Feed-Forward Neural Network를 이용한 문맥의존 철자오류 교정

황현선○, 이창기
강원대학교
{hhs4322, leeck}@kangwon.ac.kr

Context-sensitive Spelling Error Correction using Feed-Forward Neural Network

Hyunsun Hwang○, Changki Lee
Kangwon National University

요 약

문맥의존 철자오류는 해당 단어만 봤을 때에는 오류가 아니지만 문맥상으로는 오류인 문제를 말한다. 이러한 문제를 해결하기 위해서는 문맥정보를 보아야 하지만, 형태소 분석 단계에서는 자세한 문맥 정보를 보기 어렵다. 본 논문에서는 형태소 분석 정보만을 이용한 철자오류 수정을 위한 문맥으로 사전훈련(pre-training)된 단어 표현(Word Embedding)를 사용하고, 기존의 기계학습 알고리즘보다 좋다고 알려진 딥 러닝(Deep Learning) 기술을 적용한 시스템을 제안한다. 실험결과, 기존의 기계학습 알고리즘인 Structural SVM보다 높은 F1-measure 91.61 ~ 98.05%의 성능을 보였다.

주제어: Deep Learning, Word embedding, 문맥의존 철자오류

1. 서론

전상 상에서 작성자가 텍스트를 기록할 때 발생하는 오타나, 어휘적 오인 등으로 발생하는 문제를 철자오류라고 한다. 이러한 철자오류는 자연어처리의 여러 과정에서 지속적으로 오류가 누적되어 문제가 발생하게 된다. 특히 언어적 지식이 부족한 일반 사용자들을 위한 문서 편집기에서는 이러한 오류를 자동으로 잡을 수 있는 자동 교정시스템이 필요하게 된다.

철자오류는 크게 단순 철자오류(Non-word Spelling Error)와 문맥의존 철자오류(Context-sensitive Spelling Error) 두 가지로 나눌 수 있다[1]. 단순 철자오류는 “요금 결죄”에서 ‘결죄’와 “감기가 낮다”에서 ‘낮다’와 같이 단순히 사전에 검색을 하였을 때 존재하지 않는 단어임을 확인하여 쉽게 오류임을 알 수 있는 경우이다. 그러나 문맥의존 철자오류의 경우 “요금 결재”에서 ‘결재’나 “감기가 낳다”에서 ‘낳다’와 같이 국어사전에 존재하나 해당 단어만으로는 오류인지 파악을 할 수 없어, 주위의 문맥정보를 보아야만 오류인지 판단 할 수 있는 경우를 말한다.

문맥의존 철자오류를 발견하기 위해서는 해당 단어 주위의 문맥정보를 보아야 하지만, 형태소 분석 단계에서는 자세한 문맥정보를 얻기가 어려워 다양한 방법으로 연구가 되었다. 규칙 기반의 시스템의 경우 시스템 구축에 고도의 언어학적 지식을 가진 전문가의 많은 노력과 시간이 필요하며, 규칙이 추가 될수록 시스템의 복잡도가 커지게 된다. 통계 기반의 시스템의 경우 통계 모델을 만들어내기 위한 학습 데이터가 많이 필요하게 된다[2].

최근 딥 러닝(Deep Learning)을 이용한 자연어처리 연구가 많이 진행되고 있다[3]. 딥 러닝은 연결주의 인공지능에서 나온 인공 신경망 개념에 많은 수의 비선형 Hidden layer을 추가하여 복잡한 문제를 풀도록 한 기계학습 알고리즘 이다. 그러나 많은 수의 학습 파라미터로 인한 학습속도 저하와 과적합(overfitting)등의 문제가 있으나, 하드웨어의 발전과 과적합을 해결하기위한 비교사학습(unsupervised) 방식의 사전훈련(pre-training) 및 Drop-out 기술 등이 연구되어 문제를 극복할 수 있게 되었다[4]. 본 논문에서는 최근 자연어처리에 이용되는 기계학습 알고리즘인 딥 러닝을 이용한 문맥 의존 철자오류 교정 시스템을 제안한다.

2. 관련 연구

문맥의존 철자오류 교정은 크게 규칙 기반의 시스템과 통계 기반의 시스템으로 연구되었다. 규칙 기반의 시스템의 경우 정확도가 높지만 재현율이 낮으며, 시스템 구축을 위한 노력과 시간이 많이 들어간다는 단점이 있다. 통계 기반의 시스템의 경우 규칙 기반의 시스템보다 재현율이 높지만, 학습데이터 부족은 높은 성능을 내기 어렵다[2]. 그 외 문맥정보를 보기 위해 구문 분석 정보를 이용하거나, 어휘 의미망을 사용한 방법도 연구되었다[5]. 이 연구들은 모두 문맥의존 철자오류가 자주 발생하는 어휘들을 쌍으로 묶어서 해당 쌍에 대해서만 처리하는 교정 어휘 쌍을 이용한 방법[6]으로, 구성된 특정 어휘 쌍에서만 적용될 수 있는 한계가 있다. 이에 따라 최근에는 전체 어절을 대상어로 한 노이지 채널모형(noisy channel model) 기반의 통계적 방법이 연구되었다[7].

딥 러닝의 과적합 문제의 해결 방법 중 하나로 비교사 학습 방식의 사전훈련이 연구되었다. 자연어처리에서는 딥 러닝의 입력으로 단어들이 들어가기 때문에, 이 단어들을 표현할 방법으로 NNLM(neural network language model)로 사전훈련한 단어 표현(Word Embedding)을 사용하게 된다[3]. 이렇게 사전훈련된 단어 표현(Word Embedding)을 사용한 딥 러닝은 한국어 자연어처리에 적용하였을 때, 기존의 기계학습 알고리즘보다 더 성능이 좋다고 연구 되어 있다[8,9].

본 논문에서는 사전훈련된 단어 표현(Word Embedding)을 문맥정보로서 사용한 딥 러닝을 이용하여 형태소 분석 정보만을 이용한 교정 어휘 쌍을 이용한 문맥의존 철자오류 교정을 하는 방법을 제안한다.

3. 딥 러닝을 이용한 문맥의존 철자오류 교정

본 논문에서는 교정 어휘 쌍 방식의 문맥의존 철자오류 교정을 위해 기본적인 딥 러닝의 신경망 모델인 FFNN(Feed-Forward Neural Network)를 사용하였다. FFNN는 정해진 window-size 만큼의 단어들을 보는 구조이다. 먼저 형태소 분석이 끝난 문장에서 교정 어휘 쌍에 해당되는 단어를 찾는다. 그리고 해당 단어와 주위의 window-size 만큼의 형태소를 가져온다.

그림 1은 신경망으로 들어가는 window-size 2 기준으로 추출된 입력 데이터의 예시 이다. 교정 어휘 쌍에 해당되어 현재 문맥의존 철자오류가 존재하는지 알 수 없는 단어와 주위의 정해진 window-size만큼의 형태소들이 들어가게 되고, 신경망의 결과로 이것이 오류가 있는지 없는지를 판단하게 된다. 그리고 오류가 존재할 시 교정 어휘 쌍 모델의 해당 단어에 해당되는 교정 어휘로 바꿔주게 된다.

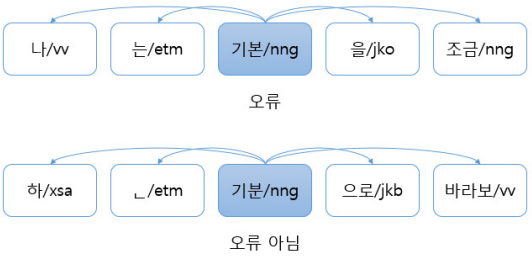

그림 1. 신경망의 입력 예시

그림 2는 교정 어휘 쌍 방식의 문맥의존 철자오류 교정을 위한 FFNN의 구조를 나타낸다. 본 논문에서는 다른 자질정보를 이용한 Feature Embedding을 사용하지 않고 입력 단어의 단어 표현(Word Embedding) 정보만을 이용한 모델로 연구를 진행 하였다. 교정 어휘 쌍을 이용하여 추출한 입력 데이터들은 딥 러닝의 입력으로 들어가게 되고 Projection layer에서 미리 학습된 단어 표현(Word Embedding)을 이용해 단어별로 해당하는 벡터 값으로 변환 되어, 자질의 차원을 줄여주게 된다. 그리고 연결된 가중치들과 곱해져서 연결된 Hidden layer로 값이 전달된다. Hidden layer에서는 비선형 변환 함수인 Sigmoid나 ReLU등을 사용하여 비선형 변환을 하게 된다. 추가적으로 과적합 문제를 해결하기 위해 Hidden layer의 임의 노드(Node)를 일정 확률로 제거하여 학습하는 Drop-out 기술을 0.5의 확률로 적용 시켰다. 신경망의 Output 노드(Node)의 수는 OX분류로 문제를 해결하기 위해 2개로 두어 2가지의 Class분류를 시도하였다. Output layer에서는 Class분류를 위해 Softmax 함수를 사용하였다. 신경망의 학습을 위해서는 학습 목표인 목적 함수(object function)를 지정하게 되는데, Softmax함수의 확률을 이용한 Class분류의 경우 Cross entropy 척도(measure)를 목적 함수로서 사용하게 된다.

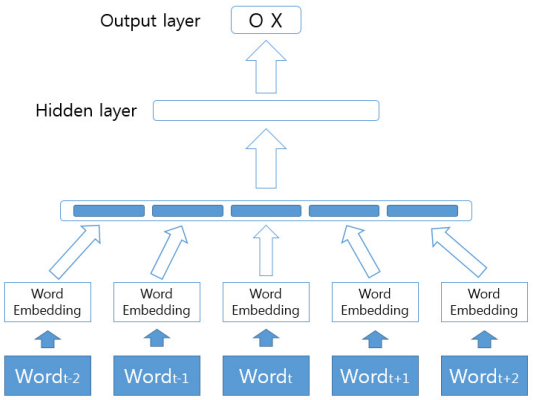

그림 2. 교정 어휘 쌍 방식의 문맥의존 철자오류 교정을

위한 신경망의 구조

4. 실험

본 논문에서는 교정 어휘 쌍 방식의 문맥의존 철자오류 교정의 실험을 위해 세종품사부착말뭉치에서 교정 어휘 쌍을 지정하여 데이터를 추출 하였다.

표 1은 실험을 위해 지정한 교정 어휘 쌍과 추출한 데이터의 문장 수를 나타낸다. 추출한 데이터는 각 어휘 쌍별로 8:2의 비율로 나누어 학습데이터와 평가데이터로 나누었다. 실험 환경은 [1]과 같은 방식으로 설정하였다. 그러나 [1]에서는 한쪽 방향으로의 교정을 테스트 하였는데, 본 논문에서는 데이터 부족으로 양방향으로의 교정을 시도하는 방법으로 실험을 진행하였다. 성능 평가는 정확도(Precision)와 재현율(Recall)을 구해 F1-measure로 성능 측정을 하였다[1].

표 1. 실험을 위해 추출한 교정 어휘 쌍의 데이터 수

단어1	단어2	단어1의 문장 수	단어2의 문장 수
낫다	낳다	1145	2633
마치다	맞히다	1581	72
마치다	맞추다	1591	2157
맞히다	맞추다	72	2157
배다	베다	412	459
집다	짚다	711	406
기본	기분	2769	2431
자식	지식	2500	2510
사정	사장	1743	2017
의지	의자	1726	890
주의	주위	2001	1588

먼저 최적의 비선형 변환 함수와 window-size를 찾기 위해 각 파라미터 별로 실험을 진행하였다. 그림 3은 교정 어휘 쌍 { '낫다', '낳다' }에 대한 실험 결과를 나타낸다. 가로축은 window-size를 나타내고 세로축은 F1-measure 성능을 나타낸다. 비선형 변환 함수로는 Sigmoid 함수보다 ReLU 함수가 더 좋은 성능을 보였으며, window-size가 작을 때에는 볼 수 있는 정보가 적어 성능이 낮음을 확인 할 수 있었다. 그러나 window-size가 너무 크면 확인하는 정보가 너무 많아 과적합 되어 성능이 떨어지는 것을 볼 수 있었다. 본 논문에서는 비선형 변환 함수가 ReLU 이고, window-size가 4일 때가 최적의 파라미터라고 가정하고 나머지 데이터에 대해서 실험을 진행 하였다.

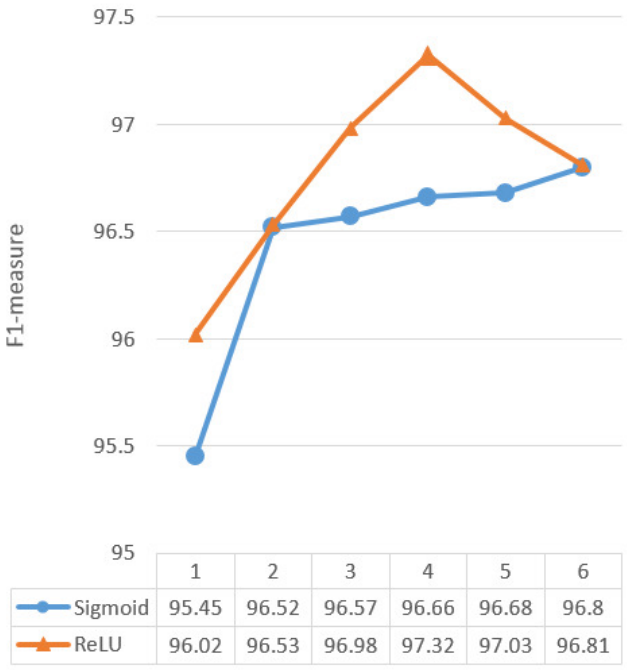

그림 3. '낫다' 와 '낳다' 의 비선형 변환 함수와 window-size별로의 성능 변화 그래프

딥 러닝이 기존의 기계학습 알고리즘보다 얼마나 성능 향상이 있는지를 확인하기 위해 Structural SVM으로 비교실험을 진행 하였다[10]. Structural SVM의 자질 정보로는 딥 러닝의 입력과 같은 단어들을 넣었고 추가로 각 형태소의 POS tag 정보도 넣었다. 표 2는 교정 어휘 쌍별로 기계학습 알고리즘의 성능을 나타낸 표이다. 위의 성능 표에서 Structural SVM의 성능이 낮게 나온 것을 볼 수 있는데, 이는 문맥의존 철자오류 교정이 기계학습으로도 풀기 힘든 어려운 문제임을 보여준다. 딥 러닝의 경우 다른 자질 정보를 사용하지 않고 입력 단어들의 단어 표현(Word Embedding) 정보만을 사용하였음에도 Structural SVM보다 높은 성능을 볼 수 있었는데, 이는 사전훈련된 단어 표현(Word Embedding)이 효과적인 문맥 정보로 이용될 수 있으며, 딥 러닝이 기존의 기계학습 알고리즘보다 복잡한 문제도 풀 수 있는 모델임을 확인할 수 있다. 그리고 일부 Structural SVM의 성능이 딥 러닝보다 높게 나온 데이터들이 존재하는데, 이는 표 1

을 보면 교정 어휘 쌍의 한쪽 단어가 비교적 적은 양의 데이터들임을 볼 수 있다. 이로서 기존의 기계학습 알고리즘인 Structural SVM의 경우 적은 데이터일 때는 딥 러닝보다 높은 성능을 낼 수 있으나, 데이터양이 많고 복잡한 데이터에 대해서는 딥 러닝보다 낮은 성능을 내는 것을 알 수 있다.

표 2. 교정 어휘 쌍별로 기계학습 알고리즘별 성능표

교정 어휘 쌍	F1-measure	
	Structural SVM	딥 러닝
낫다, 낳다	72.14	97.32 (+25.18)
마치다, 맞히다	**96.04**	97.57 (+1.53)
마치다, 맞추다	55.03	96.4 (+41.37)
맞히다, 맞추다	**96.82**	96.77 (-0.05)
배다, 베다	58.88	94.31 (+35.43)
집다, 짚다	61.81	93.92 (+32.11)
기본, 기분	47.65	98.05 (+50.4)
자식, 지식	53.8	92.41 (+38.61)
사정, 사장	51.42	91.61 (+40.19)
의지, 의자	56.15	96.78 (+40.63)
주의, 주위	45.46	96.83 (+51.37)

표 3은 기존연구[1]의 규칙기반 시스템의 성능과 딥 러닝 시스템과의 성능 비교표이다. 사용된 데이터가 달라 정확한 성능의 비교는 어렵지만 재현율이 낮은 규칙기반 시스템보다 딥 러닝 시스템이 비교적 성능이 높은 것을 확인 할 수 있다.

표 3. 기존연구와의 성능 비교표

교정 어휘 쌍	기존연구[1]			딥 러닝		
	precision	recall	F1-measure	precision	recall	F1-measure
낫다, 낳다	100.00	41.51	58.67	97.32	97.32	97.32
	100.00	19.80	33.06			
마치다, 맞히다	100.00	20.00	33.33	97.57	97.57	97.57
	-	-	-			
배다, 베다	100.00	27.00	42.52	95.40	93.25	94.31
	100.00	33.00	49.62			
집다, 짚다	100.00	23.00	37.40	92.39	95.50	93.92
	86.67	13.00	22.61			

5. 결론

본 논문에서는 딥 러닝 기술을 적용한 교정 어휘 쌍 방식의 문맥의존 철자오류 교정 시스템을 제안하였다. 딥 러닝을 적용한 시스템은 가장 간단한 모델인 FFNN에 단어 표현(Word Embedding) 정보만을 사용하여 기존의 기계학습 알고리즘인 Structural SVM보다 높은 성능을 낼 수 있음을 확인하였다.

추후 연구로는 교정 어휘 쌍 방식의 한계를 극복하기 위해 RNN(Recurrent Neural Network)나 LSTM RNN(Long Short Term Memory Recurrent Neural Network)을 이용한 Sequence2Sequence 모델을 이용한 통합적 오류 교정 방식을 연구할 예정이다.

감사의 글

이 논문은 2015년도 정부(미래창조과학부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임. (No.R0101-15-0062, 휴먼 지식증강 서비스를 위한 지능진화형 WiseQA 플랫폼 기술 개발)

참고문헌

[1] 최현수, 윤애선, 권혁철. "조사제약 조건의 완화에 의한 문맥의존 철자오류 교정의 재현율 향상 방식." 정보과학회논문지: 소프트웨어 및 응용 41.3 (2014): 249-256.

[2] 김민호, 권혁철, 최성기. "어절 N-gram 을 이용한 문맥의존 철자오류 교정." 정보과학회논문지 41.12

(2014): 1081-1089.
[3] Collobert, Ronan, et al. "Natural language processing (almost) from scratch." The Journal of Machine Learning Research 12 (2011): 2493-2537.
[4] Hinton, Geoffrey E., Simon Osindero, and Yee-Whye Teh. "A fast learning algorithm for deep belief nets." Neural computation 18.7 (2006): 1527-1554.
[5] 김민호, 최현수, 권혁철, 윤애선. "한국어 어휘의미망을 이용한 문맥 의존 철자오류 교정규칙의 일반화." 정보과학회논문지: 컴퓨팅의 실제 및 레터 20.2 (2014): 106-110.
[6] 김민호, 김경식, 권혁철. "교정 어휘 쌍을 이용한 통계적 문맥 철자오류 교정." 한국정보과학회 2013 한국컴퓨터종합학술대회 논문집 (2013): 607-609.
[7] 김경식, 최성기, 권혁철. "극한 언어사용 환경에 적응적인 문맥의존 철자오류 교정 기법." 한국정보과학회 2015 한국컴퓨터종합학술대회 논문집 (2015).
[8] 이창기, 김준석, 김정희. “딥 러닝을 이용한 한국어 의존 구문 분석.” 한글 및 한국어 정보처리 학술대회, 2014.
[9] 이창기, 김준석, 김정희, 김현기. “딥 러닝을 이용한 개체명 인식.” 한국정보과학회 동계학술발표회, 2014.
[10] 이창기. “Structural SVM을 이용한 한국어 띄어쓰기 및 품사 태깅 결합 모델.” 정보과학회논문지: 소프트웨어 및 응용, 제40권, 제11호, 2013.