DialogueRNN: An Attentive RNN for Emotion Detection in Conversations

 **순환 신경망 네트워크(RNN)**을 이용하여 개발 당사자를 추적해 *감정 분류*를 해보자!

1. Introduction

- 현재의 시스템은 대화의 다른 당사자를 구별하지 않음

  → DialogueRNN 시스템을 소개

- 감정과 관련된 세 가지 주요 측면(= 화자, 이전 발화의 맥락, 이전 발화의 감정)을 감지
  - 대화 시 당사자는 구별된 역할을 가짐

    ⇒ 맥락을 추출하기 위해서 주어진 시점에서의 화자와 청취자의 이전 턴을 **모두** 고려
- 3개의 게이트 순환 유닛(GRU)를 활용
  - 들어오는 발화는 글로벌 GRU와 파티 GRU라고 불리는 두 개의 GRU에 공급되어 맥락과 당사자 상태를 업데이트
  - 글로벌 GRU는 해당 당사자 정보를 인코딩하는 동안 발화를 인코딩
  - 업데이트된 화자 상태가 주어진 발화의 감정 표현을 디코딩하기 위해 감정 GRU에 공급됨
  - 파티 GRU는 동일한 당사자의 두 시퀀셜 상태 간의 관계를 모델링

2. Related Work

- 감정 인식은 자연어 처리, 심리학 등에서 주목받고 있으며, 다양한 방법이 연구되고 있음

- 최근에는 RNN 기반의 딥러닝이 감정 인식에 성공적으로 활용되었음
  - 대화에서 감정은 상호작용적 현상으로 중요하며, 이를 고려한 모델이 필요

    ⇒ 이를 위해 순환 신경망을 활용
  - 메모리 네트워크를 사용한 이전 연구도 대화에서 감정 인식을 개선하는 데 도움이 되었음

3. Methodology

3-1. 문제 정의

- 주어진 대화에서 M명의 참가자($p_1, p_2, ..., p_M$)가 구성하는 대화의 발화인 $u_1, u_2, ..., u_N$의 감정 라벨(= happy, sad, neutral, angry, excited, frustrated)을 예측하는 것
  - 여기서 발화 u_t는 파티(참가자) $p_s(u_t)$에 의해 발화되며, s는 발화와 해당 파티의 색인 사이의 매핑임
  - $u_t \in R^{D_m}$은 특성 추출기를 사용하여 얻은 발화 표현임

3-2. Unimodal Feature Extraction

텍스트 특성 추출

- 합성곱 신경망(CNN) 활용
  - 각 발화에서 n-gram 특성을 3, 4 및 5 크기의 세 가지 서로 다른 컨볼루션 필터를 사용하여 얻음
  - 출력은 max-pooling을 거쳐 ReLU 활성화를 거침
  - 이후 100차원의 fc layer에 공급됨 → 텍스트 발화 표현

오디오 및 비주얼 특성 추출

- 3D-CNN 및 openSMILE을 사용

3-3. Our Model(DialogueRNN)

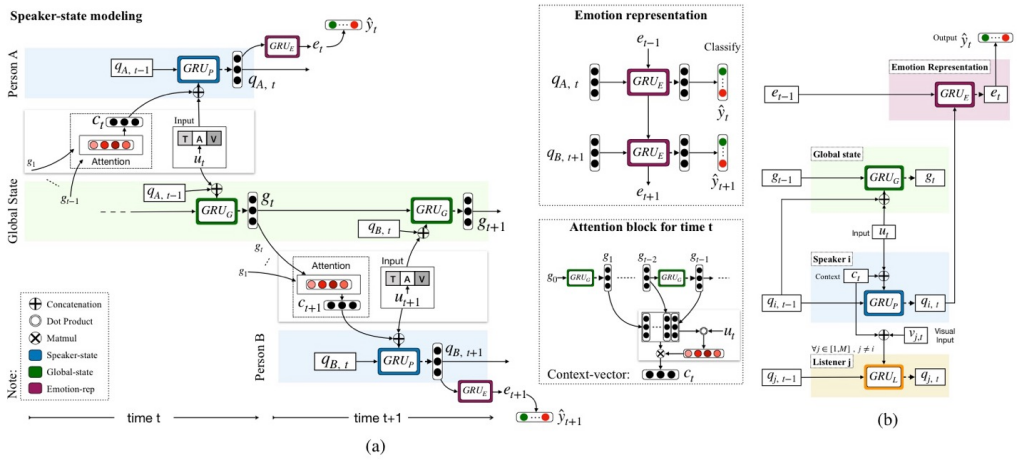

Figure 2: (a) DialogueRNN architecture. (b) Update schemes for global, speaker, listener, and emotion states for t^{th} utterance in a dialogue. Here, Person i is the speaker and Persons $j \in [1, M]$ and $j \neq i$ are the listeners.

- 해당 논문에서는 대화에서의 발화의 감정이 **세 가지 주요 요인**에 의존한다고 가정

> 💡 1. 발화자(party)
> 2. 이전 발화에서 주어진 맥락
> 3. 이전 발화에서의 감정

  - 각 파티는 발화마다 상태를 가지며, 전역 상태는 발화와 발화자 상태를 함께 인코딩하여 맥락을 캡처
  - GRU 셀을 사용하여 상태를 업데이트하고, 감정 표현은 이전 발화의 감정 표현과 현재 발화자의 상태를 기반으로 추론됨
  - 모델은 최종적으로 감정 분류를 위한 softmax 레이어로 전달됨

전역 상태(Global GRU)

- 전역 상태는 발화와 발화자 상태를 공동으로 인코딩하여 주어진 발화의 맥락을 캡처

duce improved context representation. The current utterance u_t changes the speaker's state from $q_{s(u_t),t-1}$ to $q_{s(u_t),t}$. We capture this change with GRU cell $GRU_{\mathcal{G}}$ with output size $D_{\mathcal{G}}$, using u_t and $q_{s(u_t),t-1}$:

$$g_t = GRU_{\mathcal{G}}(g_{t-1}, (u_t \oplus q_{s(u_t),t-1})), \quad (1)$$

where $D_{\mathcal{G}}$ is the size of global state vector, $D_{\mathcal{P}}$ is the size of party state vector, $W_{\mathcal{G},h}^{\{r,z,c\}} \in \mathbb{R}^{D_{\mathcal{G}} \times D_{\mathcal{G}}}$, $W_{\mathcal{G},x}^{\{r,z,c\}} \in \mathbb{R}^{D_{\mathcal{G}} \times (D_m + D_{\mathcal{P}})}$, $b_{\mathcal{G}}^{\{r,z,c\}} \in \mathbb{R}^{D_{\mathcal{G}}}$, $q_{s(u_t),t-1} \in \mathbb{R}^{D_{\mathcal{P}}}$, $g_t, g_{t-1} \in \mathbb{R}^{D_{\mathcal{G}}}$, $D_{\mathcal{P}}$ is party state size, and $\oplus$ represents concatenation.

대화 참여자 상태(Party GRU)

- DialogueRNN은 대화 전반에 걸쳐 고정 크기의 벡터 $q_1, q_2, ..., q_M$을 사용하여 각각의 발화자의 상태를 추적
  - 대화에서 발화자의 상태를 대표
  - 감정 분류와 관련
- 상태는 대화 참가자의 현재(시간: t) 역할인 발화자 또는 청취자, 그리고 들어오는 발화 u_t에 기반하여 업데이트

⇒ 모델이 각 발화의 발화자를 인식하고 이에 적절하게 대응할 수 있도록

발화자 업데이트(Speaker GRU)

- 발화자는 보통 대화에서 이전 발화인 **맥락**에 기반하여 응답을 구성

  ⇒ 발화(u_t)와 관련있는 맥락(c_t)를 다음과 같이 캡쳐

$$\alpha = \text{softmax}(u_t^T W_\alpha [g_1, g_2, \ldots, g_{t-1}]), \quad (2)$$

$$\text{softmax}(x) = [e^{x_1}/\Sigma_i e^{x_i}, e^{x_2}/\Sigma_i e^{x_i}, \ldots], \quad (3)$$

$$c_t = \alpha [g_1, g_2, \ldots, g_{t-1}]^T, \quad (4)$$

where $g_1, g_2, \ldots, g_{t-1}$ are preceding $t-1$ global states ($g_i \in \mathbb{R}^{D_{\mathcal{G}}}$), $W_\alpha \in \mathbb{R}^{D_m \times D_{\mathcal{G}}}$, $\alpha^T \in \mathbb{R}^{(t-1)}$, and $c_t \in \mathbb{R}^{D_{\mathcal{G}}}$. In Eq. (2), we calculate attention scores α over the previous global states representative of the previous utterances. This assigns higher attention scores to the utterances emotionally relevant to u_t. Finally, in Eq. (4) the context vector c_t is calculated by pooling the previous global states with α.

- GRU Cell

$$q_{s(u_t),t} = GRU_{\mathcal{P}}(q_{s(u_t),t-1}, (u_t \oplus c_t)),$$

  - 현재 발화에 대한 정보를 전역 GRU에서 가져와 발화자의 상태 $q_s(u_t)$로 인코딩 → 감정 분류에 도움이 되는 컨텍스트를 포함

청취자 업데이트(Listener Update)

- 청취자 상태는 발화자의 발화로 인한 청취자의 상태 변화를 모델링
- 두 가지 상태 업데이트 메커니즘을 시도
  1. 청취자의 상태를 그대로 유지

$$\forall i \neq s(u_t), q_{i,t} = q_{i,t-1}.$$

  2. 다른 GRU Cell 추가

$$\forall i \neq s(u_t), q_{i,t} = GRU_{\mathcal{L}}(q_{i,t-1}, (v_{i,t} \oplus c_t)), \quad (7)$$

where $v_{i,t} \in \mathbb{R}^{D_{\mathcal{V}}}$, $W_{\mathcal{L},h}^{\{r,z,c\}} \in \mathbb{R}^{D_{\mathcal{P}} \times D_{\mathcal{P}}}$, $W_{\mathcal{L},x}^{\{r,z,c\}} \in \mathbb{R}^{D_{\mathcal{P}} \times (D_{\mathcal{V}} + D_{\mathcal{G}})}$, and $b_{\mathcal{L}}^{\{r,z,c\}} \in \mathbb{R}^{D_{\mathcal{P}}}$. Listener visual features of party i at time t $v_{i,t}$ are extracted using the model introduced by Arriaga, Valdenegro-Toro, and Plöger (2017), pretrained on FER2013 dataset, where feature size $D_{\mathcal{V}} = 7$.

- 첫 번째 방법만으로 충분하기는 함
- 두 번째 접근법은 매개변수의 수를 증가시키면 성능이 향상됨
  - 청취자가 말할 때에만 대화에 연관되기 때문
  - 어떤 파티가 말할 때, 우리는 모든 이전 발화에 관련 정보를 포함하는 맥락 c_t로 상태 q_i를 업데이트

감정 표현(Emotion GRU)

- 발화 u_t에 대한 감정 관련 표현 e_t를 발화자의 상태($q_{s(u_t),t}$)와 이전 감정 상태(e_{t-1})를 통해 추론

the other party states. Hence, we model e_t with a GRU cell ($GRU_{\mathcal{E}}$) with output size $D_{\mathcal{E}}$ as

$$e_t = GRU_{\mathcal{E}}(e_{t-1}, q_{s(u_t),t}), \quad (8)$$

where $D_{\mathcal{E}}$ is the size of emotion representation vector, $e_{\{t,t-1\}} \in \mathbb{R}^{D_{\mathcal{E}}}$, $W_{\mathcal{E},h}^{\{r,z,c\}} \in \mathbb{R}^{D_{\mathcal{E}} \times D_{\mathcal{E}}}$, $W_{\mathcal{E},x}^{\{r,z,c\}} \in \mathbb{R}^{D_{\mathcal{E}} \times D_{\mathcal{P}}}$, and $b_{\mathcal{E}}^{\{r,z,c\}} \in \mathbb{R}^{D_{\mathcal{E}}}$.

- 발화자 상태는 발화자별 발화 표현 역할을 하는 전역 상태에서 정보를 얻기 때문에, 모델은 이미 다른 파티에 대한 정보에 접근할 수 있다고 주장할 수도 있음
  - 그러나 제거 연구에서 보여진 바와 같이 감정 GRU는 **직접** 이전 파티의 상태를 연결 → 성능 향상에 도움
  - 또한, 발화자 및 전역 GRU(GRU_P, GRU_G)가 함께 인코더와 유사하게 작용하는 반면, 감정 GRU는 디코더 역할을 수행

감정 분류

- 발화(u_t)의 감정 표현(e_t)으로부터 6가지 감정 클래스의 확률 c를 계산
  - 최종 softmax 레이어를 포함하는 두 개의 레이어로 구성된 다층 퍼셉트론을 사용
  - 이후 가장 확률이 높은 감정 클래스를 선택

$$l_t = \text{ReLU}(W_l e_t + b_l), \quad (9)$$

$$\mathcal{P}_t = \text{softmax}(W_{smax} l_t + b_{smax}), \quad (10)$$

$$\hat{y}_t = \underset{i}{\text{argmax}}(\mathcal{P}_t[i]), \quad (11)$$

where $W_l \in \mathbb{R}^{D_l \times D_{\mathcal{E}}}$, $b_l \in \mathbb{R}^{D_l}$, $W_{smax} \in \mathbb{R}^{c \times D_l}$, $b_{smax} \in \mathbb{R}^c$, $\mathcal{P}_t \in \mathbb{R}^c$, and $\hat{y}_t$ is the predicted label for utterance u_t.

훈련

- L2 규제가 적용된 cross entropy loss를 활용

$$L = -\frac{1}{\sum_{s=1}^{N} c(s)} \sum_{i=1}^{N} \sum_{j=1}^{c(i)} \log \mathcal{P}_{i,j}[y_{i,j}] + \lambda \|\theta\|_2, \qquad (12)$$

where N is the number of samples/dialogues, $c(i)$ is the number of utterances in sample i, $\mathcal{P}_{i,j}$ is the probability distribution of emotion labels for utterance j of dialogue i, $y_{i,j}$ is the expected class label of utterance j of dialogue i, λ is the L2-regularizer weight, and θ is the set of trainable parameters where

$$\theta = \{W_\alpha, W_{\mathcal{P},\{h,x\}}^{\{r,z,c\}}, b_{\mathcal{P}}^{\{r,z,c\}}, W_{\mathcal{G},\{h,x\}}^{\{r,z,c\}}, b_{\mathcal{G}}^{\{r,z,c\}}, W_{\mathcal{E},\{h,x\}}^{\{r,z,c\}}, b_{\mathcal{E}}^{\{r,z,c\}}, W_l, b_l, W_{smax}, b_{smax}\}.$$

- Adam 옵티마이저를 활용한 SGD 적용
- 교차 검증을 통한 하이퍼 파라미터 튜닝 수행

3-4. DialogueRNN Variants

- DialogueRNN + 청취자 상태 업데이트(DialogueRNNl)
  - 발화자 상태 $q_{s(ut),t}$로부터 파생된 청취자 상태를 업데이트
- 양방향 다이얼로그 RNN(BiDialogueRNN)
  - 양방향 RNN과 유사하며, 입력 시퀀스의 전방향 및 후방향 패스에 각각 다른 RNN을 사용
  - BiDialogueRNN에서 최종 감정 표현은 순방향 및 역방향 다이얼로그RNN을 통해 대화에서 이전 및 이후 발화의 정보를 포함하여 감정 분류를 위한 더 나은 맥락을 제공
- DialogueRNN + Attention(DialogueRNN+Att)
  - 각 감정 표현 e_t에 대해 대화에서 주변 감정 표현에 대해 attention이 적용되어 해당 표현들과 e_t를 일치시킴으로써 관련된(attention 점수에 기반한) 미래 및 이전 발화에서 맥락을 제공
- 양방향 다이얼로그RNN + 감정 주의 (BiDialogueRNN+ Att)
  - BiDialogueRNN의 각 감정 표현 e_t에 대해 대화에서 모든 감정 표현에 대해 attention이 적용되어 대화의 다른 발화로부터 맥락을 캡처

$$\beta_t = \text{softmax}(e_t^T W_\beta [e_1, e_2, \ldots, e_N]), \quad (13)$$

$$\tilde{e}_t = \beta_t [e_1, e_2, \ldots, e_N]^T, \quad (14)$$

where $e_t \in \mathbb{R}^{2D_\varepsilon}$, $W_\beta \in \mathbb{R}^{2D_\varepsilon \times 2D_\varepsilon}$, $\tilde{e}_t \in \mathbb{R}^{2D_\varepsilon}$, and $\beta_t^T \in \mathbb{R}^N$. Further, $\tilde{e}_t$ are fed to a two-layer perceptron for emotion classification, as in Eqs. (9) to (11).

4. Experimental Setting

4-1. Datasets Used

- DialogueRNN을 평가하기 위해 IEMOCAP과 AVEC 두 개의 감정 감지 데이터셋을 사용

Dataset	Partition	Utterance Count	Dialogue Count
IEMOCAP	train + val	5810	120
	test	1623	31
AVEC	train + val	4368	63
	test	1430	32

Table 1: Dataset split ((train + val) / test ≈ 80%/20%).

IEMOCAP

- 열 명의 고유한 발화자로부터의 이중 대화 비디오를 포함
- 각 비디오에는 단일 이중 대화가 포함되어 있으며, 발화로 분할됨
  - 발화는 행복, 슬픔, 중립, 화남, 흥분, 좌절 중 하나의 여섯 가지 감정 라벨 중 하나로 주석이 달려 있음

AVEC

- 인간과 인공 지능 에이전트 간의 상호 작용을 포함하는 SEMAINE 데이터베이스의 수정판
- 대화의 각 발화는 네 가지의 실수값 감성 속성으로 주석이 달려 있음
  - 감정(Valence), 흥분(Arousal), 기대(Expectancy), 권력
- 주석을 발화 수준의 주석으로 적응하기 위해 발화 기간 동안의 각 속성을 평균화

4-2. Baselines and State of the Art

DialogueRNN을 포괄적으로 평가하기 위해 아래의 baseline들과 비교

- c-LSTM
  - 주변 발화로부터 맥락을 포착하여 맥락을 고려한 발화 표현을 생성하기 위해 양방향 LSTM을 사용
  - 그러나 해당 모델은 발화자를 구별하지 않음
- c-LSTM+Att
  - 각 타임스탬프에서 c-LSTM 출력에 어텐션을 적용

    ⇒ 최종 발화 표현에 더 나은 맥락을 제공
- TFN
  - multi-modal 시나리오에 특화되어 있음
  - 텐서 외적을 사용하여 다중성과 내부성 상호작용을 포착
  - 주변 발화로부터의 맥락을 포착하지 않음
- MFN
  - 다중모달 시나리오에 특화 → 다중뷰 학습을 활용하여 뷰별 및 교차 뷰 상호작용을 모델링
  - TFN과 유사하게 맥락적 정보를 사용하지 않습니다.
- CNN
  - 텍스트 특징 추출 네트워크와 동일
  - 주변 발화로부터 맥락 정보를 사용하지 않음
- Memnet
  - 현재 발화를 메모리 네트워크에 공급하며, 이 때 메모리는 이전 발화에 해당
  - 메모리 네트워크의 출력이 감정 분류를 위한 최종 발화 표현으로 사용됨
- CMN
  - 두 발화자를 위해 두 가지 다른 GRU를 사용하여 대화 이력에서 발화 맥락을 모델링
  - 최종적으로 현재 발화를 두 발화자의 각각에 대한 두 가지 다른 메모리 네트워크에 쿼리로 공급하여 발화 표현을 얻음

4-3. Modalities

- 주로 텍스트 모달리티에서 모델을 평가
    - 다중모달 시나리오에서 모델의 효과를 입증하기 위해 다중모달 특징들과도 실험을 진행

5. Results and Discussion

- 텍스트 데이터를 기준으로 DialogueRNN과 그 변형들을 baseline과 비교

Methods	IEMOCAP												AVEC							
	Happy		Sad		Neutral		Angry		Excited		Frustrated		**Average(w)**		Valence		Arousal		Expectancy		Power	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	MAE	r	MAE	r	MAE	r	MAE	r
CNN	27.77	29.86	57.14	53.83	34.33	40.14	61.17	52.44	46.15	50.09	62.99	55.75	48.92	48.18	0.545	-0.01	0.542	0.01	0.605	-0.01	8.71	0.19
memnet	25.72	33.53	55.53	61.77	58.12	52.84	59.32	55.39	51.50	58.30	67.20	59.00	55.72	55.10	0.202	0.16	0.211	0.24	0.216	0.23	8.97	0.05
c-LSTM	29.17	34.43	57.14	60.87	54.17	51.81	57.06	56.73	51.17	57.95	67.19	58.92	55.21	54.95	0.194	0.14	0.212	0.23	0.201	0.25	8.90	-0.04
c-LSTM+Att	30.56	35.63	56.73	62.90	57.55	53.00	59.41	59.24	52.84	58.85	65.88	59.41	56.32	56.19	0.189	0.16	0.213	0.25	0.190	0.24	8.67	0.10
CMN (SOTA)	25.00	30.38	55.92	62.41	52.86	52.39	61.76	59.83	55.52	60.25	71.13	60.69	56.56	56.13	0.192	0.23	0.213	0.29	0.195	0.26	8.74	-0.02
DialogueRNN	31.25	33.83	66.12	69.83	63.02	57.76	61.76	62.50	61.54	64.45	59.58	59.46	59.33	59.89	0.188	0.28	0.201	0.36	0.188	0.32	8.19	0.31
DialogueRNN$_l$	35.42	35.54	65.71	69.85	55.73	55.30	62.94	61.85	59.20	62.21	63.52	59.38	58.66	58.76	0.189	0.27	0.203	0.33	0.188	0.30	8.21	0.30
BiDialogueRNN	32.64	36.15	71.02	74.04	60.47	56.16	62.94	63.88	56.52	62.02	65.62	**61.73**	60.32	60.28	0.181	0.30	0.198	0.34	0.187	0.34	8.14	0.32
DialogueRNN+Att	28.47	**36.61**	65.31	72.40	62.50	57.21	67.65	**65.71**	70.90	68.61	61.68	60.80	61.80	61.51	0.173	0.35	0.168	0.55	0.177	0.37	7.91	0.35
BiDialogueRNN+Att	25.69	33.18	75.10	**78.80**	58.59	**59.21**	64.71	65.28	80.27	**71.86**	61.15	58.91	63.40	**62.75**	**0.168**	**0.35**	**0.165**	**0.59**	**0.175**	**0.37**	**7.90**	**0.37**

Table 2: Comparison with the baseline methods for textual modality; Acc. = Accuracy, MAE = Mean Absolute Error, r = Pearson correlation coefficient; bold font denotes the best performances. Average(w) = Weighted average.

⇒ DialogueRNN은 두 데이터셋 모두에서 최첨단 기법인 CMN을 포함한 모든 baseline보다 우수한 성능을 보여줌

5-1. Comparison with the State of the Art

- 텍스트 모달리티를 위한 IEMOCAP 및 AVEC 데이터셋에서 DialogueRNN의 성능을 최첨단 방법인 CMN과 비교

IEMOCAP

- IEMOCAP 데이터셋에 대해 DialogueRNN은 평균적으로 CMN에 비해 2.77%의 정확도 및 3.76%의 F1 점수를 능가
- 이 향상은 CMN과 DialogueRNN 사이의 근본적인 차이에 기인한다고 고려됨
    1. GRU_P를 사용한 party 상태 모델링
    2. 발화자별 발화 처리,
    3. GRU_G를 사용한 전역 상태 캡처
- DialogueRNN은 유감 클래스를 제외한 다섯 가지의 감정 클래스에서 상당한 마진으로 CMN을 능가
    - 유감 클래스의 경우, DialogueRNN은 CMN에 비해 1.23%의 F1 점수로 뒤처지는 것으로 나타남

AVEC

- valence, arousal, expectancy 및 power 속성에 대해 CMN을 능가
    - 모든 네 가지 속성에 대해 낮은 평균 절대 오차(MAE) 및 더 높은 피어슨 상관 계수(r)를 제공

  ⇒ CMN에 누락된 party 상태와 감정 GRU 통합 덕분이라고 생각됨

5-2. DialogueRNN vs DialogueRNN Variants

DialogueRNN 및 그 변형들의 성능을 텍스트 데이터에 대해 IEMOCAP 및 AVEC 데이터셋에서 비교

- DialogueRNNI
    - 명시적 청취자 상태 업데이트를 사용하는 것은 일반적으로 DialogueRNN보다 성능이 약간 낮음
    - 하지만 IEMOCAP의 경우 행복한 감정 라벨에서만 DialogueRNN을 약간 능가
- BiDialogueRNN
    - 미래 발화로부터의 맥락을 포착하기 때문에 DialogueRNN보다 향상된 성능을 보임
- DialogueRNN+Attn
    - 현재 발화와 일치하도록 과거 및 미래 발화에서 정보를 가져와 감정적으로 중요한 맥락 발화에 관련성을 제공
    - 이로 인해 BiDialogueRNN보다 더 나은 성능을 보임
- BiDialogueRNN+Attn
    - 최상의 성능을 보이며, 다른 모든 방법들보다 뛰어난 성능을 제공
    - IEMOCAP의 경우 최첨단 CMN보다 평균적으로 6.62% 높은 F1 점수를 보임
    - AVEC 데이터셋에서도 네 가지 속성에서 최상의 성능을 제공

5-3. Multimodal Setting

- multi-modal 기능에 대해 DialogueRNN을 평가
    - uni-modal 기능의 연결을 퓨전 방법으로 사용
- DialogueRNN은 강력한 baseline이자 최첨단 방법인 CMN을 능가

Methods	IEMOCAP	AVEC			
	F1	Valence (r)	Arousal (r)	Expectancy (r)	Power (r)
TFN	56.8	0.01	0.10	0.12	0.12
MFN	53.5	0.14	25	0.26	0.15
c-LSTM	58.3	0.14	0.23	0.25	-0.04
CMN	58.5	0.23	0.30	0.26	-0.02
BiDialogueRNN+att$_{text}$	62.7	0.35	0.59	0.37	0.37
BiDialogueRNN+att$_{MM}$	**62.9**	**0.37**	**0.60**	**0.37**	**0.41**

Table 3: Comparison with the baselines for trimodal (T+V+A) scenario. BiDialogueRNN+att$_{MM}$ = BiDialogueRNN+att in multimodal setting.

5-4. Case Studies

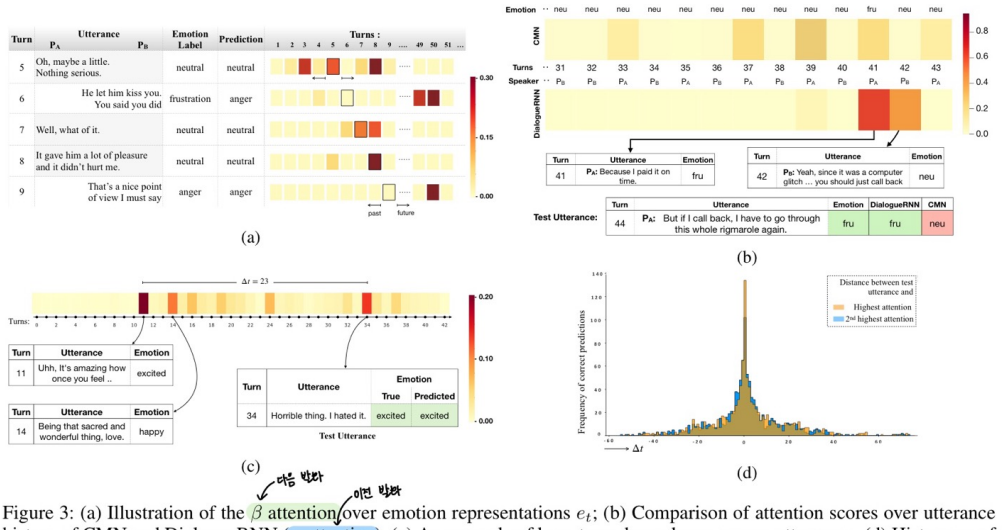

Figure 3: (a) Illustration of the β attention over emotion representations e_t; (b) Comparison of attention scores over utterance history of CMN and DialogueRNN (α attention). (c) An example of long-term dependency among utterances. (d) Histogram of Δt = distance between the target utterance and its context utterance based on β attention scores.

DialogueRNN은 이전 발화에 대한 종속성, 다음 발화에 대한 종속성, 그리고 먼 컨텍스트에 대한 종속성을 포착

- 이전 발화에 대한 종속성
  - DialogueRNN의 주의 메커니즘이 CMN보다 더 집중된 주의를 제공하며, 이는 감정적 변화를 올바르게 예측하는 데 도움
- 다음 발화에 대한 종속성
  - BiDialogueRNN+Att 모델이 미래 발화의 감정적 상태를 포착하고 이전 발화와의 상호 종속성을 확인
- 먼 컨텍스트에 대한 종속성
  - DialogueRNN이 전반적인 대화의 감정적 톤을 고려하여 발화 간의 감정적 상관 관계를 이해

⇒ 이러한 종속성을 통해 DialogueRNN은 감정적 정보를 효과적으로 모델링하고 감정 분류를 향상

5-5. Error Analysis

- 예측에서 두드러지는 추세는 관련된 감정들 간의 높은 교차 분류임
  - 모델이 행복한 감정을 잘못 분류하는 경우 대부분이 흥분된 클래스로 나타났으며, 분노와 좌절은 서로 잘못 분류되는 경우가 많음
  - 중립 클래스는 잘못된 양성 예측이 높은데, 이는 해당 감정이 클래스 분포에서 대다수를 차지하고 있기 때문임
- 대화 수준에서는, 동일한 당사자의 이전 턴과의 감정 변화가 있는 턴에서 상당한 수의 오류가 발생하는 것을 관찰하였음
  - 감정 변화가 없는 영역에서 모델의 성공률이 더 높으며, 감정 변화는 복잡한 현상으로 여겨지며 개선이 필요

5-6. Ablation Study

Party State	Emotion GRU	F1
-	+	55.56
+	-	57.38
+	+	59.89

Table 4: Ablated DialogueRNN for IEMOCAP dataset.

파티 상태와 감정 GRU의 도입 관점에서 수행

- 파티 상태가 성능에 매우 중요한 역할
  - 이 구성 요소가 없으면 성능이 4.33% 하락
- 감정 GRU도 중요한 역할을 하지만, 파티 상태보다는 영향이 적음
  - 그러나 이 구성 요소가 없으면 성능이 2.51% 하락
  - 이는 다른 당사자들의 상태를 통한 컨텍스트 흐름이 부족하기 때문으로 추측됨

6. Conclusion

- 대화에서의 감정 탐지를 위한 RNN 기반의 신경망 구조를 제시

- CMN과 대조적으로, 각 입력 발화를 스피커의 특성을 고려하여 다루어 발화에 미세한 컨텍스트를 부여
- 우리의 모델은 텍스트 및 멀티모달 설정에서 두 개의 다른 데이터셋에서 현재의 최첨단을 능가