컨볼루션 신경망 기반 대용량 텍스트 데이터 분류 기술

조휘열[O1] 김진화[2] 윤상웅[3] 김경민[1] 장병탁[1,2,3]
서울대학교 공과대학 컴퓨터공학과[1]
서울대학교 인문대학 협동과정 인지과학전공[2]
서울대학교 자연대학 협동과정 뇌과학전공[3]
{hyjo, jhkim, swyoon, kmkim, btzhang}@bi.snu.ac.kr

Large-Scale Text Classification Methodology with Convolutional Neural Network

Hwiyeol Jo[1] Jin-Hwa Kim[2] Sangwoong Yoon[3] Kyung-Min Kim[1] Byung-Tak Zhang[1,2,3]
School of Computer Science & Engineering, Seoul National University[1]
Interdisciplinary Program in Cognitive Science, Seoul National University[2]
Interdisciplinary Program in Neuroscience, Seoul National University[3]

요 약

최근 들어 많은 인기를 얻고 있는 딥러닝 알고리즘인 컨볼루션 신경망을 문서 분류 문제에 적용한 결과를 보고한다. 인터넷을 통해 뉴스 문서를 수집하였고, 문서 내용으로부터 주제를 예측할 수 있는 컨볼루션 신경망을 구축했다. 수집한 데이터 셋은 9개의 대주제, 68개의 소주제로 나눠진 60만 건 이상의 한국어 뉴스 문서로 이루어져있다. 구축한 컨볼루션 신경망은 기존 텍스트 분류 기술에서 널리 사용되는 Naive Bayes 및 Support Vector Machine에 비하여 대주제 예측과 소주제 예측 과제 모두에서 뛰어난 성능을 보였다. 한편, 단어의 의미적, 문법적 특성을 벡터로 나타내는 Word2Vec을 신경망의 단어 임베딩 초기화에 적용해보았으나, 기존에 보고된 바와 달리 큰 도움이 되지 않는 것을 관찰했다. 이러한 결과들을 통해 딥러닝 기반 한국어 문서 분류 시스템의 발전에 기여할 수 있을 것으로 기대된다.

1. 서 론

많은 데이터들이 메신저, 인터넷 뉴스, 웹페이지 등을 통해 텍스트 포맷으로 전달되고 있다. 하지만 이러한 데이터들은 매우 빠르게, 동시다발적으로 생기기 때문에 해당 데이터가 어떤 카테고리의 데이터인지 파악하여, 사용자에게 있어 유의미한 혹은 무의미한 데이터인지를 파악하는 것이 중요해졌다.

같은 맥락으로 텍스트 분류 문제는 데이터 마이닝과도 직결된다. 데이터 마이닝이란, 많은 데이터에서 유용한 상관관계 등의 정보를 발굴해내는 것[1]을 뜻하는데, 비디오, 음악 등의 다양한 포맷의 데이터들이 텍스트 형식으로 쉽게 표현 될 수 있기 때문에 유사한 방식으로 분류 한다면 해당 데이터 속 유의미한 정보를 찾는데 도움이 될 것이다.

기존 텍스트 분류 문제에서는 Support Vector Machine이 많이 사용되었으나 최근 순환형 신경망(Recurrent Neural Network)을 이용한 분류[2]뿐만 아니라 컨볼루션 신경망을 사용한 논문도 나오고 있다.컨볼루션 신경망을 활용하여 문장 분류를 하거나[3], 텍스트 데이터 셋을 분류하는[4] 연구들이 있었다. 하지만 전의 연구들은 데이터가 몇 단어 안 되는 짧은 것이거나, 데이터의 양이 적었다.

이 연구에서는 앞서 소개한 연구들보다 많은 60만개의 데이터에, 각 문서에 들어가는 단어 수도 충분했다. 이 데이터를 68개의 클래스에 단일 레이블, 다중 레이블 방법으로 각각 분류해보았다. 또한 많은 연구들이 영어를 처리하는데 집중한 반면, 본 연구는 한국어 문서를 형태소 별로 파싱하여 분석한 점에서도 의미가 있다.

표현(Representation) 측면에는 컨볼루션 신경망을 활용할 때 그 단어 표현에 따라 성능 변화가 어떻게 일어나는지도 확인해보았다. 일반적인 텍스트 분류 기술을 [그림 1]과 같이 요약할 수 있다.

본 연구에서 사용한 컨볼루션 신경망은 기존 텍스트 분류 기술에서 널리 사용되는 Naive Bayes 및 Support Vector Machine에 비하여 대주제 예측과 소주제 예측 과제 모두에서 뛰어난 성능을 보였다. 한편, 단어의 의미적, 문법적 특성을 벡터로 나타내는 Word2Vec을 신경망의 단어 임베딩 초기화에 적용해보았으나, 기존에 보고된 바와 달리 큰 도움이 되지 않는 것을 관찰했다.

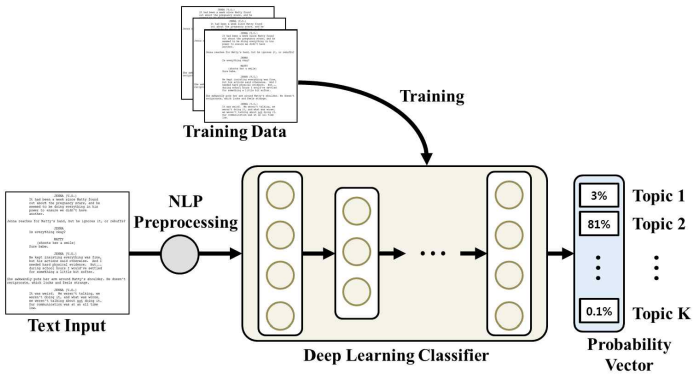

그림 1 일반적인 텍스트 분류 기술 모식도

2. 분류 모델

2.1 전처리기

단어 표현[5]이란, 각 단어들을 수학적인 객체 (e.g. 벡터)들과 연결한 것을 말한다. 각 차원들의 값들은 feature와 연결되어 있으며 심지어는 단어의 의미나 문법에 관련된 정보도 포함 할 수 있다. 특히, 자연 언어 처리에서 성능을 향상시키는 가장 간단하고 일반적인 방법은 단어 표현을 적절하게 만들어 주는 것이다.

일반적인 단어 표현 방법으로는 Brown clusters, Collobert and Weston embeddings, HLBL embeddings가 있었고 최근 Word2Vec[6]가 각광을 받고 있다.

Word2Vec은 무감독 학습 방식으로 피드포워드 신경망을 통해 학습하는 표현으로, 학습 텍스트 데이터에서 어휘를 형성하고, 각 단어의 벡터 표현을 학습한다. 각 단어의 log-likelihood를 높여가는 과정에서 어휘에 대한 구문론적, 의미론적인 정보를 표현 가능하다고 한다.

본 연구에서는 문서마다 처음 n개의 단어를 보며 빈도수를 기준으로 상위 m개까지 단어를 사전에 저장하도록 하였다. 사전에 저장된 단어는 랜덤 표현 또는 Word2Vec 표현을 하였다. 그 외에 사전에 저장되지 않은 단어는 기타단어로서 모두 동일한 표현을 가졌다.

2.2 컨볼루션 신경망

컨볼루션 신경망[7]은 사람의 신경망에서 고안한 모델로 다양한 패턴인식 문제에 사용되고 있다. 두 가지 연산 층(convolutional, subsampling 혹은 max-pooling 층)을 번갈아 수행하며, 최종적으로는 fully connected layer를 통해 분류를 수행하는 계층 모델이다. Convolutional 층은 입력 이미지에 대해 필터 뱅크를 적용하여 2D 필터링을 수행하고, subsampling layer에서 입력 이미지에 대해 지역적으로 최댓값을 추출하여 2D이미지로 매핑한다. 점차적으로 영역을 더 크게 하고, 다운 샘플링을 반복한다. 최종적으로 fully connected layer를 생성한 후, 역전파를 이용해서 입-출력간 오차를 최소화 하는 방향으로 학습을 반복해나간다. 기존엔 얼굴 인식, 손 글씨 인식 등에 사용되었으나, 최근에는 텍스트 데이터에도 많이 사용되고 있다.

텍스트에서 컨볼루션 신경망의 첫 번째 층은 문장 속 단어들을 테이블 lookup을 이용하여 단어 벡터로 만든다. 각 단어를 픽셀로 생각하고 각 문서를 문서 당 단어의 개수만큼의 채널을 가진 (|문서|)×1 벡터로 표현한 뒤 그 나머지는 이미지의 경우와 동일하게 단어 벡터를 feature 벡터로 사용한다.

3. 실 험

3.1 실험 데이터

실험을 위해 다양한 분야에 분포되어 있는 623,303개의 텍스트로 표현된 뉴스 데이터를 수집하였으며 학습 데이터, Cross-Validation 데이터, 테스트 데이터의 비율은 각각 70%, 15%, 15%로 설정하였다. 수집한 뉴스데이터의 분야별 분포는 다음과 같았다.

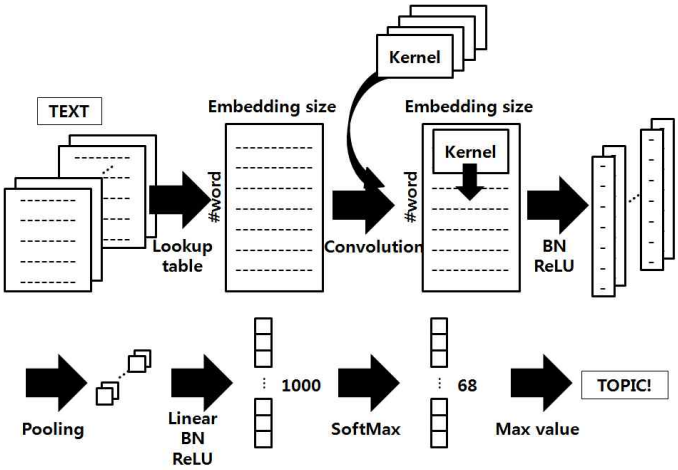

그림 2 문서 분류 컨볼루션 신경망 개요

Large Category	Small Category(#Document)
Science	Kistiscience(2,065), Science_general(3,063), Scienceskill(425)
Special Section	Esc_section(8,494)
International	Arabafrica(6,221), China(5,556), Globaleconomy(2,199) Europe(5,431), Internationalunit(564), Asiapacific(3,684), America(14,359), International_general(14,203) Globaltopic(887), Japan(6,437)
Economy	Working(2,852), Finance(6,228), Marketing(1,098) IT(3,998), Car(3,371), Stock(4,299), Heri_review(795) Biznews(6,393), Consumer(5,160), Property(5,970) Economy_general(49,679)
Politics	Bluehouse(5,856), Assembly(14,946), Politics_general(35,056), Defense(16,864) Administration(1,951), Diplomacy(2,926)
Culture	Travel(889), Movie(7,761), Book(14,068), Culture_general(12,067), Entertainment(13,054), Music(7,761), Religion(2,218), Skysea(50), Novel_salt(126), Baryprincess(126)
Society	Labor(6,305), Internalmove(459), Religious(2,448), Campus(4,338), Environment(5,249), Society_general(105,143), Women(1,604), Schooling(19,725), Health(7,484), Ngo(2,332), Rights(2,232), Obituary(4,398), Media(6,128), Area(52,459), Handicapped(797)
Opinion	Dica(1,331), Column(18,701), Because(5,462), Editorial(9,398), Argument(141)
Sports	Sports_general(24,113), Gameschedule(2,672), Baseball(11,911), Baduk(605), Scoreboard(382), Soccer(17,042), Golf(3,834)
[Total - 68 class] 623,303	

표 1 실험 데이터 문서 수

3.2 파라미터 설정

학습률, 채널 크기, 커널 크기 등 여러 파라미터에 대하여 파인 튜닝을 했다. 상당히 큰 대용량 처리이기 때문에 한 번의 학습에 걸리는 시간이 상당했다. 따라서 빠른 최적점을 찾기 위해 batch normalization[8]을 사용했다. 그 결과, 컨볼루션 신경망의 학습에 평균 7시간이 걸렸지만, 최적점을 5~7 epoch에서 찾는 것을 확인할 수 있었다. 활성 함수로는 ReLU (Rectified Linear Unit)를 사용했다.

최적의 파라미터는 문서당 단어수를 300개, 상위 50만개 단어를 사전에 저장하도록 하고, 학습률은 0.02, 학습률 decay는 선형방식으로, 채널 크기, 커널 크기, 히든 크기, 임베딩 크기는 각각 2,500, 4, 1000, 300으로 했을 때 가장 높았다.

3.3 실험 결과

컨볼루션 신경망의 분류 성능을 측정하기 위해 비교모델로 TF-IDF (Term Frequency-Inverse Document Frequency)를 사용한 Support Vector Machine과 Multinomial Naive Bayes를 사용하였다.

각 모델에서 정확도는 다음과 같다. 정확도는 가장 확률이 높은 카테고리 1개, 3개, 5개안에 정답이 있는지를 확인하도록 하였다.

Model	Accuracy (Top-1,3,5)		
MNB	0.641	0.911	0.958
SVM	0.795	0.960	0.991
CNN	**0.856**	**0.986**	**0.997**

표 2 모델 별 대주제 실험 정확도

Model	Accuracy (Top-1,3,5)		
MNB	0.399	0.679	0.794
SVM	0.614	0.851	0.906
CNN	**0.700**	**0.920**	**0.962**

표 3 모델 별 소주제 실험 정확도

Repre.	Accuracy (Top-1,3,5)		
Rand	0.856	**0.986**	**0.997**
W2V	**0.857**	0.985	**0.997**

표 4 Representation 별 대주제 실험 정확도

Repre.	Accuracy (Top-1,3,5)		
Rand	**0.700**	0.920	**0.962**
W2V	0.696	**0.921**	**0.962**

표 5 Representation 별 소주제 실험 정확도

실험 결과, 컨볼루션 신경망 모델이 가장 정확도가 높았다. 반면, 컨볼루션 신경망의 표현을 Word2Vec으로 임베딩한 결과는 성능 향상에 큰 도움을 주지 못하는 것을 발견하였다.

4. 결 론

본 논문은 대용량 텍스트와 주어진 카테고리가 있을 때, 컨볼루션 신경망을 통해서 분류를 하는 모델을 제안하는 내용을 담고 있다. 그동안 컨볼루션 신경망은 컴퓨터 비전 분야에서 많은 문제들을 해결해왔으나, 텍스트 문제에 적용해보았을 때에도 강력한 성능을 발휘한다는 것을 알 수 있었다. 반면, Word2Vec에 관한 다른 많은 논문들과 달리 대용량 텍스트 분류 문제에서 Word2Vec를 도입했을 때 성능 변화가 실험의 오차 범위 안에서 이루어졌다. 원인으로는 다양한 분야의 데이터를 사용하다보니 같은 단어라도 다른 의미를 내포할 수 있기 때문에 정성적인 문제가 발생한 것으로 추측해 볼 수 있다. 또는, 데이터가 워낙 방대하다보니 Word2Vec 표현으로 부족한 corpus를 보완해주는 것이 그다지 성능 향상에 도움이 되지 않는다고 해석 가능하다. 이는 추후 연구에서 하도록 한다.

참고 문헌

[1] Rajaraman, Anand, and Jeffrey D. Ullman. "Mining of massive datasets," Vol. 77. Cambridge: Cambridge University Press, pp. 1-17, 2012.

[2] Lai, Siwei, et al. "Recurrent convolutional neural networks for text classification." 29th Association for Advanced of Artificial Intelligence Conference on Artificial Intelligence. 2015.

[3] Kim, Yoon. "Convolutional Neural Networks for Sentence Classification", Empirical Methods on Natural Language Processing, 2014

[4] Johnson, Rie, and Tong Zhang. "Effective Use of Word Order for Text Categorization with Convolutional Neural Networks," North American Chapter of the Association for Computational Linguistics, 2015

[5] Turian, Joseph, Lev Ratinov, and Yoshua Bengio. “Word representations: a simple and general method for semi-supervised learning,” In Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, pp. 384–394, 2010

[6] Mikolov, Tomas, et al. "Efficient Estimation of Word Representations in Vector Space". In Proceedings of Workshop at International Conference on Learning Representations, 2013

[7] LeCun, Yann, and Yoshua Bengio. "Convolutional networks for images, speech, and time series," In M. A. Arbib (Ed.), The handbook of brain theory and neural networks, Cambridge, MA: MIT Press, pp. 255–258, 1995

[8] Ioffe, Sergey, and Christian Szegedy. "Batch normalization: Accelerating deep network training by reducing internal covariate shift." International Conference on Machine Learning, 2015