

양방향 장단기 메모리 신경망을 이용한 욕설 검출

Abusive Detection Using Bidirectional Long Short-Term Memory Networks

나인섭^{1†} · 이신우² · 이재학³ · 고진광⁴

조선대학교 SW융합교육원¹, 조선대학교 컴퓨터공학과², 송원대학교 전기전자공학과³, 순천대학교 컴퓨터공학과⁴

요 약

욕설과 비속어를 포함한 악성 댓글에 대한 피해는 최근 언론에 나오는 연예인의 자살뿐만 아니라 사회 전반에서 다양한 형태로 증가하고 있다. 이 논문에서는 양방향 장단기 메모리 신경망 모델을 이용하여 욕설을 검출하는 기법을 제시하였다. 웹 크롤러를 통해 웹상의 댓글을 수집하고, 영어나 특수문자 등의 사용하지 않은 글에 대해 불용어 처리를 하였다. 불용어 처리된 댓글에 대해 문장의 전·후 관계를 고려한 양방향 장단기 메모리 신경망 모델을 적용하여 욕설 여부를 판단하고 검출하였다. 양방향 장단기 메모리 신경망을 사용하기 위해 검출된 댓글에 대해 형태소 분석과 벡터화 과정을 거쳤으며 각 단어들에 욕설 해당 여부를 라벨링하여 진행하였다. 실험 결과 정제하고 수집된 총 9,288개의 댓글에 대해 88.79%의 성능을 나타내었다.

■ 중심어 : 양방향 장단기 메모리 신경망, 욕설 검출, 형태소 분석, 웹 크롤링, 불용어

Abstract

Recently, the damage with social cost of malicious comments is increasing. In addition to the news of talent committing suicide through the effects of malicious comments. The damage to malicious comments including abusive language and slang is increasing and spreading in various type and forms throughout society. In this paper, we propose a technique for detecting abusive language using a bi-directional long short-term memory neural network model. We collected comments on the web through the web crawler and processed the stopwords on unused words such as English Alphabet or special characters. For the stopwords processed comments, the bidirectional long short-term memory neural network model considering the front word and back word of sentences was used to determine and detect abusive language. In order to use the bi-directional long short-term memory neural network, the detected comments were subjected to morphological analysis and vectorization, and each word was labeled with abusive language. Experimental results showed a performance of 88.79% for a total of 9,288 comments screened and collected.

■ Keyword : Bidirectional Long Short-Term Memory Neural Network, Abusive Detection, Morphological Analysis, We Crawler, Stopwords

I. 서론

인터넷은 사용자들에게 다양한 정보를 제공하여 사람들의 삶의 양식에 큰 변화를 주었다[17,18]. 모든 사회 현상에는 양면성이 있듯이 인터넷 이용의 확산은 우리 사회에 긍정적으로만 작용하는 것은 아니다. 인터넷 익명성을 이용하여 명예 훼손, 인신공격, 사생활 침해 등과 같이 악의적으로 이용하여 사회적으로 심각한 문제를 양산하고 있다. 인터넷 게시판의 악성 댓글은 인터넷에서 발생하는 불법적인 언어나 행위와 관련하여 가장 대두되고 있는 문제이다. 댓글은 인터넷에 게시된 원문기사에 대하여 또 다른 정보나 의견을 담하여 올리는 글이며, 악성 댓글이란 인터넷의 익명성을 악용하여 상습적으로 남을 헐뜯거나 허위 사실을 퍼뜨리는 댓글을 말한다[16].

“구하라”, “설리”, “유니”, “최진실” 등 최근 우리 사회는 악성 댓글의 영향으로 사망하는 사람들이 늘어가고 있다. 지금까지 악의적인 댓글을 시스템적으로 관리하기 위한 많은 연구들이 진행되었지만 악성 댓글을 작성하는 사용자는 기존의 문법을 파괴하면서 댓글을 작성하기 때문에 변형된 악성 어휘를 인식하는데 취약하다는 단점이 존재한다[16].

이에 이 연구에서는 인터넷상에서 작성되는 방대한 양의 데이터 중에서 악의적인 정보 특히 욕설을 탐지하는 정확도를 높이는 방법을 제안하고자 한다.

이 논문은 2장에서 관련연구로 순환신경망, 장단기 메모리 신경망, 양방향 장단기 메모리 신경망을 살펴보고, 3장에서 제안하는 욕설 검출을 위한 양방향 신경망 모형에 대해 데이터 수집단계, 데이터 전처리 및 특징 생성단계, 그리고 양방향 장단기 메모리 신경망을 적용하는 단계를 살펴본다. 4장에서는 웹상에서 크롤링한 실 데이터에 대해 제안 방법을 적용한 실험한 결과를 그리고 5장 결론을 제시하였다.

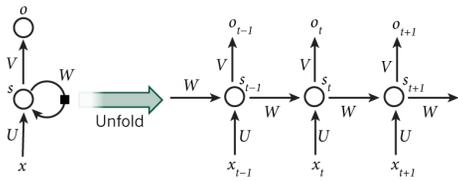
II. 관련 연구

2.1 악성 댓글 탐지

국내외에서 욕설을 포함한 다양한 형태의 악성 댓글을 탐지하려는 시도가 진행되어 오고 있다. 대부분은 특정 플랫폼을 기반으로 진행하고 있는데 해외의 경우 동영상 제공 거대 플랫폼인 유튜브가 대표적이다. 접근 방법 역시 조회수가 높은 뮤직 비디오 n가지의 댓글 데이터를 유튜브 API를 이용해 수집하고 정상적인 햄(HAM)과 비정상적인 스팸(SPAM)으로 구분하는 형태를 취한다[21]. 알베르토(Alberto)[19] 등은 결정 트리, 로지스틱 회귀분석, 베르누이 나이브 베이즈, 랜덤 포레스트, 선형 서포트 벡터 머신, 가우시안 서포트 벡터 머신 등을 기반으로 진행하였고, 툴키아르(Thulfiqar)[20]는 알베르토 방법에 인공 신경망(Artificial Neural Network) 모델을 적용하여 진행하였다.

2.2 순환신경망

순환신경망(RNNs:Recurrent Neural Networks)은 데이비드 루멜하트(David E. Rumelhart), 제프리 힌튼(Geoffrey E. Hinton), 로널드 윌리엄(Ronald J. Williams)이 네이처(nature)지에 1986년에 발표된 “Learning representations by back propagating errors”[9]에 기반하여 발달하였다. 순환 신경망은 앞과 뒤 순서가 존재하는 시계열 데이터에 적합한 모델로, 기존 신경망에서 이전 시간($t-1$)의 은닉층(hidden layer)의 출력 값을 다음 시간(t)의 은닉층으로 다시 넣는 경로가 추가된 형태이다. 그림 1에서 보여지듯, 현재 시간(t)의 결과 역시 다음 시간($t+1$)의 은닉층에 경로가 추가되어 현재 시간이 다음 시간에 영향을 미치고, 다음 시간($t+1$)의 값 또한 그 다음 시간($T+2$)의 값에 영향을 미치는 순환적 구조를 갖는다.

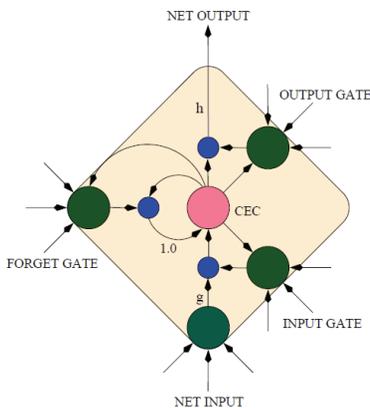


〈그림 1〉 단순한 Neural Networks를 펼친 형태(Unfold)로 바라본 RNNs[6,12]

순환 신경망 알고리즘은 시간적 순서로 입력되는 데이터 패턴을 분류하는 효과적인 알고리즘이지만, 특정 시점으로부터 먼 과거의 정보를 예측 결과에 반영하지 못하는 구조적 단점을 가진다. 이러한 문제점을 장기 의존성 문제(Long-Term Dependency Problem)라고 한다[10,11].

2.3 장단기 메모리 신경망

장단기 메모리 신경망(LSTMs:Long Short-Term Memory Networks)은 순환 신경망의 장기 의존성 문제(Long-Term Dependency Problem)를 해결하기 위해 제안된 내부 연산 구조로서, 이전 시점에 대한 정보의 유지를 담당하는 망각 게이트(Forget Gate)가 존재하는 것이 특징이다. 장단기 메모리 신경망은 기존의 순환 신경망이 은닉층의 연산 결과만 다음 시점으로 전달하는 방식과 달리 먼 과거의

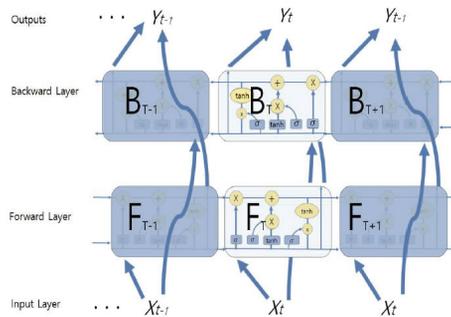


〈그림 2〉 장단기 메모리 셀의 내부 연산구조[5]

시점을 포함한 과거의 셀 상태 정보를 은닉층의 연산 결과와 함께 다음 시점의 은닉 층으로 전달하는 특징이 있다[3,11]. 그림 2는 장단기 메모리 신경망 셀의 내부 연산 구조이다.

2.4 양방향 장단기 메모리 신경망

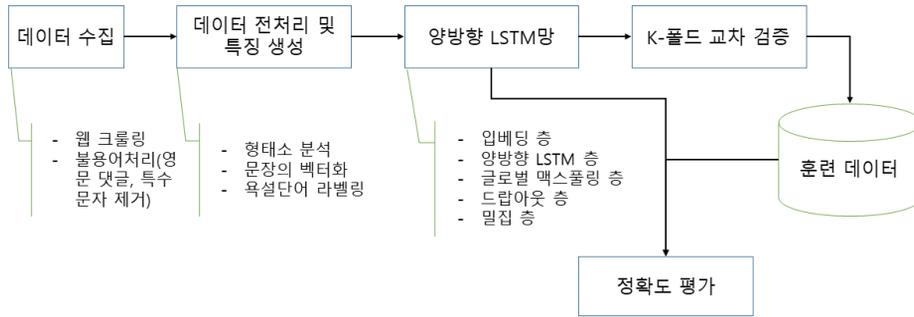
장단기 메모리 신경망 모델의 성능을 향상시키기 위해, 이전 정보를 지속 관리하는 장단기 메모리 신경망에 이후 정보를 추가하여 구성된 신경망 모델이 양방향 장단기 메모리 신경망이 되겠다. 양방향 장단기 메모리 신경망 알고리즘은 그림 3에서 보여지는 것처럼, X는 입력계층, F는 망각게이트, 입력게이트, 출력게이트로 변형한 포워드계층, B는 망각게이트, 입력게이트, 출력게이트로 변형한 백포워드계층, Y는 양방향 장단기 메모리 신경망 출력계층으로 구성된다.



〈그림 3〉 양방향 장단기 메모리 신경망 구조[13]

III. 양방향 장단기 메모리 신경망을 이용한 욕설 검출

맷글에서 욕설 검출을 위하여 제안하는 양방향 장단기 메모리 신경망 모형은 아래 그림 4와 같이 데이터 수집, 데이터 전처리 및 특징 생성, 양방향 장단기 메모리 신경망의 구성을 가지고 있다. 데이터의 검증을 위해 K-폴드 교차 검증을



〈그림 4〉 양방향 장단기 메모리 신경망 모형을 이용한 욕설 검출 구성도

시행한다. 데이터 수집은 웹 크롤링과 불용어처리(영문, 특수문자제거)과정을 거쳐 수행하고, 데이터 전처리 및 특징 생성은 형태소 분석, 문장의 벡터화 그리고 욕설단어 라벨링을 진행한다. 양방향 장단기 메모리 신경망은 임베딩 층, 양방향 장단기 메모리 신경망 층, 글로벌 맥스풀링 층, 드롭아웃 층, 밀집층으로 구성하였다.

3.1 데이터 수집

실험에 필요한 데이터는 디시인사이드(<https://www.dcsinside.com/>)의 6개의 갤러리(국내야구, 주식, 리그 오브 레전드, 해외 연예, 인터넷방송, 역사)로부터 수집하였다. 데이터 수집을 위하여, 오픈 API 중 dc api[1]를 사용하였다.

그림 5와 같이 총 9,402개의 댓글을 수집하였으며 수집된 댓글은 데이터 엔지니어링 과정을 거치며 영문 댓글과 특수문자가 제거된 후 9,288개 수집

gall_name	cmt_contents
0 baseball_nev:8	루돌루돌
1 baseball_nev:8	=====
2 baseball_nev:8	o o
3 baseball_nev:8	사도C 나 구소속 도서폰이건
4 baseball_nev:8	훈인이 것났능
...	...
9397 history	—건 가중 한습늘 거늘음
9398 history	정신은 차라 자조후.
9399 history	도타지드 한습늘 높은걸 써C지
9400 history	막걸리지만 현실성 씬은 것. >g. 말타 가 나 했을 니? > 내가 김한것 훈이다.
9401 history	화난? 다-리-내-가-에- c>다는> 습C서그래

9402 rows x 2 columns

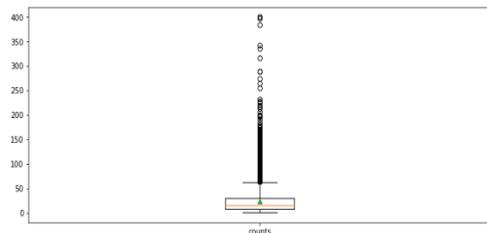
〈그림 5〉 수집된 댓글 목록

되었다.

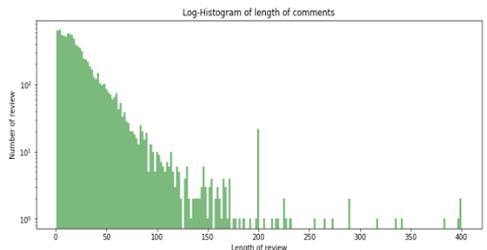
수집된 댓글은 아래 그림 6의 워드 클라우드에서 보여지는 것과 같이 “존나”, “진짜”, “ㅋㅋ”, “그냥”, “관중” 등의 단어가 많이 수집되었다.



〈그림 6〉 워드 클라우드



(a) 수집된 댓글 길이에 대한 상자 수염 그림

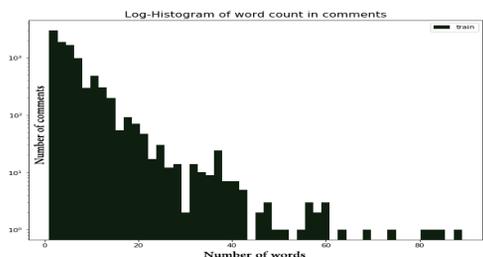


(b) 수집된 댓글 길이에 대한 로그 히스토그램

〈그림 7〉 수집된 댓글 길이 분석

수집된 댓글의 특성에 대해 그림 7 (a), (b)와 같이 상자 수염 그림(box plot)과 로그 히스토그램 분석을 하였다. 분석 결과 댓글의 길이 최댓값은 400자, 최솟값은 1자, 평균 댓글의 길이는 24.2자, 표준 편차는 27.3자, 중간값은 17.0자, 제1사분위는 8.0자, 제3사분위는 30.0자이다.

수집된 댓글 단어를 분석해 보면 그림 8에서 보는 것과 같이, 댓글 단어 개수의 최댓값은 89단어이고, 최솟값은 1단어, 평균값은 5.6단어, 표준 편차는 6.0 단어, 중간값은 4.0단어, 제1사분위는 2.0단어, 제3사분위는 7.0단어를 나타내고 있다.



〈그림 8〉 댓글의 단어 개수에 대한 로그 히스토그램

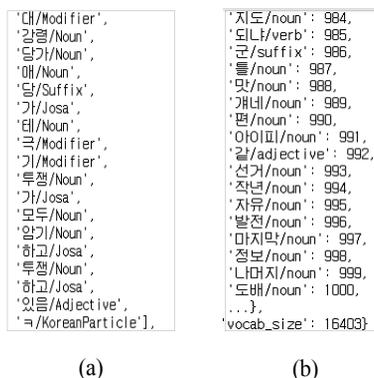
3.2 데이터 전처리 및 특징 생성

수집된 데이터에 대해 양방향 장단기 메모리 신경망(BLSTMs:bidirectional LSTM Networks)에서 사용할 특징 생성을 하기 위해, 형태소 분석, 문장의 벡터화, 욕설 단어 라벨링의 3 단계를 거쳐 코드북 사전을 작성한다.

3.2.1 형태소 분석

코드북 단어 사전의 한국어 형태소 분석을 위하여, 자연어처리(NLP:Natural Language Processing) 분야에서 널리 사용되고 속도가 빠른 KoNLPy[3,4]의 Okt(구. Twitter)를 사용한다. KoNLPy는 JAVA, C++ 등으로 구현된 형태소 분석기들을 모아 파이썬에서 사용할 수 있게 만든 오픈 패키지다. KoNLPy는 총 5개의 형태소 분석기(Hannanum, kma, Komoran, Mecab, Twitter)로 이루어져 있다.

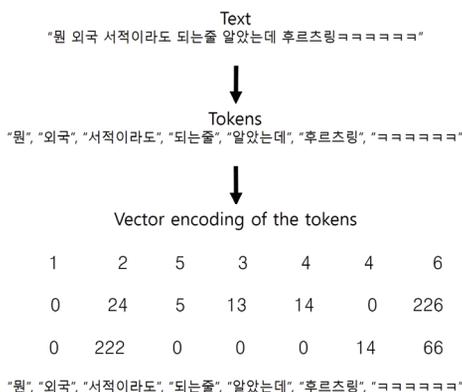
분석기는 각각의 사전(dictionary)을 가지고 있고, 사전 내용을 기반으로 형태소 분석을 진행한다[2]. 형태소 분석결과 아래 그림 9의 (a)와 같이 불용어를 제거한 단어를 추출하고, 추출된 단어에 대해 형태소를 기록한다. 그림 9의 (b)와 같이 추출된 형태소 데이터에 대해 단어 순서를 라벨링한다. 이 연구에서는 총 16,403개의 단어가 추출되었다.



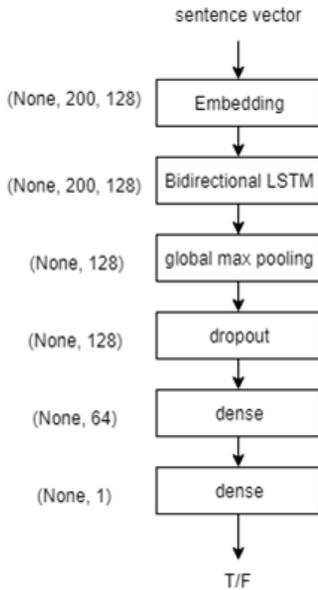
〈그림 9〉 형태소 분석 결과

3.2.2 문장의 벡터화

양방향 장단기 메모리 신경망(bidirectional LSTM Networks)의 입력 값으로 사용하기 위해 단어를 토큰화하고 정수화한다. 정수형으로 표현하기 위해 벡터화 과정을 거친다. 형태소 분석과정 중 토큰된 문장에 대해 그림 10과 같이 숫자 정보로 벡터화



〈그림 10〉 문장의 벡터화 과정



〈그림 13〉 제한된 양방향 장단기 메모리 신경망 모형

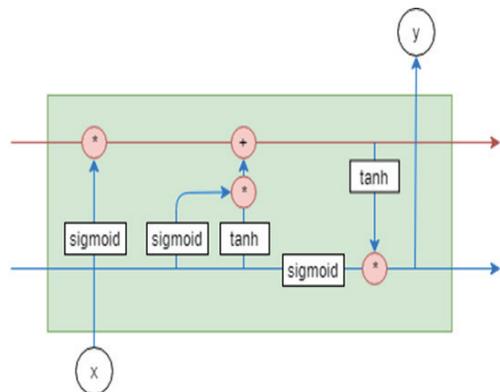
3.3.1 임베딩 층

단어의 임베딩 층에서는 단어들간의 거리를 계산하고자 할 때, 단어를 0과 1과 같은 정수의 벡터값으로 나타내다 보면, 단어 집합의 크기만큼의 고차원 희소 벡터(sparse vector) 형태로 표현되게 된다. 고차원 희소 벡터의 단점은 단어의 수만큼 벡터의 차원을 가지게 되다 보면 단어 간의 유사도가 모두 같아지는 단점을 가지게 된다. 따라서 단어 벡터의 정보를 유의미한 저차원 밀집 벡터(dense vector)의 실숫값으로 변화시킬 필요가 있다. 임베딩 층에서는 고차원 희소 벡터를 저차원 밀집벡터로 변환하는 작업을 하고 샘플(단어)의 수(number of samples), 입력 문장의 길이(length of input sequence)의 2D 정수 텐서를 입력 받아, 샘플(단어)의 수(number of samples), 입력 문장의 길이(length of input sequence), 임베딩 단어의 출력 차원(embedding word dimensionality) 즉 결과로 나오는 임베딩 벡터의 크기로 구성된 3D 텐서정보를 반환한다. 이 정보는 양방향 장단기 메모리 신경망의 입력값으로 사용한다. 이 논문에서는 3.2.1 형태소 분석결과 16,403개보

다 큰 수를 샘플(단어)의 수로 사용한다. 입력 문장의 길이는 3.1 데이터 수집에서 문장 길이 최댓값이 400자이고, 평균이 24.2자, 표준편차 27.3자로 최댓값과 대푯값과의 편차가 매우 커서 최댓값의 절반 값인 200을 입력 문장의 길이로 선정하였다. 출력 차원은 양방향 장단기 메모리 신경망의 입력값으로 128크기로 설정하였다.

3.3.2 양방향 장단기 메모리 신경망 층

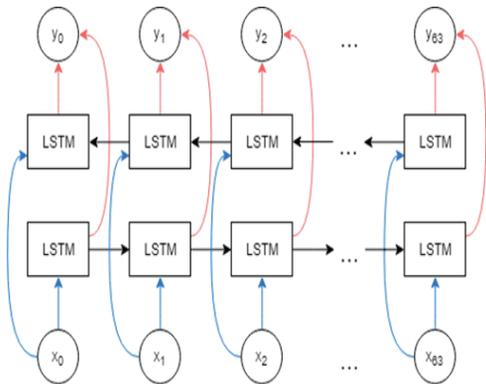
알렉스 그레이브(alex Graves)박사의 RNNs (Recurrent Neural Networks)[5]은 시계열 데이터를 효과적으로 모델링 하는 심층학습모델이다. 하지만 RNNs을 포함한 대부분의 신경망 구조는 학습이 계속 진행됨에 따라, 이전 입력 정보의 학습에 미치는 영향이 점점 감소하다 사라지는 치명적인 문제 즉 경사도가 사라지는 베니싱 그라디언트 문제(vanishing gradient problem)를 가지고 있다. 대부분의 신경망 구조가 심층학습(deep learning)을 구현할 수 없는 이유이다. 이 문제를 해결하고자 RNNs의 은닉 층(hidden layer)을 입력(input), 출력(output), 망각(forget) 3개의 게이트(gate)로 구성하는 메모리 블록(memory block)으로 대체 하는 장단기 메모리 신경망(Long Short-Term Memory)구조를 제시하였다. 우리는 장단기 메모리 신경망의 빠른 구현과 내부적으로 텐서플루어 GPU만을 사용하는 아래 그림 14와 같은 메모리 블록을 이용한



〈그림 14〉 장단기 메모리 신경망구조

cuDNN장단기 메모리 신경망을 사용하였다[7].

현재 시간에서 이전 정보를 저장, 활용하여 성능을 높이는 장단기 메모리 구조에, 추가로 이후 정보를 저장해서 활용하는 구조가 양방향 장단기 메모리 구조이다. “여자로 태어났으면 너 재미하고 있었음”라는 문장에서 “너”를 아래 그림 15의 x_0 , “재미하고”를 x_1 , “있었음”을 x_2 라 가정하면, x_1 의 상태에서 전방향 상태(forward states) 정보 x_0 와 후방향 상태(backward states) 정보 x_2 를 은닉 층으로 전달 받아 최종 결과값을 도출하는 방식이다.



〈그림 15〉 양방향 장단기 메모리 신경망 구조

3.3.3 기타 전결합층

양방향 장단기 메모리 신경망 모델의 임베딩과 양방향 장단기 메모리 신경망층외 전결합층(fully connected layer)들은 글로벌 맥스풀링, 드랍아웃, 밀집 층 등이 있다. 글로벌 맥스풀링 층에서는 여러개의 벡터 정보 중 가장 큰 벡터를 선택하여 반환하는 역할을 하게 된다. 이 곳에서는 3차원 데이터를 2차원으로 변형하게 된다. 드랍아웃 층은 과적합[8] 문제를 줄이는 방법으로 사용하였다. 전체 가중치(weight)를 모두 계산하는 것이 아니라 층(layer)에 포함된 가중치 중 일부만 참여시키는 방법이다. 우리는 0.2즉 10개 중 2개를 사용하였다. 첫 번째 밀집 층에서는 총 64개의 출력을 하게 되고 활성화 함수로는 렐루(ReLU)를 사용하였

다. 두 번째 밀집 층에서는 1개의 출력 값을 가지고 활성화 함수는 시그모이드(sigmoid)를 사용하였다. 손실 함수(loss function)는 시그모이드 교차 엔트로피 손실(sigmoid cross-entropy loss)라고도 불리는 이진 교차 엔트로피 손실(binary cross-entropy loss) 함수를 사용하였다. 이 손실 함수는 시그모이드 활성화와 교차 엔트로피 손실로 구성되어, 소프트맥스(softmax)손실과 달리 각 벡터 구성요소에 대해 독립적이고, 모든 벡터 구성요소에 대해 계산된 손실이 다른 구성 요소 값의 영향을 받지 않는 특징을 가지고 있다. 최적기(optimizer)는 단계(step)의 크기가 경사도(gradient)의 크기조정(rescaling)에 영향을 받지 않는 아담 최적기(adam’s optimizer)를 사용하였다.

3.4 K-폴드 교차 검증

데이터의 수가 적은 경우에는 훈련데이터를 많이 사용하게 되면, 검증 데이터도 전체 데이터의 일부이다 보니, 검증할 수 있는 데이터의 수가 적게 되고, 검증 데이터가 적기 때문에 검증의 신뢰도가 떨어지게 된다. 이렇게 학습할 데이터 수가 적은 경우, 과소적합 학습이 될 확률이 늘어나는 현상을 해결하기 위해 K-폴드 교차 검증(K-fold cross validation) 방법을 사용한다. 전체 원본 데이터가 {1,2,3,4}가 있다면 4를 제외한 {1,2,3}을 훈련 시키고 4를 가지고 검증하고, 다시 {1,2,4}를 훈련시키고 3을 가지고 k번만큼 반복 후 평균을 구하여 검증하는 방식이다. 우리는 원본 데이터를 20개의 구간으로 나누어 즉 k=20으로 설정하고 교차 검증을 수행하였다.

IV. 실험 및 평가

3.1 데이터 수집에서 획득된 욕설 댓글 2,089개와 욕설이 아닌 댓글 7,199개로 구성된 총 9,288개에

대해 무작위로 7,430개는 훈련과 검증에 사용하고, 1,858개는 테스트에 사용하였다. 훈련 및 테스트에 사용된 컴퓨터는 cpu는 인텔 i9, 메모리는 64G, GPU는 Nvidia RTX 2080 ti를 사용하였다. 운영체제는 윈도우즈 10을 사용하였고, 주용 사용 언어와 패키지는 아나콘다 3.7, KoNLPy, sklearn, WordCloud, STOPWORDS, eunjeon, Mecab, bs4, parse, selenium, Image, preprocessing 등이다.

정확도 평가는 아래 식과 같이, 정확히 욕설로 검출된 개수를 전체 욕설의 개수로 나누어 평가하였다.

$$\text{정확도} = \frac{\text{전체 욕설 개수}}{\text{정확히 욕설로 검출된 개수}} \times 100$$

실험결과 아래 표 2에서와 같이, K를 20으로 한 K-폴드 교차분석결과는 훈련데이터 7,059개와 테스트 371개를 기준으로 최소 73.75%, 최대 89.4%의 결과를 나타냈다.

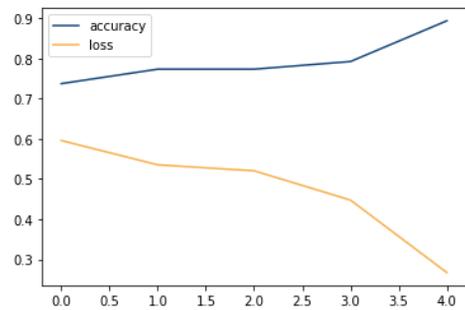
〈표 2〉 K-폴드 교차분석

K-fold (K=20)	7,430 개
훈련	7,059 개
테스트	371 개
최소 손실	26.7 %
최대 손실	59.59 %
최소 정확도	73.75 %
최대 정확도	89.4 %
평균 정확도	79 %

테스트 결과 욕설 검출 결과는 아래 표 3과 그림 16에서 보여지는 것과 같다. K-폴드 교차분석에 사용된 욕설 1,671개와 비욕설 5,759개로 구성된 댓글 7,430개를 훈련데이터 사용하고, 훈련에 사용되지 않은 욕설 418개, 비욕설 1,440개로 구성된 댓글 1,858개를 테스트 데이터로 사용하여 욕설 검출 정확도를 측정한 결과 88.79%를 나타냈다.

〈표 3〉 욕설 검출 정확도

훈련	댓글수	7,430 개
	욕설	1,671 개
	비욕설	5,759 개
테스트	댓글수	1,858 개
	욕설	418 개
	비욕설	1440 개
정확도		88.79 %



〈그림 16〉 욕설 검출률과 손실률

V. 결론

욕설과 비속어를 포함한 악성 댓글에 대한 피해는 최근 언론에 나오는 연예인의 자살뿐만 아니라 사회 전반에서 다양한 형태로 증가하고 있다. 이 논문에서는 양방향 장단기 메모리 신경망 모델을 이용하여 욕설을 검출하는 기법을 제시하였다. 웹 크롤러를 통해 웹사이트의 댓글을 수집하고, 영어나 특수문자 등의 사용하지 않은 글에 대해 불용어 처리를 하였다. 불용어 처리된 댓글에 대해 문장의 전-후 관계를 고려한 양방향 장단기 메모리 신경망 모델을 적용하여 욕설 여부를 판단하고 검출하였다. 양방향 장단기 메모리 신경망을 사용하기 위해 검출된 댓글에 대해 형태소 분석과 벡터화 과정을 거쳤으며 각 단어들에 욕설 해당 여부를 라벨링하여 진행하였다. 실험 결과 정제하고 수집된 총 9,288개의 댓글에 대해 88.79%의 성능을 나타내었다. 제안 시스템은 새

롭게 발생하는 신조 욕설과 인위적 변형 욕설 등이 계속 발생하게 되면 일일이 추가하고 훈련 한다는 한계점이 있다. 비록 이 논문에서 양방향 장단기 메모리 신경망을 이용하여 전·후 단어의 관계성을 분석하였지만, 반어법 등과 같이 언어적 특성에 따른 내용은 검출하지 못하는 한계점 역시 존재한다. 향후 문장 내의 감정 등 어감에 대한 정보를 추가하여 분석하게 된다면 제안 방법의 한계점들을 보완할 수 있을 것으로 사료된다.

참 고 문 헌

- [1] dc api 0.4.6, <https://pypi.org/project/dc-api/>, 2019. 11.
- [2] 고동우, 양정진, “KoNLPy와 Word2Vec을 활용한 한국어 자연어 처리 및 분석”, 2018년 한국컴퓨터 종합학술대회 논문집, 한국정보과학회, pp. 2140-2142, 2018.
- [3] 박은정, 조성준, “KoNLPy: 쉽고 간결한 한국어 정보처리 파이썬 패키지”, 제 26회 한글 및 한국어 정보처리 학술대회 논문집, 2014.
- [4] KoNLP, <https://konlpy-ko.readthedocs.io/ko/v0.4.3/#>, 2019.11.
- [5] Alex Graves, Supervised Sequence Labelling with Recurrent Neural Networks, Part of the Studies in Computational Intelligencebook series (SCI, volume 385), 2008.
- [6] 솔라리스, 텐서플로로 배우는 딥러닝, 2018.11
- [7] CuDNNLSTM, <https://keras.io/layers/recurrent/#cudnnlstm>, 2019.11
- [8] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, Ruslan Salakhutdinov, “Dropout: A Simple Way to Prevent Neural Networks from Overfitting”, *Journal of Machine Learning Research*, 15, pp. 1929-1958, 2014.
- [9] David E. Rumelhart, Geoffrey E. Hinton & Ronald J. Williams, “Learning representations by back-propagating errors”, *Nature*, Vol. 323, pp. 533-536, 1986.
- [10] Bengio, Yoshua, Patrice Simard, and Paolo Frasconi, “Learning long-term dependencies with gradient descent is difficult”, *IEEE Transactions on Neural Networks* Vol. 5 No. 2, pp. 157-166, 1994.
- [11] 방성혁, 배석현, 박현규, 전명중, 김제민, 박영택, “순환신경망 기반의 사용자 의도 예측 모델”, 정보과학회논문지, Vol. 45, No. 4, pp. 360-369, 2018.
- [12] Yann LeCun, Yoshua Bengio & Geoffrey Hinton, “Deep learning”, *Nature*, Vol. 521, pp. 436-444, 2015.
- [13] 정호철, 선영규, 이동구, 김수현, 황유민, 심이삭, 오상근, 송승호, 김진영, “에너지인터넷에서 1D-CNN과 양방향 LSTM을 이용한 에너지 수요 예측”, 전기전자학회논문지, Vol. 23, No. 1, pp. 134-142, 2019.
- [14] Alex Graves, Abdel-rahman Mohamed and Geoffrey Hinton, “Speech Recognition with Deep Recurrent Neural Networks”, <https://arxiv.org/pdf/1303.5778.pdf>, 2013.
- [15] Mike Schuster and Kuldip K. Paliwal, “Bidirectional Recurrent Neural Networks”, *IEEE Transactions on Signal Processing*, Vol. 45, No. 11, pp.2673-2681, 1997.
- [16] 홍진주, 김세한, 박제원, 최재현, “감성분석과 SVM을 이용한 인터넷 악성댓글 탐지 기법”, 한국정보통신학회논문지, Vol. 20, No. 2, pp.260-267, 2016.
- [17] 장원구, 이경호, “효과적인 사이버공간 작전수행을 위한 빅데이터 거버넌스 모델”, 한국빅데이터학회 학회지, Vol. 4, No. 1, pp. 39-51, 2019.
- [18] 우종필, 이병욱 등, “빅데이터를 이용한 독감, 폐렴 및 수족구 환자수 예측 모델 연구”, Vol.3,

No.1, pp.55-62, 2018.

- [19] Alberto, Tulio & Lochter, Johannes & Almeida, Tiago. "TubeSpam: Comment Spam Filtering on YouTube", 10.1109/ICMLA.2015.37., pp. 138-143, Dec. 2015.
- [20] Thulfiqar, A., Hussein, A. and Samir, Q. A. "YouTube Spam Comments Detection Using Artificial Neural Network", *Journal of Engineering and Applied Sciences*, 13: pp. 9638-9642, 2018.
- [21] 정민철, 이지현, 오하영, "머신러닝 기반 유튜브 스팸 댓글 탐지," 한국통신학회 학술대회 논문집, pp.555-556, 2019.11.

사 사

이 성과는 2018년도 과학기술정보통신부의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(NRF-2018R1A1A1A05022526).

저 자 소개



나 인 섭(In-Seop Na)

- 2008년: 전남대학교 전산학과 (이학박사)
- 2018년: 캘리포니아 주립대학 방문학자
- 2008년~현재: 조선대학교 SW융합교육원 조교수
- 관심분야: 시각지능, 인공지능, 패턴인식



이 신 우(Sin-Woo Lee)

- 2014년~현재: 조선대학교 컴퓨터공학과 학부생
- 관심분야: 인공지능, 시각지능, 영상처리
- 관심분야: 인공지능, 시각지능, 영상처리



이 재 학(Jae-Hak Lee)

- 2005년: 중앙대학교 전기공학과 (공학박사)
- 2017년~현재: 송원대학교 전기전자공학과 교수
- 관심분야: IoT Control, ICT, 인공지능



고 진 광(Jin-Gwang Koh)

- 1997년: 홍익대학교 컴퓨터공학과 (이학박사)
- 2014년: 조지아공대(GIT) 방문교수
- 1988년~현재: 순천대학교 컴퓨터공학과 교수
- 관심분야: 데이터베이스, USN