

감정 분석에서의 이모티콘 활용의 유효성

Korea University COSE461 Final Project

장영재
Department of Economics
Team 30
2014150061

육현준
Department of Control & Instrumentation Engineering
Team 30
2014270619

황수연
Department of Computer Convergence Software
Team 30
2019270605

Abstract

감정분석에 있어 이모티콘을 활용하는 것이 언어 모델의 성능을 개선한다는 연구결과가 일반적이거나, 본 프로젝트는 ㉠ 데이터 선정, ㉡ 전처리 방식, ㉢ 모델 선택 및 설계, ㉣ 평가 방식 및 척도에 따라 성능이 악화되는 경우도 있을 것으로 예측하였다.

이에 ㉠ Snapchat 어플리케이션에 대한 사용자의 평가 데이터를 분석자료로 선정하여 ㉡ 데이터내 이모티콘을 텍스트로 대체한 데이터와 이모티콘을 소거한 데이터로 양분하여 전처리한 후 ㉢ 대표적인 언어 모델인 NB, SVM, LSTM, BERT에 대해 양 데이터마다 별개로 학습시키고 ㉣ 학습된 모델마다 다시금 두 데이터를 통과시켜 정확도, 정밀도, 재현율, f1 차원에서 성능을 평가해 보았다.

그 결과, **정확도** 및 **정밀도** 차원에서는 모델에 따라 이모티콘을 텍스트로 대체한 데이터로 학습한 경우가 이모티콘을 제거한 데이터로 학습한 경우보다 더 나은 성능을 보이는 경우가 있었으나, 되려 유의미한 개선을 가져오지 못하거나 오히려 악화되는 경우도 있었다. 또한 이모티콘을 텍스트로 대체한 데이터로 학습한 모델로 이모티콘을 제거한 데이터에 대해 성능을 시험한 경우나, 이모티콘을 제거한 데이터로 학습한 모델로 이모티콘을 텍스트로 대체한 데이터에 대해 성능을 시험한 경우 모두 성능이 개선되거나 악화된 경우 모두 있었다. **재현율**과 **f1** 차원에서는 모델에 따라 다소간 차이는 있으나 대체로 이모티콘을 텍스트로 대체한 데이터로 모델을 학습하여 이모티콘을 제거한 데이터로 하는 것이 수 있었다. 한편, 기계학습 모델은 딥러닝 모델보다 현저히 낮은 재현율을 보여주었다. 결론적으로, 감정분석에 있어 이모티콘을 활용하는 것이 일반적으로 성능 개선을 가져온다기보다는 감정분석이 수반되는 테스트와 데이터의 특성과 모델 선정 및 설계, 평가 방식에 따라 그 성능이 상이하다고 할 수 있다.

1 Introduction

감정 분석(Sentiment Analysis)이란, 자연어로부터 의견, 감정, 평가, 태도 등 언어 구사자에 대한 주관적인 정보를 도출하는 것을 말한다. 감정 분석에서는 판단에 결정적이지 않은 데이터를 전처리 과정에서 제거하여 텍스트 길이를 최소화함으로써 모델의 학습시간을 단축시키고 성능을 개선하는 것이 중요한데, 종래의 연구는 이모티콘이 문장부호(Punctuation)처럼 특별한

감정가를 담지하지 않는다고 보아 이를 전처리과정에서 제거하는 것이 일반적이었다.

하지만 근래에 감정분석에 활용되는 데이터의 출처가 Twitter처럼 일상적이고 즉흥적인 언어 표현이 주류를 이루는 소셜미디어가 되기 시작하면서 자연스레 이모티콘이 데이터에서 차지하는 비중이 점차적으로 증가하기에 이른다. 더욱이 이모티콘 이미지가 개선되고 그것의 종류 역시 많아지면서, 이모티콘 자체가 감정을 표현하는 주요한 방식 중 하나라는 것을 부인할 수 없게 되었다. 이러한 변화를 반영하듯 최근에는 이모티콘을 감정 분석 과정에 활용하는 것이 모델의 성능을 개선한다고 보고하는 연구 결과가 등장하고 있다.

그러나 어떠한 언어 모델도 시공간적 상황적 맥락을 초월하여 우수하다고 단정할 수는 없으므로, 이모티콘을 감정분석에 활용하는 것이 성능 측면에서 유효하다는 주장을 몇몇 증거에 따라 세분화하여 검토해볼 필요가 있다고 할 수 있다. 따라서 본 프로젝트는 그 증거로서 크게 ㉠ 데이터 선정, ㉡ 전처리 방식, ㉢ 모델 선택 및 설계, ㉣ 평가 방식과 척도를 제시하고, 해당 증거에 입각하여 이모티콘 활용의 유효성을 살펴보고자 한다.

2 Related Work

감정 분석에 대한 연구를 살펴보면, 아직까지는 감정 분석에 이모티콘을 활용하는 것이 활발하지는 않은 것으로 보이나, 그럼에도 이모티콘을 감정분석에 활용해본 연구를 참고하여 본 프로젝트의 접근 방식을 구체화하고자 종래의 연구를 살펴보고자 한다. 이러한 연구들은 이모티콘에 대한 전처리 방식을 제시하거나 이모티콘을 활용하여 다양한 모델을 학습시키는 것이 성능을 개선시킨다는 것을 제시하는 것에 집중되어 있다. 다음은 이모티콘을 활용하여 감정분석을 진행한 연구를 정리한 것이다.

- 한 연구는 이모티콘의 의미상 모호함으로 인하여 감정 분석에 이모티콘을 활용하는 경우 모델의 정확도가 감소할 것이라고 예상했으나, 연구 결과 오히려 모든 모델에서 성능(정확도)이 향상되었다고 분석하였다. 다만 이모티콘을 단순히 다른 단어로 변환하여 모델을 훈련시키는 경우 모델의 성능이 저하될 수 있어 이모티콘을 적절한 문자로 치환하기 위한 별도의 학습이 요구된다고 보았다.[1]
- 다른 연구는 Twitter의 트윗 데이터를 활용하여 이모티콘별로 감정 점수를 도출한 후 이를 트윗별 감정점수를 도출하는데 활용하였다. 그리하여 이모티콘이 부정적인 경우보다 긍정적인 경우에 활용하는 것이 정확도 개선이 도움이 되었다는 것이다.[2]

상기한 연구는 ㉠ 데이터 선정, ㉡ 전처리 방식, ㉢ 모델 선택 및 설계, ㉣ 평가 방식과 척도 차원에서 다음과 같은 양상을 보였다.

- 데이터 선정 : 대부분 Twitter와 같은 소셜 미디어 데이터를 활용하고 있었다.
- 전처리 방식 : 감정 분석에 영향을 주지 않는 불필요한 데이터(문장부호, 불용어)는 제거하였다.
- 모델 선정 및 설계 : 대체로 Logistic Regression, Naive Bayes, SVM, LSTM 모델이 활용되고 있었으며 임베딩시 Word2Vec이 주로 사용되었다.
- 성능 평가 : 감정 분석의 특성상 정확도를 사용하는 경우가 많았다.

3 Approach

기술한 연구경향을 감안하여 다음과 같은 접근을 시도해보고자 한다.

- 데이터 선정 : 연구에 빈번히 활용되는 소셜미디어 데이터가 아닌 다른 양상의 데이터를 활용하여 당해 데이터에서도 감정분석시 이모티콘을 활용하는 것이 유의미한 성능 개선을 가져올지 분석해보고자 한다.
- 전처리 과정 : 이모티콘마다 별개의 Token Id를 부여하거나 개별적 감정가(polarity)를 도출한 후 문장의 감정점수를 도출하는 방식보다는, 모델이 이모티콘 자체를 하나의 텍스트로 인식하도록 하여 이모티콘을 활용치 않던 종래의 분석 방식과 모의상의 유사성을 높이고자 했다. 따라서 이모티콘을 문자로 대체한 데이터와 이모티콘을 소거한 데이터로 양분하여 전처리하고자 한다.

- 모델 선택 및 설계 : 최근에는 자연어처리에 있어 딥러닝 언어 모델이 주류를 차지하고 있으므로 대표적인 딥러닝 모델인 LSTM, BERT를 활용하는 대신, 비교군을 위해 종래의 머신러닝 언어 모델인 SVM, NB도 사용하고자 한다. 단, 딥러닝 모델의 중요도를 고려하여 BERT의 경우에는 BERT-Base 외에 파생 모델도 함께 분석에 활용하였다.
- 평가 방식 및 척도 : 통상적으로 학습시 활용되는 데이터와 평가시 활용되는 데이터의 전처리 방식을 일치시키거나, 이모티콘의 종류나 쓰임새가 급변한다는 특징을 감안해본다면 이모티콘을 활용하지 않고 학습한 모델에 대해서도 이모티콘이 담긴 데이터를 기입하여 평가하여야 하는 경우가 있을 것으로 보았다.
그리하여 평가 방식에 있어 학습 데이터와 평가 데이터의 전처리 방식을 반드시 일치시키지 않고 대신 이모티콘이 제거된 데이터로 학습한 모델과 이모티콘이 문자로 대체된 데이터로 학습한 모델 양자에 각각 다시금 이모티콘이 제거된 데이터와 이모티콘이 텍스트로 대체된 데이터를 각각 대입하여 성능을 평가해보는 방식을 취하기로 하였다. 한편, 평가 척도에 있어 상당수의 연구가 평가를 위해 정확도를 활용하고 있으나, 본 프로젝트는 감성 분석이 다양한 태스크에서 다른 자연어처리 분야와 복합적으로 활용될 수도 있다고 보아 정확도 외에도 정밀도, 재현율, f1을 모두 성능 평가 지표로 활용해보고자 한다.

4 Experiments

4.1 Data Selection

기존의 연구가 주로 Twitter와 같은 '소셜미디어' 데이터를 활용하고 있는데, 새로운 양상의 데이터에 대한 감성 분석시 이모티콘을 활용하는 것이 성능에 미치는 영향을 도출하고자 소셜미디어 데이터가 아닌 상품 내지 서비스에 대한 '평가' 데이터를 활용해보고자 하였다.

다만, 평가 데이터를 분석 자료로 삼는 경우 다음과 같은 점이 예견되었는데, 첫째, 일상성이 특징인 소셜미디어와 다르게 평가 데이터는 작성자가 글을 남기는 것 자체가 감정적 의도를 함축할 가능성이 있어 데이터 복잡도가 낮아 이모티콘의 활용 유효성이 낮을 것으로 예견되었다. 둘째, 논제가 광범위한 소셜미디어에 비해 특정한 상품 내지 서비스가 주제인 평가데이터는 등장하는 단어의 다양성이나 어휘가 상대적으로 단순할 가능성이 있어 이모티콘을 감성분석에 활용하지 않더라도 충분히 성능이 우수할 수도 있을 것으로 예측되었다.

이러한 예측 하에 본 프로젝트는 리뷰 데이터의 하나로서 Snapchat이라는 미국의 메신저 어플리케이션에 대한 사용자의 평가 데이터를 분석 자료로 택하였다. 데이터는 kaggle에서 습득하였으며 해당 자료는 2018.09.12에서부터 2019.11.19까지 Google Playstore에 사용자가 남긴 평가를 수집한 것이다. 원 데이터는 다음과 같은 columns으로 구성되어 있으며 32874개의 개별 item으로 이루어져 있다.

- date : review 게시 날짜
- Rating : 사용자 평점(1~5점)
- Helpful : 이 review가 얼마나 유용한지에 대한 정보
- Review Text : 평가 텍스트

4.2 Preprocessing

전처리는 다음과 같이 진행하였다. 주지할 것은 이모티콘에는 키보드 입력만으로 완성가능한 ☹️bracket형 이모티콘(e.i. :) , :(과 키보드로 입력이 불가능한 😊image형 이모티콘이 있다는 것이다. 이모티콘 유형마다 전처리 과정도 다소 상이하게 적용된다.

- 무관한 feature 제거 : 감성분석과 무관한 feature인 date와 Helpful column을 제거하였다.
- label 설정 : 1부터 5사이의 값을 가지는 Rating을 label로 설정하였으며, 1~3 사이 값을 0(negative)으로, 4~5 값을 1(positive)로 이진화하였다. 그 결과 19834개의 0(negative) 데이터와 13041개의 1(positive) 데이터로 나누어졌다.
- 불용어(stopword) 제거 : 특별한 감정가를 지니지 않는 불용어를 제거하였다. 불용어로는 조사, 접미사가 있다. NLTK 라이브러리에서 제공하는 100여개 이상의 불용어 리스트를 활용하였다.

- image형 이모티콘 처리 : image 이모티콘은 사용자가 사용하는 device에 따라서도 다양하게 달라진다. 이러한 다양성 때문에 demoji 라이브러리에서 제공하는 사전을 활용하여 전처리를 진행하였다. 학습 데이터에 존재하는 모든 image 이모티콘을 demoji 사전에서 정의한 text로 변경하여 이모티콘이 replace된 데이터를 생성하였다.
- bracket형 이모리콘 처리 : 이모지를 text로 변경했지만 감정을 충분히 나타낼 수 있는 feature인 키보드로 입력가능한 bracket형 이모티콘이 아직 data set에 남아있다. 관련된 이모티콘만 선정하여 직접 dictionary로 만들어 변환하여 data set에 감정과 관련된 이모지와 이모티콘을 text로 변환하여 replace data set을 생성하였다.
- 모델별 차등적 전처리 : NB와 SVM은 Scikit-Learn에서 제공하는 TF-IDF 방법을 이용하여 데이터를 인코딩하였다. TF-IDF단어를 갯수 그대로 카운트하지 않고, 모든 문서에 공통적으로 들어있는 단어의 경우 문서 구별 능력이 떨어진다고 보아 가중치를 축소하는 방법인데, max features를 500으로 두어 상위 500개의 word만 고려하였다.

4.3 Model Selection & Settings

최근에는 딥러닝에 기반한 감정분석 사례가 증가하고 있으므로 머신러닝 기반 모델인 SVN, NB와 딥러닝 기반 모델인 LSTM과 BERT를 선정하였다. 이 중 BERT는 SOTA 모델 중 하나로 평가받고 있어 파생 모델도 다양하기 때문에 BERT-Base ALBERT, RoBERTa, ELECTRA를 추가로 선정해보았다.

- SVM : SVM(Support Vector Machine)은 기계 학습 모델의 하나이자 지도 학습 모델이며, 주로 분류(classification)와 회귀 분석에 사용된다. 두 카테고리 중 어느 하나에 속한 데이터의 집합이 주어졌을 때, SVM 알고리즘은 주어진 데이터 집합을 바탕으로 하여 새로운 데이터가 어느 카테고리에 속할지 판단하는 비확률적 이진 선형 분류 모델을 만든다.
- NB : NB(Naive Bayes) 모델은 단순한 지도 학습 알고리즘의 한 종류로 Naive Bayes정리에 기반한 통계적 분류기법이다. Naive Bayes 알고리즘의 가장 강력한 전제는 feature들간의 독립이라는 조건이 존재한다.
단순하지만 정확도가 높고 특히 대용량 데이터에 대해서 속도가 빠른편으로 알려져 있다.
스팸메일 분류에서 간단하지만 높은 정확도를 보여주는 알고리즘으로 긍정과 부정을 분류하는 감정분석 분류 비교모델로 선정하였다.
- LSTM : 장단기 메모리(Long Short-Term Memory, LSTM) 모델은 순환신경망(Recurrent Neural Network) 모델이 지닌 기울기 소실 문제를 보완코자 당해 모델에 게이트를 추가하여 이전 단계의 웨이트를 참고하도록 설계한 모델이다. 설계 차원에서는 128개의 은닉층이 배정된 LSTM층 다음에 다시 128개로 구성된 두개의 affine층을 통과하도록 설계하였다.
- BERT : BERT(Pre-training of Deep Bidirectional Trnasformers for Language Understanding)는 RNN 후속인 트랜스포머(transformer) 아키텍처 내 일부인 인코더를 활용하는 모델로서 모든 테스트마다 개별적 훈련을 진행하는 것이 아니라 범용 목적의 사전학습을 진행한 후 사전학습된 모델을 태스크에 맞춰 재학습시키며 미세조정(fine-tuning)을 하는 모델이다. 최근에는 BERT 파생모델이 다수 등장하고 있기 때문에 Bert 모델 외 AlBERT, RoBERTa, ELECTRA를 함께 분석에 활용하였다. 네가지 모델은 모두 Huggingface에서 제공하는 Transformers를 통해 진행하였으며,
BERT-Base : 110M의 파라미터, 12 레이어, 768 은닉층으로 구성되어 있다.
ALBERT-Base : 12M의 파라미터, 12 레이어, 768 은닉층으로 구성되어 있다.
RoBERTa
ELECTRA

4.4 Evaluation method

언어 모델을 평가하는 지표로는 내부 평가(Intrinsic Evaluation)과 외부평가(Extrinsic Evaluation)이 있다. 성능을 비교해야하는 모델이 여러 개인 경우에 일일이 모델들에 작업을 시켜 정확도를 비교하는 것은 공수가 많이 들기 때문에 주로 내부 평가 지표 중 하나인 perplexity를 사용한다. 그리하여 모델 내에서 자신의 성능을 수치화하여 내놓고, 수치가 낮을 수록 언어 모델의 성능이 좋다고 평가한다. 하지만 이는 데이터 상에서 높은 정확도를 보인 것뿐, 사람이 느끼기에 좋은 언어 모델이라는 것을 의미하진 않으므로 주의할 필요가 있다. 따라서 본 프로젝트에선 perplexity보다 정확도, 정밀도, 재현율, f1을 활용하여 언어 모델을 평가하고 비교분석해보고자 한다.

- True Positive(TP) : 실제 True인 정답을 True라고 예측 (정답)
- False Positive(FP) : 실제 False인 정답을 True라고 예측 (오답)
- False Negative(FN) : 실제 True인 정답을 False라고 예측 (오답)
- True Negative(TN) : 실제 False인 정답을 False라고 예측 (정답)
- 정확도 : $accr = (TP + TN) / (TP + FN + FP + TN)$
- 정밀도 : $prec = TP / (TP + FP)$
- 재현율 : $rec = TP / (TP + FN)$
- f1-score : $f1 = 2 \times (prec \times rec) / (prec + rec)$

4.5 Results

학습시 사용한 데이터 유형을 불문하고 모든 언어 모델마다 이모티콘을 제외한 데이터와 이모티콘을 문자로 대체한 데이터를 모델에 통과시켜 성능을 평가해보았다.

Table 1: Accuracy

SVM			Naive Bayes		
	test			test	
train	remove	replace	train	remove	replace
remove	0.79	0.375	remove	0.745	0.347
replace	0.464	0.785	replace	0.406	0.756

LSTM			BERT		
	test			test	
train	remove	replace	train	remove	replace
remove	0.849	0.843	remove	0.852	0.87
replace	0.849	0.85	replace	0.873	0.854

RoBERT			ALBERT		
	test			test	
train	remove	replace	train	remove	replace
remove	0.841	0.843	remove	0.85	0.854
replace	0.841	0.843	replace	0.842	0.838

ELECTRA		
	test	
train	remove	replace
remove	0.852	0.87
replace	0.867	0.859

Table 2: Precision

SVM			Naive Bayes		
	test			test	
train	remove	replace	train	remove	replace
remove	0.818	0.6	remove	0.839	0.674
replace	0.627	0.806	replace	0.652	0.843

LSTM			BERT		
	test			test	
train	remove	replace	train	remove	replace
remove	0.84	0.829	remove	0.873	0.902
replace	0.843	0.851	replace	0.829	0.852

RoBERT			ALBERT		
	test			test	
train	remove	replace	train	remove	replace
remove	0.834	0.862	remove	0.839	0.859
replace	0.834	0.862	replace	0.813	0.831

ELECTRA		
	test	
train	remove	replace
remove	0.84	0.848
replace	0.857	0.854

Table 3: Recall

SVM			Naive Bayes		
test	remove	replace	test	remove	replace
train	0.764	0.273	train	0.671	0.234
remove	0.368	0.764	remove	0.295	0.685
replace			replace		

LSTM			BERT		
test	remove	replace	test	remove	replace
train	0.766	0.761	train	0.734	0.758
remove	0.762	0.754	remove	0.849	0.77
replace			replace		

RoBERT			ALBERT		
test	remove	replace	test	remove	replace
train	0.743	0.722	train	0.771	0.763
remove	0.743	0.722	remove	0.784	0.75
replace			replace		

ELECTRA		
test	remove	replace
train	0.775	0.811
remove	0.802	0.786
replace		

Table 4: f1-score

SVM			Naive Bayes		
test	remove	replace	test	remove	replace
train	0.79	0.375	train	0.745	0.347
remove	0.464	0.785	remove	0.406	0.756
replace			replace		

LSTM			BERT		
test	remove	replace	test	remove	replace
train	0.801	0.794	train	0.798	0.823
remove	0.801	0.799	remove	0.839	0.809
replace			replace		

RoBERT			ALBERT		
test	remove	replace	test	remove	replace
train	0.786	0.786	train	0.839	0.808
remove	0.786	0.786	remove	0.798	0.788
replace			replace		

ELECTRA		
test	remove	replace
train	0.806	0.829
remove	0.829	0.819
replace		

5 Analysis

학습한 모델에 대해 성능 시험을 진행해본 결과 어떤 모델이건 전반적으로 이모티콘을 제거 하든 활용하든간에 80퍼센트대의 정확도를 보였는데, 본 프로젝트가 활용한 데이터 자체가 분석 난이도가 크게 높지는 않았기 때문인 것으로 추측된다. 이는 소셜미디어 데이터와 다르게 평가 데이터의 경우에는 감정 분석시 이모티콘의 활용의 유효성이 상대적으로 낮을 수도 있음을 암시 하는 대목이다. 다만, 그럼에도 평가 척도별로 결과를 세분하여 분석해보면 다음과 같은 유의미한 점을 도출할 수 있었다.

- **정확도** : 머신러닝 모델(SVM, NB)은 모델 학습 당시의 데이터 유형(이모티콘 제거 혹은 대체)과 다른 데이터 유형으로 성능을 시험한 경우 현저하게 성능이 저하되는 양상을 보였으나, 딥러닝 모델(LSTM, BERT)은 같은 데이터 유형으로 성능을 시험한 케이스와 사실상 성능이 차이가 없거나 더 나은 경우가 목격되었다. 이는 정확도 차원에서는 머신러닝 모델보다 딥러닝 모델을 사용하는 것이 유리할 것임을 암시한다. 또한 머신러닝 모델을 사용하는 경우에는 데이터를 모델에 대입하기 전에 반드시 모델 학습시 취했던 전처리 과정을 동일하게 적용해야 함을 알 수 있다.
한편, 딥러닝 모델은 학습 당시의 데이터 유형과 시험시 데이터 유형이 다른 경우 외려 좋은 성능을 보이는 경우가 더러 목격되었는데, 이를 통해 보았을 때 딥러닝 모델 사용시 이모티콘을 분석에 지속적으로 활용하다. 이모티콘을 제거하거나 활용하는 것을 바꾸어 사용하는 것이 더 나은 성능을 보여줄 수 있을 수도 있는 것으로 보인다.
- **정밀도** : 머신러닝 모델은 모델 학습 당시의 데이터 유형과 같은 데이터 유형으로 성능을 시험한 경우에 다른 데이터 유형으로 시험한 경우보다 현저하게 성능이 좋은 양상을 보였다. 하지만 딥러닝 모델에서는 그 둘의 차이가 거의 없거나 다른 데이터 유형으로 시험한 경우에 더 좋은 성능을 보이는 경우도 있었다. 따라서 정밀도 역시 머신러닝보다 딥러닝 모델을 사용하는 것이 정밀도를 높이는 면에선 더 유리할 것이라고 해석할

수 있다. 특히 BERT 모델의 경우 시험하는 데이터 유형에 상관 없이 가장 우수한 정밀도를 보여주었다. 또한 머신러닝 모델을 사용할 경우, 학습한 것과 같은 데이터 유형을 사용하여 시험하는 것이 정밀도의 측면에서 유리하다.

- 재현율 : 머신러닝 모델에서는 학습 데이터와 같은 전처리과정을 거친 평가 데이터가 현저히 높게 나왔다. 하지만 딥러닝 알고리즘인 LSTM부터는 같은 전처리과정을 거친 평가 데이터뿐만 아니라 다른 전처리과정을 거친 평가 데이터에서도 비슷한 재현율을 보여주었다.
주목할만한 점은 BERT 모델에서 이모티콘을 텍스트로 대체한 학습 데이터로 학습한 경우 이모티콘을 제거한 평가 데이터로 평가한 경우 같은 전처리과정을 거친 평가 데이터보다 약 3 ~ 4%정도 재현율이 증가하는 것을 확인하였다. 이후 BERT를 base로한 모델인 ROBERTa와 ALBERT, electra도 bert만큼은 아니지만 텍스트로 대체한 데이터로 학습한 경우 같은 전처리 과정을 거친 평가 데이터의 결과보다 이모티콘을 제거한 평가 데이터에서 비슷하거나 높게 재현율이 증가하였다.
- f1 : f1은 대체로 재현율과 비슷한 양상을 보였다. 즉, 머신러닝 모델에서는 학습시 데이터 유형과 시험시 데이터 유형이 다른 경우 30 ~ 40퍼센트대의 값을 보여주며 급격하게 성능이 저하되었지만, 딥러닝 모델에서는 어느 데이터유형이든 그리고 어느 데이터유형으로 학습했든 유사한 성능치를 보여주었다.

6 Conclusion

상술한 분석을 토대를 통해 보았을 때 데이터와 태스크의 성질, 전처리 과정, 모델의 선택 및 설계, 성능 평가 방식과 척도에 따라 감정분석에서 이모티콘을 활용하는 것이 상이한 성능을 보임은 알 수 있었다. 다만, 본 프로젝트는 다음과 같은 한계가 있다 할 수 있다.

- label 설정의 임의성 : label을 전처리하는 과정에서 평점이 3점인 경우 label을 0(negative)로 간주하였던 것이 지나치게 태스크를 단순화하는 문제가 있었다. label을 -1(negative), 0(neutral), 1(positive)로 세분화하는 것이 타당했을 것으로 보인다.
- 이모티콘 문자 대체의 단순성 : 이모티콘을 문자로 대체하는 과정에서 모 이모티콘 딕셔너리를 참고하였는데, 단순히 딕셔너리를 참고하여 대체하는 것보다는 문맥상 가장 적합한 문자로 대체하는 것이 타당하다.
- 모델의 설명가능성 : 은닉층을 사용하는 딥러닝 언어 모델의 경우 머신러닝 언어 모델보다 성능 평가 결과 원인을 설명하는 것이 어렵다.
- 성능 평가의 비일관성 : 언어 모델의 성능을 평가할 때마다 결과값이 다소 요동치는 양상을 보여 성능에 대한 평가를 일률적으로 단정지을 수 없다는 한계가 있었다. 태스크의 난이도가 지나치게 쉽거나 모델의 설계상 문제로 인한 것으로 예측된다.

이를 통해 보았을 때 다음과 같은 과제가 도출되었다.

- label 재설정 : label을 단순히 이진화하여 분석하기보다는 부정-중간-긍정이나 매우나쁨-나쁨-중간-좋음-매우좋음과 같이 label을 n진화하여 분석해볼 필요가 있다.
- 최적 대체어 탐색 알고리즘 구현 : 이모티콘을 다른 텍스트로 대체하는 과정에서 단순히 이모티콘 딕셔너리를 활용하는 것보다는 word2vec 등을 사용하여 문맥에 가장 적합한 단어로 대체한 후 분석하는 것이 타당할 것으로 보인다.
- 연속된 이모티콘에 대한 가중치 부여 : 이모티콘이 반복 사용되는 경우 반복된 이모티콘 묶음 자체를 토큰화함으로써 빈도에 가중치를 부여하는 것도 고려할 만하다.
- 데이터 재설정 : 특정 상품 내지 서비스에 대한 평가 데이터보다는 Amazon과 같이 좀더 광범위한 상품내지 서비스를 다루는 유통 및 물류 서비스에 대한 평가 데이터를 활용해보는 것도 시도해볼 만하다.

References

- [1] Chuchu Liu, Fan Fang, Xu Lin, Tie Cai, Xu Tan, Jianguo Liu, and Xin Lu. Improving sentiment analysis accuracy with emoji embedding. *Journal of Safety Science and Resilience*, 2(4):246–252, 2021.
- [2] Mohammed Shiha and Serkan Ayvaz. The effects of emoji in sentiment analysis. *Int. J. Comput. Electr. Eng.(IJCEE.)*, 9(1):360–369, 2017.

A Appendix: Team contributions

- 장영재 : LSTM, BERT-Base 및 ELECTRA 모델 설계
- 육현준 : 이모티콘 전처리, Naive Bayes 및 ALBERT 모델 설계
- 황수연 : 관련 논문 분석, 데이터 전처리, SVM 및 RoBERTa 모델 설계