
Triaffine Attention을 이용한 엔터티 타이핑 기반 엔터티 링크

2020.12.21

최형준¹, 나승훈¹, 김현호², 김선훈², 강인호²
¹전북대학교, ²NAVER

Contents

- Background
- Related work
 - Connecting Embeddings for Knowledge Graph Entity Typing [Y. Zhao et al. 2020]
- Entity typing model – based biaffine
- Entity linking model – based triaffine
- Experiment Result
- Conclusion



엔터티 링킹

Barack Obama



From Wikipedia, the free encyclopedia
(Redirected from Barack obama)

"Barack" and "Obama" redirect here. For other uses, see Barack (disambiguation) and Obama (disambiguation).

Barack Hussein Obama II (/bɑːrɑːk huːˈseɪn oʊˈbɑːmɑː/ (listen)[ⓘ][Ⓘ] born August 4, 1961) is an American politician who served as the 44th President of the United States from 2009 to 2017. The first African American to assume the presidency, he was previously the junior United States Senator from Illinois from 2005 to 2008. He served in the Illinois State Senate from 1997 until 2004.

Obama was born in 1961 in Honolulu, Hawaii, two years after the territory was admitted to the Union as the 50th state. Raised largely in Hawaii, Obama also spent one year of his childhood in Washington State and four years in Indonesia. After graduating from Columbia University in New York City in 1983, he worked as a community organizer in Chicago. In 1988 Obama enrolled in Harvard Law School, where he was the first black president of the *Harvard Law Review*. After graduation, he became a civil rights attorney and professor, and taught constitutional law at the University of Chicago Law School from 1992 to 2004. Obama represented the 13th District for three terms in the Illinois Senate from 1997 to 2004, when he ran for the U.S. Senate. Obama received national attention in 2004 with his unexpected March primary win, his well-received July Democratic National Convention keynote address, and his landslide November election to the Senate. In 2008, Obama was nominated for president a year after his campaign began and after a close primary campaign against Hillary Clinton. He was elected over Republican John McCain and was inaugurated on January 20, 2009. Nine months later, Obama was named the 2009 Nobel Peace Prize laureate, accepting the award with the caveat that he felt there were others "far more deserving of this honor than I."



- **Context** : *After campaigning on the promise of health care reform, President Obama gave a speech in March 2010 in Pennsylvania*
- **Mention** : [President Obama](#)
- **Entity** : *Barack Obama* (http://wikipedia.org/wiki/Barack_Obama)

엔터티 링킹은 엔터티의 중의성을 해결하기 위한 작업으로, 문서에 나타난 개체 표현과 부합하는 지식 베이스에 있는 개체를 연결해주는 기술

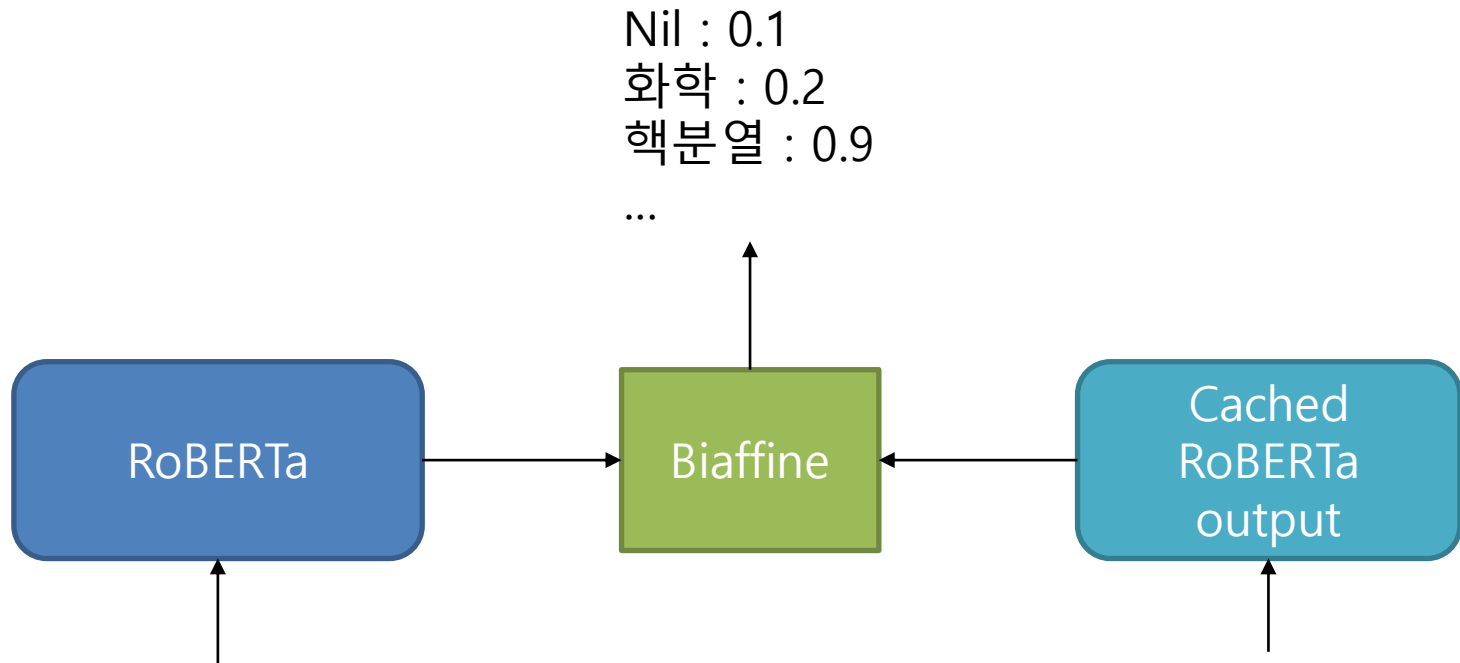


멘션 임베딩을 이용한 nil 멘션 탐지와 개체 연결의 통합 모델 [홍승연 et al' 20]

- 엔터티 링킹 시 Nil 엔터티를 두어 엔터티 링킹과 멘션 탐지를 동시에 수행
 - Nil 멘션은 문장 내의 엔터티가 아닌 부분을 의미
 - Nil 멘션인 경우 Nil 엔터티를, non-Nil 멘션인 경우 해당하는 엔터티를 예측
- RoBERTa를 통해 엔터티 설명을 인코딩, 출력의 첫 번째 토큰인 <s>에 해당하는 부분을 엔터티 임베딩으로 사용
 - 엔터티 임베딩은 학습되지 않도록 고정시켜 사용
- 문장 내의 멘션에 해당하는 부분을 Concatenate 한 것과 각각의 엔터티 후보를 Biaffine 연산을 통해 점수를 계산
- 엔터티 후보는 멘션-엔터티 사전을 통해 추출
 - 데이터셋에서 등장한 모든 멘션-엔터티 쌍을 통해 사전 구성
 - 사전은 각각의 멘션과 등장한 모든 엔티티로 이루어짐



멘션 임베딩을 이용한 nil 멘션 탐지와 개체 연결의 통합 모델 [홍승연 et al' 20]



핵융합은 물리학에서 핵분열과 상반되는...

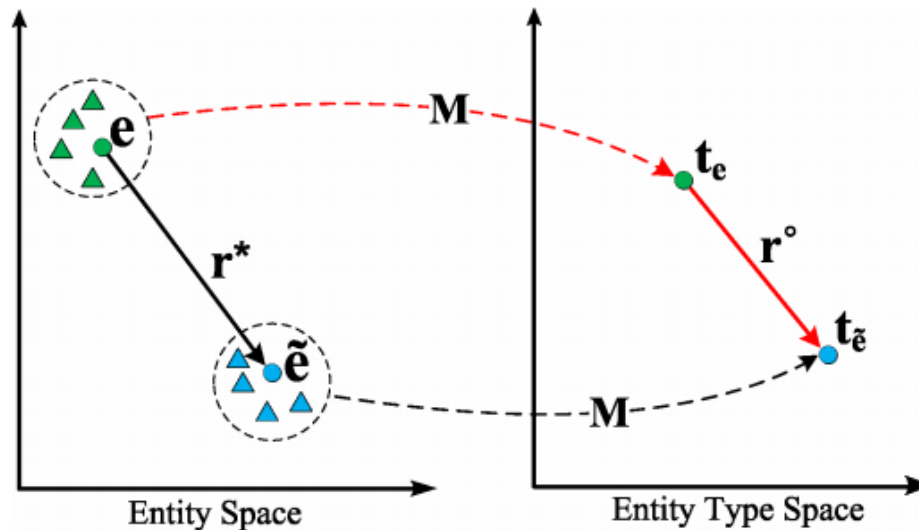
핵분열, 핵융합, 화학...

- RoBERTa로 인코딩된 문장에서 멘션에 해당하는 부분을 Concatenate, 각각의 엔터티 임베딩과 Biaffine을 통해 점수 계산
 - 엔터티 임베딩은 각 엔터티의 설명을 RoBERTa로 인코딩 하여 사용
- 단순히 RoBERTa를 통해 인코딩 하기 때문에 엔터티 임베딩은 최적화 되지 않은 상태



Connecting Embeddings for Knowledge Graph Entity Typing [Y. Zhao et al. 2020]

- 엔티티 임베딩과 지식 그래프를 통해 타입 임베딩을 구축
 - 임베딩이 해당 엔티티에 내포된 타입에 따라 응집한다는 것을 이용
 - 엔티티 임베딩을 타입 임베딩에 투영, 지식 그래프상의 관계를 통해 최적화



제안 모델 - Triaffine Entity Linking

- 엔티티 타이핑과 엔티티 링킹을 결합
 - 엔티티 타이핑 결과를 엔티티 링킹시 활용, 엔티티 링킹 성능을 향상시킬 방법을 모색
 - 엔티티 타입 정보와 엔티티의 표상, 멘션의 표상을 결합하기 위해 Triaffine attention 사용
- 엔티티 타이핑 모델과 엔티티 링킹 모델로 구성
 - 엔티티 타이핑 모델은 문장 내의 멘션 표상과 엔티티 임베딩을 Biaffine attention을 통해 각 엔티티에 대한 점수를 계산
 - 타이핑 모델의 출력을 타입 임베딩 내의 각 표상에 weighted sum 시행
 - 각 타입 표상에 출력값을 곱한 후 더함
 - 엔티티 링킹 모델은 weighted sum을 시행한 타입 정보와 각 엔티티의 표상, 멘션의 표상을 Triaffine 어텐션을 통해 점수 계산



엔티티 타이핑 모델 – Biaffine 기반

- 하나의 RoBERTa와 LSTM, 타입 임베딩으로 구성
 - RoBERTa와 LSTM은 문장을 인코딩 하기 위해 사용
 - 입력 문장을 RoBERTa로 인코딩 한 후, LSTM을 RoBERTa의 출력에 적용
 - 각 멘션 스팬의 시작점과 끝 점을 **Concatenate**하여 멘션 스팬에 대한 표상을 획득
- 각 타입에 대한 표상은 TransE를 통해 미리 계산하여 사용
 - 타입 임베딩 $Emb(t)$ 은 각 타입 id를 TransE 표상으로 변환하는 임베딩 행렬
 - 타입 임베딩은 모델 학습 중 학습되지 않도록 값을 **고정시켜** 사용
 - $F(m)$ 은 멘션 m 이 가질 수 있는 타입으로, 멘션이 m 일때 등장한 **모든 엔티티의 타입에 대한 합집합**



엔티티 타이핑 모델 – Biaffine 기반

- 타입에 대한 점수는 타입 임베딩과 멘션의 표상을 **Biaffine Attention**을 적용 한 결과를 사용

$$X' = \text{LSTM}(\text{RoBERTa}(X)) = [x'_1, \dots, x'_{start}, \dots, x'_{end}, \dots, x'_l]$$

$$\mathbf{m} = \text{concat}([x'_{start}, x'_{end}])$$

$$\text{Score}_{type}(t) = \frac{\exp(\text{Biaffine}(\mathbf{m}, \text{Emb}(t)))}{\sum_{t' \in F(m)} \exp(\text{Biaffine}(\mathbf{m}, \text{Emb}(t')))}$$

$$\text{Biaffine}(\mathbf{e}, \mathbf{d}) = \mathbf{dU} + \mathbf{dW}_d + \mathbf{eW}_e + \mathbf{B}$$



엔티티 타이핑 모델 – Biaffine 기반

- 타이핑 기반 Nil 탐지
 - Nil 탐지 점수를 계산하여 판정

$$\text{Score}_{nil}(m) = \log P(\text{nil}|m) - \log \sum_{t \in F(m)} P(t|m)$$

- $\text{Score}_{nil}(m)$ 이 **threshold**를 초과하거나, 타이핑 결과 중 상위 10개 타입과 $F(m)$ 사이의 **교집합이 없는 경우** m 을 Nil 멘션으로 판단

$$\begin{cases} Nil, & \text{if } \text{Score}_{nil}(m) > thr \\ Nil, & \text{if } F(m) \cap \text{Topk}_{\text{type}}(\text{Score}_{\text{type}}(t)) = \emptyset \\ nonNIL, & \text{else} \end{cases}$$



엔티티 링킹 모델 - Triaffine기반

- 엔티티 링킹 모델은 타입 임베딩과 엔티티 임베딩, 문장내의 멘션의 표상을 Triaffine 하여 각 엔티티에 대한 점수를 계산
 - 멘션에 대한 표상은 엔티티 타이핑 모델에서 사용한 것을 재사용
 - 타입 임베딩은 모든 타입 임베딩 내의 표상을 엔티티 타이핑 모델의 출력을 **weighted sum**하여 계산
 - 엔티티 임베딩은 **RELIC**을 통해 사전학습을 한 것을 사용

$$\mathbf{m} = \text{concat}([x'_{start}, x'_{end}]), \mathbf{e} = \text{emb}(e)$$

$$\mathbf{t} = \sum \text{score}_{\text{type}}(t) \cdot \text{emb}(t)$$

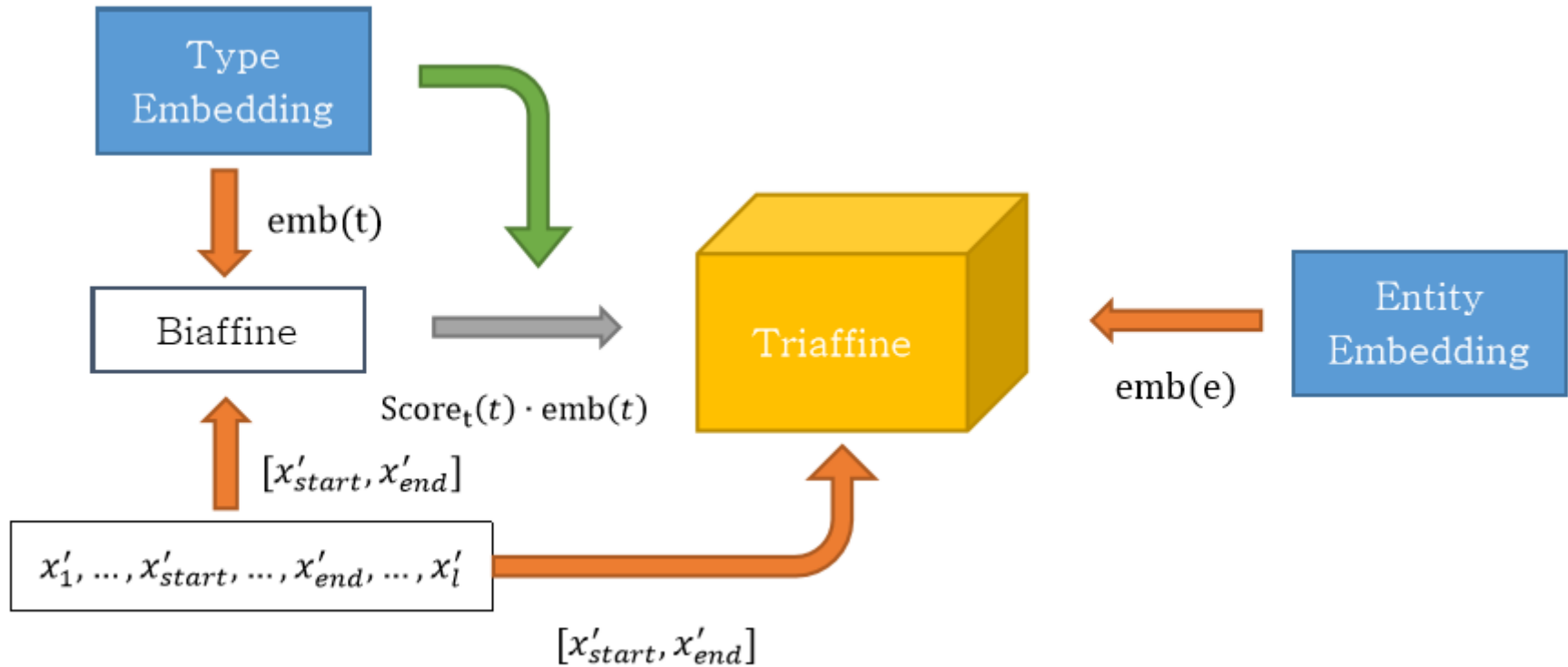
$$\mathbf{m}^\circ = \text{MLP}^{\mathbf{m}^\circ}(\mathbf{m}), \mathbf{t}^\circ = \text{MLP}^{\mathbf{t}^\circ}(\mathbf{t}), \mathbf{e}^\circ = \text{MLP}^{\mathbf{e}^\circ}(\mathbf{e})$$

$$\text{Score}_{\text{entity}}(e) = \frac{\exp(\text{Triaffine}(\mathbf{t}^\circ, \mathbf{m}^\circ, \mathbf{e}^\circ))}{\sum_{e' \in E(m)} \exp(\text{Triaffine}(\mathbf{t}^\circ, \mathbf{m}^\circ, \mathbf{e}'^\circ))}$$

$$\text{Triaffine}(\mathbf{t}, \mathbf{m}, \mathbf{e}) = (\mathbf{t}^\circ + \mathbf{b}_t) \cdot \mathbf{W}^{\text{triaffine}} \cdot \mathbf{m}^\circ \cdot (\mathbf{e}^\circ + \mathbf{b}_e)$$



엔티티 링킹 모델 - Triaffine기반



실험

- 엔티티 링킹 학습 데이터는 총 6만개의 문장으로 구성, 3만개를 train, 2만개를 test, 1만개를 dev 셋으로 사용
 - 데이터셋에 포함된 타입의 개수는 6016개, 엔티티의 개수는 56055개
 - 각 엔티티에 대한 타입 정보는 Wikidata를 통해 획득
- RELIC은 위키백과 전체 데이터를 사용하여 사전학습 시행
- 타이핑 기반 Nil 탐지를 위한 threshold값은 -8.8을 사용, 타이핑 결과 중 0.0001미만인 값은 0으로 취급



실험

	정밀도	재현율	F1	Nil 탐지 F1
링킹	82.12%	81.51%	81.84%	88.85%
타이핑	65.70%	88.62%	75.46%	87.30%

- Nil 탐지 F1은 모델이 예측한 결과가 Nil인지 여부에 대한 성능
 - 링킹 시 Nil 탐지 f1은 예측된 엔티티가 Nil 엔티티인 경우 nil, 아닌 경우를 nonNil로 취급하여 계산



결론 및 향후 연구

- 결론

- 엔티티 타이핑을 시행하고 그 결과를 활용하는 방향으로 엔티티 링킹 수행
- F1기준 멀티 타입 타이핑 성능은 75.46%, 링킹은 81.84%가 산출됨

- 향후 연구

- 타이핑 과정의 loss 함수를 정교화 – loss함수와 Nil 점수 계산식에 따라 성능이 크게 변화
- 엔티티 링킹을 일종의 coreference resolution 문제로 보고 문장 내의 멘션을 클러스터링, 각 클러스터에 적합한 엔티티로 링킹 함으로써 엔티티 링킹을 구현



Q/A

