한국어 스타일 변환 기반 데이터 증강을 이용한 감성 분류 성능 향상

고은우○, 이은찬, 안상태*
경북대학교 전자공학부, 경북대학교 전자전기공학부
*stahn@knu.ac.kr

Improving Performance of Sentiment Classification using Korean Style Transfer based Data Augmentation

Eunwoo Go○, Eunchan Lee, Sangtae Ahn*
School of Electronics Engineering, Kyungpook National University
School of Electronic and Electrical Engineering, Kyungpook National University

요 약

텍스트 분류는 입력받은 텍스트가 어느 종류의 범주에 속하는지 구분하는 것이다. 분류 모델에 있어서 좋은 성능을 나타내기 위해서는 충분한 양의 데이터 셋이 필요함을 많은 연구에서 보이고 있다. 이에 따라 데이터 증강기법을 소개하는 많은 연구가 진행되었지만, 실제로 사용하기 위한 모델에 곧바로 적용하기에는 여러 가지 문제점들이 존재한다. 본 논문에서는 데이터 증강을 위해 스타일 변환 기법을 이용하였고, 그 결과 기존 방법 대비 한국어 감성 분류의 성능을 높였다.

주제어: 스타일 변환, 데이터 증강, 감성 분류

1. 서론

자연어 처리 문제 중 대표적이면서 많이 접하는 태스크인 텍스트 분류 태스크는 특정 텍스트를 여러 범주 중 어느 범주에 속하는지 분류하는 태스크이다. 텍스트 분류(Text Classification) 종류에는 감정분류, 스팸분류 등 많은 종류의 분류가 있다. 이러한 분류 모델에 있어서 좋은 성능을 나타내고, 오버피팅(overfitting)을 해결하기 위한 방안으로 충분한 양의 데이터 셋이 필요하다[1][2]. 하지만 현실에서는 충분한 다량의 데이터를 구하지 못하는 상황이 많다. 따라서 데이터의 양을 증대시키는 연구가 머신러닝, 딥러닝 영역 전반에 걸쳐서 이루어져 왔다. 컴퓨터 비젼 분야에서는 [3]의 연구가, 자연어처리 분야에서는 [4]의 연구가 그 예이다. [4]에서는 텍스트 분류 태스크에 있어서 성능을 증대시키기 위한 데이터 증강기법 4가지를 소개하였다. 그러나, 이전의 연구에서 소개된 증강기법을 현실의 데이터 셋에 곧바로 적용하기에는 많은 어려움이 있다. 이에 따라, 본 논문에서는 한국어 텍스트를 다양한 말투로 변환시키는 한국어 텍스트 스타일 변환기법을 이용하여 텍스트 데이터 증강(text data augmentation)을 수행하여 텍스트 분류의 성능을 높이고자 하였다. 텍스트 스타일 변환을 진행하는 과정에서 KoBART[5] 모델을 이용하여 입력 문장을 여러가지 다른 문체의 문장으로 변환하는 기능을 구현하였다. 그리고 이를 검증하기 위하여 한국어 텍스트 분류에 있어서 가장 잘 알려진 데이터 셋 중 하나인 네이버영화 감성리뷰 데이터 셋에 변환된 스타일의 문장들을 추가시켜 데이터 셋을 증강시킨 후, LSTM을 활용한 분류 문제에 적용시켜 성능 변화를 직접 실험해보고 결과를 제시하였다.

2. 관련 연구

2.1 텍스트 데이터 증강

데이터를 증강하는 기법에 대한 연구도 이루어져 있는데, 대표적으로 컴퓨터 비전 분야에서는 뒤집기(flipping), 자르기(cropping), 크기 조정(scaling), 회전(rotating) 등과 같은 기하학적 변환을 이용해 데이터 셋을 증강해 오버피팅(overfitting)을 줄이고, 성능을

높였다[6]. 자연어 처리 분야에서도 데이터 셋 증강을 위한 연구가 진행되었는데, [7]에서는 데이터 증강(data augmentation)을 공식화하기 위한 시도로 다음과 같은 규칙들을 제시하였다.

1. 만족도에 대한 규칙 : 의미론적 단계에서 data의 의미에 변화를 주지 않고 패턴인식의 관점에서 "새로운 형태"를 학습할 수 있도록 해야 한다.

2. 의미 불변의 규칙 : 데이터 증강(data augmentation)은 반드시 의미 불변 변화(transformation)를 사용해야 한다.

이 규칙들을 위한 기술로 "문맥적 잡음" 삽입(textual noise injection)과 잘못된 철자 삽입 등이 있다. 이것들은 단어의 일부 철자 제거, 잘못된 철자 삽입, 대·소문자의 변화 등과 같은 것으로 잡음 삽입(noise injection)은 위의 규칙을 따르면서 text의 형태에 변화를 주는 것이다[7]. 또한, [4]에서는 텍스트 분류 태스크에서의 성능 향상을 위해 데이터 증강기법 4가지를 제시한다. 그것은 다음과 같다.

SR: Synonym Replacement, 특정 단어를 유의어로 교체
RI: Random Insertion, 임의의 단어를 삽입
RS: Random Swap, 문장 내 임의의 두 단어의 위치를 바꿈
RD: Random Deletion: 임의의 단어를 삭제

이 4가지는 합성곱신경망과 순환신경망에서 성능을 향상시키는데, 특히 데이터가 더 적은 경우에 강력한 성능 향상을 보여준다. 또한, [4]에서 한 문장 당 얼마나 증강하는 게 성능 향상에 좋은지에 대해 실험을 하였는데, 다음은 한 문장당 증강한 문장 수(aug)에 따른 성능 향상(Performance Gain)을 그래프로 나타낸 것이다. 여기서 N은 기존 데이터 개수이고, y축 절편은 성능 변화(단위: %)이다.

그림 1에 따르면, 기존 데이터 셋의 크기가 작을수록 데이터 증강을 했을 때 더 큰 성능 향상 효과를 볼 수 있고, 한 문장당 증강한 문장 수(aug)가 일정 이상보다 커지면 성능 향상의 크기가 오히려 작아진다는 것을 알 수 있다.

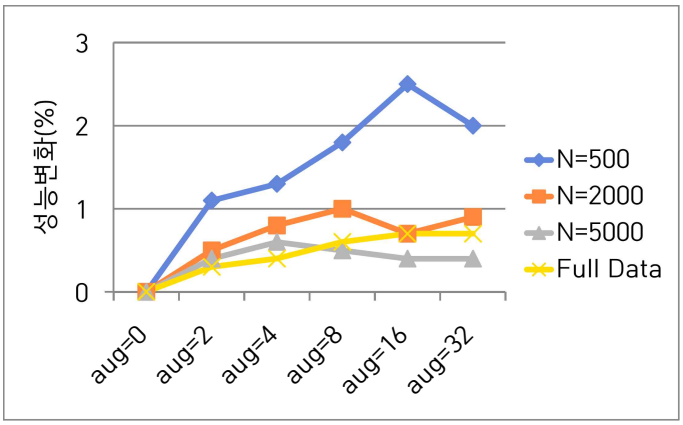

그림 1. 데이터 증강에 따른 성능 변화

2.2 텍스트 스타일 변환

텍스트 스타일 변환(text style transfer)은 입력문장에 대해 문장의 의미는 보존하고, 스타일(style)만 바꾸는 것이다. 최근에는 딥러닝(deep learning)을 이용한 텍스트 스타일 변환(text style transfer)연구가 이루어지고 있는데, 트랜스포머 기반의 딥러닝 모델을 이용해 높임말을 낮춤말로 바꾸는 연구[8], 해요체를 합쇼체로 스타일을 변화시킨 연구[9]와 같이 어미와 높임말의 체계가 존재하는 한국어의 특성을 이용한 텍스트 스타일 변환(text style transfer) 연구도 이루어지고 있다.

3. 텍스트 스타일 변환 설계 및 구현

본 연구에서는 [10]의 데이터 셋을 언어 모델에 학습한 후, 학습된 모델을 통해 임의 추출된 네이버 영화리뷰 데이터들을 여러 문체의 스타일로 변환시켰다.

3.1 데이터 셋

본 논문에서는 텍스트 스타일 변환을 하기 위해 [11]의 한국어 대화 스타일 변환 데이터 셋을 사용하였다. 이 데이터 셋은 한 문장에 대해 17개의 서로 다른 문체의 문장으로 스타일 변환(style transfer)된 대화 데이터를 약 3,700쌍(pair)으로 구축된 데이터 셋이다. 아래는 1개의 문장에 대해 17개의 서로 다른 문체로 이루어진 데이터 2쌍의 예시이다.

표 1. 한국어 텍스트 스타일 변환 데이터 셋

formal	informal	android	azae	chat	choding	emoticon	enfp	gentle
고양이를 6마리나요? 키우는거 안 힘드세요?	고양이를 6마리나? 키우는거 안 힘들어?	고양이. 6마리. 양육. 번거로운가.	아니 무슨 고양이를 6마리나? 거 키우는 거 안 힘든가?	엥? 6마리나? 안힘듬?ㅋㅋㅋㅋ	6마리? 에바아니냐 안 힘듦?	고양이를 6마리나?!! w(ﾟДﾟ)w 키우는거 안 힘듬?? (⊙_⊙;)	고양이를 6마리나? 완전 대박~ 키우는 거 안 힘들어?!	고양이를 6마리나 키우십니까? 안 힘드신지,,
뭐하고 계세요? 뭐하고 있어?	뭐하고 있어?	휴먼. 지금. 뭐하는가.	뭐하고 있는감?	뭐해?	뭐함?	뭐하고 있엉?? (⊙_⊙)?	안뇽안뇽~ 지금 뭐해뭐해~?	뭐하고 계십니까?

halbae	halmae	joongding	king	naruto	seonbi	sosim	translator
고양이를 6마리나? 키우는거 힘들지 않는가?	니기럴 털만 날리는 거 키우기 안 힘들데?	아니 고양이를 6마리나? 안힘드냐?	고양이를 6마리나? 키우는게 수고스럽진 않소?	고양이를 6마리나? 키우는거 힘들지 않냐니깐?	고양이를 6마리나 키우고 있는 것이오? 힘들지 않소?	고양이..6마리나? ㅠ 키우는건 혹시 안힘들어..?	6마리의 고양이? 당신은 그들로부터 지치지 않습니까?
뭐하는 거요?...	뭘 또 하고 자빠졌나	뭐하는데	무엇하는가?	뭐하고 있냐니깐! ㅋㅋ	뭐하고 있소?	혹시… 뭐해..?	무엇 당신은 한다 지금?

표 2. KoBART 모델의 학습환경

Epochs	batch	Optimizer	learning rate	Validation Split Rate	Max_Sequence_Length	Loss Function
20	8	Adamw	5e-05	10%	256	cross entropy

[표 1]과 같이, 어미의 변화, 노이즈 주입(noise injection)을 통하여 의미는 변질하지 않고, 형태만 변화되도록 스타일이 변환되어 있음을 확인할 수 있다.

3.2 모델

최근 Transformer 기반의 사전 학습 모델을 사용한 텍스트 생성 모델이 많은 자연어 처리 태스크에서 높은 성능을 보이고 있다. BART는 Sequnece-to-Sequence 트랜스포머(Transformer)기반의 모델로, 노이즈(noise)로 손상된 텍스트를 복구하는 오토인코더(autoencoder) 형태로 학습이 이루어진다[11]. 그 중, KoBART는 [11]에서 사용된 노이즈 함수를 사용하여 40GB이상의 한국어 텍스트에 대해 학습한 한국어 인코더-디코더 모델이다[5]. 본 논문에서는, 한국어 텍스트 생성을 위해 스타일 변환 태스크에 있어 KoBART[5] 모델을 사용하였다.

3.3 학습 및 텍스트 스타일 변환

[12]의 KoBART 모델을 이용해서 [11]의 스타일 변환 데이터셋을 학습시켰다. 학습 환경은 표 2와 같다.

본 논문에서는 텍스트 스타일 변환을 이용한 데이터 증강 실험을 네이버 영화리뷰 감성분류 태스크에 적용했다. 먼저, [13]의 데이터 셋의 약 20만 개의 데이터 중 학습하기에 충분하지 않은 정도의 적은 양의 데이터를 랜덤 추출 후, 각각의 데이터들에 대해 학습된 KoBART모델을 이용해 9가지의 서로 다른 문체로 변환시켰다. 추출된 데이터 셋의 크기는 각각 150, 375, 750, 1500개가 실험에 사용되었다. 다음은 스타일 변환된 [14]의 데이터 중 하나를 나타낸 것이다.

표 3. 입력 문장에 따른 스타일 변환 결과

style	text
입력 문장	런닝타임 20~30분 정도 더 늘렸으면 하는 아쉬움이 있지만, 짧고 강렬하네요 애니...
formal	네,,,,,,,,,,,,,, 런닝 시간을 조금 더 늘렸으면 하는 아쉬움도 있지만, 짧고 강렬함이 느껴집니다.
android	시간. 20분. 증가. 아쉬움. 그러나. 짧음. 강렬함.
chat	ㅇㅇ ㅇㅈㅇ...런닝 시간 좀 더 길게 했으면 좋았을텐데 아쉽네...
emoticon	ㅇㅇ,,,,,,,,,,, (´`) 좀 더 길게 했으면 좋겠지만 짧고 강렬하네...(⊙_⊙;)
gentle	네,,, 런닝 시간을 조금 더 늘렸으면 좋겠습니다만, 짧고 강렬하게 들리는 것 같습니다.
halmae	엥닝 시간을 좀 더 길게 했으면 하는 염병할 아쉬움도 있구만 짧고 강렬하구만
king	그렇소. 한 시간 정도 더 정진했으면 하는 아쉬움도 있으나, 짧고 강렬한 인상을 남겨주오.
seonbi	소생은 런닝 시간을 20분 정도 더 늘렸으면 하는 아쉬움도 있소만, 짧고 강렬하다고 생각되는 것이오!
translator	따라서, 런닝 시간을 20분 정도 더 늘렸으면 하는 아쉬움도 있지만, 그것은 짧고 강렬한.

앞서 2.1에서 언급한 문맥적 잡음 삽입(context noise injection)을 통해 의미가 변질되지 않고 형태만 바뀌는 것을 위의 표 3에서 확인할 수 있었다. 또한 [4]에서 소개한 대표적인 증강 기법 중 RI, RS, RD 기법이 스타일 변환을 통해 적용되어 있음을 확인할 수 있었다. 그러나, 입력 문장에 따라 다음의 [표 4]와 같이 의미가 변질되는 스타일 변환 결과도 있었다.

표 4. 의미가 변질된 스타일 변환 결과

style	text
입력 문장	속편이 이정도면,,, 그리 악평이 줄을 이을껀 아닌듯한데,,
formal	네, 속편이 이렇게 좋은데, 그렇게 악평을 받을 일은 아닌 것 같아요.
android	속편. 이정도. 악평. 없음. 추측.
chat	° ° 속편이 이렇게 나오면 그렇게 악평할 이유가 없는 거 같음
emoticon	이 정도이야...(⊙_⊙;) 그렇게 악평할 일은 아닌 것 같아 (⊙o⊙)
gentle	제 생각편이 이렇게 좋은데,,,, 그렇게 악평을 들을 이유가 없네요.
halmae	이 새끼 놈아 이거 지랄이여
king	그렇소. 속편이 그리 좋지 않소. 그리 악평을 들을 일은 없는 것 같소.
seonbi	그렇소! 속편이 이렇게 좋지 않소! 그렇게 악평을 할 이유가 없는 것이오!
translator	그것은 사실, 그것은 그다지 나쁜 평가는 아닙니다.

표 4의 king, seonbi 문체는 입력 문장에 대해 의미가 변질됨을 볼 수 있고, halmae체는 의미가 완전히 바뀌어 버림을 확인할 수 있다.

5. 실험 및 실험 결과

본 장에서는 스타일 변환된 데이터들을 기존의 임의추출된 영화리뷰 데이터들에 추가해 증강하고, 증강 이전의 데이터들과 증강 이후의 데이터들을 네이버 영화리뷰 감성분류 태스크에 적용해 성능 변화를 실험하였다.

데이터 증강 및 분류 태스크 적용

먼저, 증강되지 않은 기존의 임의 추출된 데이터 셋을 순환신경망 모델에 학습시켜 태스크에 적용하였다. 모델은 다 대 일 구조의 LSTM을 사용하였으며, 마지막 시점에서 두 개의 선택지 중 한 가지를 예측하는 이진 분류 문제(binary classification task)를 수행하도록 하였다. 활성화 함수로는 시그모이드 함수를 사용하고, 손실함수로 크로스 엔트로피 함수를 사용하였다. 배치 크기(batch size)는 64이며, 15 에포크(epoch)를 수행하였다. 이후, 기존의 임의 추출된 데이터 셋에 변환된 9가지 서로 다른 문체의 리뷰 문장 데이터 셋을 더해 10배로 데이터의 양을 증강하고 동일한 환경에서 태스크에 적용하였다. 다음은 150, 375, 750, 1500개의 기존 데이터 셋을 10배의 크기로 증강하기 전의 태스크에서의 성능과 증강 후의 태스크에서의 성능을 표로 나타낸 것이다.

표 5. 증강 전과 후의 텍스트 분류 성능 비교

증강 전 데이터셋 크기	증강 전 분류 성능	증강 후 데이터셋 크기	증강 후 분류 성능
150	0.5712	1500	0.6723
375	0.6778	3750	0.6884
750	0.7081	7500	0.7228
1500	0.7325	15000	0.7255

이와 같이 기존 데이터셋의 양(N)이 적을수록 증강시킨 데이터셋의 성능 향상이 크게 나타났다. 이것은 앞서 2.1에서 언급한 그림 1의 결과와 유사하다. 하지만, 기존 데이터셋 크기가 1500개일 때는 오히려 성능이 줄어듦을 확인할 수 있는데, 스타일 변환 시 [표 4]와 같이 의미가 변질되는 문장이 많이 생겨서 분류 태스크에서 저하된 성능을 보인 것으로 생각된다.

6. 결론

본 논문에서는 텍스트 스타일 변환을 통해 한국어 데이터를 증강하는 기법을 제안했다. 텍스트 스타일 변환은 텍스트 의미의 변질을 최소화한다. 따라서, 의미의 변화 없이 형태와 스타일만 바꿔 데이터 셋 증강을 하고자 하는 이들은 본 논문에서 제안하는 방법을 통해 그 목적을 달성할 수 있다. 기존 데이터의 개수가 적을수록 이 기법은 성능을 극대화하며, 실험을 통해 그 결과를 확인하였다. 따라서 본 논문에서는 데이터를 증강하는 기법 중 하나로 스타일 변환 기법을 제안하며 또한, 앞으로 여러 태스크에서 데이터 증강 시 성능 변화를 극대화 할 수 있는 텍스트 스타일 변환 연구를 진행할 예정이다.

Acknowledgement

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2022R1A4A1023248).

참고문헌

[1] Anaby-Tavor, Ateret, et al. "Do not have enough data? Deep learning to the rescue!." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 34. No. 05. 2020.

[2] Ying, Xue. "An overview of overfitting and its solutions." Journal of physics: Conference series. Vol. 1168. No. 2. IOP Publishing, 2019.

[3] Perez, Luis, and Jason Wang. "The effectiveness of data augmentation in image classification using deep learning." arXiv preprint arXiv:1712.04621 (2017).

[4] Wei, Jason, and Kai Zou. "Eda: Easy data augmentation techniques for boosting performance on text classification tasks." arXiv preprint arXiv:1901.11196 (2019).

[5] https://github.com/SKT-AI/KoBART

[6] Taylor, Luke, and Geoff Nitschke. "Improving deep learning with generic data augmentation." 2018 IEEE Symposium Series on Computational Intelligence (SSCI). IEEE, 2018.

[7] Coulombe, Claude. "Text data augmentation made simple by leveraging nlp cloud apis." arXiv preprint arXiv:1812.04718 (2018).

[8] Hong, Taesuk, et al. "Korean Text Style Transfer Using Attention-based Sequence-to-Sequence Model." Annual Conference on Human and Language Technology. Human and Language Technology, 2018.

[9] 박다솔 et al. “TGST : 트랜스포머 기반의 한국어 생성적 스타일 변환”, 한국정보과학회 2020 한국컴퓨터종합학술대회 (KCC 2020), VOL 47 NO. 01 PP. 0353 ~ 0355 2020. 07

[10]https://github.com/smilegate-ai/korean_smile_style_dataset

[11] Lewis, Mike, et al. "Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension." arXiv preprint arXiv:1910.13461 (2019).

[12] https://huggingface.co/gogamza/kobart-base-v2

[13] https://github.com/e9t/nsmc/