

---

# Easy-First Deep Biaffine Attention을 이용한 한국어 의존 파싱

---

2019.10.12

홍승연<sup>1</sup>, 나승훈<sup>1</sup>, 신종훈<sup>2</sup>, 김영길<sup>2</sup>

<sup>1</sup>전북대학교, <sup>2</sup>ETRI

# Contents

---

- Background
- 관련 연구
- Easy-First Deep Biaffine Attention Parsing
- 결론



# Dependency Parsing(의존파싱)

- 의존 파싱은 파싱(구문분석)의 한 갈래로 단어와 단어 사이의 의존 관계에 따라 문장의 구조를 구성해나가는 방식

## 전이 기반 방식

버퍼와 스택으로부터 자질을 추출한 후 모델을 적용하여 다음 전이 액션을 결정한 후 버퍼와 스택을 갱신해나가는 방식

지역적인 모델



## 그래프 기반 방식

가능한 모든 단어 쌍의 의존관계를 찾아서 점수화하여 가장 높은 점수를 갖는 의존 트리를 찾는 방식

전역적인 모델

# 관련 연구

---

## 영문

### **Arc-swift: A Novel Transition System [Peng Qi, 2017]**

- 전이 기반 모델. arc-eager, arc-standard 이외에 새롭게 제안한 전이 시스템
- PTB 데이터 셋에서 UAS : 94.3%, LAS : 92.2% 의 성능을 보임

### **Deep Biaffine Attention for Neural Dependency Parsing[T. Dozat, 2017]**

- 그래프 기반 모델. Head와 Dep에 대한 표상에 Biaffine Attention 수행
- UAS : 95.74%, LAS : 94.08% [PTB 데이터 셋].

### **Stack-Pointer Networks for Dependency Parsing[X. Ma, 2018]**

- 전이 기반 모델. Pointer Network에 내부 Stack을 만들어 의존소의 위치를 얻어 의존 트리 구성.
- UAS : 95.87%, LAS : 94.19% [PTB 데이터 셋]. 영문 최고 성능



# 관련 연구

---

## 국내

### **Deep Biaffine Attention을 이용한 한국어 의존 파싱[나승훈, 이건일, 2017]**

- Biaffine Attention Parsing[T. Dozat, 2017]을 한국어에 맞게 확장한 모델
- UAS : 91.78%, LAS : 89.76% [세종 데이터 셋]

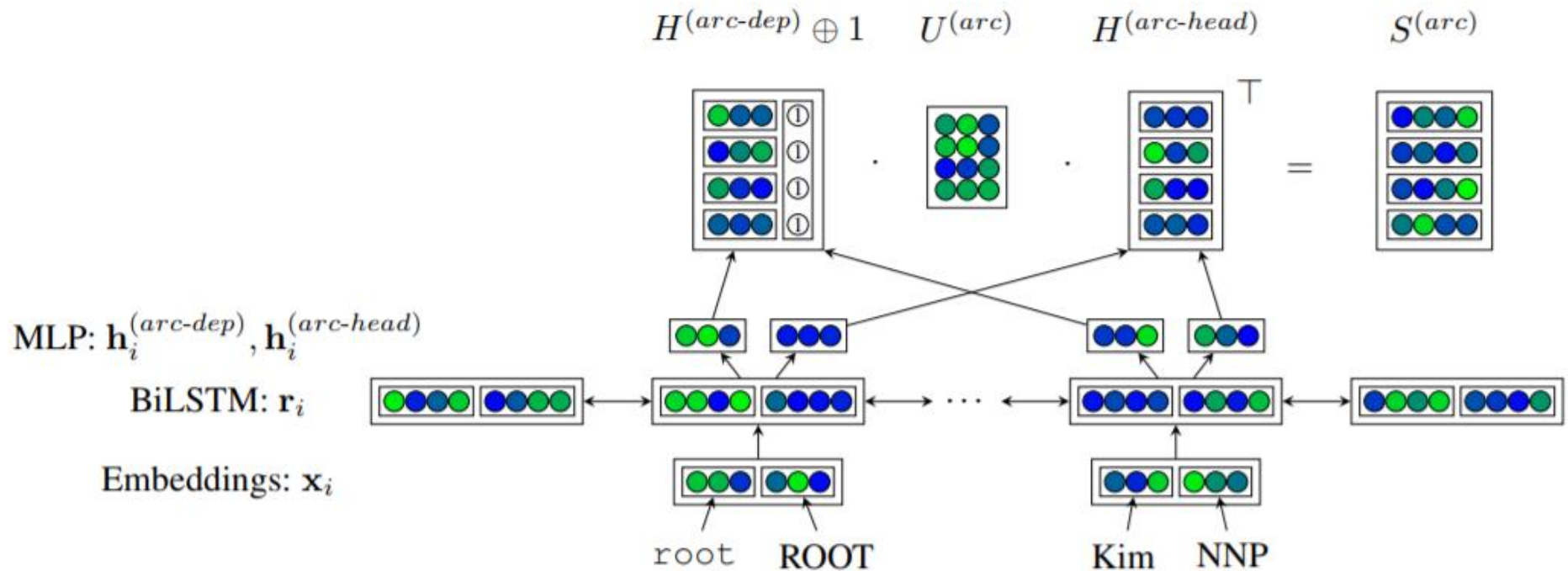
### **포인터 네트워크를 이용한 한국어 의존 구문 분석[박천음, 이창기, 2017]**

- 주의집중 메커니즘을 통해 입력열의 위치를 출력하는 모델
- UAS : 92.16%, LAS : 89.88% (+ Stacked Pointer Network) [세종 데이터 셋]



# Deep Biaffine Attention for Neural Dependency Parsing

[T.Dozat, ICLR '17]



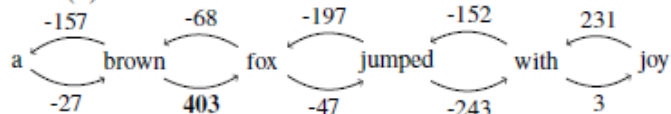
$$S_i^{(arc)} = h_i^{(arc-dep)} U h^{(arc-head)} + w^T h^{(arc-head)}$$

- 그래프 기반 방식
- 지배소 표상과 의존소 표상을 같은 표상으로부터 각기 다른 MLP를 적용하여 얻고 Biaffine Attention을 이용하여 모든 단어 쌍에 대한 점수를 계산

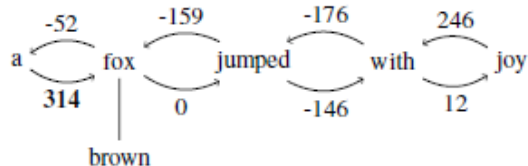


# An Efficient Algorithm for Easy-First Non-Directional Dependency Parsing[Y. Goldberg, ACM'10]

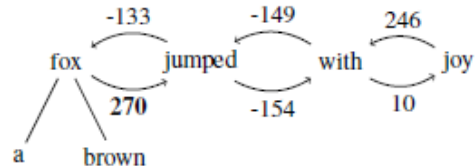
(1) ATTACHRIGHT(2)



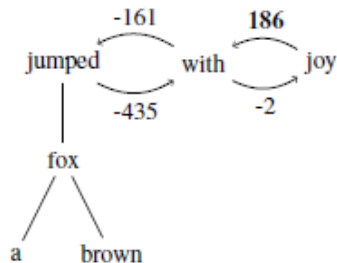
(2) ATTACHRIGHT(1)



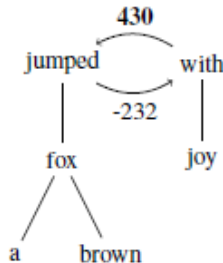
(3) ATTACHRIGHT(1)



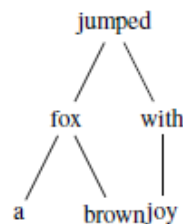
(4) ATTACHLEFT(2)



(5) ATTACHLEFT(1)



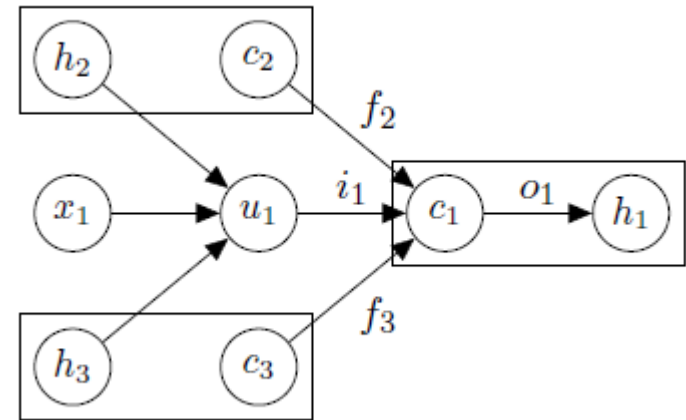
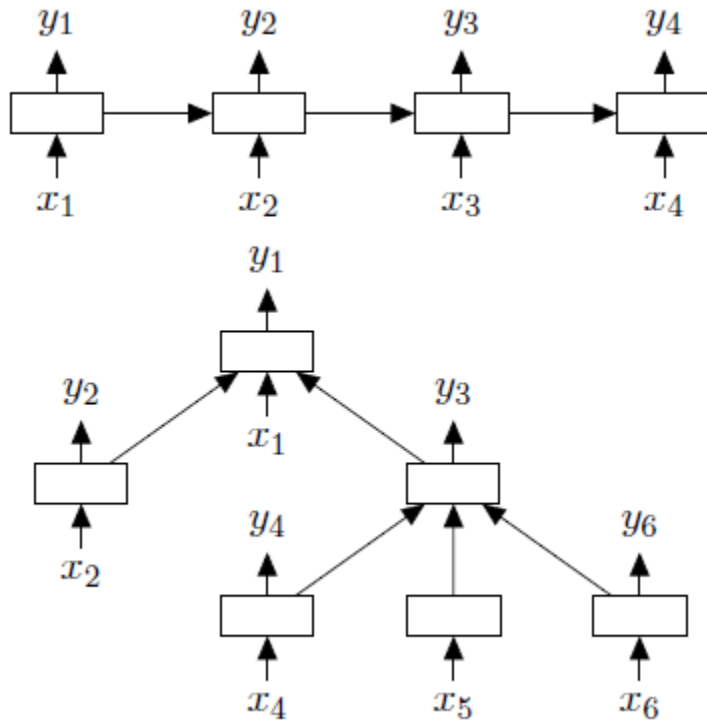
(6)



- 모델이 쉽다고 생각되는 액션부터 수행하고 어려운 액션은 앞의 액션으로부터 생성되는 트리 정보를 이용하여 액션 결정.
- 액션 중에 가장 높은 점수를 가지는 액션을 먼저 수행.



# Improved Semantic Representations From Tree-Structured Long Short-Term Memory Networks[S. Tai, ACL'15]

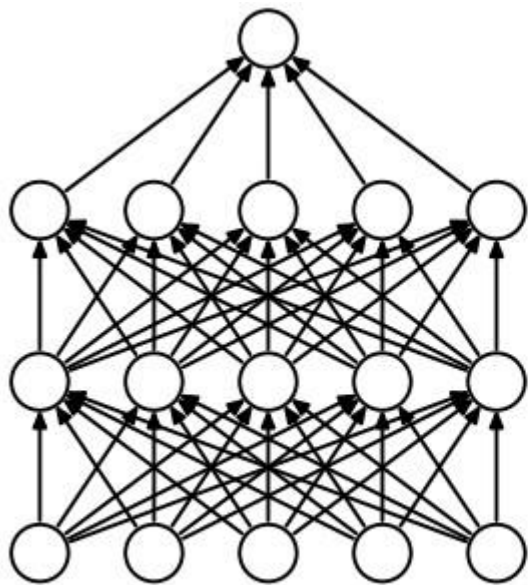


- 기존의 LSTM의 응용으로 자식 노드들의 정보를 활용하여 학습 가능하도록 확장된 LSTM 제안
- 자식 노드의 은닉 표상을 합 정보를 이용하는 Child-Sum Tree-LSTM를 나타냄.

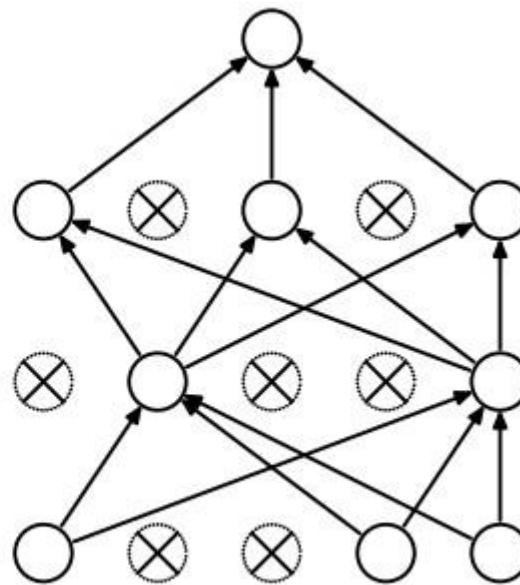




# Dropout as a Bayesian approximation: Representing model uncertainty in deep learning[Y. Gal, JMLR'16]



(a) Standard Neural Net

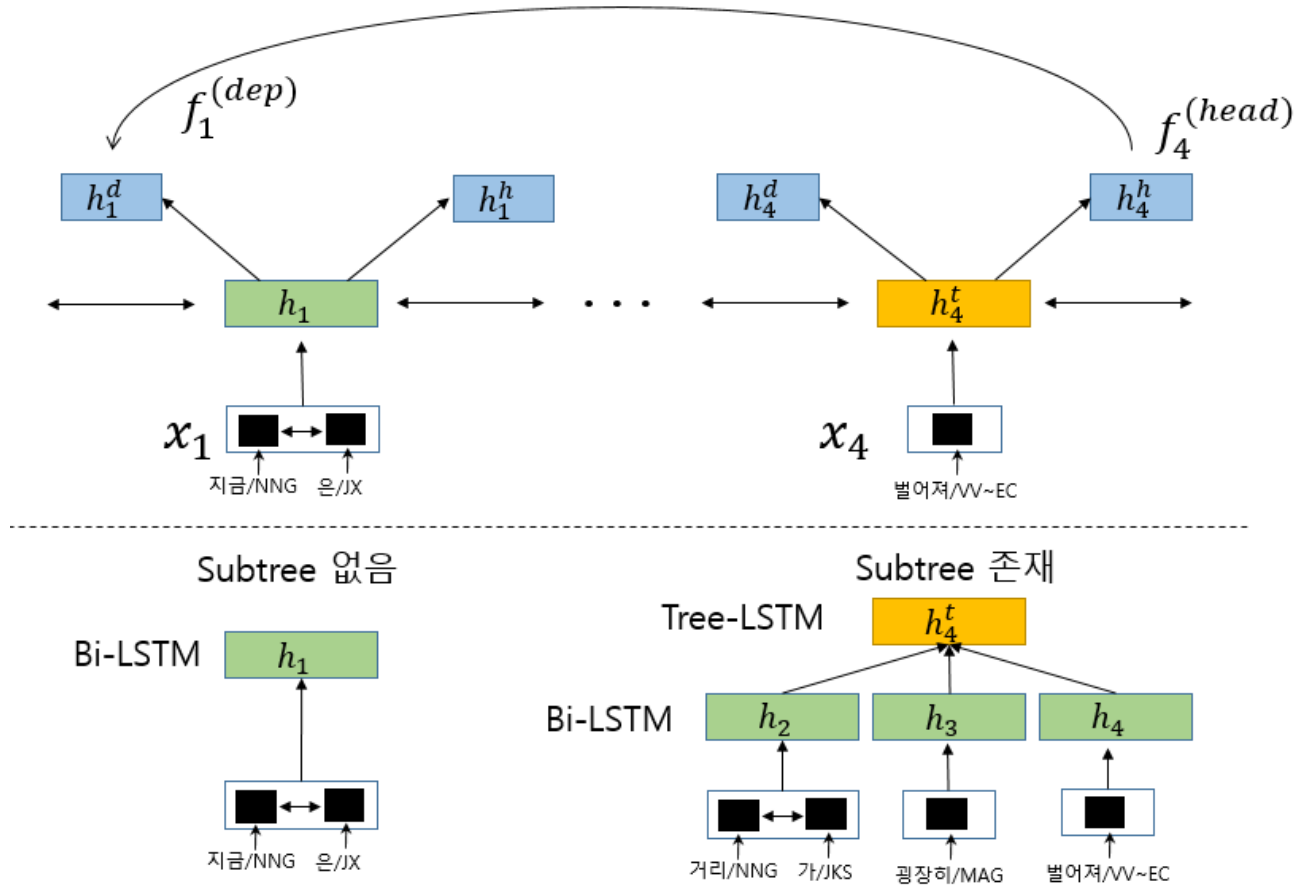


(b) After applying dropout.

- 기존의 딥러닝 모델은 같은 입력이 들어오면 항상 똑같은 출력을 내지만 dropout을 적용하여 불확실성을 부여.
- Dropout은 랜덤하게 노드간 연결을 끊어주기 때문에 확률 모델의 특성을 부여할 수 있다.



# Easy-First Deep Biaffine Attention을 이용한 한국어 의존 파싱

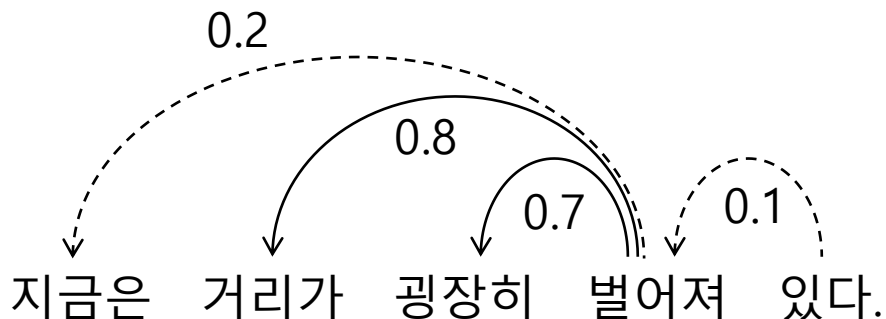


- 사전 학습된 정보를 이용하여 예측하기 쉬운 의존성은 먼저 결정하고 이를 통해 각 단어의 subtree를 구축하여 기존의 그래프 기반 모델이 트리 구조의 특성도 반영할 수 있도록 제안한 모델.



# 예측하기 쉬운 의존성 결정

1. 어텐션 점수 기반하여 쉬운 의존성 판단
  - 사전 학습된 모델에서 어텐션 점수를 추출하여 점수에 따라 높은 점수는 쉬운 의존성, 낮은 점수는 어려운 의존성이라고 판단
2. 확신도에 기반하여 쉬운 의존성 판단
  - 사전 학습된 모델에 MC-Dropout을 적용하여 모델이 불확실성을 갖도록하고 이를 통해 sampling을 통해 의존성 별로 확신도를 측정하여 점수에 따라 높은 점수는 쉬운 의존성, 낮은 점수는 어려운 의존성이라고 판단.
  - 확신도는 sampling 100번 중에 하나의 특정 의존성을 80번 선택했다면 그때 확신도는 80%이다.

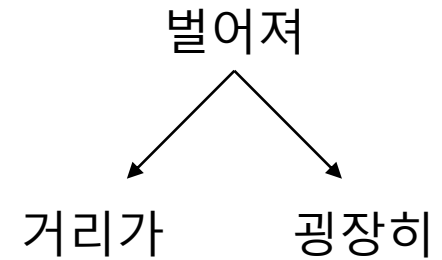


- 실선은 쉬운 의존성, 점선은 어려운 의존성을 나타낸다



## SubTree 구성

지금은 거리가 굉장히 벌어져 있다.



- 쉬운 의존성이 결정 되었을때 결정된 의존성을 통해 트리를 구성
- 위 그림은 하나의 Subtree만 구성 가능



# Easy-First Deep Biaffine Attention을 이용한 한국어 의존 파싱

$h_1, \dots, h_n = BiLSTM(x_1, \dots, x_n)$       부분 트리가 없는 경우 그대로 사용

$h_i^t = TreeLSTM(h_i, child(h_i))$       부분 트리가 있는 경우

- 구성된 트리 정보를 효율적으로 학습하기 위해 Tree-LSTM사용하였고 부분 트리가 없는 경우 Bi-LSTM에서 얻은 결과를 그대로 사용하였다. Child는 해당 표상에 자식 표상 정보를 의미
- 새롭게 얻어진 표상에 MLP를 적용하여 지배소 표상과 의존소 표상을 통해 기존 Biaffine모델과 동일하게 의존성을 결정한다.



# 실험 셋팅

## • 데이터 셋

### – SPMRL'14 한국어 데이터 셋

- 데이터 셋 통계

	Train	Dev	Test
문장수	23010	2066	2187

- CRF를 이용한 자동 형태소 분석 결과(F1: 97.60%) 이용

### – 모델 세팅

- 해당 형태소 임베딩은 Glove로 사전 학습한 300차원의 임베딩 벡터 사용
- 합성된 단어 임베딩은 300차원 임베딩 벡터 사용
- 모든 히든 레이어의 차원 수는 1200차원
- 드랍아웃은 0.2로 설정



# 실험 셋팅

---

## - 쉬운 의존성 결정

- 예측하기 쉬운 의존성 결정은 한문장에서 점수나 확신도를 보고 큰 순서대로 50%를 채택하여 쉬운 의존성이라고 판단하고 의존성을 결정 후 사용.
- 쉬운 의존성만 성능 테스트 하였을 때 Score50%의 성능은 uas 92.63%, Confidence50%의 성능은 uas 97.91%로 쉬운 의존성이기 때문에 높은 성능을 보였다.



## 실험 결과

- 실험결과(base Model과 비교 쉬운 의존성 제외한 평가)

	UAS	LAS
Biaffine Model-Score50%	85.87%	85.29%
EasyFirst-Score50%	86.38%	85.86%
Biaffine Model-Score50%	85.90%	85.86%
EasyFirst-confidence50%	86.37%	85.88%

- Biaffine Model은 트리 구조를 만들지 않고 평가한 모델
- 제안 모델 0.5%정도 개선된 성능을 보이고 있음





## 실험 결과

- 실험결과(base Model과 비교 모든 의존성)

	UAS	LAS
Biaffine Model	91.68%	91.08%
EasyFirst-Score50%	91.94%	91.36%
EasyFirst-confidence50%	91.92%	91.35%

- Base Biaffine Model 성능보다 UAS : 0.26, LAS 0.26%  
높은 성능을 보여 개선된 성능을 보임



# 실험 결과

- 실험결과(기존 모델과 비교)

	UAS	LAS
EasyFirst-Score50%	91.94%	91.36%
DeepBiaffine + dual-decomposition[KCC'18 민진우]	91.07%	N/A
Multilevel Biaffine [KSC'18 홍승연]	91.84%	91.29%
Multilevel Biaffine ELMO, BERT [KCC'19 홍승연]	93.46%	92.85%
Biaffine BERT[KCC'19 박천음]	93.86%	93.30%

- Base 모델과 비교하였을때 개선된 성능을 보이고 있지만 BERT, ELMO를 적용한 모델에 비해 낮은 성능을 보임.



# Q&A

