Attention Mechanism에 따른 포인터 네트워크 기반 의존 구문 분석 모델 비교

한미래[O1], 박성식[1], 김학수[2]
건국대학교 인공지능학과[1], 컴퓨터공학부[2]
{future26[1], a163912[1], nlpdrkim[2]}@konkuk.ac.kr

Comparison of Pointer Network-based Dependency Parsers Depending on Attention Mechanisms

Mirae Han[O1], Seongsik Park[1], Harksoo Kim[2]
Konkuk University, Department of Artificial Intelligence[1], Computer Science and Engineering[2]

요 약

의존 구문 분석은 문장 내 의존소와 지배소 사이의 관계를 예측하여 문장 구조를 분석하는 자연어처리 태스크이다. 최근의 딥러닝 기반 의존 구문 분석 연구는 주로 포인터 네트워크를 사용하는 방법으로 연구되고 있다. 포인터 네트워크는 내부적으로 사용하는 attention 기법에 따라 성능이 달라질 수 있다. 따라서 본 논문에서는 포인터 네트워크 모델에 적용되는 attention 기법들을 비교 분석하고, 한국어 의존 구문 분석 모델에 가장 효과적인 attention 기법을 선별한다. KLUE 데이터 셋을 사용한 실험 결과, UAS는 biaffine attention을 사용할 때 95.14%로 가장 높은 성능을 보였으며, LAS는 multi-head attention을 사용했을 때 92.85%로 가장 높은 성능을 보였다.

주제어: attention 기법, 포인터 네트워크, 의존 구문 분석

1. 서론

의존 구문 분석은 문장 내 의존소(dependent)와 지배소(head) 사이의 관계를 기반으로 문장 구조를 분석하여 문장의 구조적, 의미적 중의성 문제를 해소하는 작업으로, 상호 참조 해결, 개체명 인식, 기계 번역 등 다양한 자연어처리 태스크에 활용된다[1].

최근 의존 구문 분석은 딥러닝(Deep-learning)을 활용하는 방법이 주로 연구되고 있다. 딥러닝을 활용한 한국어 의존 구문 분석은 대부분 다음의 세 단계로 구분하여 수행된다. 첫 번째는 의존소와 지배소 간의 관계를 인식하는데 필요한 자질들을 추출하여 어절의 벡터 표현을 얻어내는 어절 표상 단계이다. 최근 대용량 말뭉치로 사전 학습한 언어 모델[2]이 자연어처리 연구 전반에서 큰 성과를 거두면서 한국어 의존 구문 분석 모델의 어절 표상 단계에서도 많이 사용되고 있다. 어절 표상 단계에서는 사전 학습 언어 모델을 통해 얻어낸 풍부한 언어 자질이나 형태소 품사 정보 등의 의미적 자질이 활용된다. 두 번째는 각 어절의 주변 문맥을 고려하는 문맥 반영 단계이다. 문맥 반영 단계에서는 어절 표상 단계에서 얻은 언어 자질을 순환 신경망(Recurrent Neural Network, RNN)에 입력하여 문맥을 인코딩한다. 마지막으로, 세 번째는 각 어절의 지배소를 파악하고 의존소와 지배소 간의 관계를 식별하는 지배소 및 의존 관계 결정 단계이다. 지배소 및 의존 관계 결정 단계에서는 포인터 네트워크(Pointer Network)[3]를 활용하는 방법이 주로 연구되고 있다. 포인터 네트워크는 모델의 각 출력 스텝마다 입력 열의 특정 위치를 포인팅하도록 설계된 기술이다. 포인터 네트워크는 내부적으로 attention 기법을 응용해 동작한다. 즉, 어떤 attention 기법을 사용하느냐에 따라 포인터 네트워크의 성능이 달라질 수 있다. 본 논문에서는 포인터 네트워크 기반 한국어 의존 구문 분석의 지배소 및 의존 관계 인식 단계에서 결정적 역할을 하는 attention 기법들을 비교 분석한다.

본 논문의 구성은 다음과 같다. 2장에서는 기존 한국어 의존 구문 분석 연구들에 대해 설명하고, 3장에서는 attention의 비교에 사용할 의존 구문 분석 기본 모델 구조와 비교에 사용할 attention 기법들을 설명한다. 4장에서는 이에 대한 실험 및 결과를 기술하고, 5장에서는 결론 및 향후 연구에 관해 논의한다.

2. 관련 연구

기존 한국어 의존 구문 분석은 포인터 네트워크를 활용하는 연구가 활발히 진행되었다. [4]는 추가적인 순환 신경망 계층 없이 사전 학습 언어 모델 내의 self-attention만을 활용하여 문맥을 인코딩한 후 의존 구문 분석을 수행했으며, [5]는 Sequence-to-Sequence 방식의 포인터 네트워크와 multi-head attention을 이용하여 의존 구문 분석을 수행하였다. [6]은 사전 학습 언어 모델인 BERT를 사용하였고 LSTM 계층에서 문맥 인코딩을 진행한 후, biaffine attention을 적용해서 높은 성능을 보였다. [7]은 ELECTRA 모델과 Specific-Abstraction 인코더 모델에 bilinear attention을 적용하여 지배소와 의존 관계를 예측했다. [8]은 스택 포인터 네트워크(Stack Pointer Network)와

biaffine attention을 적용한 deep biaffine network의 성능을 비교한 연구이며, [9]는 biaffine attention을 적용한 Left-To-Right 포인터 네트워크 모델의 학습 시 손실 함수에 따른 성능 차이를 비교한 연구이다. 기존 연구들은 각기 다른 attention 기반의 포인터 네트워크를 사용해 연구를 진행하였지만, 어떤 attention 기법이 의존 구문 분석에서 가장 좋은 효과를 보이는지는 알 수 없다. 따라서 본 논문에서는 포인터 네트워크 기반 한국어 의존 구문 분석에 가장 효과적인 attention 기법을 선별하기 위한 비교 실험을 진행한다.

3. 포인터 네트워크 기반 의존 구문 분석 모델

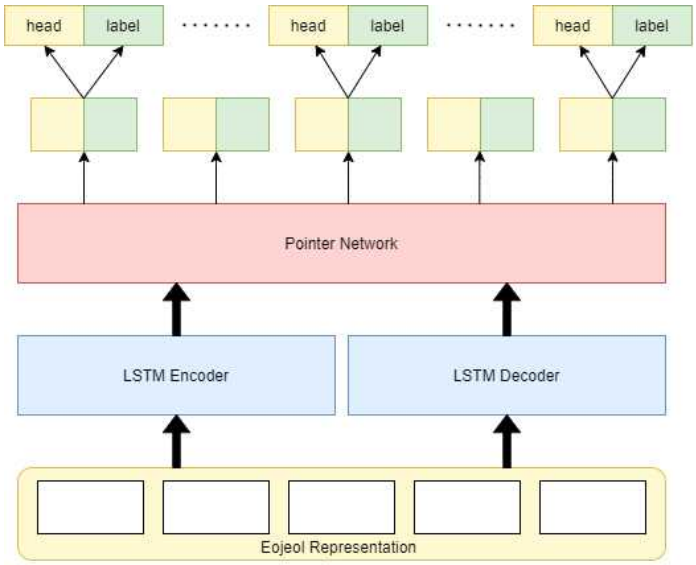

그림 1. 모델의 전체 구조

포인터 네트워크의 attention 기법 비교를 위한 기본 모델의 전체 구조는 그림 1과 같다. 사전 학습 언어 모델을 기반으로 어절 표상을 진행하고, LSTM 인코더-디코더 계층을 통해 문맥 인코딩을 수행한다. 이후 문맥 인코딩 벡터를 입력으로 포인터 네트워크에서 지배소 및 의존 관계를 결정한다. 포인터 네트워크 계층에서 다양한 attention 기법을 적용하여 비교를 진행한다.

3.1 어절 표상 및 문맥 반영

본 논문에서는 어절 표상 단계에서 사전 학습 언어 모델을 사용하였다. 어절 표상 과정은 그림 2와 같다. wordpiece 단위로 분절된 입력 문장을 언어 모델에 입력하여 출력된 벡터들의 평균을 구해 어절 임베딩 벡터를 구성한다. 이후, 형태소 품사 임베딩 벡터와 연결하여(concatenate) 어절을 표상한다. 이때 각 어절의 첫 번째 형태소의 품사 정보만을 어절 표상을 위한 자질로 사용하고, 얻어낸 어절 표상을 bi-LSTM(bidirectional LSTM) 계층에 통과시켜 각 문장의 구조적인 정보를 반영한다.

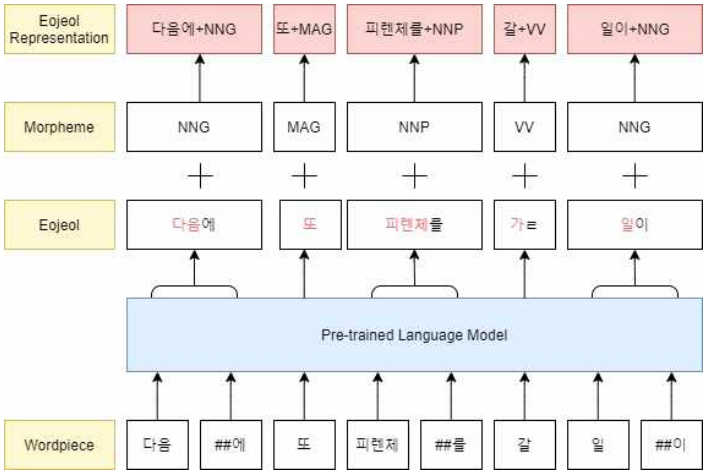

그림 2. 어절 표상 과정

3.2 지배소 및 의존 관계 결정

지배소 및 의존 관계 결정 단계에서는 포인터 네트워크 모델을 사용한다. 포인터 네트워크는 attention 기법을 통해 디코더 출력의 각 스텝에 대응되는 인코더 출력값을 포인팅하도록 한다. LSTM의 인코더와 디코더의 출력이 가지는 상호 의존성을 attention 기법을 통해 계산하여 각 디코딩 스텝마다 가장 관련성이 높게 측정되는 인코딩 위치를 해당 어절의 지배소로 선택하고, 의존 관계 인식을 위해서는 attention 기법의 결과 값을 사용한다. 포인터 네트워크의 성능 증감을 확인하기 위해 scaled dot-product attention, multi-head attention, biaffine attention 기법을 각각 적용한다. attention 기법의 공통적인 연산 과정은 수식 (1)과 같다.

$$S = f(Q, K)$$
$$O = Softmax(S) \cdot V \quad (1)$$

Q는 LSTM 디코더의 출력값, K와 V는 LSTM 인코더의 출력값에 해당한다. attention 연산을 통해 현재 시점의 디코더 출력값과 모든 시점의 인코더 출력값 사이의 연관성을 계산하며, attention의 결과 값인 S와 O는 각각 attention 점수와 단어 확률 분포를 반영한 attention 출력 벡터를 의미한다. $f(Q,K)$의 연산 결과인 S가 가장 높게 측정된 어절을 지배소로 선택한 후, $Softmax(S) \cdot V$의 연산 결과인 O를 이용해 각 어절과 지배소 간의 의존 관계를 판단한다. scaled dot-product attention의 수식은 (2)와 같다.

$$f(Q,K) = \frac{QK^T}{\sqrt{d_k}} \quad (2)$$

scaled dot-product attention에서의 $f(Q,K)$는 내적(dot-product) 연산을 통해 주어진 Q에 대해서 모든 K와의 유사도를 구하여 지배소 선택을 위한 S를 얻어낸다. $\sqrt{d_k}$는 attention 점수 정규화를 위한 값으로, Q, K의 히든 사이즈를 의미한다.

multi-head attention은 scaled dot-product attention을 병렬적으로 여러 번 수행하여 다양한 특징에 집중한 정보를 수집한다. multi-head attention의 수식은 (3)과 같다.

$$
\begin{aligned}
f_i(Q,K) &= \frac{Q_i K_i^T}{\sqrt{d_k}} \\
head_i &= softmax(f(Q_i,K_i))V_i \\
f(Q,K) &= \sum_{i=1}^{n} f_i(Q,K) \\
O &= concat(head_0, ..., head_n)W
\end{aligned}
\quad (3)
$$

Q, K, V를 사전에 정의한 헤드의 개수(n)로 분할하고, 분할된 Q, K, V에 대해서 헤드의 개수만큼 독립적인 scaled dot-product attention 연산을 수행한다. 병렬 attention 연산을 끝마친 후에는 산출된 n개의 $head_i$를 모두 연결하고, linear 계층에 입력하여 최종 attention 결과인 O를 출력한다. W는 linear 계층 학습을 위한 파라미터이다. 각 헤드 attention에서의 결과값 $f_i(Q,K)$를 모두 더해서 지배소 위치 예측에 사용하고, O는 의존 관계명 예측에 사용한다.

biaffine attention은 bilinear 연산을 통해 입력 열에 대한 의존소와 지배소의 문장 구조 및 의존 관계 점수를 계산한다. biaffine attention의 수식은 (4)와 같다.

$$f(Q,K) = QU^TK + W^T(Q+K) + b \quad (4)$$

U, W, b는 학습 가능한 파라미터로, U는 bilinear 연산의 가중치, W는 linear 계층의 가중치, b는 편향(bias)에 해당한다.

4. 실험

4.1 실험 환경

표 1. 모델의 하이퍼 파라미터

하이퍼 파라미터	값
형태소 품사 표상 차원 수	128
RoBERTa large 차원 수	1,024
bi-LSTM 차원 수	1,024
인코더 은닉층 차원 수	1,024
디코더 은닉층 차원 수	1,024
배치 크기(batch size)	64
학습률(learning rate)	0.0005
multi-head attention의 헤드 수	8
드랍 아웃(drop out)	0.1

본 논문은 실험 및 평가를 위해 KLUE 데이터 셋[10]을 사용한다. KLUE 데이터 셋은 Wikitree와 Airbnb에서 추출한 문장들로 구성되었으며, Wikitree의 문장들은 사전 검수를 거쳐 문법 오류가 적은 반면, Airbnb의 문장들은 사전 검수를 거치지 않아 상대적으로 구어체에 가깝다. Wikitree는 7,250 문장, Airbnb는 7,250 문장, 총 14,500 문장으로 이루어져 있으며, 이중 10,000 문장이 훈련(train) 집합, 2,000 문장이 개발(development) 집합, 2,500 문장이 평가(test) 집합에 해당한다. 어절 표상에 사용한 사전 학습 언어 모델은 KLUE 데이터 셋과 함께 공개된 KLUE RoBERTa-large 모델이다. 평가 척도는 Unlabeled Attachment Score(UAS), Labeled Attachment Score(LAS)를 사용하였고, 모델의 하이퍼 파라미터는 표 1과 같다.

4.2 실험 결과

표 2. attention 기법의 종류에 따른 성능

attention 기법의 종류	UAS	LAS
Scaled dot-product	94.87	92.09
Multi-head	94.83	**92.85**
Biaffine	**95.14**	92.68

표 2는 attention 기법의 종류에 따른 모델의 성능 측정 지표이다. biaffine attention이 scaled dot-product attention과 비교하여 UAS가 약 0.27%, multi-head attention과 비교하여 약 0.31% 높은 성능을 보인다. 따라서 포인터 네트워크 모델에 biaffine attention을 적용하는 것이 어절의 지배소를 결정하는데 가장 효과적인 것으로 보인다. 어절과 지배소 사이의 의존 관계를 인식하는 과정에서는 multi-head attention이 가장 높은 성능을 보이며, scaled dot-product attention과는 LAS가 약 0.76%, biaffine attention과는 약 0.17%의 차이를 보인다. 따라서 의존 관계 인식을 위해서는 다양한 시각에서 정보를 수집하는 방식의 multi-head attention이 가장 효과적인 것으로 판단된다.

표 3. 구문 분석 모델에 따른 성능

구문 분석 모델	UAS	LAS
KLUE-BERT-base	89.96	88.05
KLUE-RoBERTa-small	90.04	88.14
KLUE-RoBERTa-base	93.04	88.32
KLUE-RoBERTa-large	93.48	88.36
Our Model	**95.14**	**92.68**

표 3은 구문 분석 모델에 따른 성능 측정 지표이다. 본문에서 제안하는 biaffine attention을 적용한 포인터 네트워크 기반 의존 구문 분석 모델이 가장 높은 성능을 보인다. KLUE-RoBERTa-large 모델은 의존 구문 분석을 sequence tagging 문제로 보고, biaffine attention을 통한 sequence labeling으로 의존 구문 분석을 수행했다. 제안 모델은 KLUE-RoBERTa-large 모델과 비교하여 UAS가 약 1.66%, LAS가 약 4.32% 높은 성능을 보인다.

5. 결론

본 논문에서는 포인터 네트워크 기반 한국어 의존 구문 분석 연구에 주로 사용된 attention 기법을 비교 분석하였다. 포인터 네트워크를 활용하는 가장 기본적인 모델 구조를 사용하고, attention 기법의 종류만 변경하며 비교 실험을 진행하였다. 이 단계에서 scaled

dot-product attention, multi-head attention, biaffine attention을 교차 적용하며 성능을 비교하였다. 실험 결과, 지배소 결정에는 biaffine attention이, 의존 관계 인식에는 multi-head attention이 가장 좋은 성능을 보였다. 향후 연구로는 지배소와 의존 관계 결정 단계를 독립적인 두 과정으로 분리하고, 각 과정에 특화된 attention 기법을 분석해보고자 한다.

감사의 글

이 논문은 2020년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No. 2013-0-00131, (엑소브레인-총괄/1세부) 휴먼 지식증강 서비스를 위한 지능진화형 WiseQA 플랫폼 기술 개발). 또한 본 연구는 과학기술정보통신부 및 정보통신기획평가원의 대학ICT연구센터육성지원사업의 연구결과로 수행되었음(IITP-2021-2016-0-00465)

참고문헌

[1] 박천음, 이창기, “멀티 태스크 학습 기반 포인터 네트워크를 이용한 한국어 의존 구문 분석”, 한국정보과학회 학술발표 논문집, pp.440-442, 2021.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, “BERT:Pre-training of Deep Bidirectional Transformers for Language Understanding”, arXiv:1810.04805, 2018.

[3] Oriol Vinyals, Meire Fortunato, Navdeep Jaitly, “Pointer Networks”, Neural Information Processing Systems(NIPS), pp.2692-2700, 2015.

[4] 임준호, 김현기, “사전학습 언어모델의 토큰 단위 문맥 표현을 이용한 한국어 의존 구문분석”, 정보과학회논문지 제48권 제1호, pp.27-34, 2021.

[5] 박성식, 오신혁, 김홍진, 김시형, 김학수, “ELMo와 멀티헤드 어텐션을 이용한 한국어 의존 구문 분석”, 제30회 한글 및 한국어 정보처리 학술대회 논문집, pp.682-684, 2018.

[6] 박천음, 이창기, 임준호, 김현기, “BERT를 이용한 한국어 의존 구문 분석”, 한국정보과학회 학술발표 논문집, pp.530-532, 2019.

[7] 김봉수, 황태선, 김정욱, 이새벽, “사전 학습 모델과 Specific-Abstraction 인코더를 사용한 한국어 의존 구문 분석”, 제32회 한글 및 한국어 정보처리 학술대회 논문집, pp.98-102, 2020.

[8] Hwijeen Ahn, Minyoung Seo, Chanmin Park, Juae Kim, Jungyun Seo, “Extensive Use of Morpheme Features in Korean Dependency Parsing”, IEEE International Conference on Big Data and Smart Computing (BigComp), 2019.

[9] 이진우, 최맹식, 이충희, 이연수, “의존 구문 분석에 손실 함수가 미치는 영향: 한국어 Left-To-Right Parser를 중심으로”, 제32회 한글 및 한국어 정보처리 학술대회 논문집, pp.93-97, 2020.

[10] Sungjoon Park, Jihyung Moon, Sungdong Kim, Won Ik Cho, Jiyoon Han, Jangwon Park, Chisung Song, Junseong Kim, Yongsook Song, Taehwan Oh, Joohong Lee, Juhyun Oh, Sungwon Lyu, Younghoon Jeong, Inkwon Lee, Sangwoo Seo, Dongjun Lee, Hyunwoo Kim, Myeonghwa Lee, Seongbo Jang, Seungwon Do, Sunkyoung Kim, Kyungtae Lim, Jongwon Lee, Kyumin Park, Jamin Shin, Seonghyun Kim, Lucy Park, Alice Oh, Jung-Woo Ha, Kyunghyun Cho, “KLUE: Korean Language Understanding Evaluation”, arXiv:2105.09680, 2021.