의존 경로와 음절단위 의존 관계명 분포 기반의 Bidirectional LSTM CRFs를 이용한 한국어 의존 관계명 레이블링

안재현○, 이호경, 고영중
동아대학교 컴퓨터공학과
{anjaehyun17, hogay88, youngjoong.ko}@gmail.com

Korean Dependency Relation Labeling Using Bidirectional LSTM CRFs Based on the Dependency Path and the Dependency Relation Label Distribution of Syllables

Jaehyun An○, Hokyung Lee, Youngjoong Ko
Donga University, Department of Computer Engineering

요 약

본 논문은 문장에서의 어절 간 의존관계가 성립될 때 의존소와 지배소가 어떠한 관계를 가지는지 의존 관계명을 부착하는 모델을 제안한다. 국내에서 한국어 의존구문분석에 관한 연구가 활발히 진행되고 있지만 의존 관계만을 결과로 제시하고 의존 관계명을 제공하지 않는 경우가 많았다. 따라서 본 논문에서는 의존 경로(Dependency Path)와 음절의 의존 관계명 분포를 반영하는 음절 임베딩를 이용한 의존 관계명 부착 모델을 제안한다. 문장에서 나올 수 있는 최적의 입력 열인 의존 경로(Dependency Path)를 순차 레이블링에서 좋은 성능을 나타내고 있는 bidirectional LSTM-CRFs의 입력 값으로 사용하여 의존 관계명을 결정한다. 제안된 기법은 자질에 대한 많은 노력 없이 의존 경로에 따라 어절 및 음절 단어표상(word embedding)만을 사용하여 순차적으로 의존 관계명을 부착한다. 의존 경로를 사용하지 않고 전체 문장의 어절 순서를 바탕으로 자질을 추출하여 CRFs로 분석한 기존 모델보다 의존 경로를 사용했을 때 4.1%p의 성능향상을 얻었으며, 의존 관계명 분포를 반영하는 음절 임베딩을 사용한 bidirectional LSTM-CRFs는 의존 관계명 부착에 최고의 성능인 96.01%(5.21%p 개선)를 내었다.

주제어: 의존 파싱, 한국어 구문분석, 의존관계명, 딥러닝

1. 서론

의존구문분석(Dependency Parsing)은 문장에서 어절 간의 구조를 찾아내는 것을 말하며, 모든 어절은 지배소 혹은 의존소가 될 수 있고 어절 사이 관계를 파악하는 것을 말한다. 의존구문분석(Dependency Parsing)의 연구는 전이기반(Transition Based)[1] 방식과 그래프 기반 (Graph Based) [2]방식이 존재하며, 국내에서 적용한 연구는 전이기반 방식과 딥러닝을 활용한 연구[3]이 있고, 그래프 기반과 온라인 학습을 이용한 연구[4,5]가 있다.

본 논문은 한국어 의존구문 분석 이후 의존소와 지배소의 관계명을 부착하는 연구이다. 의존관계가 주어의 관계인지 목적어의 관계인지 등을 부착하여, SRL(Semantic Role Labeling)에서 분석된 의존 관계명(Dependency relation label)[6]을 사용하여 성능을 개선하는 등, 의존관계와 의존 관계명은 전반적인 자연어 처리(Natural Language Processing) 분야에서 중요한 정보로 활용되고 있다.

기존 의존 관계와 의존 관계명 부착[3,7]에 대한 연구에서 이전 의존관계의 정보가 현재 의존 관계명을 부착하는데 중요한 자질로 사용되고 있다. 그러나 기존의 연구에서 의존 관계명을 부착하는데 입력된 어절의 순서대로 넣기 때문에 이전 의존관계의 정보를 정확하게 사용하지 못하는 단점이 있었다[7]. 앞서 말한 순차적인 의존관계의 정보를 잘 활용하기 위해서 본 논문에서는 최적의 입력 열인 의존경로(Dependency Path)를 추출하여 의존 관계명을 분석하는 기법에 대해서 연구한다. 의존경로는 의존 구문트리에서 최하위 잎 노드(의존소)에서 부모노

이 논문은 2016년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(No. NRF-2015R1D1A1A01056907)

드(지배소)를 방문하여 최상위 Root까지 노드들의 입력 열을 의미한다. 본 연구에서 제안하는 의존 경로를 CRFs으로 학습하여 기존 연구[7]에서 제안한 문장의 입력 어절 순서를 사용하는 시스템의 성능보다 4.1%p 높은 성능을 획득하였다.

본 논문에서는 주어, 목적어, 보어 등의 의존 관계명 부착을 위해 Bidirectional Long Short Term Memory Conditional Random Fields(bi-LSTM-CRFs)를 사용한다. 그리고 의존소와 지배소의 관계를 표현하기 위해 의존 구문트리를 구축하고, 의존 구문트리에서의 의존 경로를 추출하여 최적의 입력 열로 표현하였다. bi-LSTM-CRFs의 입력 값으로 의존 경로로 표현된 어절을 사용하며 경로 상에 있는 어절의 단어표상(word embedding), 품사 출현 벡터와 음절 기반 단어벡터를 사용하였다. 품사 출현 벡터는 어절에서 출현한 모든 형태소들의 품사를 출현벡터 표현하였다. 음절기반 임베딩 벡터는 대량의 말뭉치에서 한 음절이 출현하는 관계명 분포를 음절 출현 분포를 구하여 활용한다. 본 연구에서는 말뭉치에서 미리 계산한 음절 당 관계명 분포를 구하고 의존 경로에서의 어절을 음절 단위로 나누어 미리 구해진 분포를 적용하여 LSTM의 입력으로 사용하여 구해진 단어벡터를 추가적인 자질로 사용하여, 어절의 단어 표상과 품사 열을 사용한 기본 bi -LSTM-CRFs모델보다 개선된 성능을 보였다.

제안된 기법은 실험을 통해서 bi-LSTM-CRFs, 어절 단어표상, 어절의 품사 출현 벡터, 음절 기반 단어벡터를 활용하여 96.01%의 의존 관계명 부착 정확도를 가지는 모델을 구축할 수 있었다.

논문의 구성은 다음과 같다. 2장에서는 관련된 연구에 대하여 소개한다. 3장에서는 본 논문에서 제안하는 bi-LSTM-CRFs에 대해 설명을 하고, 4장에서는 실험방법 및 그 결과를 살펴보고 5장에서 결론으로 끝맺는다.

2. 관련 연구

자연어에 대한 의존구조 파싱 기술과 관련하여 McDonald가 제안한 그래프기반 의존파싱이 있고, Nivre가 제안한 전이기반 의존파싱이 있다.

McDonald가 제안한 알고리즘은 일정한 자질 집합을 정의하고 각 어절의 의존관계마다 그 자질 집합을 만든다. 그 후, 생성된 모든 가능한 의존 관계를 이용하여 간선을 포함하는 그래프를 만들고 그 안에서 가장 점수가 높은 최대 신장 트리를 이용하여 파스트리를 결정하는 것이다[2,4-5]. 이러한 방식을 그래프 기반 모델이라고 불리는데 전역적 학습 모델로써 $O(n^2,n^3)$의 시간 복잡도를 가진다. 반면 Nivre가 제안한 알고리즘은 가능한 의존 구문트리의 일부에서만 탐색을 진행하여 전이 과정에서 어떠한 행위(action)을 취할지 결정을 내리는 메카니즘이 필요하며 주로 Structural Perceptron을 이용한 Beam Search 방식을 사용한다[3,8-9].전이 기반 방식은 지역적 학습 모델로써 $O(n)$의 시간 복잡도를 가진다.

최근 국내 연구에서는 전이 기반 알고리즘과 단어표상(Word Embedding), 딥러닝을 이용한 의존 구문분석[3]으로 의존파서의 성능을 개선하였고, Stack LSTM(Long Short Term Memory)을 이용한 의존구문분석[10]에서 LAS(Labeled Attachment Score)성능이 개선되었다. 그리고 의존 관계명 부착을 위한 연구[7]는 CRFs를 활용한 연구로 의존소와 지배소의 내용어, 기능어의 어휘정보와 품사를 자질로 부착한 연구가 있다.

3. 제안 방법

3.1 bi-LSTM-CRFs

RNN(Recurrent Neural Networks)[11]은 순환신경망으로 입력 층, 은닉 층, 출력 층을 거치는 일반적인 신경망과 달리 현재 은닉 층의 결과가 다시 은닉 층의 입력으로 들어가는 신경망이다.

LSTM(Long Short Term Memory)은 순차데이터의 처리가 유리한 RNN의 경사도가 소실(Vanishing) 혹은 발산(Exploding)되는 문제점을 보완한 모델이다. LSTM에서는 오류역전파(Backpropagation) 과정에서 오류의 값이 소실, 발산 하지 않고 잘 유지되는데 LSTM의 게이트가 부착된 셀의 기능으로 정보를 가져올지, 상태를 다음 노드로 전이할지 혹은 유지할지를 결정한다.

bi-LSTM-CRFs[12]는 forward로 전파하는 LSTM에서 Backward로 전파하는 LSTM을 붙여 전체 입력된 데이터의 정보를 이용할 수 있다.

아래 그림 1은 bi-LSTM-CRFs와 입력 형태, 출력 형태에 대한 도식화한 그림이다.

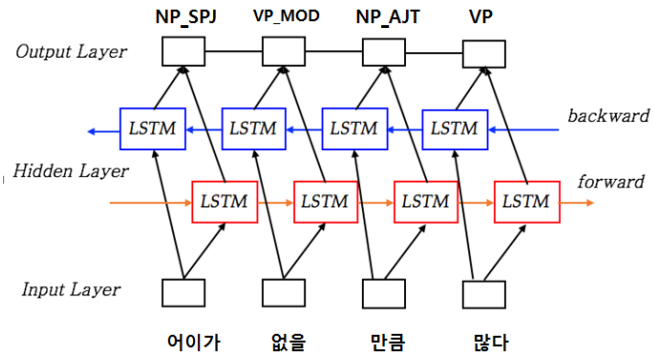

그림 1. bi-LSTM-CRFs

입력의 형태는 어절과 품사 출현 벡터, 음절 기반 임베딩 벡터를 사용하였는데, 의존소 어절을 표현하기 위해 어절 임베딩 벡터를 사용하였다. 어절 임베딩 벡터는 대량의 말뭉치에서 어절 기준으로 Word2Vec의 CBOW[13]모델을 사용하여 구하였다. 의존소의 품사 열은 품사 출현벡터의 형태로 표현하였다. 품

사 출현벡터는 품사의 출현을 벡터로 표현하는 방법이다. 예를 들어 어절에서 출현한 모든 품사가 "NNG, JKS" 인 품사 열이 있을 때 아래 표 1과 같이 품사 출현벡터를 추출할 수 있다.

표 1. 품사 빈도 벡터(46차원)의 예

품사	NNG	...	NNP	JKS
빈도	1	...	0	1

음절 기반의 임베딩 벡터는 해당 음절에 대한 모든 관계명의 분포로 입력된 어절을 음절 단위로 나누어 각각의 벡터를 LSTM의 입력으로 사용하여 음절 기반 임베딩 벡터로 표현했다. 아래 표 2는 제안하는 모델에서 사용한 자질이다.

표 2. 의존 관계명 부착 시스템의 자질

자질(의존 경로)	
자질 1	의존소의 어절 단어표상(Word embedding)
자질 2	의존소의 모든 품사(품사 출현 벡터)
자질 3	의존소 음절 기반 임베딩 벡터

3.2. 의존 경로(Dependency Path)

본 논문에서 제안하는 의존 경로(Dependency Path)는 어절 사이 의존 관계가 결정 되었을 때 의존 구문트리로 표현하여 최하위 잎노드(의존소)가 부모노드(지배소)를 방문하여 최상위 Root까지 노드들의 최적의 입력 열(Sequence)을 의미한다. 한 의존 구문트리에서 추출 할 수 있는 모든 의존 경로를 그림으로 도식화하면 아래 그림 2와 같다.

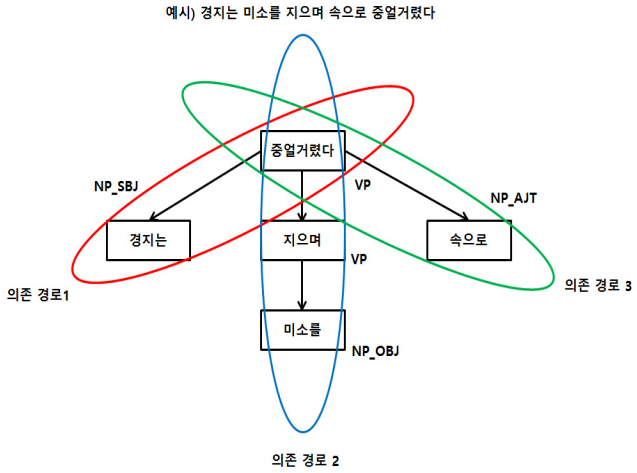

그림 2 의존 경로 예시

의존 경로를 추출하면 의존 경로를 순차 레이블링 문제라고 볼 수 있고, 현재 관계명을 부착할 때 이전의 의존관계 자질들을 추가로 활용할 수 있다는 장점이 있다. 최적의 입력 열을 추출하기 위해 의존 경로를 추출해야 한다. 본 연구에서는 의존 구문트리에서 경로가 다른 모든 의존경로를 추출하여 모든 의존 경로에 대하여 학습을 하였다. 이와 같은 방식으로 학습을 하면 중복된 의존관계가 많이 포함되기 때문에 의존 관계명에 대한 평가 방법이 정확하다고 볼 수 없다. 평가의 정확성을 위해 모든 의존 트리에서 중복된 의존 관계를 제거하여 성능을 평가 했다.

기존의 의존 관계명 부착은 의존 구문트리에서 모든 어절을 순서대로 의존소와 지배소의 자질을 추출하여 의존 관계명을 부착하는 방법[7]이다. 예를 들어 "경지는 미소를 지으며 속으로 중얼거렸다" 라는 문장이 있을 때 아래의 표 3과 같이 의존관계가 추출된다.

표 3. 의존관계 예시

	의존소	지배소	의존관계명
의존관계1	경지는	중얼거렸다	NP_SBJ
의존관계2	미소를	지으며	NP_OBJ
의존관계3	지으며	중얼거렸다	VP
의존관계4	속으로	중얼거렸다	NP_AJT
의존관계5	중얼거렸다	ROOT	VP

이와 같이 의존관계가 추출 되었을 때 [7]의 연구에서는 CRFs를 활용하여 의존 관계명을 부착하였다. 어절의 순서대로 모델을 학습하였고, 현재의 의존 관계에서 의존소와 지배소의 자질을 추출하여 학습하므로, 최적의 입력 열을 활용하지 않아 오류를 포함하고 있다. 본 연구에서는 의존 관계명을 부착하기 위해 이전 의존관계의 정보가 중요한 자질로 사용된다. 그러므로 본 연구에서 의존경로를 제안한다. 순차 레이블링 문제임을 입증하기 위해 다음과 같은 실험을 진행하였다. 첫 번째는 다중 클래스 분류 문제와 순차 레이블링 문제를 비교하기 위해 먼저 다중 클래스 분류로 가정하고 실험을 진행하였다. 실험을 위해 SVM(Support Vector machine)을 사용하였다. SVM의 입력으로 의존소와 지배소의 어휘와 품사를 사용하였고, [7]의 성능과 유사하였으나 조금 낮은 성능을 보였다. 두 번째는 순차 레이블링 문제로 가정하고 모든 의존관계를 의존 구문 트리로 표현하여 최적의 입력 열인 의존경로를 추출하였다. 의존경로에서 어절들을 CRFs로 학습한 결과 성능이 다중 클래스 분류(SVM)에서 보다 6.18%p 개선되었고 최적의 입력 열을 사용하지 않은 [7]의 연구보다 성능 면에서 4.1%p 개선되었다. 그리하여 다중 클래스 분류문제라기 보다 순차 레이블링 문제라고 보는 것이 타당하다는 결론을 맺었다.

본 논문에서는 의존 경로의 어절을 bi-LSTM-CRFs의 입력으로 사용하였다. 예시문장으로 “경지는 미소를 지으며 속으로 중얼거렸다” 이며, 트리의 잎(leaf)노드의 수만큼 경로가 생성되고, 자식노드는 의존소, 부모노드는 지배소가 되어 최적의 입력 열이 되고 각 어절마다 의존 관계명을 부착한다. 아래 표 4는 문장에서의 추출할 수 있는 모든 의존경로이다.

표 4. 문장에서 추출할 수 있는 의존경로

	모든 의존 경로
경로 1	경지는 -> 중얼거렸다
경로 2	미소를 -> 지으며 -> 중얼거렸다
경로 3	속으로 -> 중얼거렸다

3.3 음절 기반 임베딩 벡터

음절 기반 임베딩 벡터란 학습 코퍼스에서 출현한 어절을 음절 단위로 잘라 해당 음절이 어떤 관계명으로 많이 사용되었는지 분포 값을 미리 구해 어절을 표현하는 것을 말한다. 아래 그림 3은 음절 기반 임베딩 구하는 bi-LSTM-CRFs을 도식화 한 것이다.

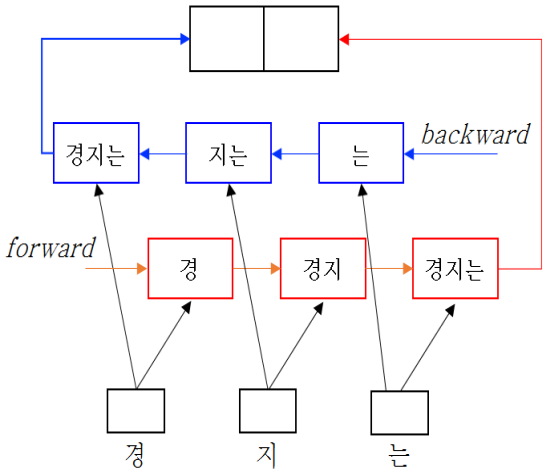

그림 3. 음절 기반 임베딩 벡터 bi-LSTM-CRFs

음절 당 분포의 값의 예시는 표5와 같다. 음절 당 분포의 벡터는 “경지는” 이라는 어절이 NP-SBJ라는 의존관계명과 NP_OBJ라는 관계관계명이 존재했을 때 어절을 음절 단위로 나누어 “경” 이라는 음절은 B-NP-SBJ, B-NP-OBJ이고 “지” 는 I-NP-SBJ, B-NP-OBJ “는” 은 I-NP-SBJ, I-NP-OBJ이다. B(단어의 시작)과 I(단어의 시작이 아닌)로 의존 관계명이 36개에서 총 72차원의 의존관계명의 분포를 구하였다. 다음 표 5는 음절 당 관계명의 분포의 예이다.

표 5. “경지는” 어절의 음절 당 관계명 분포의 예

음절	분포 벡터 차원
경	[0.1124.. 0.0024... 0.0135.. 0.0201...]
지	[0.0854... 0.0135... 0.0043 0.0109...]
는	[0.0132... 0.0043... 0.0135... 0.0020]

본 논문에서의 음절 기반 임베딩 벡터는 사전학습(pretraining)된 어절이 bi-LSTM-CRFs의 입력으로 들어가기 전 어절을 음절 단위로 잘라 각 음절의 관계명 분포 값을 LSTM(Long Short Term Memory)의 입력으로 넣어 어절을 표현한다.

4. 실험

4.1 실험 데이터

세종 구구조 말뭉치에서 의존구조로 변형된 코퍼스를 사용하였고 전체 데이터에서 23,002개의 문장을 임의로 추출하여 18,403개는 학습데이터로, 4,599개의 문장은 평가 데이터로 사용하였고, 개발 데이터로 학습 데이터에서 1,300문장을 임의로 추출하여 사용하였다.

4.2 다중 클래스 분류와 순차 레이블링

앞서 제안한 의존 경로(Dependency Path)에서 의존소와 지배소의 의존 관계명을 부착하기 위해 순차 레이블링 문제와 의존 경로의 타당성을 증명하기 위한 실험이다. [7]의 경우 의존 관계명을 부착하기 위해서 세종계획 구구조 말뭉치를 의존구조 말뭉치로 변경하여 사용하였고 자질은 의존 관계에서의 의존소, 지배소의 기능어, 내용어 어휘 및 품사를 추출하여 자질로 사용하였다. 입력된 어절의 순서대로 CRFs의 입력으로 사용했기에 오류를 포함하고 이전의 의존 관계에 대한 정보를 사용하지 않기 때문에 의존 관계명 부착은 순차 레이블링 문제로써 타당성을 검증하는 실험을 진행하였다. 본 실험에서 SVM은 다중 클래스 분류를 위해서 사용되었고, CRFs는 순차 레이블링을 위해서 사용되었다.

그리하여 SVM 모델의 경우 의존소와 지배소의 모든 어휘와 모든 품사를 20차원벡터로 표현하여 학습을 하였다. CRFs 모델의 경우 의존 경로를 이용하여 의존 관계명 부착을 하였다. 의존 경로를 사용할 경우 여러 경로가 존재 할 때 동일한 어절이 포함 되어 있으므로 중복을 제거하여 원래의 데이터로 재구축을 하는 작업이 필요하다.

표 6 에서의 성능은 재구축을 하여 중복을 제거한 성능이다.

표 6. 실험 1의 성능 비교

Model	Micro-F1
[가] [10]정석원	90.80%
[나] SVM(의존소,지배소 어휘,품사)	88.72%
[다] CRFs(의존 경로,어절,품사)	94.90%

모델[가], [나]와 모델[다]의 성능을 비교 하였을 때 의존 관계명은 순차 레이블링을 적용했을 때 더 높은 성능을 얻을 수 있음을 확인할 수 있었다. 결과적으로 본 논문에서 제안한 최적의 입력 열인 의존 경로를 사용하는 것이 가장 높은 성능을 나타낸다.

4.3 bi-LSTM-CRFs를 이용한 의존 경로에 대한 실험

bi-LSTM-CRFs와 의존 경로를 사용하여 의존 관계명을 부착한 실험의 결과가 표 7에서 확인할 수 있다. 의존 경로를 사용한 실험에서는 여러 경로가 존재 할 때 동일한 의존관계가 여러 번 포함되어 있으므로 중복을 제거, 재구축 하여 성능을 내었다.

표 7. bi-LSTM-CRFs를 이용한 의존 경로를 적용한 모델

Model	Micro-F1
[라] bi-LSTM-CRFs (어절 임베딩 + 품사 빈도벡터)	95.16%
[마] bi-LSTM-CRFs (어절 임베딩 + 품사 빈도벡터 + 음절 임베딩벡터)	**96.01%**

그림 4는 [가],[라],[마]에 따른 F1 성능 비교를 나타낸다. 기존 연구[가]의 성능 보다 의존 경로와 bi-LSTM-CRFs를 사용한 [라]에서 4.6%p 향상되었고, [마]에서 의존 경로와 bi-LSTM-CRFs, 음절기반 임베딩을 사용하였을 때 5.21%p 향상된 성능을 보였다.

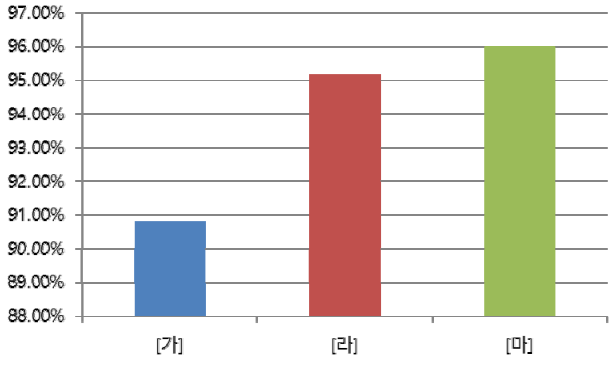

그림 4 실험 별 F1 성능 비교

결과적으로 의존 구문트리에서 추출한 최적의 입력 열인 의존 경로를 사용하여 의존 관계명을 부착하는 것이 기존의 연구와 같이 어절을 입력 순서와 동일하게 학습하는 것 보다 개선된 성능을 보였고, 음절 임베딩 벡터를 활용하는 bi-LSTM-CRFs 모델이 성능 면에서 개선되었다. 그리고 어절의 단어 표상과 품사 출현 벡터만을 사용하는 방법보다 추가적으로 음절에 대한 의존 관계명의 분포를 사용한 경우가 해당 어절의 의존 관계를 잘 분석하기 위해서 좋은 자질임을 알 수 있다.

5. 결론

기존 연구에서는 의존 관계명 부착을 위해 CRFs와 SVM 등을 활용하였다. 기존의 연구는 이전 의존 관계의 자질을 현재의 의존 관계명 부착에 활용하지 않는 문제가 있다. 그러나 최적의 입력 열과 CRFs를 활용하여 의존 관계명을 부착하였을 때 기존의 연구인 CRFs를 사용한 실험과 SVM을 사용한 실험의 성능보다 각각 4.1%p, 6.8%p를 향상하였다. 또한, 제안된 의존 경로와 음절 기반 임베딩 벡터를 사용하는 bi-LSTM-CRFs 모델은 최종적으로 기존의 연구보다 5.21%p 높은 성능(96.01%)을 보였다.

참고문헌

[1] Nivre J, “An efficient algorithm for projective dependency parsing”, *In Proceedings of the 8th International Workshop on Parsing Technologies 2003, pp. 149-160,* 2003.

[2] Ryan McDonald, Fernando Pereira, Jan Haji˘c, and Kiril Ribarov, “Non-projective dependency parsing using spanning tree algortihms“, *In Proceedings of NAACL-HLT 2005, pp. 523-530,* 2005.

[3] 이창기, 김준석, 김정희 “딥 러닝을 이용한 한국어 의존 구문 분석”, 한글 및 한국어 정보처리 학술대회, pp.87-94, 2014.

[4] 이용훈, 이종혁, “온라인 학습을 이용한 한국어 의존구문분석”, *한국컴퓨터종합학술대회 논문집, 37권, 1호, pp.299-304,* 2010.

[5] 김성진, 옥철영, “한국어 의존관계 분석과 자질 집합 분할을 이용한 기계학습의 성능 개선”, *전자공학회논문지 51권, 8호, pp.66-74,* 2014.

[6] Roth, Michael, Mirella Lapata, “Neural Semantic Role Labeling with Dependency Path Embeddings”, *arXiv preprint arXiv:1605.07515,* 2016.

[7] 정석원, 최맹식, 김학수, “CRFs를 이용한 의존구조 구문 레이블링”, *한글 및 한국어 정보처리 학술대회, pp.137-138,* 2013.

[8] M.Collins, B.Roark, “Incremental Parsing with the Perceptron Algorithm”, *In Proceedings of Association for Computational Linguistics 2004, pp. 111-118,* 2004.
[9] L. Huang, K, Sagae, “Dynamic Programming for Linear-Time Incremental Parsing”, *In Proceedings of Empirical Methods in Natural Language Processing 2013, pp.647-657,* 2013.
[10] 이건일, 이종혁, 순환 신경망을 이용한 전이 기반 한국어 의존 구문 분석”, *정보과학회 컴퓨팅의 실제 논문지, 21권, 8호, pp.567-571,* 2015.
[11] James Martens, Ilya Sutskever, “Learning Recurrent Neural Networks with Hessian-Free Optimization”, *In Proceedings of International Conference on Machine Learning 2011, pp. 1033-1040,* 2011.
[12] 이창기, “Long Short-Term Memory 기반의 Recurrent Neural Network를 이용한 개체명 인식”, *한국컴퓨터종합학술대회 논문집, pp.645-647,* 2015.
[13] word2vec, [Online]. Available: http://code.google.com/archive/p/word2vec/