices & Memory Banks. We investigate the advantage of AGRU over Soft Attention here. As presented in Table 5, with the UniF memory bank, AGRU attains better results than Soft Attention on both datasets. The bidirectional variant of AGRU extends the advantage even further. It is noteworthy that we also include the results of Soft Attention with the BiF memory, which are considerably better than that with UniF. This further verifies that BiF can produce better memory representation.

Context-Window Size. We plot the performance trend of UniF-BiAGRU and BiF-AGRU on both datasets when varying the context-window size K for building the memory bank. On both datasets, the two models follow similar trends such that the performance increases at first and falls then with the increase of K. On IEMOCAP, the best results are obtained at $K = 40$ for BiF-AGRU and $K = 50$ for UniF-

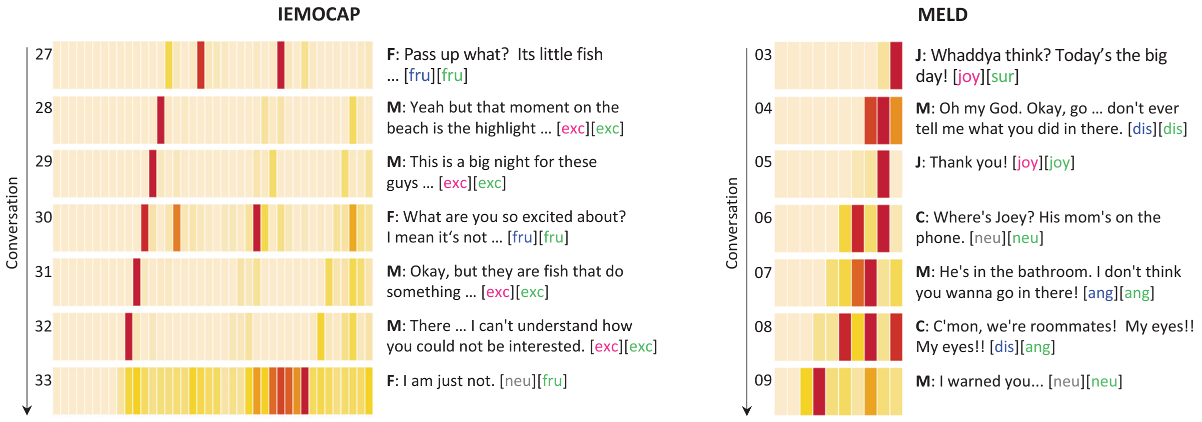

Figure 4: Evolution of memory selection as a conversation develops. The attention weights come from BiF-AGRU for both IEMOCAP and MELD. Each utterance is tagged with two labels, the first is ground truth and the second is the prediction by BiF-AGRU. IEMOCAP: M: Male, F: Female; MELD: J: Joey, M: Monica, C: Chandler.

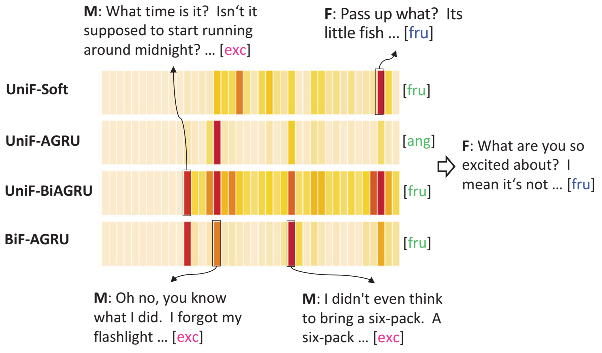

Figure 5: Comparison of memory selection between UniF-Soft and UniF-AGRU, UniF-BiAGRU, and BiF-AGRU.

BiAGRU, which align with the average length of conversations in the dataset (see Table 1). On MELD, the best values of K are 5 and 10, respectively, which are much lower than that on IEMOCAP. We speculate that it is because the data of MELD comes from the Friends TV sitcom with more rapid emotion fluctuation. Therefore, a longer context may result in model confusion. In contrast, the emotion state in IEMOCAP evolves much more gently.

Case Study

Attention Evolution. We find that the selection of memory differs between speakers as a conversation develops, though we do not distinguish the speakers explicitly in our model. In Fig 4, we visualize the attention weights of BiF-AGRU tested on a conversation fragment of IEMOCAP and MELD, respectively. For IEMOCAP, the Male is *excited* from the beginning so that the four utterances presented here pay the most attention to the first utterance of the conversation. In contrast, the attention of the Female is distributed over several historical utterances, including the first one and some intermediate ones (see Fig 5) that make her *frustrated*. As for MELD, Joey is joyful all the time and his attention is paid to his last *joyful* utterance. Monica pays her most attention to the conversation between her and Joey, providing clues for Chandler. Chandler focuses on Joey and where Joey is.

Attention Comparison. The selection of memory also varies between different attention mechanisms. In Fig 5, given a query utterance expressed by the Female with *frustrated* emotion, Soft Attention focuses on the utterance that also expresses *frustrated*. AGRU pays most attention to one utterance that could be the reason for the Female's *frustrated* emotion, but it classifies the query emotion as *angry* in this example. BiAGRU can sense both kinds of clues, providing more comprehensive memory. BiF improves the memory representation and helps AGRU to extract the memory as comprehensively as UniF-BiAGRU.

Error Analysis. In Fig 4, the 33rd utterance of IEMOCAP is recognized as *frustrated* not *neutral*. We argue that the original annotation might not be accurate. Given several turns of *frustrated* and the latest response (34th) from the Male, the Female could express *frustrated* again. Still, we cannot deal with minority classes very well on MELD, such as *disgust*. With more data or multimodal features to disambiguate with other emotions, this issue might be better resolved.

Conclusions

We propose an Attention Gated Hierarchical Memory Network (AGHMN) for Real-Time Emotion Recognition. Firstly, the proposed Hierarchical Memory Network improves the quality of utterance features and memories. Then, the proposed Attention GRU summarizes better contextual information than the commonly used Soft Attention. We conduct extensive experiments on two emotion conversation datasets, and our models outperform the state-of-the-art approaches with significant margins. Lastly, ablation studies and attention visualization demonstrate the efficacy of each component of our AGHMN models.

Acknowledgments

This work is supported by the Research Grants Council of the Hong Kong Special Administrative Region, China (No. CUHK 14208815 and No. CUHK 14210717 of the General Research Fund). We thank the three anonymous reviewers for their insightful suggestions on various aspects of this work.

References

Abdul-Mageed, M., and Ungar, L. H. 2017. Emonet: Fine-grained emotion detection with gated recurrent neural networks. In *ACL*, 718–728.

Bahdanau, D.; Cho, K.; and Bengio, Y. 2015. Neural machine translation by jointly learning to align and translate. In *ICLR*.

Busso, C.; Bulut, M.; Lee, C.; Kazemzadeh, A.; Mower, E.; Kim, S.; Chang, J. N.; Lee, S.; and Narayanan, S. 2008. IEMOCAP: interactive emotional dyadic motion capture database. *Language Resources and Evaluation* 42(4):335–359.

Chen, L. S.; Huang, T. S.; Miyasato, T.; and Nakatsu, R. 1998. Multimodal human emotion/expression recognition. In *FG*, 366–371.

Cho, K.; van Merrienboer, B.; Gülçehre, Ç.; Bahdanau, D.; Bougares, F.; Schwenk, H.; and Bengio, Y. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *EMNLP*, 1724–1734.

Ghosal, D.; Majumder, N.; Poria, S.; Chhaya, N.; and Gelbukh, A. F. 2019. Dialoguegcn: A graph convolutional neural network for emotion recognition in conversation. *CoRR* abs/1908.11540.

Hazarika, D.; Poria, S.; Mihalcea, R.; Cambria, E.; and Zimmermann, R. 2018a. ICON: interactive conversational memory network for multimodal emotion detection. In *EMNLP*, 2594–2604.

Hazarika, D.; Poria, S.; Zadeh, A.; Cambria, E.; Morency, L.; and Zimmermann, R. 2018b. Conversational memory network for emotion recognition in dyadic dialogue videos. In *NAACL-HLT*, 2122–2132.

Hochreiter, S., and Schmidhuber, J. 1997. Long short-term memory. *Neural Computation* 9(8):1735–1780.

Hsu, C.; Chen, S.; Kuo, C.; Huang, T. K.; and Ku, L. 2018. Emotionlines: An emotion corpus of multi-party conversations. In *LREC*.

Jiao, W.; Yang, H.; King, I.; and Lyu, M. R. 2019. Higru: Hierarchical gated recurrent units for utterance-level emotion recognition. In *NAACL-HLT*, 397–406.

Kim, Y. 2014. Convolutional neural networks for sentence classification. In *EMNLP*, 1746–1751.

Kingma, D. P., and Ba, J. 2015. Adam: A method for stochastic optimization. In *ICLR*.

Ling, W.; Tsvetkov, Y.; Amir, S.; Fermandez, R.; Dyer, C.; Black, A. W.; Trancoso, I.; and Lin, C. 2015. Not all contexts are created equal: Better word representations with variable attention. In *EMNLP*, 1367–1372.

Majumder, N.; Poria, S.; Hazarika, D.; Mihalcea, R.; Gelbukh, A. F.; and Cambria, E. 2019. Dialoguernn: An attentive RNN for emotion detection in conversations. In *AAAI*, 6818–6825.

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. S.; and Dean, J. 2013. Distributed representations of words and phrases and their compositionality. In *NeurIPS*, 3111–3119.

Olson, D. 1977. From utterance to text: The bias of language in speech and writing. *Harvard educational review* 47(3):257–281.

Poria, S.; Cambria, E.; Hazarika, D.; Majumder, N.; Zadeh, A.; and Morency, L. 2017. Context-dependent sentiment analysis in user-generated videos. In *ACL*, 873–883.

Poria, S.; Hazarika, D.; Majumder, N.; Naik, G.; Cambria, E.; and Mihalcea, R. 2019a. MELD: A multimodal multi-party dataset for emotion recognition in conversations. In *ACL*, 527–536.

Poria, S.; Majumder, N.; Mihalcea, R.; and Hovy, E. H. 2019b. Emotion recognition in conversation: Research challenges, datasets, and recent advances. *IEEE Access* 7:100943–100953.

Poria, S.; Cambria, E.; and Gelbukh, A. F. 2015. Deep convolutional neural network textual features and multiple kernel learning for utterance-level multimodal sentiment analysis. In *EMNLP*, 2539–2544.

Sukhbaatar, S.; Szlam, A.; Weston, J.; and Fergus, R. 2015. End-to-end memory networks. In *NeurIPS*, 2440–2448.

Tang, D.; Qin, B.; and Liu, T. 2015. Document modeling with gated recurrent neural network for sentiment classification. In *EMNLP*, 1422–1432.

Tong, E.; Zadeh, A.; Jones, C.; and Morency, L. 2017. Combating human trafficking with multimodal deep models. In *ACL*, 1547–1556.

Wilson, T.; Wiebe, J.; and Hwa, R. 2004. Just how mad are you? finding strong and weak opinion clauses. In *AAAI*, 761–769.

Wu, C.; Chuang, Z.; and Lin, Y. 2006. Emotion recognition from text using semantic labels and separable mixture models. *ACM TALIP* 5(2):165–183.

Yang, C.; Lin, K. H.; and Chen, H. 2007. Emotion classification using web blog corpora. In *WI*, 275–278.

Zhang, D.; Wu, L.; Sun, C.; Li, S.; Zhu, Q.; and Zhou, G. 2019. Modeling both context- and speaker-sensitive dependence for emotion detection in multi-speaker conversations. In *IJCAI*, 5415–5421.