

Emotion Recognition from Korean Text

(Abstract) 이 연구의 목적은 한국어 텍스트로부터 감정을 인식하고 이를 알맞게 분류하는 것이다. 다양한 방법으로 한국어 텍스트를 분류하고 각각의 방법에 따른 정확도를 분석한다. 감정 label의 종류는 행복, 사랑, 놀람, 화남, 슬픔, 무서움, 중립 7가지로 나눈다. 한국어 텍스트 데이터는 대중적인 SNS인 Twitter에서 수집한다. 감정을 인식하는 방법은 텍스트 전처리 방법, 특징 추출 기준, 다양한 분류기 사용으로 달라질 수 있다.

1. INTRODUCTION

한국어 텍스트를 읽어와 기계 학습을 이용하여 행복, 사랑, 놀람, 화남, 슬픔, 무서움, 중립 이 7가지 감정으로 분류한다. 다양한 텍스트 전처리, 특징 추출 방법, 분류기를 이용하여 감정을 분류하고 각각의 방법을 평가할 것이다.

이를 통하여 한 인물, 물건, 회사, 사건, 주제 등에 대한 대중의 전반적인 생각을 파악할 수 있다. 또한 경제적, 환경적, 사회적 요인에 따라 달라지는 시민의 행복도를 판단하고 관찰하는 데 사용될 수 있다. 뿐만 아니라 각종 SNS, 대화메신저에서 사용자의 기분을 파악할 수 있으며 그를 이용하여 유용한 어플리케이션을 새롭게 개발할 수 있다.

한국어 텍스트 데이터는 대중적이며 감정 표현이 많이 나타나는 SNS 매체인 Twitter에서 수집한다.

2. REQUIREMENT ANALYSIS

감정의 종류는 행복, 사랑, 놀람, 화남, 슬픔, 무서움, 중립 이 7가지로 선정하였다. 텍스트는 반드시 한가지의 감정으로 분류되도록 한다. 기계 학습을 수행할수록 정확도는 더욱 높아져야 할 것이다.

텍스트를 통해 분석한 감정은 실제 감정에 부합하는 결과값이어야 한다. 또한 텍스트에 유행어, 맞춤법이 어긋난 어구, 이모티콘 등을 포함하고 있더라도 이를 인식하고 이에 대한 감정 인식을 알맞게 수행하여야 한다.

데이터를 입력하면 그에 따라 학습을 하고 각각의 결과값을 계산하여 각 분류 방법과 분류기에 따른 정확도를 출력하도록 한다.

3. HIGH-LEVEL AND DETAILED DESIGN

한국어 텍스트 감정 인식을 더욱 정확하게 하기 위한 방법으로 기본 데이터를 제작한다. 기본 데이터는 감정을 나타내는 핵심 단어들과 단어들이 나타내는 감정 label을 매칭시킨 감정 사전이다. 이

데이터는 기계 학습을 시작할 때에 항상 학습시키도록 한다.

한국어 텍스트 감정 인식의 과정은 크게 3가지 단계로 나뉜다. 첫번째로 각 문장을 나누는 단위를 정하고 이에 따라 문장을 나누는 전처리 과정이 필요하다. 나누어진 어구 파편들은 벡터에 저장하여 다음 단계에서 사용한다. 두번째 단계에서는 전처리과정을 거친 문장의 파편에서 감정에 대한 특징을 추출한다. 마지막으로 추출한 특징들을 이용하여 분류기로 감정을 분류하고 정확도를 계산한다.

3.1. 텍스트 전처리

텍스트 전처리 과정에서 문장을 나누는 방법으로 어절 단위로 나누느냐, 형태소 단위로 나누느냐, 형태소 단위로 나눈 것에 단독으로 등장하는 자음(예를 들어 ‘ㄱㄱ’, ‘ㅍㅍ’ 와 같은 것이 있다.)까지 인식을 하느냐, 이렇게 3가지 방법이 있다. 이에 따라 한 문장을 나누고 나누어진 어구 파편들을 벡터에 저장한다. 벡터는 전체 텍스트 데이터에 대한 파편들을 저장한다.

3.2. 텍스트 특징 추출

전처리한 텍스트에서 특징을 추출하는 방법으로는 감정을 판단할 때에 어구의 개수를 몇개(n-gram)씩 묶어서 세는가에 따른 방법과 각 n-gram 묶음을 세는 방법을 고려해볼 수 있다.

어구를 n-gram씩 묶어서 판단하는 것은 n-gram의 어구들 사이의 연관성을 이용하여 문장의 감정을 인식하는 데에 효과적이다.

묶음을 세는 방법으로는 2가지가 있다. 먼저 텍스트 전처리 과정에서 나눈 벡터 안의 어구들 중에서 문장에서 등장하는 어구와 같은 것을 1로 count, 등장하지 않으면 0으로 count하는 방법(frequency)이 있다. 또 다른 방법으로는 문장에서 등장하는 어구를 나오는 만큼 count를 하는 방법(TFIDF)이 있다. 예를 들어, 문장이 “오늘

날씨가 너무 좋아서 기분이 너무 좋다.” 라면 ‘좋’이라는 어구를 2로 count하는 것이다.

3.3. 분류기

분류기는 Naïve Bayes Classifier, Support Vector Machine, Recurrent Neural Netwok(RNN)의 Gated Recurrent Unit(GRU)를 사용한다. Support Vector machine의 kernel은 linear와 poly로 나뉜다.

각 분류 방법에 따른 감정 인식 결과를 평가하는 방법으로는 5-fold cross validation을 사용한다.

4. IMPLEMENTATION

기본 데이터는 감정에 관련된 426개의 단어로 구성하였으며 각 단어마다 7가지 감정 label 중 알맞은 label을 붙였다.

감정 인식에 쓰일 훈련 데이터와 테스트 데이터는 Twitter에 크롤링하였다. 최종적으로 수집한 데이터는 1384개의 텍스트이다. 데이터는 약 500개씩 세차례에 걸쳐서 수집을 하였다. 수집한 데이터에 label을 붙이는 것은 다음과 같이 진행하였다. 첫번째의 500개는 기본 데이터를 이용하여 학습시켜 label을 붙이고 이를 직접 수정한다. 두번째의 500개는 기본 데이터와 첫번째에서 수집한 500개의 데이터로 학습시켜 label을 붙이고 이를 수정한다. 세번째도 이전의 데이터와 기본 데이터를 이용하여 label을 붙이고 수정한다.

수집한 데이터를 학습 데이터와 테스트 데이터로 나누어 학습과 그에 따른 결과를 측정한다. 수집한 데이터를 5등분하여 4/5의 데이터와 기본 데이터로 학습을 하고 남은 1/5의 데이터로 테스트를 한다. 이를 각각의 1/5의 데이터로 5번 반복하여 5번의 정확도를 측정하고 평균 정확도를 구한다.

각 분류기, 기본 데이터의 사용 유무, 전처리 과정, 어구 count 방법, N-gram에 따라 측정하고 비교한 정확도 그래프는 다음과 같다.

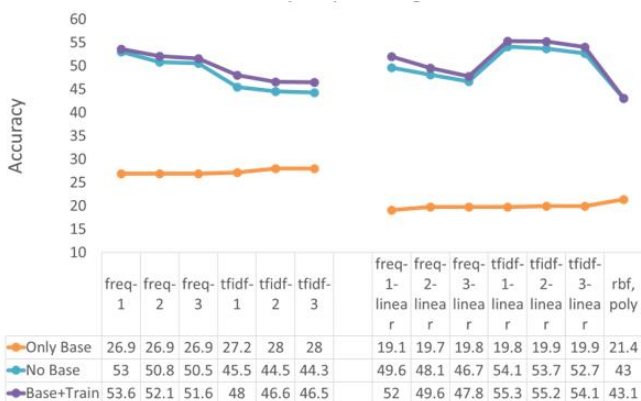


Fig. 1. 기본 데이터에 따른 NBC와 SVM의 정확도.

기본 데이터만을 훈련 데이터로 사용하였을 때에는 Twitter에서 수집한 데이터를 포함하였을 때보다 현저히 낮은 정확도를 보여준다. 반면에 Twitter에서 수집한 데이터에 기본 데이터를 함께 사용하였을 때와 사용하지 않았을 때의 정확도는 약 2% 정도의 차이로 Twitter에서 수집한 데이터의 유무에 대한 차이와 비교하였을 때 미미한 차이이다. 하지만 기본 데이터를 사용하였을 때에 항상 정확도를 높일 수 있다는 것을 알 수 있다.

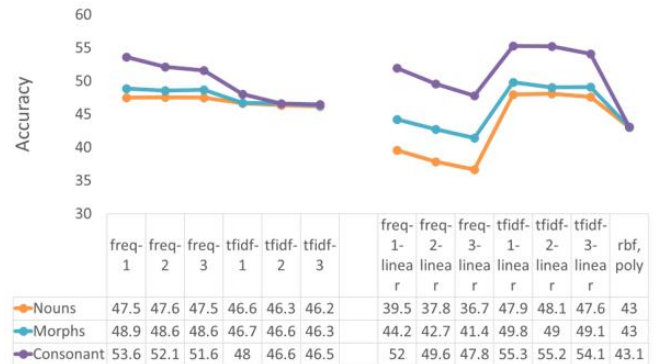


Fig. 2. 어절, 형태소, 자음 포함 형태소로 나누는 전처리 과정에 따른 NBC와 SVM의 정확도.

문장을 어절 단위로 나누느냐, 형태소 단위로 나누느냐, 자음을 포함하여 형태소 단위로 나누느냐에 따른 정확도 결과는 위 그래프와 같다. 어절 단위로 나누었을 때 정확도가 가장 낮다. 그 이유는 한국어의 특성에 있다. 한국어에는 조사에 따라 어절의 형태가 달라지는데 문장을 어절 단위로 나누면 조사를 처리하지 못하므로 같은 어근에 대하여 다른 어구로 처리하게 된다. 예를 들어, ‘좋다’, ‘좋아’, ‘좋습니다’를 모두 다른 어구로 생각하게 되는 것이다. 한국어에는 자음으로 감정을 표현할 때가 많기 때문에 자음을 포함하여 문장을 나누었을 때 가장 높은 정확도를 보여준다.

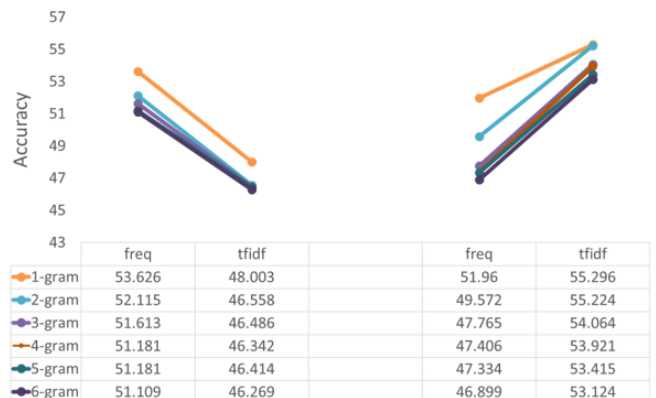


Fig. 3. N-gram에 따른 NBC와 SVM의 정확도.

N-gram에 따른 정확도는 1-gram일 때 가장 높았고, gram이 늘어날수록 정확도가 낮아졌다. 이렇게 나온 이유는 Twitter의 특성상 의미 없는 단어의 나열이 잦다. 또한 문장을 작성하는 사람이 한 사람이 아니라 매우 다양하므로 어구와 어구 사이의 연관성을 찾는 것이 정확도에 부정적인 영향을 미치게 될 때가 종종 있을 것이다.

위 3가지 그래프에서 각 gram를 1로 count한 왼쪽의 그래프와 어구의 개수만큼 count를 한 오른쪽의 그래프의 결과는 분류기에 따라 달랐다. Naïve base classifier에서는 gram을 1로 count하는 것이 정확도가 더 높았고, Support Vector Machine에서는 개수만큼 count를 하는 것이 정확도가 더 높았다.

1384개로 늘었을 때, 1-gram TFIDF linear SVM의 정확도는 52.686 %에서 55.296 %로 약 5 % 증가하였다. 데이터의 양을 많이 늘린다면 더 높은 정확도의 결과를 가져올 수 있을 것이다.

만약 다양한 이모티콘과 마침표와 같은 특수문자들을 인식하여 감정 분류에 사용한다면 감정 인식 정확도는 더욱 높아질 것이다.

REFERENCE

- [1] Cecilia Ovesdotter Alm, Dan Roth, and Richard Sproat, "Emotions from text: machine learning for text-based emotion prediction" (2005)
- [2] Chetan R. Chopade, "Text Based Emotion Recognition: A Survey", International Journal of Science and Research (IJSR), 409-414 (2013)
- [3] The Emotion Machine, <http://www.theemotionmachine.com/classification-of-emotions>.
- [4] Microsoft, <https://www.microsoft.com/developerblog/real-life-code/2015/11/30/Emotion-Detection-and-Recognition-from-Text-using-Deep-Learning.html>.

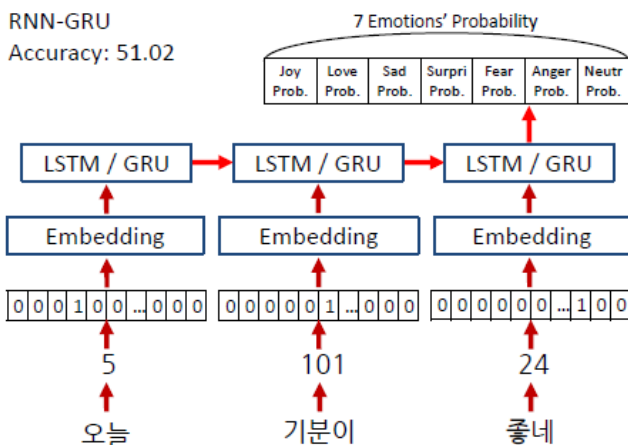


Fig. 4. GRU의 처리 과정과 GRU의 정확도.

마지막으로 이용한 분류기인 deep learning을 이용한 GRU는 형태소와 자음을 인식하는 전처리 과정을 거친 후 실행하였다. GRU의 정확도는 51.02 %였다.

예상과는 다르게 deep learning을 통한 인식의 결과값이 다른 분류기의 결과값에 비해 월등히 높지 않았고 가장 높은 결과값도 아니었다. 이에 대한 이유는 테스트 데이터의 크기가 1384개로 deep learning을 하기에 충분하지 않은 양이었기 때문이다.

5. CONCLUSION

모든 정확도를 비교하였을 때 1-gram TFIDF를 이용한 linear SVM이 가장 높은 정확도 결과를 보여주었다.

기계 학습이 모델을 학습하고 이를 이용하여 새로운 모델을 예측하는 것이기 때문에 데이터의 양은 기계 학습의 성능을 올리는 것에 큰 영향을 끼친다. 이 프로젝트에서 데이터의 양이 635개에서