

효율적인 트랜스포머를 이용한 팩트체크 자동화 모델

Automated Fact Checking Model Using Efficient Transformer

Hee Seung Yun¹ · Jason J. Jung^{2*}

¹Graduate Student, Department of Computer Engineering, Chung-Ang University, Seoul, 06974 Korea

²Professor, Department of Computer Engineering, Chung-Ang University, Seoul, 06974 Korea

ABSTRACT

Nowadays, fake news from newspapers and social media is a serious issue in news credibility. Some of machine learning methods (such as LSTM, logistic regression, and Transformer) has been applied for fact checking. In this paper, we present Transformer-based fact checking model which improves computational efficiency. Locality Sensitive Hashing (LSH) is employed to efficiently compute attention value so that it can reduce the computation time. With LSH, model can group semantically similar words, and compute attention value within the group. The performance of proposed model is 75% for accuracy, 42.9% and 75% for F1 micro score and F1 macro score, respectively.

Keywords : Automated fact checking, Locality sensitive hashing, Natural language processing, Transformer

I. 서론

최근 트위터, 페이스북과 같은 소셜 미디어가 발달하면서 뉴스도 이를 통해 접하는 경우가 많아졌다. 이러한 경향과 더불어 진실과 다른 정보를 퍼뜨리는 가짜 뉴스도 확산되고 있다.

이에 누리꾼들 사이에서 뉴스에 대한 신뢰도가 떨어

지고 있는바, 일부 언론이나 시민단체 등에서 뉴스들이 진실인지 거짓인지 판별하는 팩트체크를 하고 있다. 하지만 작업이 현재 작업자의 수작업에 의존하고 있는 나머지, 모든 뉴스에 대해서 실시간으로 검토하기엔 역부족이다. 이에 일부 연구에서는 이러한 공정을 자동화하는 모델을 제시하고 있다[1].

최근 관련 연구에서는 팩트체크 자동화 모델에 트랜스포머(Transformer) 모델을 접목하고 있다[2]. 하지만 트랜스포머 모델의 특성상 계산 복잡도가 커 이를 효율화하는 연구도 진행되고 있다.

본 논문에서는 Locality Sensitive Hashing(LSH) 알고리즘을 이용해 의미가 유사한 단어들을 묶은 후, 이러한 집합 별로 attention 값을 계산하는 방식으로 트랜스포머 모델의 계산 복잡도를 감소시키는 모델을 제안할 것이다.

II. 관련 연구

2.1. 팩트체크 자동화 모델

기존의 팩트체크 모델은 주로 텍스트 자체의 분석을 주안점으로 두고 발전했는데, 초기에는 TF-IDF를 이용하여 팩트 뉴스와 가짜 뉴스에서 각각의 단어 빈도수들을 분류하는 방식으로 진행되었다. 다른 연구에서는 텍스트 데이터가 여러 단어로 이뤄진 순차 데이터(sequential data)인 점에서 착안해 LSTM과 같은 순환형 신경망 모델을 이용한다[1]. 또한, 텍스트 분석 수준을 넘어 텍스트의 맥락을 벡터로 만들고, 여기에서 나오는 특징을 통해 뉴스를 분류하는 작업을 하는 모델도 제안되고 있다.

2.2. 트랜스포머

트랜스포머[3]란 어텐션 함수에 기반한 자연어 처리 모델로, 단어 간의 관련도를 계산하여 맥락을 벡터로 생성하는 방식으로 모델 학습이 진행된다.

Received 16 July 2021, Revised 23 July 2021, Accepted 29 July 2021

* Corresponding Author Jason J. Jung(E-mail:j3jung@cau.ac.kr, Tel:+82-2-820-5136)

Professor, Department of Computer Engineering, Chung-Ang University, Seoul, 06974 Korea

Open Access <http://doi.org/10.6109/jkiice.2021.25.9.1275>

print ISSN: 2234-4772 online ISSN: 2288-4165

여기서 어텐션(attention) 함수란 소프트맥스 함수를 변형한 공식이다. 수식으로 나타내면 다음과 같다.

$$\alpha_{ij} = softmax\left(\frac{Q \cdot K}{\sqrt{d}}\right) \cdot V \quad (1)$$

이 식에서 Q는 입력에 해당하는 쿼리(Query) 벡터, K는 쿼리 벡터와 비교 대상에 해당하는 키(Key) 벡터, V는 후에 소프트맥스 값과 곱해지는 밸류(Value) 벡터를 의미한다. 그리고 d는 벡터 차원을 뜻한다. 즉, 어텐션 함수는 Q와 K 간의 유사성을 계산하고, \sqrt{d} 를 나눠 정규화하는 단계를 거쳐 softmax 함수에 넣은 후 결과값과 V와의 내적으로 결과값을 산출하는 방식으로 진행되는 것이다.

트랜스포머 모델의 또다른 특징인 인코더-디코더 구조에 있다. 인코더-디코더 구조는 sequence-to-sequence 모델로 분류되는데, 문장을 입력 데이터로 받은 후 출력 데이터로써 문장을 받기 때문이다. 영한번역 케이스를 예로 들자면, 인코더는 번역 대상이 되는 영문 문장을 분석한다. 이후 여러 어텐션 함수로 묶인 인코더 층을 통해 나온 기울기 값을 디코더에 전달한다. 이때 디코더는 그 값을 받아서 영문 문장에 해당하는 한국어 문장을 출력한다.

이러한 구조의 트랜스포머는 머신번역이나 자연어 생성 등의 다양한 분야에서 응용되었다. 그리고 인코더나 디코더만 사용하는 방식으로 모델 연구가 진행되고 있는데, 이 중 각각 BERT[4]나 GPT가 있다.

2.3. Locality Sensitive Hashing (LSH)

LSH[5]란 높은 유사도를 가진 데이터 값끼리 같은 해시값을 부여하는 함수를 말한다. 이를 통해 기존에 해싱되지 않은 때에 비해 모델에 들어가는 데이터의 양을 줄일 수 있다.

우선 LSH 알고리즘에서 하는 것은 단어를 집합 단위로 묶는 것이다. 이를 위해 영어의 경우 각 단어를 알파벳 단위로 쪼갠 후 n-gram 방식으로 묶어서 유사도를 계산하는 과정을 거친다. 이를 통해 의미적으로 비슷한 단어끼리 묶어서 하나의 집합으로 만든다. 이후 집합에 주소에 해당하는 숫자를 매기는 과정을 거치는데, 여기에서 해시값이 매겨지게 된다.

이러한 LSH 알고리즘을 활용한 트랜스포머도 연구되고 있다. 리포머(Reformer)[6] 구조가 그 예인데, LSH

를 이용해 관련도가 높은 단어끼리 같은 해시값을 부여한 후, 그 집합 별로 어텐션 값을 계산한다.

III. 제안 모델

3.1. 모델 구조

본 논문에서 제안하는 모델은 트랜스포머의 인코더 부분만 쓰는 BERT를 골격으로 하여, LSH attention 층을 임베딩 층 위에 얹는 방식으로 구성했다. 이리하여 문장을 벡터로 만든 후에 단어 간 유사도가 높은 것끼리 같은 해시값을 부여하여, 같은 해시값을 공유하는 집합끼리 어텐션 값을 계산하게 하는 것이다. 이러한 내용을 요약해서 도식화하면 그림 1과 같다.

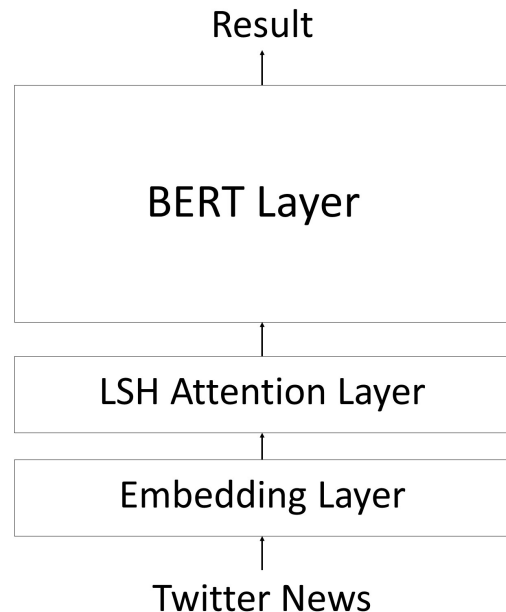


Fig. 1 The architecture of proposed model

제안 모델은 LSH에 기반하여 의미적으로 유사한 단어 벡터끼리 어텐션 값을 연산하므로, 모든 단어쌍에 대해 어텐션 값을 일일이 연산하는 기존의 트랜스포머 모델과 비교할 때 계산 시간을 아낄 수 있다는 이점을 갖고 있다. 또한, 리포머 모델은 언어 모델을 학습시키는데 목적이 있어 취급하는 문장의 길이가 긴 반면, 이 모델은 상대적으로 짧은 트윗을 학습시키는 것이 목적이므로 데이터 크기가 작은 편이다.

IV. 실험 환경 및 실험 결과

4.1. 실험 환경

먼저 제안 모델 생성에 활용한 데이터부터 설명하자면, PHEME 데이터셋[7]을 사용했다. 이 데이터셋은 샤를리 에브도 총격 사건과 같은 이슈에 대해서 트위터에 공유된 뉴스나 뉴스를 공유한 리트윗으로 구성되어 있는데, 여기에는 rumour/non-rumour로 라벨이 달려 있어 가짜 뉴스와 진짜 뉴스를 가리는 모델을 학습하는 데 적합하다.

제안하는 모델을 학습시키기 위해서 훈련 데이터와 검증 데이터, 테스트 데이터를 8:1:1 비율로 나누었고, 폴더별로 JSON 파일이 산재되어 있는 특성상, 윈도우 명령어 프롬프트 명령어를 통해 자동으로 파일을 합치는 과정을 거쳤다. 이후 전처리할 때, 리트윗의 경우 출처 트윗이 없는 것은 제거하였다. 데이터의 크기를 라벨 별로 나눠서 정리하자면 표 1과 같다.

Table. 1 Dataset summary.

	source tweet	retweet
non-rumour	2588	24632
rumour	1461	12809

또한, 제안 모델의 성능을 비교 측정하기 위해 벤치마크 모델을 만들었다. 양방향 RNN, LSTM, GRU는 텍스트 데이터가 순차 데이터라는 점을 고려해 선정했고, BERT의 경우 제안 모델의 골격이 되는 모델이라 선정하게 되었다.

또한, 제안 모델 및 벤치마크 모델을 모두 5 에포크씩 훈련 학습을 실행했고, AdamW 옵티마이저를 이용해 모델 최적화를 실행하였다. Learning rate는 0.00001로 설정했다. BERT의 경우 임베딩하는 문장의 최대 길이는 256으로 설정했다.

4.2. 제안 모델의 성능

성능 측정 지수의 경우 정확도(Accuracy)와 F1 score를 썼다. F1 score는 precision과 recall의 조화평균을 뜻하는데, 산출하는 방식에 따라 micro와 macro로 나뉜다. 전자의 경우 집단과 관계없이 모든 분석 결과를 기반으로 계산하고, 후자는 집단별로 세분화해서 계산한다.

Table. 2 Performance comparison on proposed model.

	Accuracy	F1 score (Micro)	F1 score (Macro)	Train Time (s)
Proposed	0.75	0.429	0.75	1245
Bi-LSTM	0.656	0.656	0.396	216
Bi-GRU	0.656	0.656	0.396	216
Bi-RNN	0.656	0.656	0.396	216
BERT	0.999	0.999	0.999	3833

표 2와 같이 순환형 신경망 및 BERT 모델과 성능 비교한 결과, 제안한 모델이 BERT보다는 뛰어나지 않지만, 순환형 신경망 모델에 비해 높은 성능을 보이는 것으로 나타났다. 하지만 제안한 모델이 BERT에 비해 훈련 시간이 적은 것을 확인할 수 있다.

V. 결론

본 논문에서는 최근 들어 가짜 뉴스가 소셜 미디어를 통해 가짜 뉴스가 퍼져 팩트체크에 대한 수요가 많아진 바, 팩트체크 과정을 자동화하기 위해 LSH를 이용한 트랜스포머 모델을 응용하는 방안을 제시했다.

이러한 방식으로 자동화함으로써 보다 많은 뉴스를 실시간으로 모니터링 할 수 있다는 장점과, 이를 통해 언론의 신뢰도를 높이는 데 기여할 수 있다는 장점을 기대할 수 있게 되었다.

비록 제안 모델이 BERT에 비해 정확도가 낮지만 훈련 시간이 적음을 확인할 수 있어, 향후 연구에서 모델 고도화를 하게 되면 높은 정확도와 적은 훈련 시간 모두를 얻어, 정밀하고 빠른 실시간 팩트체크에 공헌할 수 있을 것을 기대할 수 있을 것이다.

ACKNOWLEDGEMENT

This work was supported by the Ministry of Education of the Republic of Korea and the National Research Foundation of Korea (NRF-2017 S1A6A3A01078538).

REFERENCES

- [1] N. Kotonya and F. Toni, "Explainable Automated Fact-Checking: A Survey," *Proceedings of the 28th International Conference on Computational Linguistics*, Barcelona (online), pp. 5430-5443, 2020.
- [2] L. Wu, Y. Rao, Y. Zhao, H. Liang, and A. Nazir, "DTCA: Decision Tree-based Co-Attention Networks for Explainable Claim Verification," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, online, pp. 1024-1035, 2020.
- [3] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention Is All You Need," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017*, pp. 6000-6010, 2017.
- [4] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, Minneapolis: MN, pp. 4171-4186, 2019.
- [5] Wikipedia, Locality Sensitive Hashing [Internet]. Available: https://en.wikipedia.org/wiki/Locality-sensitive_hashing.
- [6] N. Kitaev, L. Kaiser, and A. Levskaya, "Reformer: The Efficient Transformer," in *International Conference on Learning Representations*, online, 2020.
- [7] E. Kochkina, M. Liakata, and A. Zubiaga. "All-in-one: Multi-task Learning for Rumour Verification," *Proceedings of the 27th International Conference on Computational Linguistics*, Santa Fe: NM, pp. 3402-3413, 2018.