

한국어에서 T5를 사용한 속성 기반 감성 분류 모델

류상연^o, 강상우
 가천대학교 AI · 소프트웨어학부
 teryas@gachon.ac.kr, swkang@gachon.ac.kr

Generative-model based Aspect-Based sentiment Analysis

Sangyeon YU^o, Sang-Woo Kang
 School of computing, Gachon University

요약

인터넷과 소셜미디어 사용량의 급증으로, 제품 리뷰, 온라인 피드백, 소셜 미디어 게시물 등을 통해 고객의 감정을 파악하는 것이 중요해졌다. 인공지능이 활용되어 고객이 제품이나 서비스의 어떤 부분에 만족하거나 불만을 가지는지를 분석하는 연구를 ABSA라고 하며 이미 해외에서는 이런 연구가 활발하게 이루어지는 반면, 국내에서는 상대적으로 부족한 상황이다. 이 연구에서는 ABSA의 두 개의 주요 작업인 ACD와 ASC에 대해 생성 모델 중 하나인 T5 모델을 사용하는 방법론을 제시한다. 이 방법론은 기존 판별 모델을 사용하는 것에 비해 시간과 성능 측면에서 크게 향상되었음을 보여준다.

주제어: 자연어 생성, 속성 기반 감성 분류, 감성 분석

1. 서론

감성분석(Sentiment Analysis)은 텍스트 데이터에서 긍정적이거나 부정적인 감성 또는 감정을 탐지하고 분류하는 과정을 말한다. 자연어 처리의 감성 분석은 다음과 같은 주요 카테고리로 분류할 수 있다. 텍스트가 긍정적인지 부정적인지 중립인지를 판단하는 극성(Polarity) 분석, 텍스트에서 여러 가지 감정을 탐지하고 강도를 측정하는 감성 스코어링(Emotion scoring), 텍스트에서 특정 속성이나 측면에 대한 감성을 분석하는 속성 기반 감성 분석(ABSA, Aspect-Based Sentiment Analysis)[1]이 있다.

인터넷 및 소셜미디어의 사용량이 증가함에 따라 제품 리뷰, 온라인 피드백, 소셜 미디어 게시물 등을 통해 고객의 감정 상태를 이해하는 것이 일반화되고 있다. 기존의 감성 분석은 텍스트 또는 문장이 전체적으로 긍정적인지, 부정적인지를 판별하는 것을 중심으로 삼았다. 그러나 마케팅 전문가들은 감성 분석을 활용하여, 단순히 긍정적이거나 부정적인 평가 이상의 정보, 즉 고객이 어떤 부분에서 제품이나 서비스에 만족하거나 불만을 가지는지에 대한 상세한 정보를 요구하고 있다. 이렇게 여러 특성을 추출하여 특성 별 감성 분석을 실시하는 것이 ABSA이다.

이러한 속성별 감성 분석이 최근에 주목받는 이유는 고객의 선호도를 특성별로 파악하고, 이를 통해 더욱 세밀하고 정확한 통찰력을 얻어 분석 결과의 유용성을 극대화할 수 있기 때문이다.

ABSA는 여러한 하위 태스크로 나뉘는데, 그중 본 논문에서 다룰 태스크는 Aspect Category Detection (ACD)와 Aspect

표 1. S가 입력 되었을때 각 작업 별로 출력되는 예시

S : 디자인도 예뻐서 인테리어소품 활용까지 제 몫 톡톡히 해내니 예뻐 죽겠네요.)		
Subtask	Input	Output
ACD	S	제품 전체#디자인
ASC	S	긍정

Sentiment Classification (ASC)이다.

ACD는 주어진 문장에서 어떤 속성이나 면이 언급되었는지를 찾는 작업이다. 예를 들어 표 1에 나온 “디자인도 예뻐서 인테리어소품 활용까지 제 몫 톡톡히 해내니 예뻐 죽겠네요:”)이라는 리뷰에서 ACD는 제품 전체라는 속성을 탐지하고 이러한 작업은 문장에서 어떤 주제가 중점적으로 다루어졌는지를 파악하는데 유용하다.

ASC는 반면에 문장에서 언급된 특정 범주에 대한 감성을 판별하는 작업이다. 문장 내의 범주에 대한 일반적인 감정이 긍정적인지, 부정적인지, 또는 중립적인지를 결정한다. 표 1의 예시에서 제품 전체 디자인에 대한 감성은 긍정적으로 판별될 수 있다.

해외에서는 ABSA와 관련하여 다양한 연구가 이루어지고 있고 (LSA, Local Sentiment Aggregation paradigm)[2], (LCF, Local Context Focus)[3], Instruction learning등 다양한 방법과 Laptop, Restaurant 데이터셋 외에도 다양한 데이터를 가지고 연구가 이루어지고 있으나, 그러나 국내에서는 BERT기반 속성 기반 감성분석을 이용한 영화 리뷰[4], Restaurant, Laptop에 대한 연구만이 이뤄지고 있는 실정이다. 이러한 방식은 각 범주와 극성을 분류하는 방식으로 이루어져 있어 분석을 해야 하는 범주의 양이 많아질수록 훈련을 하는데 많은 시간이

결린다는 단점이 있다.

본 논문은 이러한 단점을 해결하기 위하여 ABSA의 하위 작업 중 ACD와 ASC를 생성 모델 중 T5[5]를 기반으로 기존 쇼핑몰 리뷰 데이터셋의 감성 분류를 제안한다.

2. 관련 연구

2.1 언어 모델

다양한 언어 모델들 중 대표적인 생성 모델로는 Google이 개발한 자연어 처리(NLP)모델인 T5(Text-to-Text Transfer Transformer)는 인코더-디코더 구조의 Transformer[6] 모델을 기반으로 한다. 기존의 Transformer 모델들과는 달리 T5는 모든 자연어 처리 작업을 생성 문제로 재구성한다. 이러한 접근 방식을 통해, T5는 자연어 이해(NLU)와 자연어 생성(NLG) 작업 모두를 동일한 프레임 워크 내에서 처리할 수 있다.

T5는 사전 학습 단계에서 비지도 학습 방법을 사용한다. 이때 입력 텍스트에서 일부 단어나 문구를 마스킹하고, 이 마스킹된 부분을 예측하는 방식으로 학습이 이루어진다. 이는 BERT에서 사용한 방법과 유사하나, 텍스트-투-텍스트 형식을 사용하기 때문에, 마스킹된 부분을 별도로 표시하지 않고, 모델이 자연스럽게 마스킹된 부분을 예측하도록 한다.

2.2 ABSA

ABSA는 2010년대 초에 등장하였고, 텍스트와 감성 분석 분야에서 중요한 연구 주제로 인식되었다. SemEval(Semantic Evaluation) 2014[7]가 이 분야의 주요 데이터셋으로, 랩톱과 레스토랑 리뷰에서 중요한 속성을 식별하고, 해당 속성에 연관된 감정 극성을 부여하는 작업을 포함하고 있다. 전통적인 감성 분석은 문서나 텍스트 조각 전체의 감성을 판단하는 반면, ABSA는 텍스트 내 특정 속성이나 요소에 초점을 맞춘 감성 분석을 수행한다. 예를 들어, '드라이버 업데이트는 잘 됐지만, BIOS 업데이트가 시스템을 얼리고 컴퓨터가 꺼졌다.'라는 문장에서 '드라이버'에 대한 감성은 긍정적이고, 'BIOS'에 대한 감성은 부정적임을 알 수 있다. 하지만 전통적인 감성 분석에서는 '잘 됐지만'라는 긍정적인 단어 때문에 전체 문장이 긍정적인 감성으로 판단될 수 있다. 따라서 ABSA의 장점은 문장 내에서 언급된 특정 속성에 대한 감성을 더 세밀하게 파악할 수 있다는 점이다. 이는 사용자의 특정 특징이나 요소에 대한 반응을 보다 정확하게 이해하는 데 도움이 된다

2.3 인코더 기반 감성 분류

전통적인 속성 기반 감성 분류 모델은 주로 BERT[8], Roberta[9], Electra[10] 인코더만을 사용하면서 인코더-디코더 구조를 활용하지 않았다. 그러나, 속성 간의 감성이 상호 연관성을 가질 수 있다는 연구 결과에 따라, 이러한 감성들이 주변

속성과 유사성을 보이는 '속성 감성 일관성'이라는 새로운 개념이 제안되었다. 이를 바탕으로, 연구자들은 속성 감성 일관성의 모델링에 초점을 맞추어 이전 연구에서 놓쳤던 간극을 해결하는 LSA를 도입하였다. 또한, LCF 메커니즘을 기반으로, 속성 용어 추출과 그 극성을 동시에 추론하는 새로운 접근법이 제시되었다. 이러한 접근법은 기존 연구에서 주로 속성 용어의 극성에만 초점을 맞추던 방식과는 달리, 속성 용어의 추출에 중점을 두는 새로운 시각을 제공하였다.

2.4 Instruction learning 기반 감성 분류

InstructABSA[11]는 전통적인 인코더 기반 감성 분류 방법과 대비하여 Instruction learning을 활용하는 방법을 제안했다. 이는 Tk-Instruction 모델[12]을 기반으로 하였는데, 학습 과정에서는 SUPER-NATURALINSTRUCTIONS라는 자연어 처리(NLP) 태스크 분류 벤치마크를 활용하여 T5 모델에 Instruction learning 기법을 적용한 것이다. 이로 인해 모델이 다양한 컨텍스트에서 지시문을 이해하고 따르는 능력을 향상되었다. 이를 바탕으로 InstructABSA는 ABSA에 적합한 Instruction Prompt를 작성하여 Tk-Instruction 모델에 적용하는 새로운 접근 방식을 제안하였다.

3. 제안 모델

ACD, ASC 둘 다 기존 연구에서는 인코더만 사용하는 BERT 기반 모델을 사용하여 ACD의 경우 속성 쌍이 문장 내에 존재하는지 비교하는 분류하는 작업으로 진행하였고, ASC의 경우 문장 내의 감성이 긍정인지 부정인지를 판별하는 작업으로, 두 작업 모두 분류의 한 종류로 진행되었다.

그러나 속성쌍의 개수가 증가할 때마다 훈련하는 시간이 증가한다는 단점을 해결하기 위해 본 논문은 하나의 문장에 대한 속성쌍을 디코더에서 생성 해내는 방식을 제안하였고 이를 통해 생성해낸 속성을 인코더에 입력으로 넣어 극성을 디코더에서 생성하는 방식을 제안한다.

3.1 ACD

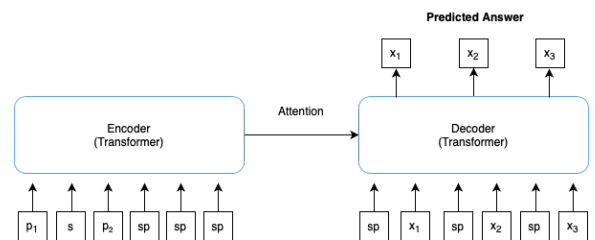


그림 1. ACD의 모델 구조

T5 모델을 학습하기 위해서는 프롬프트를 이용한다. 인코더에는 입력 문장 s 를 특정 프롬프트에 추가하며, 이에 해당하는

정답을 생성할 수 있도록 특수 토큰을 사용한다. 디코더는 이 특수 토큰을 활용하여 정보를 전달받고, 이를 기반으로 정답을 생성하도록 학습된다. 본 연구에서는 최대 3개의 특수 토큰을 사용하여 학습을 진행하였다.

예를 들어, “어제 샀는데 오늘 안 되더군.”이라는 문장을 입력으로 가질 때 다음과 같은 과정을 거친다.

첫 번째 프롬프트는 “문장에서 속성을 찾으시오:”의 형태를 가지며, 그다음에 입력 문장을 추가하여, 모델이 이를 바탕으로 속성을 추출하도록 유도한다.

다음으로, 속성을 추출하기 위한 두 번째 프롬프트인 ‘이 문장의 속성은’을 사용한다. 그리고, (<extra_id_N>)을 추가하여, 모델이 해당 토큰 위치에 속성을 채워 넣도록 학습한다. 최종 입력 문장은 “문장에서 속성을 찾으시오: 어제 샀는데 오늘 안 되더군. 이 문장의 속성은 <extra_id.1> <extra_id.2> <extra_id.3>”의 형태를 가진다.

다음 단계에서는, 디코더에 정확한 답안을 제공한다. 각각의 답안은 <extra_id_N> 토큰에 연결되며, 이는 “<extra_id.1> 본품 품질<extra_id.2> <extra_id.3>”과 같은 형태를 가진다.

마지막으로, 디코더는 최종 logits을 출력하고, 이를 입력된 레이블과 Cross-Entropy loss function을 이용하여 최종 loss를 계산한다.

$$L(y, p) = - \sum (y_i * \log(p_i)) \quad (1)$$

이러한 방식은 각각의 요소를 분리하고 디코더가 문장의 특정 속성을 인식하는 데 도움이 되며 이 접근법을 통해, 디코더는 문장의 의미와 구조를 이해하고, 해당 문장의 속성을 정확하게 추출하는 데 필요한 지침을 제공한다.

3.2 ASC

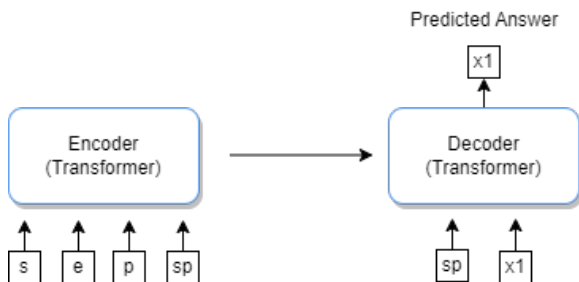


그림 2. ASC의 모델 구조

T5 모델을 ASC에 적용하는 방식은 ACD 모델의 방식과 비슷하게, 인코더와 디코더를 함께 사용하여 디코더에서 답변을 생성하는 방식을 제안하나, 프롬프트의 형식이 다르다.

예를 들어, “어제 샀는데 오늘 안 되더군.”이라는 문장에 대한 속성이 “본품 품질”일 경우, 다음과 같이 입력을 준비한다.

먼저, 주어진 문장(s)인 “어제 샀는데 오늘 안 되더군”과 속성(e)인 “본품 품질”을 연결해 “어제 샀는데 오늘 안 되더군의 본품 품질”의 형태로 만든다.

그다음, 극성(p)을 표현하는 프롬프트인 “극성은 <extra_id.1> 이다”를 앞에 문자에 연결해 “어제 샀는데 오늘 안 되더군의 본품 품질의 극성은 <extra_id.1> 이다”의 형태로 만들어 인코더에 입력한다.

이어서 디코더 단계에서는 정답을 입력하며, 예를 들면 “<extra_id.1> 부정”과 같은 형태로 넣어준다.

$$L(y, p) = - \sum (y_i * \log(p_i)) \quad (2)$$

이러한 고유의 프롬프트 형식을 통해 문장의 속성에 대한 극성을 정밀하게 분석하여 디코더는 최종 logits을 출력하며, 이를 입력된 레이블과 Cross-Entropy loss function을 이용해 최종 loss를 계산한다.

4. 실험 및 결과

4.1 데이터셋

본 논문에서 사용한 데이터 셋은 “국립국어원 속성 기반 감성 분석 말뭉치 2021”중 제품 도메인 데이터 셋을 사용하여 학습하였다. 제품 도메인의 데이터와 그에 해당하는 개체 속성쌍은 아래 표 2. 표 3. 과 같다.

표 2. 제품 도메인 종류 및 데이터 개수

	제품		
	화장품 세정제	전자 기기	제품 기타
합계	842	191	640

표 3. 제품 도메인의 개체 속성 쌍

	일반	가격	디자인	품질	편의성	다양성	인지도
제품 전체	0	0	0	0	0		0
본품	0		0	0	0	0	
패키지.구성품	0		0	0	0	0	
브랜드	0	0	0	0			0

4.2 사전 학습 모델

베이스 라인으로는 pko-t5[13]를 사용하였으며 pko-t5는 sentencepiece 대신 OOV(Out-Of-Vocabulary)가 없는 BPE(Byte Pair Encoding)를 사용했으며 한국어 데이터를 T5의 span corruption task를 사용해서 비지도 학습만 적용하여 학습을 진행하였다.

4.3 실험 환경

ACD 데이터에 대한 미세조정은 학습률 3e-6, 배치 사이즈 8으로 20 에폭 진행하였다. ASC 데이터에 대한 미세조정은 학습률 5e-6, 배치 사이즈 8으로 20 에폭 진행하였다. GPU 자원은 Titan RTX 2기를 사용하였다.

4.4 실험 결과

표 4.와 표 5.는 모델 평가 결과이다. ACD, ASC 모두 실험에서 제안 모델은 비교 모델 보다 높은 성능을 보였다.

표 4. Aspect Category Detection 모델 성능 평가

Model	hidden layer	Accuracy	F ₁ -score	Elapsed Time(s)
Klue-Bert	768	0.6399	0.6940	1103.365
XLM-Roberta	768	0.5972	0.6346	1324.277
Electra	768	0.7023	0.6886	1139.297
Ours(T5-base)	768	0.7203	0.7242	111.864
Ours(T5-large)	1024	0.7354	0.7380	320.432

ACD에서 모델이 잘 학습했는지 확인하기 위해 평가 지표를 정확도, F₁-score, 한 에폭 당 걸린 평균 시간을 비교하였다.

표 4. 에서 제안 모델은 비교 모델에 비해 정확도에서 최대 13%의 성능 향상을, F₁-score에서는 최대 10%의 성능 향상을 보였다. 또한 제안 모델은 학습시간에서 T5-base 모델은 약 10배, T5-large 모델은 약 3배의 시간이 단축되었다.

비교 모델은 도메인의 모든 개체 속성쌍에 대하여 각각 비교를 한 것에 반면에 제안 모델은 해당하는 도메인의 개체 속성쌍을 생성하는 방식으로 진행하였고, 이 때문에 한 에폭 당 걸리는 시간이 최대 약 10배 정도 차이 나는 유의미한 결과가 도출되었다.

또한 정확도와 F₁-score에서 성능 향상을 통해 제안한 방식의 개체 속성쌍을 생성하는 능력이 비교 모델인 인코더 기반 모델의 속성쌍을 분류하는 능력보다 좋음을 확인할 수 있었다.

표 5. Aspect Sentiment Classification 모델 성능 평가

Model	hidden layer	Accuracy	F ₁ -score (macro)	Elapsed Time
Klue-Bert	768	0.9777	0.6174	56.137
XLM-Roberta	768	0.9740	0.5394	67.102
Electra	768	0.9750	0.6027	57.425
Ours(T5-base)	768	0.9743	0.6132	70.672
Ours(T5-large)	1024	0.9786	0.6193	160.243

표 5. 에서 제안 모델은 비교 모델과 비교했을 때, T5-large 모델을 활용한 제안 모델은 정확도에서 최대 0.01%의 미세한 성능 향상과 F₁-score에서 0.002의 소폭의 성능 향상을 보였다. 그러나 T5-base 모델에서는 비교 모델을 뛰어넘는 성능 향상이

나타나지 않았다. 더욱이, 한 에폭 당 학습 시간은 비교 모델에 비해 약 2배 길었다.

ASC의 경우 ACD의 모든 제품 도메인과 비교하는 것과 다르게 정답의 극성이 부정, 긍정, 중립으로 이루어져 있기 때문에 일반 인코더 모델에 비하여 파라미터 수가 많은 T5 모델이 한 에폭 당 걸리는 시간이 조금 더 걸렸음을 확인하였고 성능 자체도 비교 모델 보다 낮음을 확인할 수 있었다.

5. 결론

본 논문에서는 기존의 인코더 기반 방식이 아닌, 인코더-디코더를 활용한 생성 모델 기반의 방식을 ABSA에 적용해 보았다. 이 방식을 통해 ABSA의 하위 태스크인 ACD에서는 성능 평가 결과, 정확도와 F₁-score에서 성능 향상을 확인할 수 있었다. 또한, 한 에폭 당 학습 시간의 단축을 확인할 수 있었다. 그러나, ASC에서의 성능은 비교 모델에 비해 향상되지 않았다.

이러한 문제를 해결하기 위해, 앞으로는 다양한 모델 구조를 탐색해 보는 방향으로 연구를 진행할 계획이다. Instruction learning에 기반을 두고, 한국어 고유의 Task-instruction을 이용하여 ACD와 ASC 태스크에 적용해 볼 계획이다. 그리고 텍스트 기반의 속성 감성 분석에 국한되지 않고, 이미지와 같이 활용하는 멀티 모달에도 적용할 예정이다.

감사의 글

이 성과는 2023년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No. NRF-2022R1A2C1005316)

참고문헌

- [1] 이희재 and 임희석, “속성 기반 감성 분석을 활용한 bert 감성 분류 모델의 성능 개선,” *한국정보과학회 학술발표 논문집*, pp. 332–334, 2022.
- [2] H. Yang and K. Li, “Improving implicit sentiment learning via local sentiment aggregation,” *arXiv preprint arXiv:2110.08604*, 2021.
- [3] H. Yang, B. Zeng, J. Yang, Y. Song, and R. Xu, “A multi-task learning model for chinese-oriented aspect polarity classification and aspect term extraction,” *Neurocomputing*, Vol. 419, pp. 344–356, 2021.
- [4] 이유린 and 안현철, “Bert 기반의 속성기반 감성분석을 활용한 다기준 영화 추천 모형,” *한국컴퓨터정보학회논문지*, Vol. 27, No. 3, pp. 201–207, 2022.
- [5] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, “Exploring the limits of transfer learning with a unified text-

- to-text transformer,” *The Journal of Machine Learning Research*, Vol. 21, No. 1, pp. 5485–5551, 2020.
- [6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” *Advances in neural information processing systems*, Vol. 30, 2017.
- [7] D. Kirange, R. R. Deshmukh, and M. Kirange, “Aspect based sentiment analysis semeval-2014 task 4,” *Asian Journal of Computer Science and Information Technology (AJCSIT) Vol.*, Vol. 4, 2014.
- [8] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” *arXiv preprint arXiv:1810.04805*, 2018.
- [9] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, “Roberta: A robustly optimized bert pretraining approach,” *arXiv preprint arXiv:1907.11692*, 2019.
- [10] K. Clark, M.-T. Luong, Q. V. Le, and C. D. Manning, “Electra: Pre-training text encoders as discriminators rather than generators,” *arXiv preprint arXiv:2003.10555*, 2020.
- [11] K. Scaria, H. Gupta, S. A. Sawant, S. Mishra, and C. Baral, “Instructabsa: Instruction learning for aspect based sentiment analysis,” *arXiv preprint arXiv:2302.08624*, 2023.
- [12] Y. Wang, S. Mishra, P. Alipoormolabashi, Y. Kordi, A. Mirzaei, A. Arunkumar, A. Ashok, A. S. Dhanasekaran, A. Naik, D. Stap *et al.*, “Supernaturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks,” *arXiv preprint arXiv:2204.07705*, 2022.
- [13] D. Park, “pko-t5: Paust korean t5 for text-to-text unified framework,” 5 2022. [Online]. Available: <https://github.com/paust-team/pko-t5>