碩 士 學 位 論 文

CNN-LSTM 복합 모델을 이용한 대화의 사용자 발화 감정 분류

Emotion Classification in Dialogue Utterances Using CNN-LSTM Complex Model

高麗大學校 大學院

컴퓨터學科

申 東 元

2017 年 1 月 5 日

林 海 彰 敎 授 指 導

碩 士 學 位 論 文

CNN-LSTM 복합 모델을 이용한 대화의 사용자 발화 감정 분류

Emotion Classification in Dialogue Utterances Using CNN-LSTM Complex Model

이 論文을 工學 碩士學位 論文으로 提出함

2017 年 1 月 5 日

高麗大學校 大學院

컴퓨터學科

申 東 元

申東元 의 工學 碩士學位論文
審査를 完了함

2017 年 1 月 5 日

委員長 임 해 창 (印)

委 員 임 희 석 (印)

委 員 이 도 길 (印)

요 약

최근 딥 뉴럴 네트워크 기술의 급속한 발전이 이루어지면서, 인공 지능에 대한 다양한 분야들에서 연구 및 활용이 이루어지고 있다. 특히 대화형 시스템은 자연어를 기반으로 하여, 사용자가 필요로 하는 정보를 나타내는 질의에 대해 적절한 정보를 제공하거나, 사용자와 지속적으로 대화를 진행해 가는 시스템으로, 인공 지능 연구에 있어 중요한 연구 분야들 중 하나이다.

대화형 시스템에서는 자연어를 통해 커뮤니케이션이 이루어지므로, 사용자의 상황과 목적에 따라 시스템의 응답이 적절한 형태의 자연어로 표현되어야 할 필요성이 있다. 대화형 시스템의 성능이 아무리 좋더라도 부적절한 형태로 표현이 이루어진다면, 사용자의 만족도는 낮아질 수 밖에 없을 것이다. 자연어의 적절한 표현을 위해서는 다양한 요소들이 고려되어야 하며, 특히 사용자가 나타내는 감정은 자연어의 표현을 결정하는데 있어 필수적으로 고려되어야 하는 부분이라고 할 수 있다.

본 논문은 대화 상황에서 사용자들이 나타내는 발화가 어떤 감정을 나타내는 지를 특정한 감정 분류 체계에 따라 분류하고자 한다. 감정은 매우 주관적인 영역이기 때문에, 동일한 텍스트에 대해서도 해당 텍스트에서 나타나는 감정에 대해 사람마다 해석이 엇갈릴 수 있다. 또한

일반적인 텍스트가 아닌 대화라는 상황적 특성을 고려해야 정확한 감정 분류를 할 수 있다.

본 논문에서는 대화에서 이루어지는 발화의 감정 분류를 위해 전통적인 자질(feature) 기반의 분류 모델을 사용하는 대신, 최근 자연어 처리 관련 연구에서 활발하게 사용되고 있는 딥 뉴럴 네트워크 모델을 사용하는 방법을 제안한다. 이를 통해, 기존의 사람이 직접 만들어야 했던 분류를 위한 자질들을 딥 뉴럴 네트워크 모델을 통해 자동으로 추출하게 하고, 대화라는 상황적 특성을 딥 뉴럴 네트워크 모델의 구조적 특성을 통해 반영할 수 있다. 실험 결과, 딥 뉴럴 네트워크 모델의 사용이 대화에서의 감정 분류에 있어 효과적임을 보여준다.

Contents

List of Figures

List of Tables

Chapter 1

서 론

본 논문에서는 대화에서 나타나는 발화 텍스트에 내재된 사용자의 감정을 분류하고자 한다. 대화 시스템은 사용자가 나타내는 감정의 인식을 활용하여, 사용자가 필요로 하는 응답과 서비스를 좀 더 적절한 형태로 제공할 수 있다. 로봇의 표정을 변화시키거나, 사용자의 감정 상태에 알맞은 영화나 음악을 추천하는 등을 그 예로 들 수 있다. 최근 Affective Computing과 관련된 연구 결과에 따르면, 사용자의 감정을 인지하고 이에 따라 에이전트가 적극적인 대응을 할 때 사용자도 시스템을 더욱 효과적으로 이용할 수 있게 된다고 한다[1].

본 연구에서는 Plutchick의 8분류 체계[2]에서 설명되는 기쁨(Joy), 신뢰(Trust), 두려움 (Fear), 놀람(Surprise), 슬픔(Sadness), 혐오(Disgust), 화남(Anger), 기대(Anticipation)의 8가지 감정에, 감정 없음(None), 미안함(Sorriness), 부러움(Enviousness)의 추가적인 감정 카테고리를 사용하여 총 11가지의 감정 카테고리로 대화 내

발화들이 나타내는 다양한 감정을 분류하고자 한다. 기존의 감정 분류 연구들은 대부분 대화가 아닌 일반 텍스트에서 긍/부정으로 나타나는 감성 분류(Sentiment Classification)와 같은 극성 분류에 초점을 맞추었다. 긍/부정 외에 기쁨, 슬픔, 두려움 등 다양한 감정 분류에 관한 연구는 많지 않으며, 특히 대화적 특징을 반영한 대화 속 감정 분류 연구 역시 매우 드물다.

일반적인 텍스트와는 달리, 대화 상황에서는 둘 또는 그 보다 많은 화자와 청자가 서로 발화를 주고받으며 진행되는 형태로 이루어진다. 대화 참여자들이 나타내는 감정은 다른 사람들의 발화에 의해 자신의 감정이 바뀌기도 하고, 한 번 발생된 감정이 대화 상황에서 계속 지속되기도 한다. 그리고 감정을 직접적인 어휘를 통해 드러내기도 하지만, 대화의 발화 속에서 간접적으로 나타내기도 한다. 예를 들어 그림 1과 같은 경우는 “재밌다”, “속상해”와 같은 보다 직접적으로 감정을 나타내는 어휘에 의해 하나의 발화 내에 감정이 드러난 경우이지만, 그림 2와 같은 경우는 보다 간접적인 정황을 통해 드러나며, 한번 나타난 감정이 지속될 수 있는 상황을 보여준다.

따라서 대화 속 감정 분류 시스템은 다양한 감정 어휘들뿐만이 아니라, 대화 문맥 속에서 나타나는 분위기나 주제

등 감정을 유발하는 간접적인 자질들이 반영되어야 하며, 이에 더해서 대화의 현재 발화와 함께 대화 참여자들의 이전 발화들 또한 감정 분류에 효과적으로 반영되어야 할 필요성이 있다.

그림 1 감정적인 대화 예시 1

“A: 이거 진짜 재밌다.” (기쁨)
“B: 나 지금 정말 속상해.” (슬픔)

그림 2 감정적인 대화 예시 2

“A: 어제 혼났다며”
“B: 휴… 담임 샘한테 맨날 지각한다고 혼났지”(슬픔)
“A: 헐.. 많이 혼났어?”
“B: 어” (슬픔)

본 연구에서는 이러한 대화 상황에서 나타나는 특징들과 간접적으로 나타나는 감정 표현을 감정 분류에 활용하기 위한 방법으로 딥 뉴럴 네트워크 모델의 응용인 CNN-LSTM (Convolutional Neural Network-Long Short Term Memory) 모델을 사용한다. CNN을 통해 입력으로 들어온 발화 텍스트의 자질을 추출할 수 있도록 자동으로 학습 시킨 뒤, LSTM[3]을 통해 대화

상황에서의 문맥이 반영되도록 한다. 또한 적은 양의 학습 데이터로부터 효과적으로 딥 뉴럴 네트워크 구조를 학습시키기 위해 대량의 한글 구어체 코퍼스로부터 입력 텍스트의 임베딩에 사용되는 임베딩 레이어를 사전 훈련(Pre-training) 시키는 방법을 제안한다.

Chapter 2

관련 연구

2.1 감정 분류

사용자의 감정 인식 및 전달에 대한 연구는 주로 Affective computing 분야에서 활발하게 이루어져 왔다. 전통적으로 감정에 대한 분류 체계로는 주로 행복, 슬픔, 화남, 혐오, 놀람, 공포로 이루어진 Ekman의 6가지 분류 체계[4]가 사용되어 왔으며, 이미지나 영상 등에서 표정을 나타내는 facial landmark point 등을 추출하거나, head pose, eye gaze, prosody, nonlinguistic vocalization (laugh, cry 등) 와 같은 특징들을 이용하여 사용자의 현재 감정 상태를 분류한다[5].

이와는 별개로 텍스트를 대상으로 한 감정 분류 연구들은 주로 사용자가 발화를 통해 표현한 어휘를 통해 감정을 인식하고자 한다. [6]의 경우 Ekman의 6가지 감정 카테고리에 대해 감정 단어 사전 및 감정 이모티콘 사전을 구축하여 이들 자질을 기반으로 SVM(Support Vector Machine) 분류기를 이용하여 블로그에서 감정을

인식하고자 하였다. [7]의 경우는 소셜 네트워크 서비스인 Twitter의 트윗 데이터를 분석 대상으로 하였으며, 감정 사전을 구축하여 감정 분류에 활용하는 대신 다량의 uni-gram 자질을 사용하였다. [8]의 경우 음식, 사랑, 음악과 관련된 대화 시스템에서 사용자 발화의 감정을 분석하고자 하였다. 이 연구에서는 대상이 구어체 대화라는 점을 고려하여 광범위한 n-gram 뿐 아니라 마지막 어미, 관용 현 사전, 이전 발화에서의 분석된 감정 정보 등을 자질로서 사용하였다. [9]의 연구 역시 이와 유사하게 현재 발화, 전 발화, 전전 발화에 대해 많은 수의 n-gram을 자질로서 사용하였다.

그러나 위와 같은 다량의 n-gram 어휘 자질을 사용한 지도 학습 방식의 분류에서 가장 문제가 되는 것은 데이터 부족(Data Sparseness) 문제이다. 다양한 감정 카테고리와 각 카테고리 별로 사용되는 어휘 자질이 다양하게 나타나기 때문에 이러한 어휘 자질들을 이용해 적절한 수준의 학습이 이루어 질 수 있을 만큼의 충분한 학습 데이터를 구축하는 것이 매우 어려운 문제이다. 또한 사용자가 대화 상황에서 감정 사전에서 나타나는 구체적인 어휘들을 사용해 자신의 감정을 표현할 수 있지만, 간접적이고 상황적인 표현을 통해 감정을 나타내는 경우도 있기 때문에 데이터 부족 문제는 더 심각해진다.

[10]의 연구는 이러한 데이터 부족 문제를 해결하기 위해

대량의 원시 코퍼스를 사용한 Word Embedding 모델을 감정 분류 자질에 활용하는 방법을 제안하였다. 이를 통해 문제가 어느 정도 해소 되었다고 하나, 분류 모델 내에서 구조적으로 대화 문맥을 반영한 것이 아니기 때문에, 여전히 대화 문맥을 감정 분류에 활용하기 위한 자질 튜닝(Feature Engineering) 과정을 여러 번 거쳐야 한다는 문제점이 있다.

2.2 워드 임베딩

워드 임베딩(Word Embedding)은 하나의 단어를 일반적으로 수백 개 정도의 낮은 차원의 실수 벡터로 표현하는 것이다. 주어진 단어를 연속적이고 분산된 표현으로 나타내는 것은 단어에 대해 유사도 관점의 분석을 가능하게 하고 다양한 벡터 연산을 통해 단어들 간의 관계를 유추할 수 있도록 해준다. 전통적으로 LSA[11]나 LDA[12] 방법이 제안되어왔으나 최근 들어 neural network 기반의 언어 모델 학습을 통한 벡터 학습 방법이 좋은 성능을 보여주고 있다[13][14]. 본 연구에서는 [15]에서 제안한 Word2Vec 방법을 사용하였다. Word2Vec은 기존의 neural network 모델에서 hidden layer를 제거함으로써 기존 방식에 비해 성능의 하락 없이 대략 1000배 이상의 빠른 학습 속도를 보여주고 있다. 또한 다른

의미 분석과 관련된 지도 학습에 대한 입력으로 최근 좋은 성능을 보여주고 있다[16].

Word2Vec에서 워드 임베딩은 대량의 코퍼스를 통해 문장별로 문맥에 나타난 단어들이 주어졌을 때 언어 모델과 유사하게 단어의 발생 확률을 예측함으로써 학습된다. 따라서 유사한 용례를 가진 단어들이 유사한 위치로 학습된다. 예를 들어 의미적으로 유의어인 '아빠', '아버지'의 경우 코사인 유사도(cosine similarity) 값이 매우 높게 나타난다. 혹은 문법적으로 쓰임새가 유사한 단어들이 가깝게 학습되기도 한다. 더불어 벡터 연산을 통해 단어 간 관계를 유추할 수도 있다.

이러한 특징을 이용하여 최근에는 임베딩 벡터를 자연어처리의 의미 분석 태스크에 활용하는 연구들이 나타나고 있다. [17]의 경우에는, 분류 문제에서 대개 발생하는 일부 클래스들에 대한 학습 데이터 부족 문제를 워드 임베딩을 활용한 오버 샘플링(over-sampling)을 적용하여 해결을 시도했다. 즉, 학습에 사용될 수 있는 새로운 데이터를 임베딩 벡터의 조합으로 생성함으로써 희소 클래스의 분류 성능을 향상시킬 수 있음을 보였다.

2.3 딥 뉴럴 네트워크

딥 뉴럴 네트워크(Deep Neural Network)는 인공 신경망(Artificial Neural Network)의 입력과 출력 레이어 사이에 여러 개의 은닉(hidden) 레이어가 추가된 구조이다. 이러한 추가 은닉 레이어를 통해 비슷한 작업을 하도록 설계된 인공 신경망에 비하여 더 적은 수의 유닛들로 복잡한 데이터를 모델링 할 수 있다는 장점이 있다[18].

최근 [19]의 CNN 모델은 단순한 Convolution과 Pooling 과정만으로 문장 및 문서 분류에서 좋은 성능을 보인 연구이다. 또한 [3]의 LSTM 모델은 연속적인 입력 정보에 대해 순서를 고려한 자질의 학습과 생성 측면에서 기계번역 영역뿐 만 아니라 여러 문제에서 좋은 성능을 보여주고 있다. [20]에서 사용된 모델은 텍스트 중에서도 특히 문서 속 문장들, 대화 속 발화와 같은 길이가 짧고, 연속적인 텍스트를 대상으로 하여 RNN(Recurrent Neural Network)과 CNN을 사용해 표현하고 이를 분류에 사용하여 화행(Dialogue Act) 분류에서 state-of-the-art 성능을 나타낸 연구이다.

본 논문에서 제안하는 방법은 이와 유사하게 분류 모델 내에서 대화 상황에서의 문맥을 반영하고, 대량의 어휘 자질을 추출해 자동으로 학습할 수 있도록 대화에서의 발화 감정 분류에

CNN-LSTM 모델을 사용하는 것이다.

Chapter 3

제안하는 방법

본 챕터에서는 기존 자연어 처리 연구들에서 좋은 성능을 보인 딥 뉴럴 네트워크 모델들인 RNN 모델과 CNN 모델을 대화에서의 사용자 발화 감정 분류를 위한 모델로 어떻게 활용하였는지를 다룰 것이다. 또한 감정 분류 문제에 있어 각 모델들의 한계점을 살펴보고, 이를 보완하기 위해 대화 상황에서 나타나는 발화들의 자질과 문맥들을 감정 분류에 효과적으로 활용하기 위한 CNN-LSTM 모델에 대해 자세히 살펴볼 것이다.

3.1 RNN을 이용한 감정 분류 모델

RNN 모델의 입력으로 발화의 텍스트가 들어오게 되면, 이를 RNN 레이어의 입력으로 사용하기 위해 단어 단위의 벡터로 변환하게 된다. 입력 텍스트를 구성하는 변환된 단어 벡터들은 순차적으로 RNN의 Hidden layer에 입력되어 모델의 학습이 이루어지게 된다. RNN 모델의 Hidden layer에서는 매 스텝의 상태

값을 결정하는 데에 있어 이전 상태 값과 현재의 입력을 모두 고려하게 된다. 또한 이전 입력의 출력 정보 역시 현재 입력의 학습 및 분류에 반영되는 구조로 구성되어 있다. 특히 RNN 모델의 변형인 LSTM은 RNN에서 나타나는 gradient vanishing 문제를 해결하여 Long-term dependency를 좀 더 효과적으로 학습 할 수 있다.

n번째 문장 S_n을 구성하는 t번째 단어 w_t는 LSTM 모델에서 다음과 같은 수식들을 통해 처리된다

$$i_t = \sigma(W_t x_t + U_i h_{t-1} + b_i)$$

$$f_t = \sigma\left(W_f x_t + U_f h_{t-1} + b_f\right)$$

$$\tilde{c}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o)$$

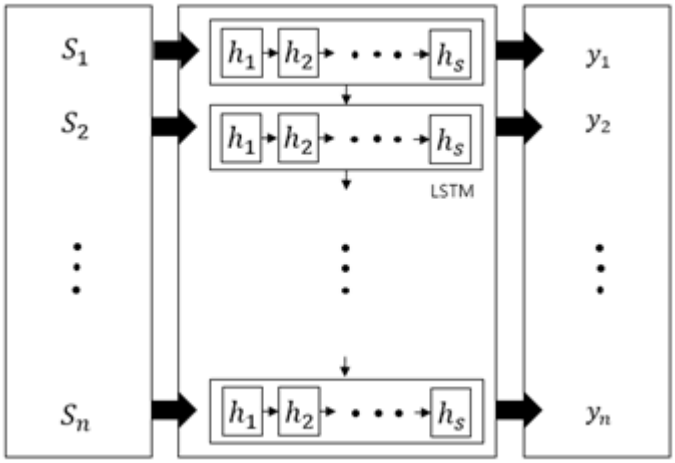
$$h_t = o_t \odot \tanh(c_t)$$

w_t는 임베딩 레이어를 통해 x_t로 나타내어 진다.x_t, h_{t-1}, c_{t-1}을 입력으로 하여, h_t, c_t를 출력한다. W, U는 가중치 행렬이며, b는 바이어스(bias) 벡터를 나타낸다. σ는 sigmoid 함수를 나타낸다.

이러한 LSTM 모델의 구조적 특성을 이용해, 대화 상황에서 현재 발화의 감정을 결정하는데 이전 발화 문장들과 감정 분류 결과들을 반영할 수 있게 된다. 전체 과정을 그림 3으로

나타내었다.

그림 3 LSTM을 이용한 감정 분류 모델



3.2 CNN을 이용한 감정 분류 모델

CNN 모델에서도 마찬가지로, 우선 입력된 발화의 텍스트를 벡터로 변환해 이를 Convolutional layer에 입력 한다. Convolutional layer 에서는 Convolution operation을 통해, 입력된 텍스트를 잘 나타내는 특징들이 추출된다. 이렇게 추출된 특징들은 마지막으로 Pooling layer를 거쳐 중요한 정보만을 남긴 후, 최종적으로 선정된 특징들 만이 다음 레이어에 들어갈 입력으로 출력 된다. Pooling 방법에는 Mean Pooling, Average Pooling, Max Pooling 등 다양한 방법들이 있지만, 일반적으로 분류를 위한 모델에서는 convolution을

통해 추출된 특징들 중 가장 큰 값만을 선택하는 Max Pooling이 사용 된다.

n 번째 문장 S_n 을 구성하는 단어들 $\{w_1, \dots, w_t\}$ 은 임베딩 레이어를 통해 $\{x_1, \dots, x_t\}$ 로 나타내어지고, 단어들은 다음과 같은 수식들을 통해 CNN 모델에서 처리된다.

$$x_{1:t} = x_1 \oplus x_2 \oplus \dots \oplus x_t$$

$$c_i = f(w \cdot x_{i:i+h-1} + b)$$

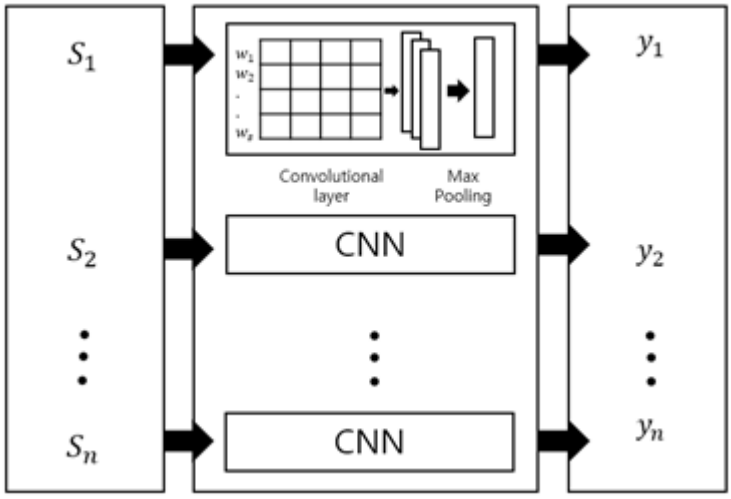
$$c = [c_1, c_2, \dots, c_{t-h+1}]$$

$$\hat{c} = \max\{c\}$$

$\oplus$ 는 벡터들을 이어 붙이는 concatenation 연산자이며, w 는 Convolutional 연산자에 포함되는 filter를 나타낸다. h 는 window size를 나타내며, 특징을 추출할 때 문장 내에서 한번에 몇 개의 단어를 사용할지를 결정하는 파라미터이다. 한 문장에서 총 $t-h+1$ 개의 특징이 추출되어, Max pooling을 거쳐 해당 문장에 대한 특징으로 $\hat{c}$을 출력하게 된다.

이러한 과정들을 거쳐 추출된 특징들을 현재 발화의 감정 분류에 사용하게 된다. 전체 과정을 그림 4로 나타내었다.

그림 4 CNN을 이용한 감정 분류 모델



3.3 CNN-LSTM을 이용한 감정 분류 모델

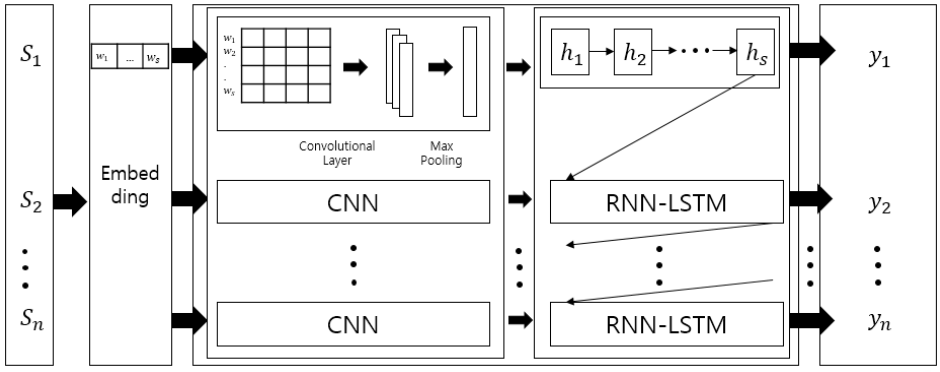
앞서 설명한 두 뉴럴 네트워크 모델은 텍스트 분류 문제에 있어 각각의 단점을 가지고 있다. LSTM 모델의 경우, 분류에 필요한 텍스트를 입력 받는 데에 있어 전적으로 임베딩에 의존하기 때문에 어휘적 특징들을 충분히 포착하기 어렵다는 단점이 존재한다. 임베딩 모델의 구축에서 충분한 양의 코퍼스와 효과적인 학습이 이루어지지 못하면, 이를 이용한 텍스트의 벡터 표현이 텍스트의 특징을 잘 표현하지 못하게 된다. 또한 CNN 모델의 경우에는 한 번에 한 발화의 텍스트 만을 입력으로 받기 때문에,

대화 상황에서 현재 발화를 분류하는데 활용될 수 있는 이전 발화들의 정보와 같은 Long-term dependency를 구조적으로 반영하기 어렵다는 문제점이 있다.

이러한 각 모델의 단점을 서로 상호 보완하기 위해, 두 종류의 뉴럴 네트워크를 함께 활용하는 CNN-LSTM 모델을 대화에서의 감정 분류에 사용하였다. CNN 모델을 이용하여 텍스트의 특징들을 잘 나타내는 벡터를 추출하고, 이를 LSTM 모델의 입력으로 하여 대화 상황에서의 Long-term dependency가 반영되도록 분류 모델을 학습시킨다. 즉 3.1의 LSTM 모델의 입력으로 문장을 구성하는 단어의 임베딩 벡터 $x = \{x_1, \dots, x_t\}$ 대신, 3.2의 문장으로부터 추출된 특징 벡터 $\hat{c}$을 사용하는 것이다.

그림 5는 이러한 과정을 그림으로 나타낸 것이다.

그림 5 CNN-LSTM을 이용한 감정 분류 모델



Chapter 4

실험 및 평가

4.1 실험 데이터 구축 방법

실제로 대화 시스템에서 사용되는 대화 데이터를 수집하는 것은 현실적으로 매우 어려울뿐더러 프라이버시 침해 문제 또한 발생할 수 있다. 이러한 문제를 피하면서도 실제로 있을 법한 대화 코퍼스를 구축하기 위해 본 연구에서는 영어 회화 코퍼스를 번역한 데이터와, 해외 드라마의 한국어 자막 코퍼스를 재가공하여 대화 코퍼스를 구축하였다. 세부적으로는 자막 코퍼스에서 대화 상황에 두 명의 참여자만 존재하는 1:1 대화 부분만을 선택하여, 화자 간의 관계를 설정한 다음 실제 대화에 가까운 자연스러운 구어체 대화의 형태를 갖도록 표현을 변환하고 각 발화에 대한 감정을 부착하였다.

대화 코퍼스의 발화 마다 감정을 부착하는 작업은 총 3명의 작업자에 의해 수행되었다. 각 작업자가 감정을 부착한 결과를 다수결을 통해 1차 감정을 결정하고, 감정의 모호성을 해소하기 위해 2차 감정까지 결정하였다. 3자의 부착 결과가 불일치한

5.9%의 발화에 대해서만 감정을 다시 부착하였다. 1차 감정에 대한 모든 작업자의 동의율은 48.3%로 매우 낮았지만, 최소 2명까지의 동의율은 92.1%로 높게 나타났으며, 2차 감정까지 포함하면 동의율은 94.1%로 나타났다. 감정 부착에 대한 동의가 이뤄지지 않은 발화들 대부분은 '슬픔', '분노', '싫어함'과 같은 부정적인 감정을 나타내는 발화에서 나타났다. 예를 들면, "아 오늘도 시험이라니"와 같은 발화는 한 명의 작업자의 경우에도 '슬픔', '분노', '싫어함'과 같은 감정들에 대해 모호성이 있다고 판단하였으며, 각 작업자들 간에서도 2차 감정까지의 감정 부착의 결과가 모두 다르게 나타났다.

이러한 과정을 통해 구축된 전체 감정 부착 대화 코퍼스는 총 2,587개의 대화, 22,005개의 발화로 구성되어 있다. 대화 코퍼스의 1차 감정에 대한 감정 카테고리 별 발화 개수 분포는 표 1과 같다.

표 1 감정 카테고리 별 발화 수 및 비율

감정 카테고리	발화 수 (개)	비율 (%)
감정 없음	15171	68.9
기쁨	1644	7.5
놀람	1463	6.6
화남	855	3.9
두려움	738	3.4
슬픔	725	3.3
싫어함	519	2.4
바람	504	2.3
미안함	272	1.2
부러움	67	0.3
신뢰	47	0.2
전체	22006	

4.2 임베딩 모델 학습

텍스트 데이터가 학습 모델에 사용되기 위해서는 벡터로 변환되어야 한다. 이를 위해 사용되는 임베딩 레이어에서는 벡터 초기화를 위해 모든 벡터 값을 0이나 작은 상수, 또는 난수로 초기화 하거나, Xavier Initialization[21] 과 같은 초기화 기법을 사용하게 된다. 하지만 본 연구에서는 Word2Vec[15]을 사용해 구축한 임베딩 모델을 이용해 딥 뉴럴 네트워크 모델의 입력으로 사용되는 임베딩 벡터의 초기화에 사용하여 입력 텍스트가 좀 더 효과적인 벡터 표현이 될 수 있도록 하였다. 이를 위한 임베딩 모델의 학습에 사용되는 코퍼스는 표 2와 같다.

표 2 임베딩 모델 학습에 사용된 코퍼스

코퍼스	크기	문장 수 (개)
한국어 뉴스 기사 코퍼스	2.069 GB	11,835,341
해외 드라마 자막 한국어 번역 코퍼스	18.3 MB	361,480
전체	2.087 GB	12,196,821

임베딩 모델 학습에 사용되는 알고리즘인 Word2Vec에 사용되는 주요 파라미터들은 표 3과 같다. 각 파라미터들의 최적

값은 실험적으로 결정하였다.

표 3 Word2Vec 파라미터

파라미터	값
Architecture	skip-gram [cbow, skip-gram]
Window size	9 [5 – 13]
Dimension	300 [100 – 500]
Training Algorithm	Hierarchical Softmax
Min-count	10 [5 – 15]

4.3 실험 설계 및 평가 척도

본 연구에서는 대화 코퍼스를 대상으로 하여 딥 뉴럴 네트워크 모델을 사용해 감정 분류기를 학습시켰다. 또한 제안한 모델의 성능을 비교하기 위하여 [10]의 SVM과 임베딩 자질을 사용한 감정 분류 모델을 베이스라인으로 사용하였다. 각 딥 뉴럴 네트워크 모델에 사용된 파라미터들은 표 5와 같다. 각 파라미터들의 값은 실험적으로 결정하였다.

딥 뉴럴 네트워크 모델을 학습하는 데에는 많은 시간이 소요되므로, 모델의 분류 성능을 검증하는데 있어 교차 검증을

사용하기 어렵다는 문제가 있다. 때문에 대화 코퍼스에서 '감정 없음' 카테고리와 나머지 감정 카테고리들의 비율을 최대한 유사하게 하도록 대화들을 선택하여, 학습/검증/테스트 데이터를 각각 대화 2000개 (발화 17062개), 293개 (2512개), 294개 (2431개)로 구성하였다.

표 4 뉴럴 네트워크 모델 파라미터

모델	파라미터	값
	Number of epoch	10000
	Learning rate	$10^{-5}[10^{-4} - 10^{-5}]$
	Dropout rate	0.5 [0-1]
LSTM	Output dimension	300 [100 – 500]
CNN	Number of filters	500 [100 – 500]
	Filter height	4 [2 – 5]

모델의 감정 분류 성능을 평가하기 위한 평가 척도는 다음과 같다. 앞서 대화 코퍼스의 감정 부착에서 살펴본 바와 같이, 대화 속에서 사용자들이 나타내는 발화가 어떤 감정을 나타내고 있는지에 대해서는 모호성이 다수 존재한다. 하지만 감정 부착 결과 1,2차 감정 중 적어도 한 감정에 대해서는 작업자들의 동의율이

높게 나타난다는 점을 반영하여 분류 대상 발화에 정답이 2차까지 있는 경우에는, 분류 모델이 두 정답 중 하나를 맞추면 맞은 것으로 하는 평가 척도를 사용 한다.

$$\text{Accuracy} = \frac{\text{1,2}\textit{차 정답 중 하나와 모델의 분류 결과가 일치한 개수}}{\textit{전체 발화 개수}}$$

4.4 실험 결과 및 분석

딥 뉴럴 네트워크 모델의 사용이 대화에서의 감정 분류에 있어 어떤 성능을 나타내는지 평가하기 위하여, 베이스라인 모델과 LSTM, CNN, CNN-LSTM 모델의 성능을 비교하였다.

표 6을 보면, 전체 감정 카테고리에 대한 분류 성능은 LSTM, SVM, CNN 순으로 나타나며, LSTM과 CNN을 함께 사용한 CNN-LSTM 모델이 가장 좋은 성능을 나타낸 것을 확인할 수 있다.

표 5 분류 모델 별 감정 분류 성능

분류 모델	Accuracy (%)
SVM	81.69
LSTM	81.04
CNN	82.35
CNN-LSTM	**82.93**

분류 성능을 좀 더 자세히 분석하기 위하여, 각 분류 모델의 감정 카테고리 별로 정답을 얼마나 출력 했는지를 살펴보았다. 표 7을 보면, '감정 없음' 카테고리에서는 기존의 SVM 모델이 가장 많이 정답을 맞혔지만, 나머지 감정 카테고리 들에 대해선 딥 뉴럴 네트워크 모델들이 SVM 모델보다 더 많이 맞춘 것을 확인할 수 있다.

SVM 모델에서 '감정 없음' 카테고리를 가장 많이 맞춘 것은 학습 데이터에 대한 과적합(Over-fitting) 때문이라고 해석할 수 있다. 반면에 딥 뉴럴 네트워크 모델들은 Dropout과 같은 네트워크가 학습 데이터에 과적합 되는 것을 방지할 수 있는 기법을 적용할 수 있어, '감정 없음' 카테고리의 분류 성능은 다소 감소하지만, 감정이 나타나는 발화들의 감정 분류 성능은 베이스라인에 비해 상당히 향상된 것으로 보인다. 다만 '부러움', '신뢰' 와 같이 감정을 나타내는 발화들의 수가 충분히 많지 않은 발화들을 분류하는 데에 있어서는 모든 모델들이 정답을 거의 맞히지 못해 좋지 않은 성능을 보였다.

다음으로 표 8을 보면, LSTM 모델과 CNN 모델을 비교했을 때 LSTM 모델에서는 '감정 없음' 카테고리의 분류 성능이 상당히 떨어졌지만, 나머지 감정 카테고리 들을 가장 많이 맞춘 것을

확인할 수 있다. 반면에 CNN 모델에서는 '감정 없음' 의 분류 성능을 높이면서, 동시에 감정이 있는 발화들에 대한 분류 성능도 베이스라인에 비해 향상되어 전체적으로 좋은 감정 분류 성능을 보였다

표 6 감정 카테고리 별 모델 분류 성능

감정 카테고리	정답 개수 (개)			
	SVM	**LSTM**	**CNN**	**CNN-LSTM**
감정 없음	1835	1772	1821	1824
기쁨	70	88	76	81
놀람	26	27	33	32
화남	5	11	8	9
두려움	1	4	2	4
슬픔	35	44	43	39
싫어함	0	2	0	2
바람	5	8	4	10
미안함	9	14	15	15
부러움	0	0	0	0
신뢰	0	0	0	0
총합	1986	1970	2002	2016

또한 두 딥 뉴럴 네트워크 모델들의 단점을 상호 보완한 CNN-LSTM 모델은 LSTM 모델보다 '감정 없음' 의 분류 성능을 높이면서, 동시에 CNN 모델보다 감정이 있는 발화들의 분류 성능까지 높여 전체적인 분류 성능을 향상시켰다. 그 결과 실험에 사용된 모델 들 중 가장 좋은 분류 성능을 보였다. CNN-LSTM 모델이 문장으로부터 감정 분류에 효과적인 특징들을 추출해 활용하며, 동시에 대화에서의 문맥을 효과적으로 고려하여 대화에서의 감정 분류 문제에 있어 효과적인 모델임을 확인 할 수 있었다.

표 7 분류 모델 별 감정 없음/감정 분류 성능

분류 모델	감정 없음 (개)	감정 (개)
SVM	**1835**	151
LSTM	1772	**198**
CNN	1821	181
CNN-LSTM	1824	192

마지막으로 CNN-LSTM 모델에서 성능이 얼마나 개선되었는지를 더 자세히 분석하기 위하여 각 감정 카테고리들의 F1 Score을 계산하여 베이스라인인 SVM 모델과 비교하였다. 비교에는

각 모델에서 정밀도나 재현율을 계산할 수 없어 비교할 수 없는 '두려움', '싫어함', '부러움', '신뢰'를 제외한 7개의 감정 카테고리만을 비교에 사용하였다. 또한 감정 카테고리에서 '감정 없음'의 비율이 과반수를 넘게 차지하기 때문에, '감정 없음'을 분석에 포함한 경우와 포함하지 않은 경우 모두에 대해 F1 Score를 계산하였다.

표 9를 보면, '감정 없음'을 제외했을 때의 '화남', '슬픔' 카테고리를 제외한 모든 경우에서 F1 Score가 상승하여 CNN-LSTM 모델의 사용이 감정 분류에 효과적임을 확인할 수 있었다.

표 8 SVM/CNN-LSTM 모델의 F1 Score

감정 카테고리	F1 Score (%)		F1 Score (%) (감정 없음 제외)	
	SVM	**CNN-LSTM**	**SVM**	**CNN-LSTM**
감정 없음	84.58	85.40		
기쁨	51.74	61.83	88.16	94.19
놀람	27.32	33.33	75.76	84.21
화남	8.76	11.27	66.67	64.00
슬픔	47.41	48.57	84.21	81.93
바람	11.59	25.97	66.67	80.00
미안함	51.43	71.43	94.74	100.00
Macro	75.97	77.56	83.38	87.59
Micro	40.40	48.26	79.37	84.05

Chapter 5

결론 및 향후 연구

본 논문은 대화에서의 발화 감정 분류를 위해 딥 뉴럴 네트워크 모델들을 어떻게 적용할 수 있는지를 연구하였다. 사용자들간의 대화에서 감정이 나타나는 양상을 반영하고, 대화 문맥을 고려할 수 있는 효과적인 딥 뉴럴 네트워크 모델을 적용하여, 다른 다양한 모델과 감정 분류 성능을 비교 분석하였다. 또한 모델의 입력으로 쓰이는 발화 텍스트의 임베딩을 통해 데이터 부족 문제를 완화하고 Drop-out 을 통해 분류 모델의 과적합 문제를 최소화 하도록 하였다.

실험 결과, 감정 분류를 위한 딥 뉴럴 네트워크 모델의 사용이 “감정 없음" 라벨의 분류 성능은 다소 감소시키지만, 이외 다른 감정들에 대한 분류 성능을 전체적으로 향상시킬 수 있음을 보였다. 또한 대화 문맥을 반영한 감정 분류에 있어 기존의 다양한 자질들을 사용한 SVM 모델이나 단순한 구조의 LSTM, CNN 을 사용한 모델에 비해 CNN-LSTM 모델의 사용이 적합하다는 것을 발견할 수 있었다.

향후에는 Attention Mechanism 을 적용한 LSTM 구조를 사용하여, 발화의 어떤 부분, 대화에서 어떤 발화가 감정 분류에 결정적인 영향을 미치는지 분석할 수 있는 여지가 있다. 또한 딥 뉴럴 네트워크 구조의 개선 및 파라미터 최적화를 통해 성능을 더 향상 시킬 수 있을 것이다.

References

[1] Picard, Rosalind W., and Roalind Picard. “Affective computing”. Vol. 252. Cambridge: MIT press (1997)

[2] Robert Plutchik, “A general psychoevolutionary theory of emotion”, In Emotion: Theory, Research, and Experience: Vol. 1. Theories of Emotion (1980)

[3] Hochreiter, Sepp, and Jürgen Schmidhuber, "Long short-term memory." Neural computation, pp.1735-1780 (1997)

[4] Paul Ekman, "An Argument for Basic Emotions", In Cognition and Emotion (1992)

[5] Valstar, Michel F., et al., "The first facial expression recognition and analysis challenge." Automatic Face & Gesture Recognition and Workshops 2011 IEEE International Conference (2011)

[6] Saima Aman and Stan Szpakowicz, "Identifying Expressions of Emotion in Text", In Proceedings of 10th International Conference on Text, Speech and Dialogue (2007)

[7] Matthew Purver and Stuart Battersby, "Experimenting with Distant Supervision for Emotion Classification", In Proceedings of EACL 2012

(2012)

[8] 강상우, “대화 시스템을 위한 사용자 발화 문장의 감정 분류”, 인지과학, 제21권, 제4호, pp. 459-480, (2010)

[9] Takayuki Hasegawa et al., "Predicting and Eliciting Addressee’s Emotion in Online Dialogue", In Proceedings of ACL 2013 (2013)

[10] 신동원 외, “임베딩 자질을 이용한 대화의 감정 분류”, 제27회 한글 및 한국어 정보처리 학술대회, pp. 109-114, (2015)

[11] Dumais, Susan T. "Latent semantic analysis." Annual review of information science and technology 38.1, pp. 188-230 (2004)

[12] Blei, David M et al., "Latent dirichlet allocation" the Journal of machine Learning research 3, pp. 993-1022 (2003)

[13] Bengio, Yoshua, et al., "A neural probabilistic language model" The Journal of Machine Learning Research 3, pp. 1137-1155 (2003)

[14] Mikolov, Tomas, et al., "Recurrent neural network based language model" INTERSPEECH 2010, 11th Annual Conference of the International Speech Communication Association (2010)

[15] Mikolov, Tomas, et al. "Efficient estimation of word representations in vector space." arXiv preprint arXiv:1301.3781 (2013)

[16] Baroni et al., "Don’t count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors.”, Proceedings of the 52nd

Annual Meeting of the Association for Computational Linguistics. Vol. 1 (2014)

[17] XU, Ruifeng, et al., "Word Embedding Composition for Data Imbalances in Sentiment and Emotion Classification", Cognitive Computation 7.2, pp. 226-240 (2015)

[18] Y. Bengio, A. Courville, and P. Vincent., "Representation Learning: A Review and New Perspectives," IEEE Trans. PAMI, special issue Learning Deep Architectures (2013)

[19] Y. Kim, “Convolutional Neural Networks for Sentence Classification”, Conference on Empirical Methods in Natural Language Processing (2014)

[20] J. Lee and Franck Dernoncourt, “Sequential Short-Text Classification with Recurrent and Convolutional Neural Networks”, Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (2016)

[21] X. Glorot and Y. Bengio., “Understanding the difficulty of training deep feedforward neural networks”, In International Conference on Artificial Intelligence and Statistics (2010)