딥러닝 기법을 이용한 가짜뉴스 탐지

이동호*, 이정훈**, 김유리***, 김형준****, 박승면*****, 양유준******, 신웅비*******
*성균관대학교 컴퓨터교육과, **경기대학교 통계학과
***한성대학교 산업경영공학과, ****연세대학교 경영학과
*****주식회사 SV, ******가천대학교 소프트웨어학과
*******태릉고등학교
e-mail : danny911kr@skku.ac.kr

Fake news detection using deep learning

Dong-Ho Lee*, Jung-Hoon Lee**, Yu-Ri Kim***, Hyeong-Jun Kim****,
Seung-Myun Park, Yu-Jun Yang, Woong-Bi Shin
*Dept. of Computer Education, Sungkyunkwan University
**Dept. of Application Statistics, Kyonggi University
***Dept. of Industrial & Management Engineering, HanSung University
****School of Business, Yonsei University

요 약

SNS가 급속도로 확산되며 거짓 정보를 언론으로 위장한 형태인 가짜뉴스는 큰 사회적 문제가 되었다. 본 논문에서는 이를 해결하기 위해 한글 가짜뉴스 탐지를 위한 딥러닝 모델을 제시한다. 기존 연구들은 영어에 적합한 모델들을 제시하고 있으나, 한글은 같은 의미라도 더 짧은 문장으로 표현 가능해 딥러닝을 하기 위한 특징수가 부족하여 깊은 신경망을 운용하기 어렵다는 점과, 형태소 중의성으로 인한 의미 분석의 어려움으로 인해 기존 모델들을 적용하기에는 한계가 있다. 이를 해결하기 위해 얕은 CNN 모델과 음절 단위로 학습된 단어 임베딩 모델인 'Fasttext'를 활용하여 시스템을 구현하고, 이를 학습시켜 검증하였다.

1. 서론

2010년대 이후 페이스북, 트위터와 같은 SNS가 급속도로 확산되며 거짓 정보를 언론으로 위장한 형태인 가짜뉴스가 유포되기 시작했다. 그리고 이는 2016년 미국 45대 대선 과정에 큰 영향을 미치며 뜨거운 화두로 떠올랐다.[1] 미 대선 중 페이스북을 통해 확산된 가짜뉴스들은 주로 특정 후보지지, 광고 수익 창출이 목적인 것으로 밝혀졌다.[2] 이후 가짜뉴스 확산을 막기 위해 세계 유수 미디어들이 연합하여 독자들에게 기사 신뢰지표를 제공하고, 미디어에서 가짜뉴스를 전담하여 판별하는 인원을 채용하고 있다.[3] 이를 기술적인 접근을 통해 해결하려는 시도 또한 다양하게 이루어지고 있는데, 대표적인 예로 인공지능 기반과 이상 확산 패턴 감지 기반 탐지 기법이 있다.[4]

그 중, 인공지능 기반 탐지 기법은 데이터를 기반으로 학습된 모델을 이용하여 가짜 여부를 판별한다. 이 기법은 머신러닝 기반 자연어 처리로 분류된다. 해외에서는 이 기법들 중 하나인 신경망 모델이나 결정 트리 등을 사용하여 80% 이상의 정확도를 도출해낸 결과가 여럿 있으나[5][6], 이를 한국어에 그대로 적용하기에는 다음과 같은 문제점이 있다. 첫째, 형태소 중의성과 용언의 불규칙한 변화로 인해 영어에 비해 형태소 분석이 어렵다. 둘째, 같은 의미 문장이어도 한국어 문장 단어수가 영어 문장 단어수보다 적어 깊은 신경망을 운용할 수 있는 특징수가 부족할 수 있다.

본 연구에서는 이러한 제한점을 해결하고 한국어에 적합한 가짜뉴스 탐지 모델을 제안하고자 컨볼루션 신경망과 단어 임베딩 모델인 'Fasttext'를 활용한 시스템을 구현하였다. 그리고 다양한 유형의 가짜뉴스 중 소위 '낚시성 기사'라 불리는 뉴스들 중 기사 제목과 본문이 부정합한 경우를 임무1, 뉴스 본문 중 맥락에 관계없는 내용이 있는 경우를 임무2로 규정하고, 이 두 가지 경우에 대해 가짜뉴스 여부를 판별한다.

2. 관련 연구

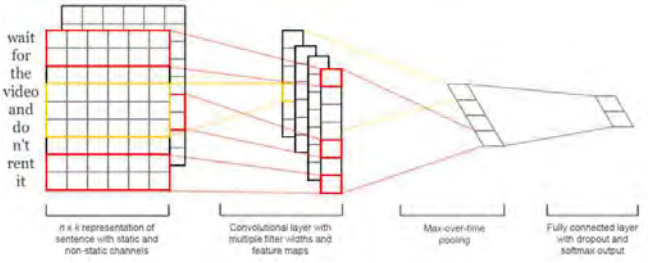
2.1. 컨볼루션 신경망(Convolutional Neural Network)

컨볼루션 신경망은 뇌의 시각피질이 이미지를 처리하고 인식하는 원리를 차용한 신경망이다.[7] 두 가지 연산층(convolution, pooling 층)을 번갈아 수행하며, 최종적으로는 fully connected layer를 통해 분류를 수행하는 계층 모델이다. convolution 층은 입력 이미지에 대해 필터를 적용하여 필터링을 수행하고, pooling layer에서 입력 이미지에 대해 지역적으로 최댓값을 추출하여 2D이미지로 매핑한다. 마지막으로 fully connected layer(affine layer)를

생성한 후, 역전파를 이용해서 입출력간 오차를 최소화 하는 방향으로 학습을 반복해나간다. 본래 CNN은 이미지 처리를 위해 만들어진 아키텍처이나, 최근에는 이를 텍스트에 적용하는 연구도 활발하게 이루어지고 있다. 본 연구는 그 중 Yoon Kim(2014)의 모델을 변형하여 활용한다.[8]

2.1.1. Shallow-and-wide CNN(Yoon Kim, 2014)[8]

모델 구조는 (그림 1)과 같다. 첫 번째 레이어는 단어를 저차원 벡터로 임베드하여 룩업테이블(Look-up Table)을 구축한다. 이후 테이블의 임베디드 단어 벡터에 대해 여러 필터로 합성곱(convolution) 연산을 수행한다. 그 다음 합성곱 레이어 결과를 max-pooling 하여 softmax layer를 통한 결과로 분류를 수행한다.



(그림 1) Shallow-and-wide CNN architecture

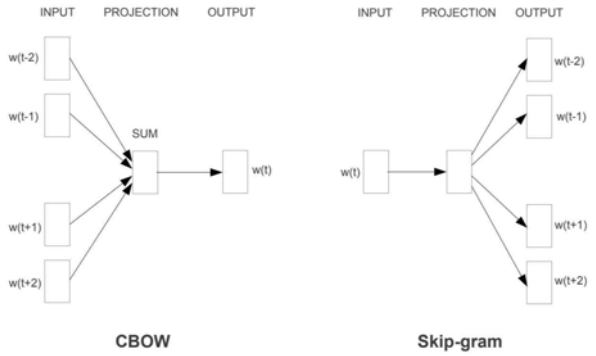
역전파를 전달하는 방식에는 static과 non-static 두 가지 방식이 있다. static은 이미 학습된 단어 벡터들을 사용하여 룩업테이블까지 역전파를 전달하지 않는 방식이고, non-static은 랜덤하게 초기화된 단어 벡터들을 사용하여 역전파가 룩업테이블까지 전달되는 방식이다. 이전 연구에서는 짧은 단문들로 이루어진 dataset으로 감정분석을 진행하였을 때 static과 non-static 두 방식 결과가 대등했으나, non-static을 사용했을 때 단어 의미를 더 잘 파악하고 있었다.[8] 하지만 non-static만 사용하게 되면 새로 등장한 단어가 이 모델에 맞게 과적합이 될 수 있어, 단어의 일반성을 확보하기 위해 두 방식 모두 사용되는 multi-channel 방식도 운용된다. 본 연구에서는 단어 임베딩으로 'Fasttext'를 사용하여 이미 학습된 단어 벡터들을 사용하는 static 방식으로 운용한다.

2.2 단어 임베딩(Word Embedding)

단어 임베딩이란 텍스트를 구성하는 단어 하나를 수치화하는 방법의 일종이다. 전통적 방법인 'Discrete Representation'에는 하나의 차원만 1이고, 나머지 모든 차원들은 0으로 채우는 표기법인 'one-hot vector' 표기법이 있다. 하지만 이런 표기법들에는 문맥을 반영하지 못하고 동의어, 반의어에 대한 처리가 부족한 문제점을 가지고 있어 최근에는 단어를 하나의 '벡터'로 표기하여 벡터 안의 모든 값들이 단어를 특정 하는데 필요한 방법인 'Distributed Representation'이 등장하였다. 본 연구에서는 다양한 표기법 중 'Word2vec'와 'Fasttext'를 소개하고, 이를 활용한다.

2.2.1. Word2vec[9]

'Word2vec'은 신경망을 이용해 단어 임베딩을 구하는 기법이다. 학습방법에는 CBOW와 skip-gram 두 종류가 있다. skip-gram 정확도가 더 높다고 알려져 더 많이 쓰인다.[9] (그림 2)의 skip-gram을 예시로 t번째에 등장한 단어 W를 통해 전후 등장한 단어들로 단어 뭉치를 만들고, t-2번째, t-1번째, t+1번째, t+2번째에 등장하는 단어들을 추측할 수 있는 가중치를 학습시켜 단어 의미를 표현한다. 특정 문맥에서 유추할 수 있는 단어 확률을 최대화하는 방법으로 학습을 시키기 때문에 유사한 단어들은 비슷한 벡터 위치를 가지게 되고 유사도가 높아지게 된다.[10]



(그림 2) Word2vec architecture

수식으로 표현하면 다음과 같다. w가 중심 단어일 때 c는 주변 단어이다. K는 전체 단어 수이다.

$$p(c|w) = \frac{e^{h_w^\top v_c}}{\sum_{k=1}^{K} e^{h_w^\top v_k}}$$

<수식 1> Skip-gram 수식

2.2.2 Fasttext[11]

'Word2vec'에 부분단어(subword) 개념을 추가한 기법이다. 각 단어를 문자 n-gram으로 표현하고, 그 벡터 표현을 skip-gram 혹은 CBOW 형태로 학습한다. 따라서 <수식 1>에서 중심 단어 벡터가 다음과 같이 표현될 수 있다.

$$h_w = \sum_{g \in w} x_g$$

<수식 2> 중심 단어 벡터 수식

3. 신경망 구성 및 학습

본 연구는 다양한 유형의 가짜뉴스 중 소위 '낚시성 기사'라 불리는 뉴스들 중 기사 제목과 본문이 정합하지 않은 경우를 임무1, 뉴스 본문 중 맥락에 관계없는 내용이 있는 경우를 임무2로 규정한다.

임무1	임무2
카타르 축구팀 '도하 참사'... 한국에 33년 만에 패배	'고양이처럼 아님 말고'작가 시연회가 열린다
한국 축구대표팀은 카타르에 결승골을 내주며 2-3으로 무릎을 꿇었다.	'고양이처럼 아님 말고'의 작가 남씨의 사인회가 마련됐다. ... 한편 배연 김지연이 패션쇼에 참석해 과감한 노출 패션으로 시선을 모았다.

<표 1> 각 임무별 기사 예시

3.1. 데이터셋

데이터셋으로는 중앙일보, 동아일보, 조선일보, 한겨레, 매일경제를 크롤링하여 가져온 10만개 뉴스기사를 사용하였다. 언론사별로 경제, 사회, 정치, 연예, 스포츠로 카테고리를 나누어 동일 비율로 기사를 수집하였다. 이 중 임무 1은 31000여개, 임무 2는 68000여개 기사를 사용하였다. 임무별로 진짜 기사와 직접 가공한 가짜뉴스 비율은 1:1로 구성되었다. 그리고 학습데이터와 검증데이터는 90:10 비율로 구성하였으며, 신경망의 최종 정확도는 학습과 검증 데이터에 포함되지 않은 최신 기사(2018년 3월 기준) 350개를 진짜, 가짜 비율이 1:1이 되도록 가공한 후, 이를 기준으로 도출하였다.

3.2. 단어 임베딩

본 연구에서 한국어에 적합한 단어 임베딩을 찾기 위해 10만개 기사에 대해 'Word2vec'와 'Fasttext'를 이용해 학습시켰다. 이에 대한 결과는 <표 2>과 같다.

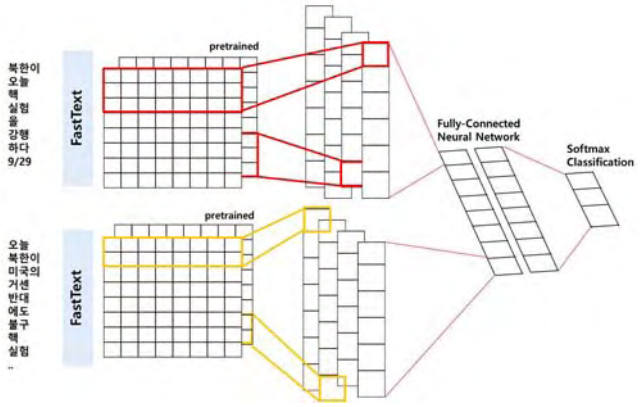
		한국 : 문재인 김정은 : ?	박근혜 : 새누리당 민주당 : ?	한국 : 서울 도쿄 : ?
Word2vec	Batch : 5000 Epochs : 50	미국, 데니스, 로드먼	문재인, 이명박, 대통령	일본 0.71
	Batch : 20000 Epochs : 100	평양, 북한, 미국	문재인, 대통령, 이명박	일본 0.73
Fasttext	Epochs : 5	미국, 북한, 중국	대통령, 문재인, 추미애	일본 0.76
	Epochs : 100	미국, 북한, 중국	대통령, 문재인, 추미애	일본 0.63

<표 2> 'Word2vec'와 'Fasttext' 학습 결과

학습 결과, 정확도 측면에서 'Fasttext' 성능이 더 좋다고 판단되어 본 연구에서는 'Fasttext'를 사용한다.

3.3. 신경망의 구성

본 연구에서 설계한 신경망은 (그림 3)과 같이 단어 임베딩으로 'Fasttext'에 의해 미리 학습된 벡터들을 사용한 컨볼루션 신경망이다. 제목과 본문에서 CONV(Convolution) layer을 통해 특징 맵을 추출한 후, 추출된 특징 맵들을 POOL(Pooling)을 통해 하나로 만든다. 이후 FC(Fully Connected) layer를 거쳐 분류를 진행한다. CONV와 FC의 활성함수는 ReLU, 분류를 위한 활성함수는 Softmax를 사용하였다. 하이퍼파라미터는 <표 3>와 같이 설정하였다. 설계 시 고려사항으로는 기사 제목과 본문 내용 사이 텍스트 수 차이가 큰 것을 고려하여 필터 개수를 이에 비례하게 설정하였다.



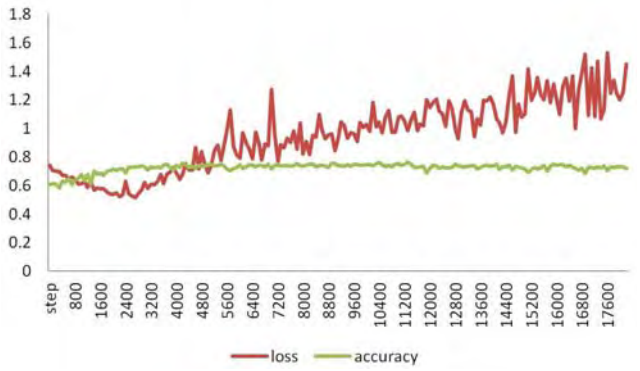
(그림 3) 신경망 구조

Label	Description	Optimized
filter_size	필터의 크기	3
num_filters	필터 개수	256(기사 제목) 1024(기사 본문)
dropout	드롭아웃	0.5
L2_alpha	학습률	0.1
batch_size	학습 미니배치 크기	64
embedding_Dim	단어 임베딩 벡터 차원 수	128

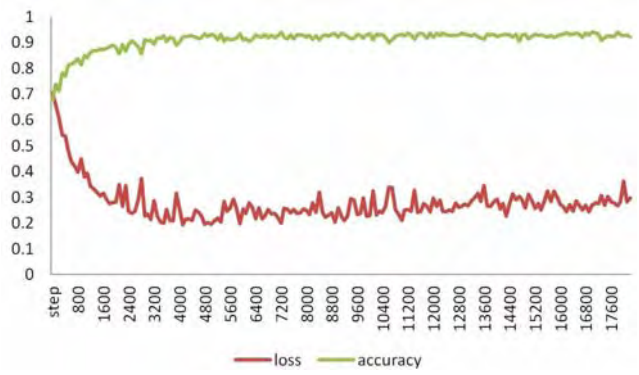
<표 3> 하이퍼파라미터 설정

4. 연구 결과

임무별로 학습이 진행됨에 따라 도출되는 검증 데이터 손실과 정확도는 (그림 4), (그림 5)와 같다.



(그림 4) 임무 1 스텝별 손실과 정확도



(그림 5) 임무 2 스텝별 손실과 정확도

이를 토대로 손실이 가장 적고, 정확도가 가장 높은 임무 1의 2800 스텝 모델과, 임무 2의 4900 스텝 모델을 사용하여 최신 기사들에 적용한 결과를 AUROC를 적용하여 도출해낸 결과는 <표 4>와 같다.

	AUROC
임무 1	0.5280
임무 2	0.7202

<표 4> AUROC 결과

5. 결론

본 연구에서는 가짜뉴스를 검출하기 위한 딥러닝 모델을 구축하고 판별 정확도를 도출하였다. 본 연구 결론을 요약하면 다음과 같다.

첫째, 기사 본문 중 맥락에 관계없는 내용이 있는 가짜뉴스들로 이루어진 임무 2에 대한 판별 정확도는 AUROC 점수 0.72 정도로 비교적 높게 도출되었다. 하지만 본문과 기사 내용이 정합하지 않는 가짜뉴스들로 이루어진 임무 1에 대한 판별 정확도는 0.52 정도로 낮게 도출되었다. 이에 대한 원인은 다음과 같이 유추할 수 있다 : 첫째, 임무 2는 여러 기사들의 본문 일부를 서로 섞는 형태로 많은 가공데이터를 확보할 수 있었으나, 임무 1은 하나하나 가공해야 하는 한계가 있어 상대적으로 학습데이터 양이 적었다. 이로 인해 발생한 학습데이터 양 차이가 정확도 차이를 일으켰다고 유추할 수 있다. 둘째, 컨볼루션 신경망이 텍스트 전역적인 정보를 추출해내고 이를 판별하기 때문에 전역적인 정보 변화가 있는 임무 2에서는 정확도가 높게 나왔으나, 지역적인 정보 변화만 있는 임무 1에서는 정확도가 낮게 나왔다는 것을 유추할 수 있다.

둘째, 한글 단어 유사도 측면에서는 'Fasttext'가 'Word2Vec'보다 성능이 더 뛰어났다. 이에 대한 원인은 다음과 같이 유추할 수 있다 : 한글은 다른 언어와 달리 단어를 이루는 음절이 뜻을 가지고 있다. 예를 들어, '대학'이라는 단어는 '크다'는 의미의 '대'와 '배우다'라는 의미의 '학'이 합쳐져 있다. 이로 인해 음절 단위로 학습이 된 'Fasttext'가 단어 단위로 학습이 된 'Word2Vec'보다 단어 유사도 성능이 뛰어났다고 유추할 수 있다.

본 연구는 가짜뉴스를 판별하고 유의미한 딥러닝 모델을 제안하였다. 연구 한계는 맥락에 관계없는 내용이 있는 경우에 대해서는 유의미한 결괏값이 도출되었으나, 제목과 본문이 부정합한 경우에 대해서는 정확도가 낮았다는 점이다. 향후, 이 모델에 RNN 등 다른 모델을 결합하여 정확도를 향상시키고자 한다.

참고문헌

[1] 운영석, et al. "페이크 뉴스 탐지 기술 동향과 시사점." 정보통신기술진흥센터 주간기술동향 2017. 10. 4.

[2] BuzzFeed, "This Analysis Shows How Viral Fake Election News Stories Outperformed Real News On Facebook.: 2016. 11. 17.

[3] Thetrustproject.org. (2018). The Trust Project - News With Integrity. [online]

[4] S. Kwon, et al. "Prominent features of rumor propagation in online social media." 2013 IEEE 13th International Conference: 1103-1108 (2013).

[5] Largent, W. (2018). Talos Targets Disinformation with Fake News Challenge Victory. [online]

[6] Medium. (2018). Team Athene on the Fake News Challenge - Andreas Hanselowski - Medium. [online]

[7] 조휘열, et al. "컨볼루션 신경망 기반 대용량 텍스트 데이터 분류 기술." 한국정보과학회 학술발표논문집 (2015): 792-794.

[8] Kim, Yoon. "Convolutional neural networks for sentence classification". arXiv preprint arXiv:1408.5882 (2014).

[9] Mikolov, Tomas, et al. "Efficient estimation of word representations in vector space." arXiv preprint arXiv:1301.3781 (2013).

[10] 김우주, 김동희, and 장희원. "Word2vec 을 활용한 문서의 의미 확장 검색방법." 한국콘텐츠학회논문지 16.10 (2016): 687-692.

[11] Bojanowski, Piotr, et al. "Enriching word vectors with subword information." arXiv preprint arXiv:1607.04606 (2016).

소스코드 : https://github.com/2alive3s/Fake_news