포트폴리오

KAIST 전산학과 석사과정
이민호

목차

- 연구실 소개
- 석사 연구
  - 개체 발견을 활용한 자가확장형 개체 연결과 개체 등록 절차에 대한 연구
- 기타 프로젝트

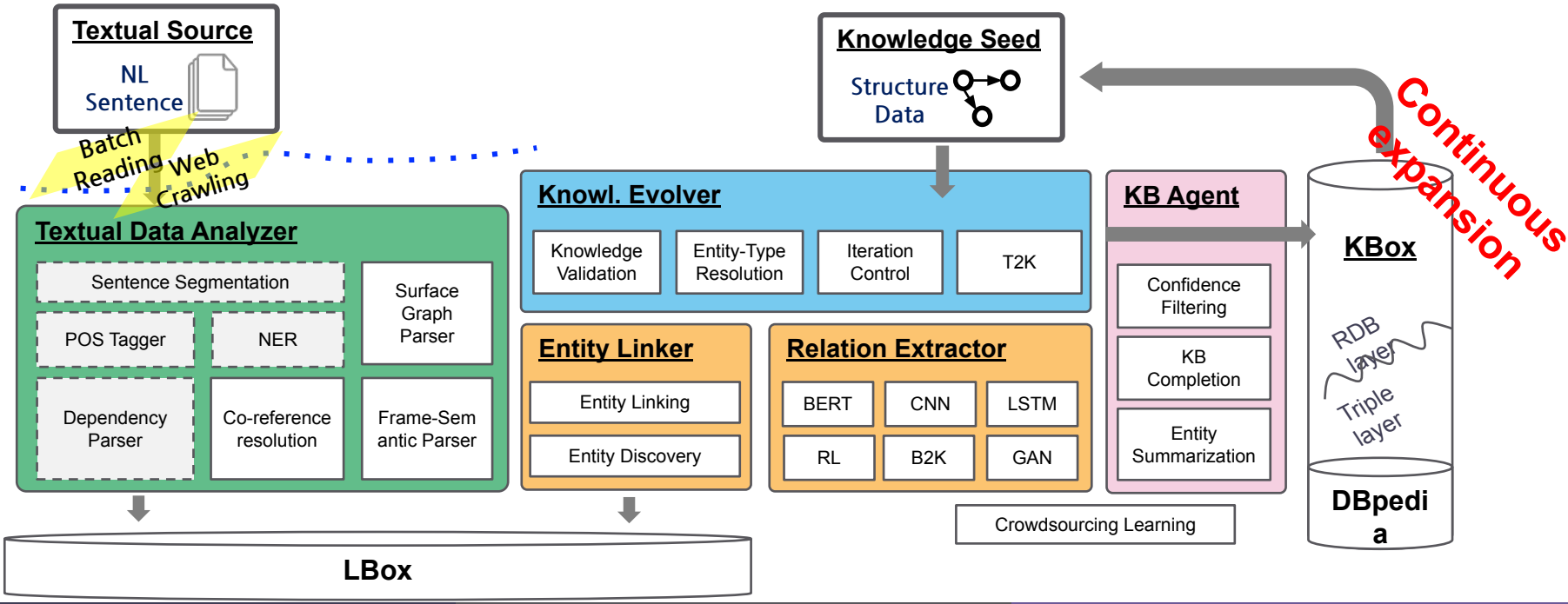
연구실 소개

연구실 프로젝트

- Exobrain(WiseKB): 빅데이터 기반으로 자가학습형 지식베이스를 구축하는 프로젝트
- VTT(Video Turing Test): 비디오에 대한 인간 수준의 이해도를 갖추는 지능 개발 프로젝트
- Flagship: 상대방의 감성을 추론, 판단하여 그에 맞는 대화를 하는 감성 지능 개발 프로젝트
- PACS: 폐, 간, 심질환 판독문을 분석하여 의료지식을 획득하는 프로젝트

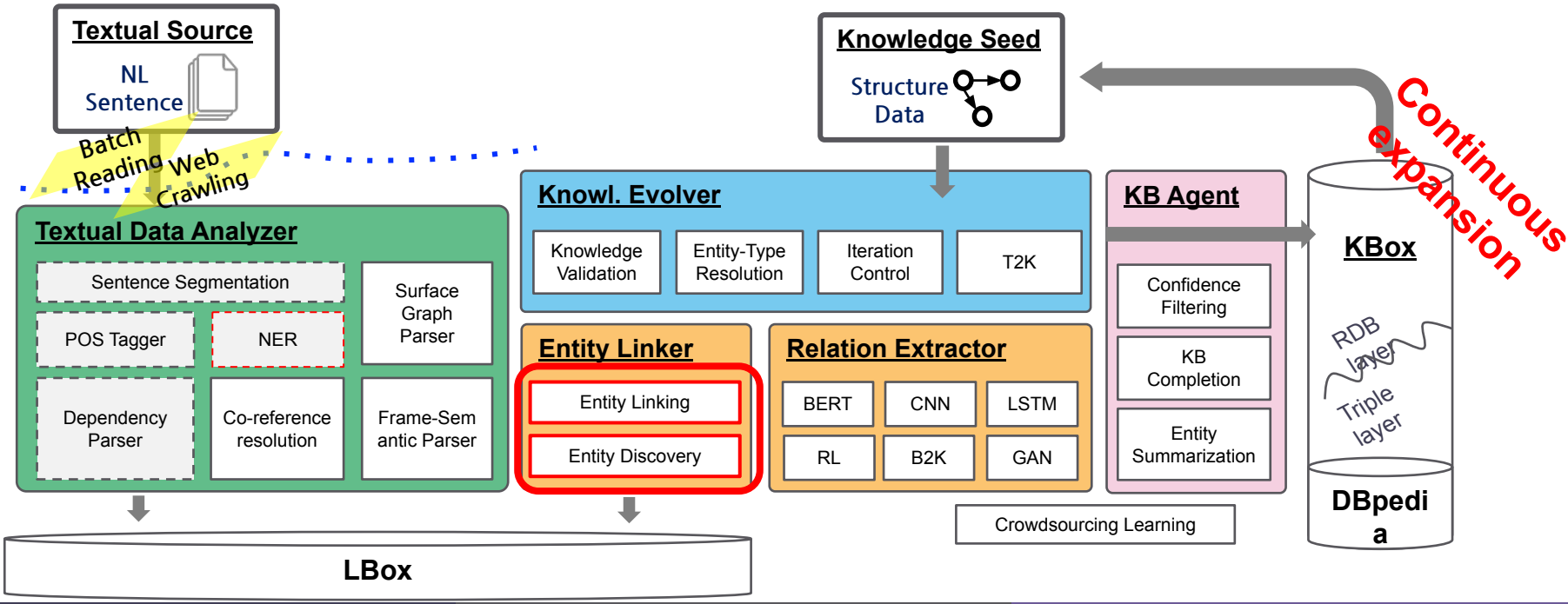
Knowledge Box

- 지식 학습 및 추출을 반복
  - 1) KBox 지식을 증강
  - 2) 지식 학습 모델 강화



Knowledge Box

- 지식 학습 및 추출을 반복
  - 1) KBox 지식을 증강
  - 2) 지식 학습 모델 강화

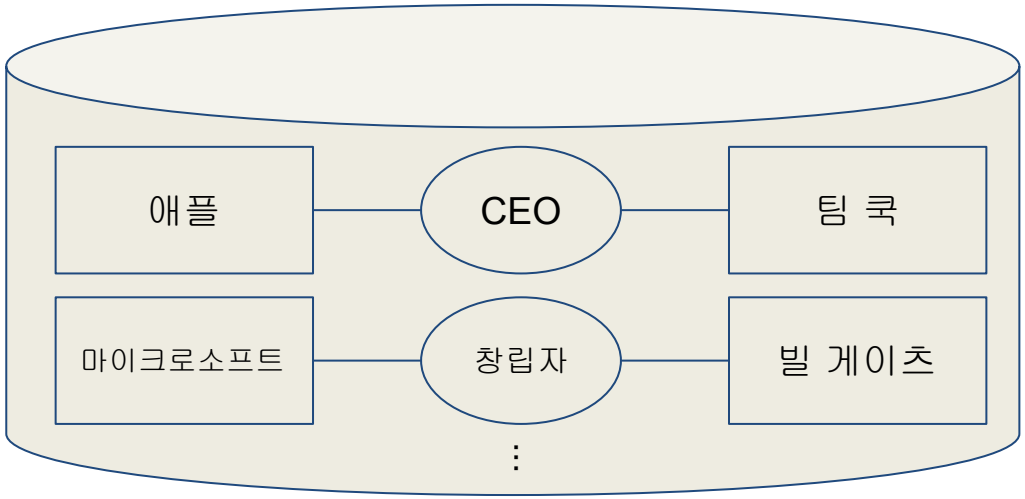


석사 연구

개체 발견을 활용한 자가확장형 개체 연결과 개체 등록 절차에 대한 연구

배경: 지식베이스

- 지식을 구조화하여 사용이 용이하게 만들어 놓은 데이터베이스
  - 개체와 개체 간의 관계로 이루어져 있음
  - <애플, CEO, 팀 쿡> 과 같은 삼항관계로 저장되어 있음



배경: 개체 연결

- 텍스트 내에서 개체를 나타내는 문자열을 지식베이스 내의 특정 URI(자원 식별자)로 연결하는 작업
  - [애플]이라는 개체가 기업 애플을 나타내는 것인지, 사과를 나타내는 것인지 구분하는 것

애플의 팀 쿡 CEO는 애플 파크의 스티브잡스 극장에서 신제품 공개 행사를 열었다.

<URI:애플_(기업)>의 <URI:팀_쿡> <URI: 최고경영자>는 애플 파크의 스티브잡스 극장에서 신제품 공개 행사를 열었다.

기존의 개체 연결의 한계

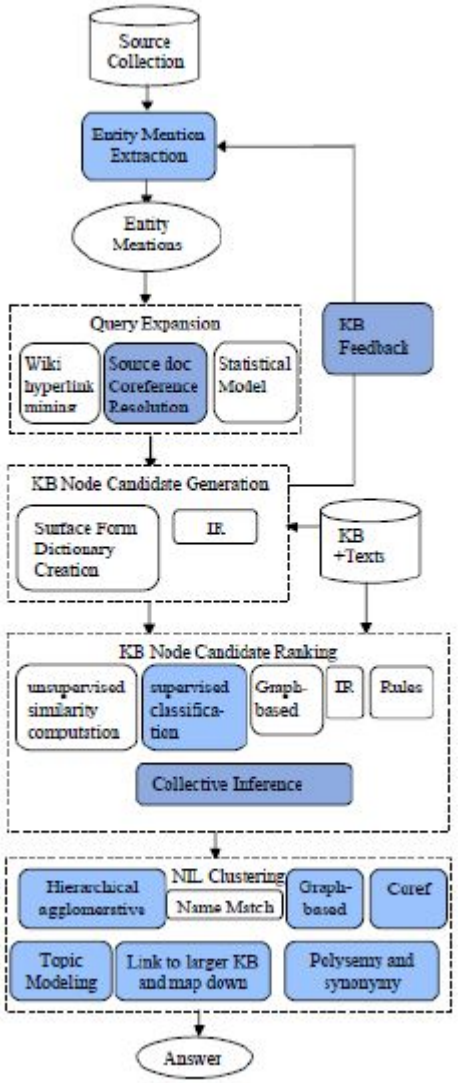
- 지식베이스 내에 없는 개체의 경우 연결 불가
  - [애플 파크], [스티브잡스 극장]과 같은 개체는 연결할 수 없다.
  - 따라서 지식베이스의 크기에 큰 영향을 받는다
  - 영어 지식베이스와 한국어 지식베이스는 개체 수로 볼 때 12배 정도의 크기 차이가 난다.

<URI:애플_(기업)>의 <URI:팀_쿡> <URI:최고경영자>는 애플 파크의 스티브잡스 극장에서 신제품 공개 행사를 열었다.

위키(지식베이스)	영문	한국어
문서(개체) 개수	5,941,263	470,824

관련 연구

- [1]에서는 지식베이스 연결에 실패한 개체들(NIL Entity)을 같은 것을 나타내는 것끼리 묶어 내는 태스크를 수행하였다.
- [2]에서는 문장이 나타난 주변 문맥, 개체 정보, 문서의 주제 등을 활용하여 개체를 나타내는 URI가 지식베이스에 존재하는지를 판단하였다.
- 본 논문에서는 [2]의 문제 정의를 차용하여 개체를 나타내는 URI가 지식베이스에 있는지를 판단하고, 없는 경우 지식베이스에 등록하여 지식베이스를 확장해 나간다.
  - 새로 등록된 개체에 연결하는 것으로 [1]에서의 NIL Entity clustering문제를 함께 풀 수 있다.

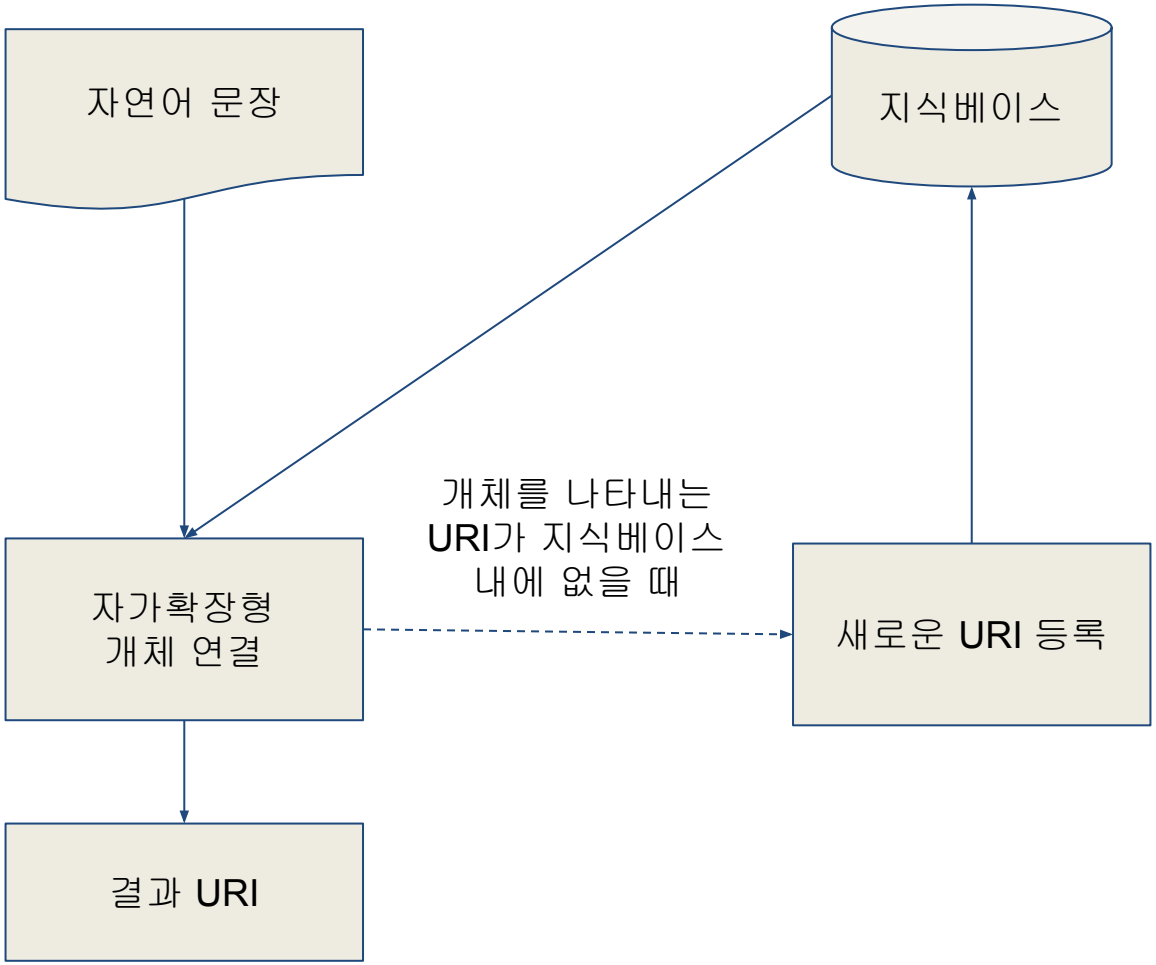


[1] Heng Ji et al., Overview of TAC-KBP2014 Entity Discovery and Linking Tasks, 2014
[2] Zhaohui Wu et al., Exploring Multiple Feature Spaces for Novel Entity Discovery, AAAI, 2016

문제 정의

- 지금의 개체 연결은 지식베이스에 없는 개체를 연결하지 못한다.
  a. 지식베이스에 없는 개체는 수동으로 등록해 주어야 연결이 가능하다.
- 지식베이스에 없는 개체 역시 등록 및 연결하여 개체 연결이 찾을 수 있는 대상을 늘린다.
- 3가지 sub-task로 문제 풀이
  a. 지식베이스에 없는 개체 구분
  b. 지식베이스에 없는 개체 연결
  c. 지식베이스에 없는 개체 등록

개념도

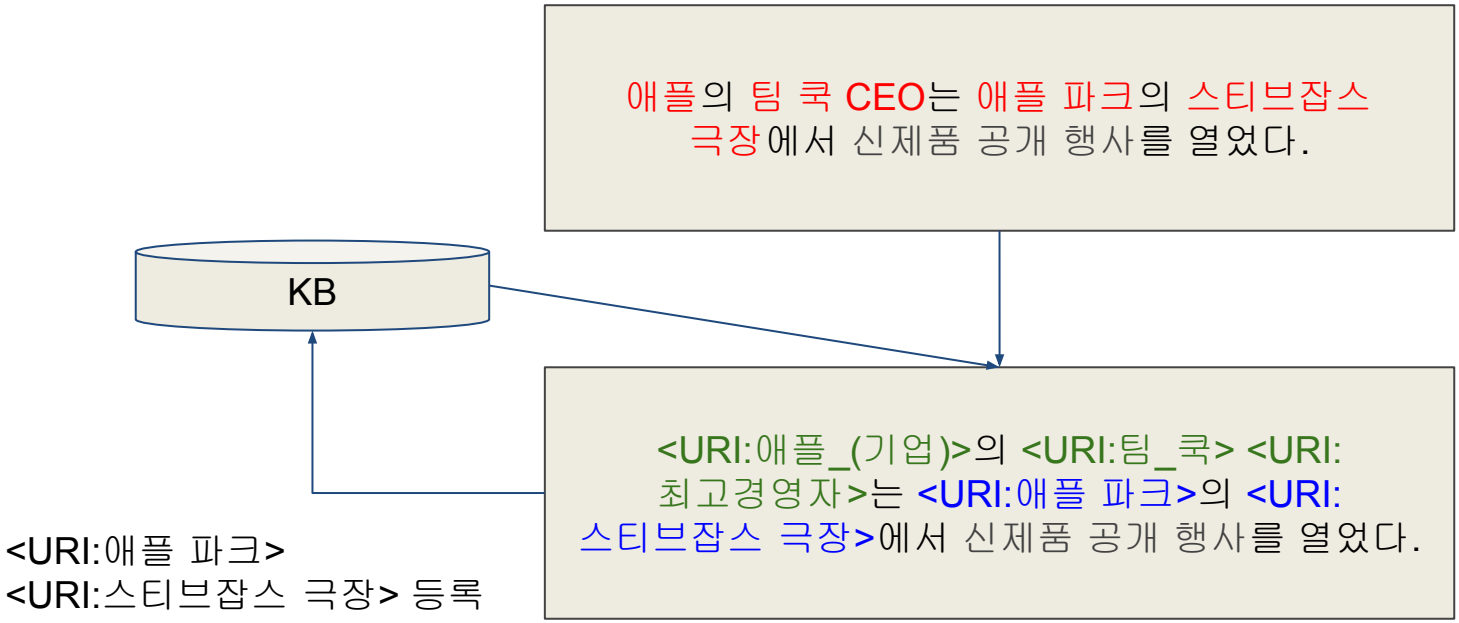

자연어 문장
지식베이스
개체를 나타내는
URI가 지식베이스
내에 없을 때
자가확장형
개체 연결
새로운 URI 등록
결과 URI

Challenge

- 개체가 지식베이스 내의 URI로 연결되어야 하는지 구분해야 한다.
- 한번 등록한 URI를 다시 연결해야 한다.
  - 같은 표현형을 가지지만 다른 개체일 수 있고, 다른 표현형을 가지지만 같은 개체일 수 있다.
- 등록 직후의 URI는 개체 연결의 Resource(개체명 사전 등)을 바로 사용할 수 없다.
  - 기존의 개체 연결은 위키피디아 등에서 링크 정보를 수집하여 표현형과 URI간 통계 사전을 사용한다.
  - 새로 등록된 개체는 표현형과 URI의 연결 통계가 단 1건밖에 없으므로 바로 적용할 수 없다.
  - 따라서 캐시 KB를 사용하여 새로 등록된 URI를 임시로 저장하고, 두 KB의 연결에 다른 Resource를 사용한다.

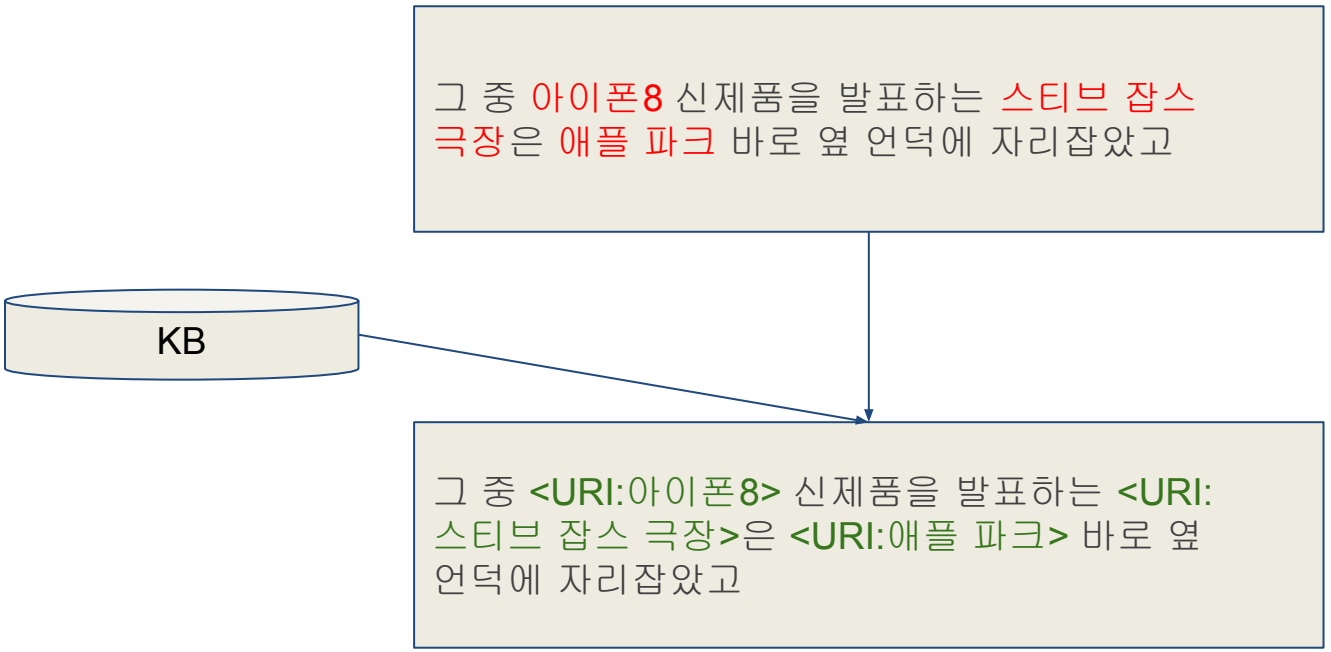
예시

[애플], [팀 쿡], [CEO]는 지식베이스에 해당하는 URI가 존재하지만, [애플 파크], [스티브잡스 극장]은 지식베이스에 존재하지 않으므로 새로운 URI를 등록한다.

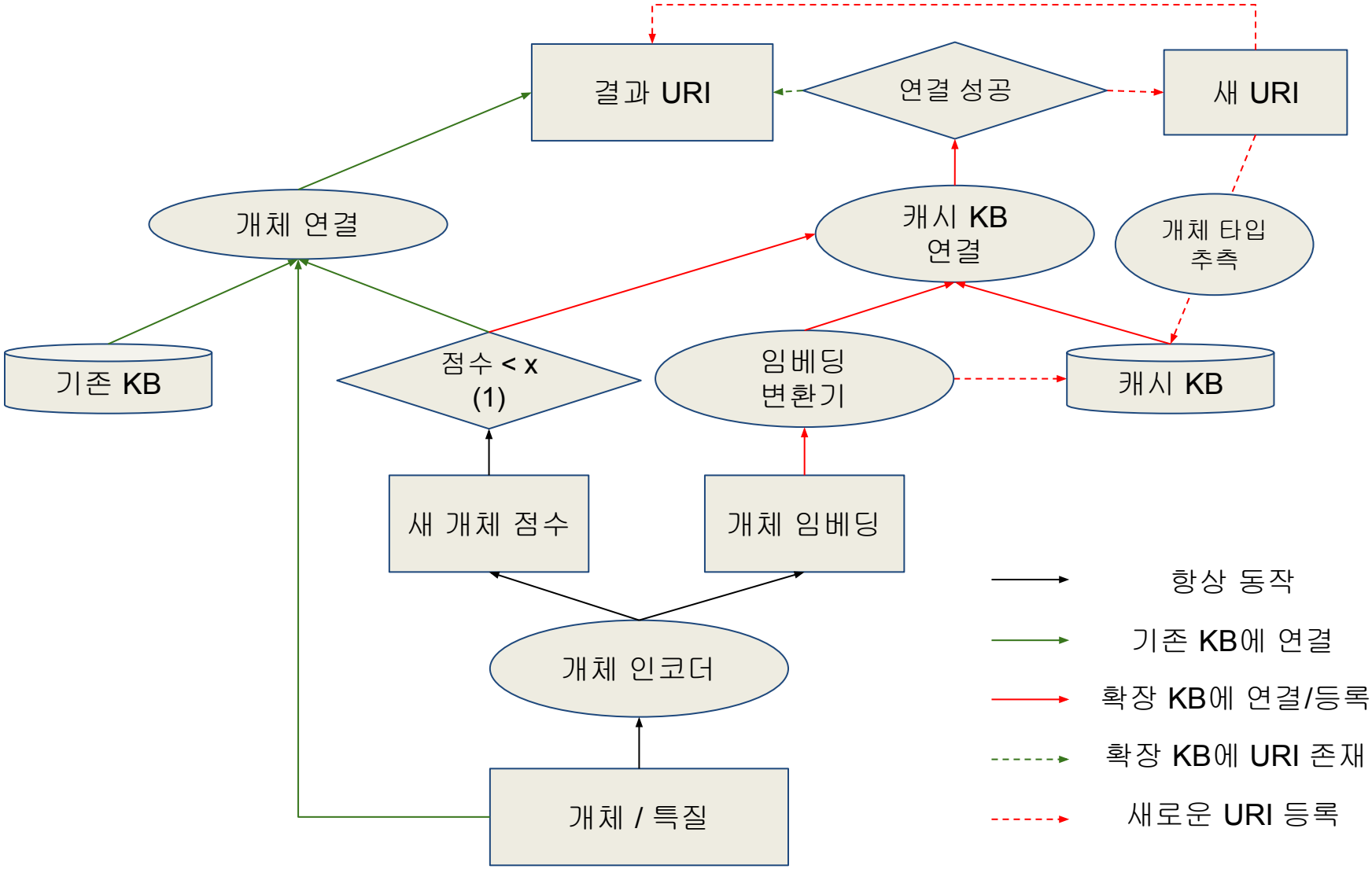


예시

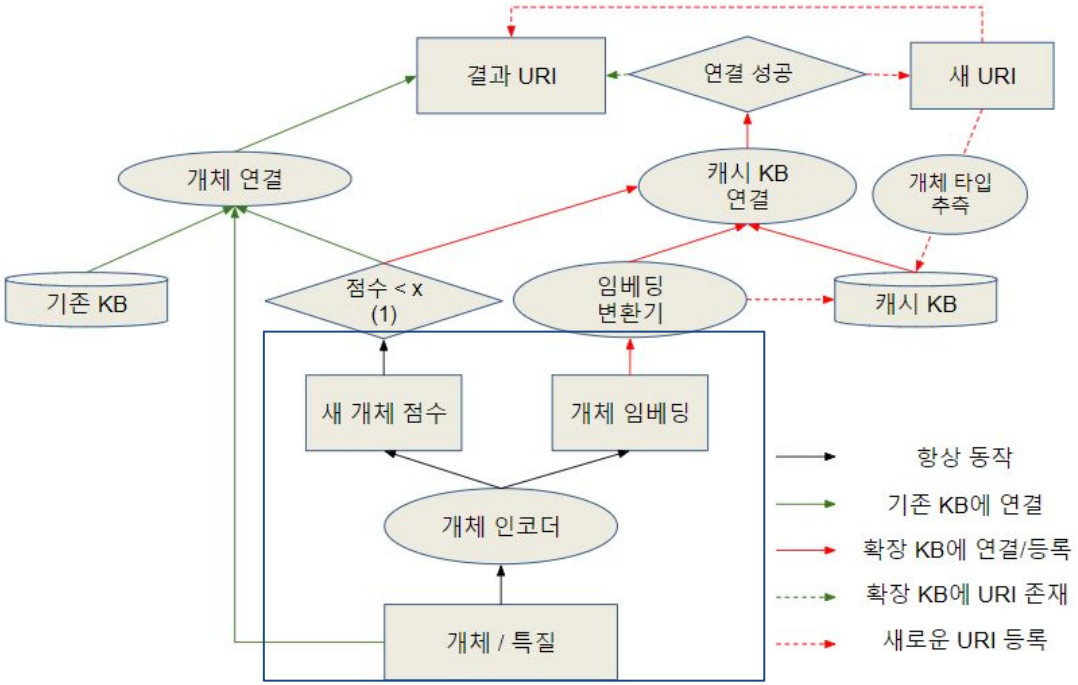
[스티브 잡스 극장]과 [애플 파크]는 이전 문서에서 등록되었으므로, 새 문서가 들어왔을 때 [스티브 잡스 극장]을 <URI:스티브 잡스 극장>으로, [애플 파크]를 <URI:애플 파크>로 연결한다.



전체 모델

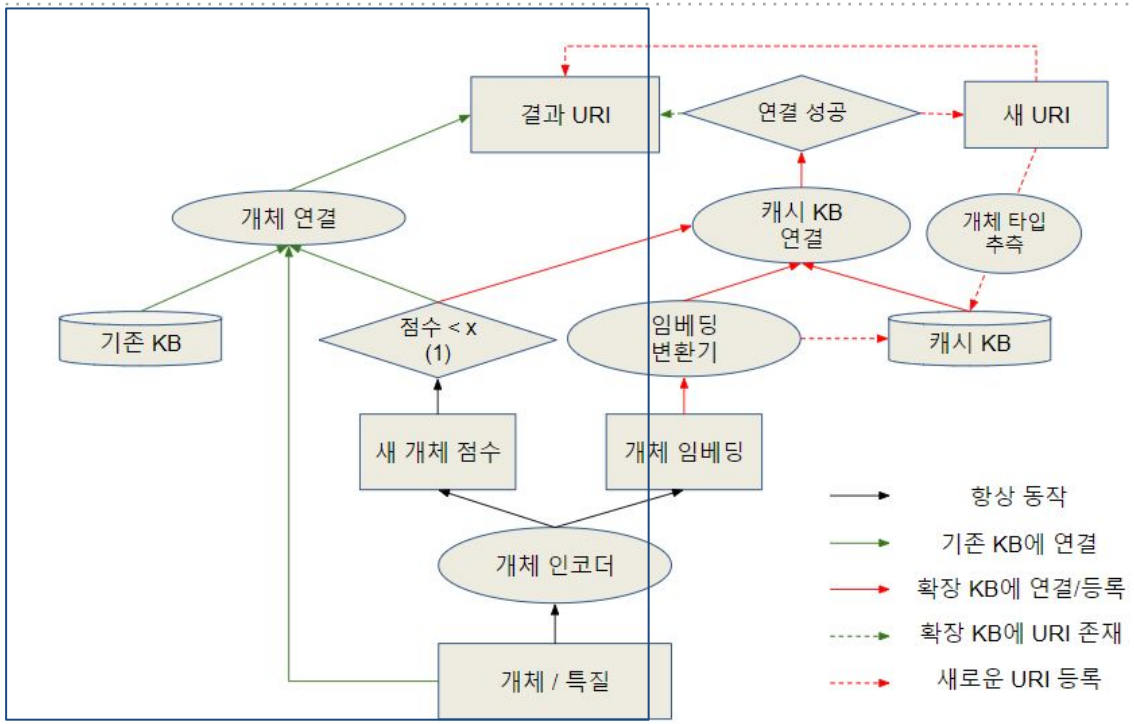

결과 URI
연결 성공
새 URI
개체 연결
캐시 KB
연결
개체 타입
추측
기존 KB
점수 < x
(1)
임베딩
변환기
캐시 KB
새 개체 점수
개체 임베딩
개체 인코더
개체 / 특질
항상 동작
기존 KB에 연결
확장 KB에 연결/등록
확장 KB에 URI 존재
새로운 URI 등록

모델 흐름도



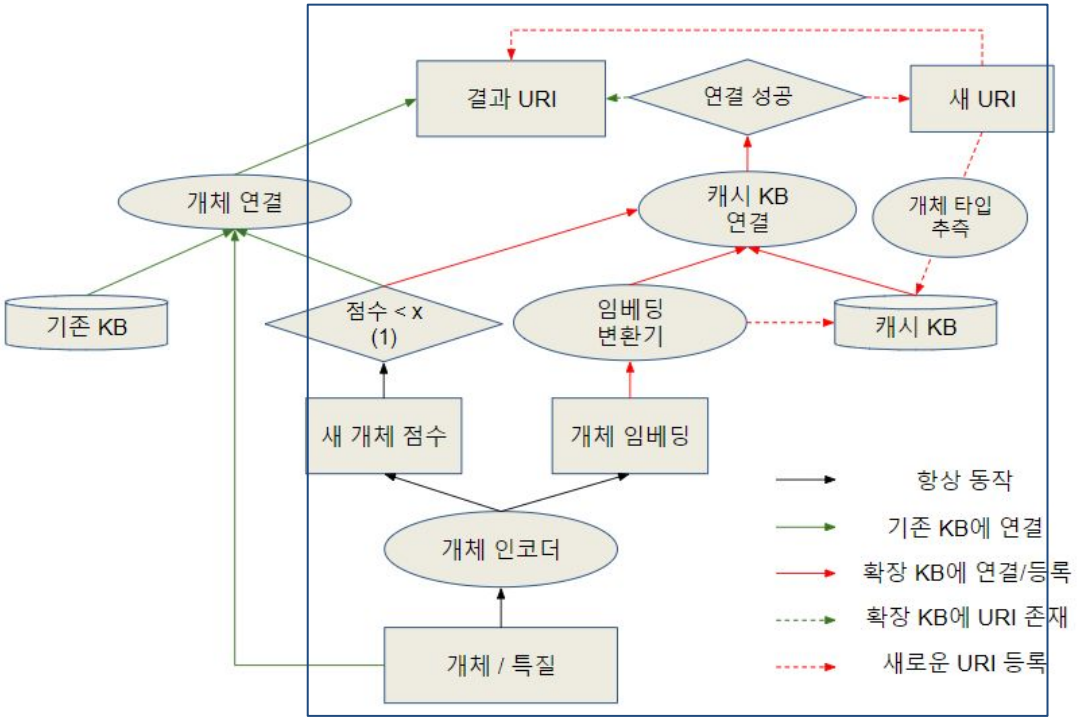
- 공통 과정(검은색 실선)
- 개체와 개체의 Feature들을 개체 인코더를 통과시켜 문맥에 맞게 인코딩한다.
- 인코딩된 문맥 정보를 사용하여 개체가 나타내는 URI가 기존 KB 안에 있는지를 파악한다.
- Score 비교를 통해 In-KB / Out-KB Linking을 수행한다.

모델 흐름도



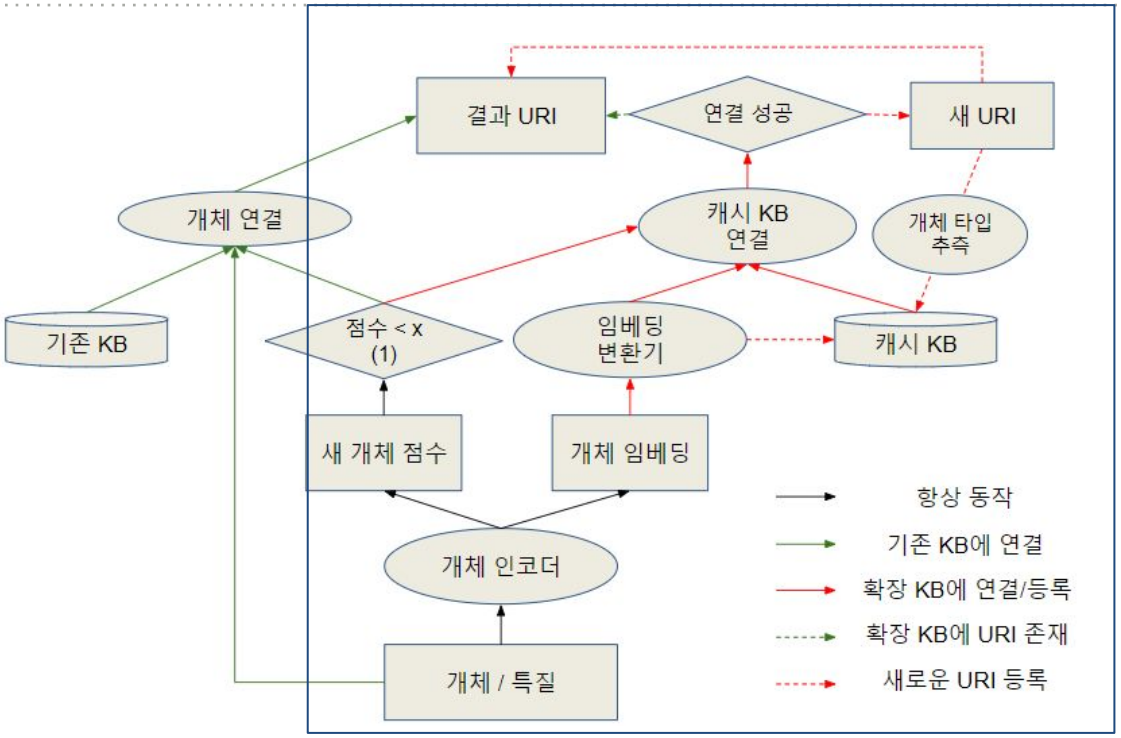
- Case 1: 기존 KB 내로 연결(초록색 실선)
- 새 개체 점수가 일정 이하일 때, 기존의 KB와 개체 간의 개체 연결을 수행한다.
- 개체 연결 모델은 개체명 사전을 활용하여 후보를 추출하고, 개체 간 관계를 고려한 모델을 사용하여 후보 중

모델 흐름도



- Case 2: 확장 KB에 등록된 URI로 연결(빨간색 실선, 초록색 점선)
- 새 개체 점수가 일정 이상일 때, 공통 과정에서 구한 인코딩된 개체 정보를 CNN을 사용하여 300차원의 개체 임베딩으로 변환한다.
- 확장 KB에 등록된 개체 임베딩과의 유사도를 구해 최대 유사도를 구한다.
- 유사도가 일정 이상인 경우 해당 URI를 결과로 반환한다.

모델 흐름도



- Case 3: 확장 KB에 등록되지 않은 URI(빨간색 실선, 빨간색 점선)
- 새 개체 점수가 일정 이상일 때, 공통 과정에서 구한 인코딩된 개체 정보를 CNN을 사용하여 개체 임베딩으로 변환한다.
- 확장 KB에 등록된 개체 임베딩과의 유사도를 구해 최대 유사도를 구한다.
- 유사도가 일정 이하인 경우, 새 URI를 등록하고, 해당 URI와 개체 임베딩을 확장 KB에 등록한다.

Features

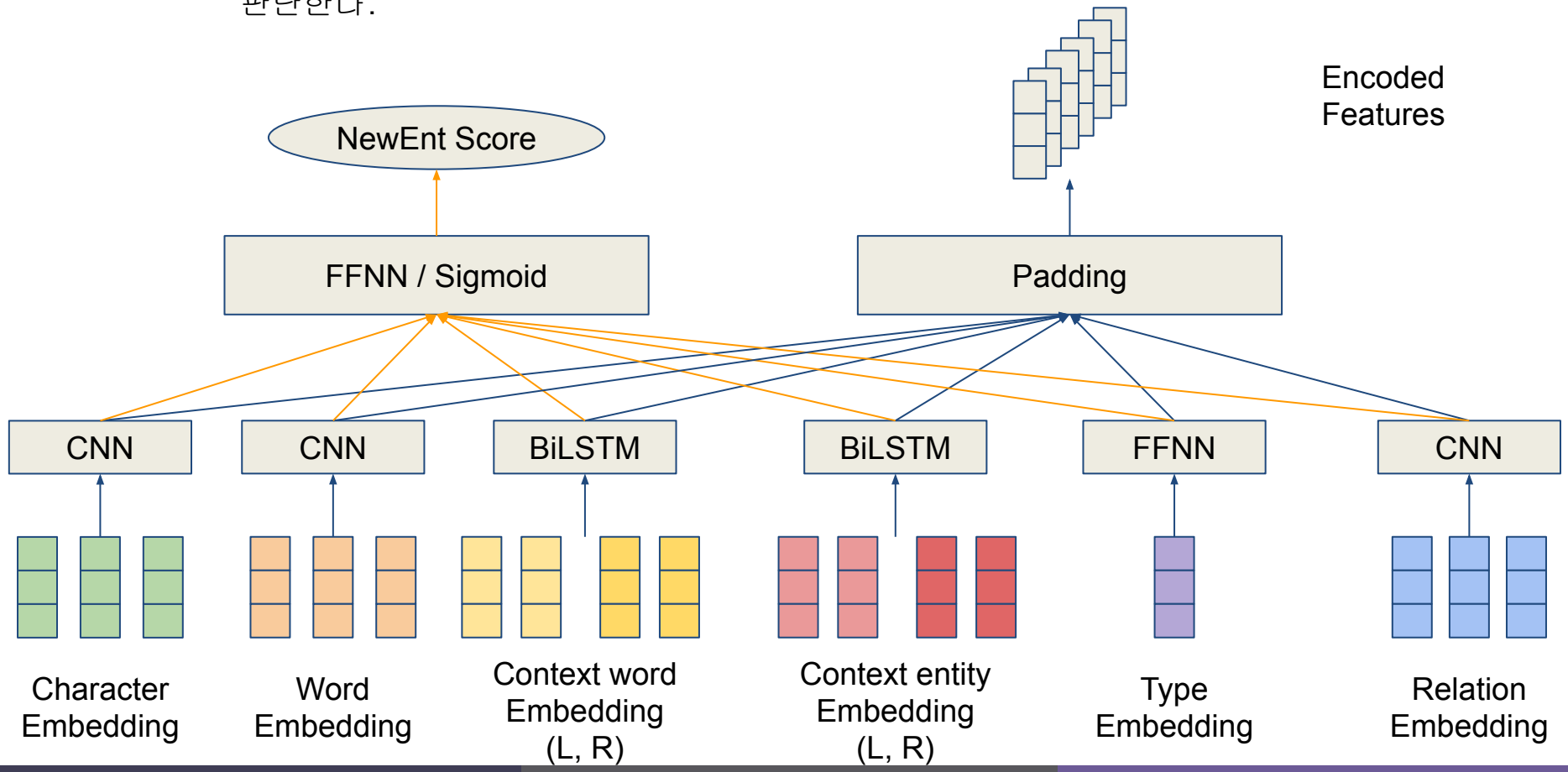
- 각각의 개체는 6가지의 feature를 가진다.
  - Character: 개체의 표현형을 자모 단위로 분해한 것
  - Word: 개체의 표현형을 형태소 단위로 분해한 것
  - Context word: 개체 주변의 형태소
  - Context entity: 개체 주변의 다른 개체
  - Type: 개체명 인식에서 얻은 타입 정보
  - Relation: 다른 개체와의 관계 정보

[애플]의 본사는 **[애플 파크]**이다.

- Character: ㅇ, ㅐ ,e,ㅍ,ㅡ,ㄹ,ㅍ, ㅏ ,e,ㅋ,ㅡ,e
- Word: 애플, 파크
- Context word: [애플, 의, 본사, 는], [이다, .]
- Context entity: [애플], []
- Type: LC
- Relation: <애플, location, 애플 파크>

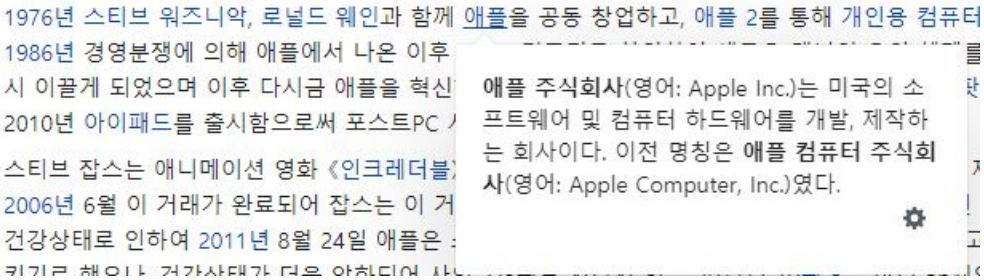
1. 개체 인코딩

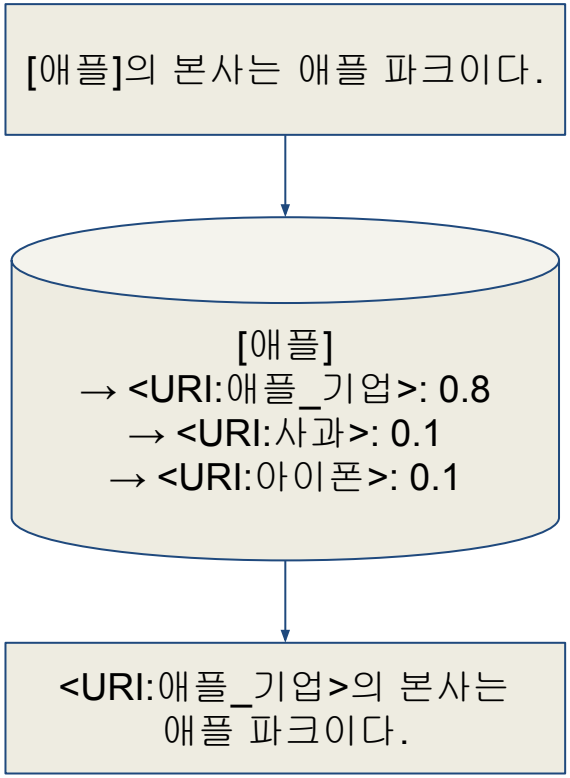
- 개체의 구성 요소와 문맥 정보를 받아 하나의 벡터로 만드는 과정
  - 문맥이 중요하지 않은 Character, Word, Relation은 CNN으로, 문맥이 중요한 Context word, entity는 BiLSTM을 사용하여 인코딩한다.
  - 인코딩된 Feature에 FFNN을 적용하여 개체가 지식베이스 내에 포함되어 있는지 판단한다.



2. 개체 연결

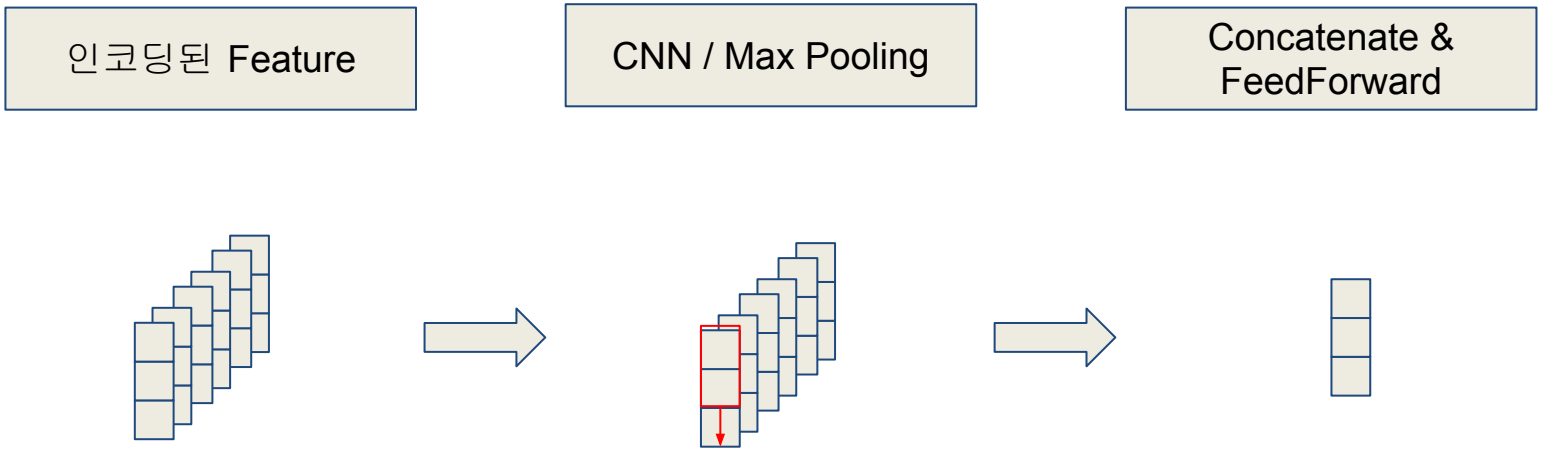
- 통계적 p(entity|mention) 사용
- 위키피디아 링크 텍스트와 링크 대상을 수집하여 가장 많이 링크된 대상으로 연결





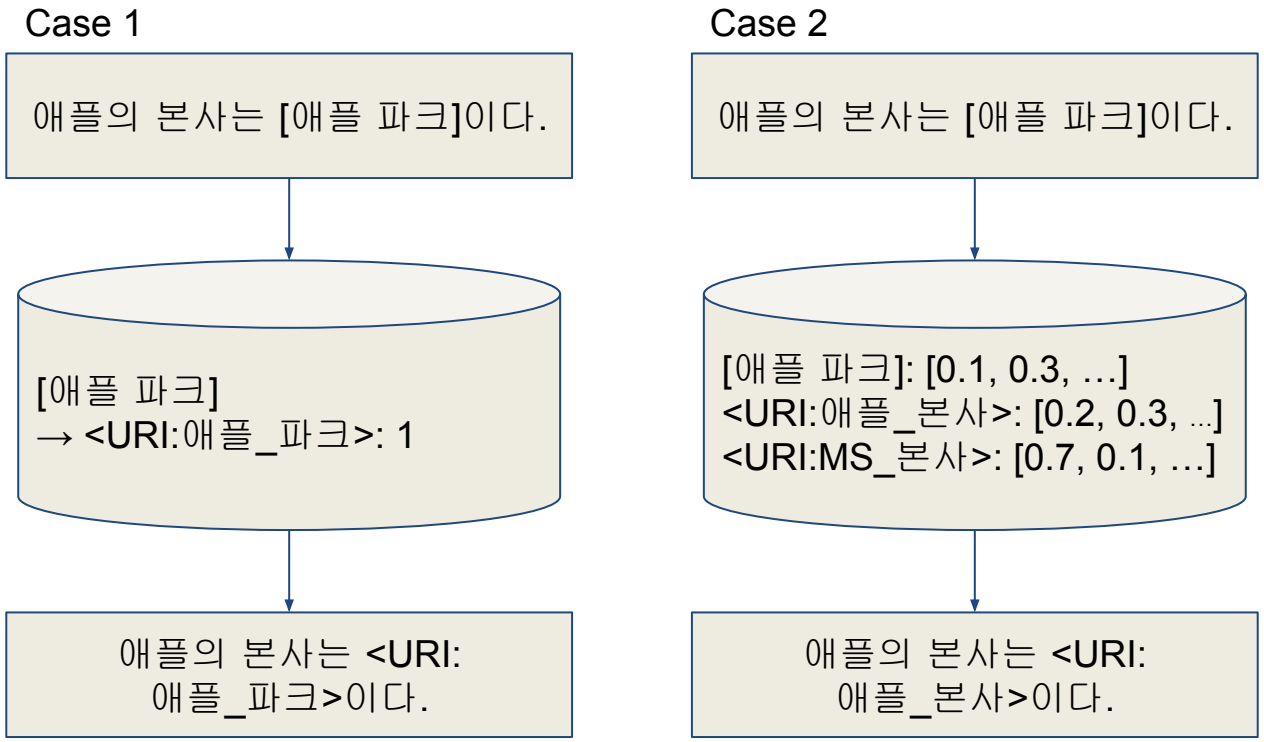
3. 임베딩 변환

- 개체 인코딩에서 각각의 Feature별로 인코딩된 벡터를 가지고 CNN과 Max pooling을 수행한다
  - 각각의 Feature의 특징을 학습하고, 가중치를 주기 위해 CNN 사용
  - 3개의 filter로 학습 뒤 FFNN을 사용하여 개체 임베딩과 같은 차원으로 변환



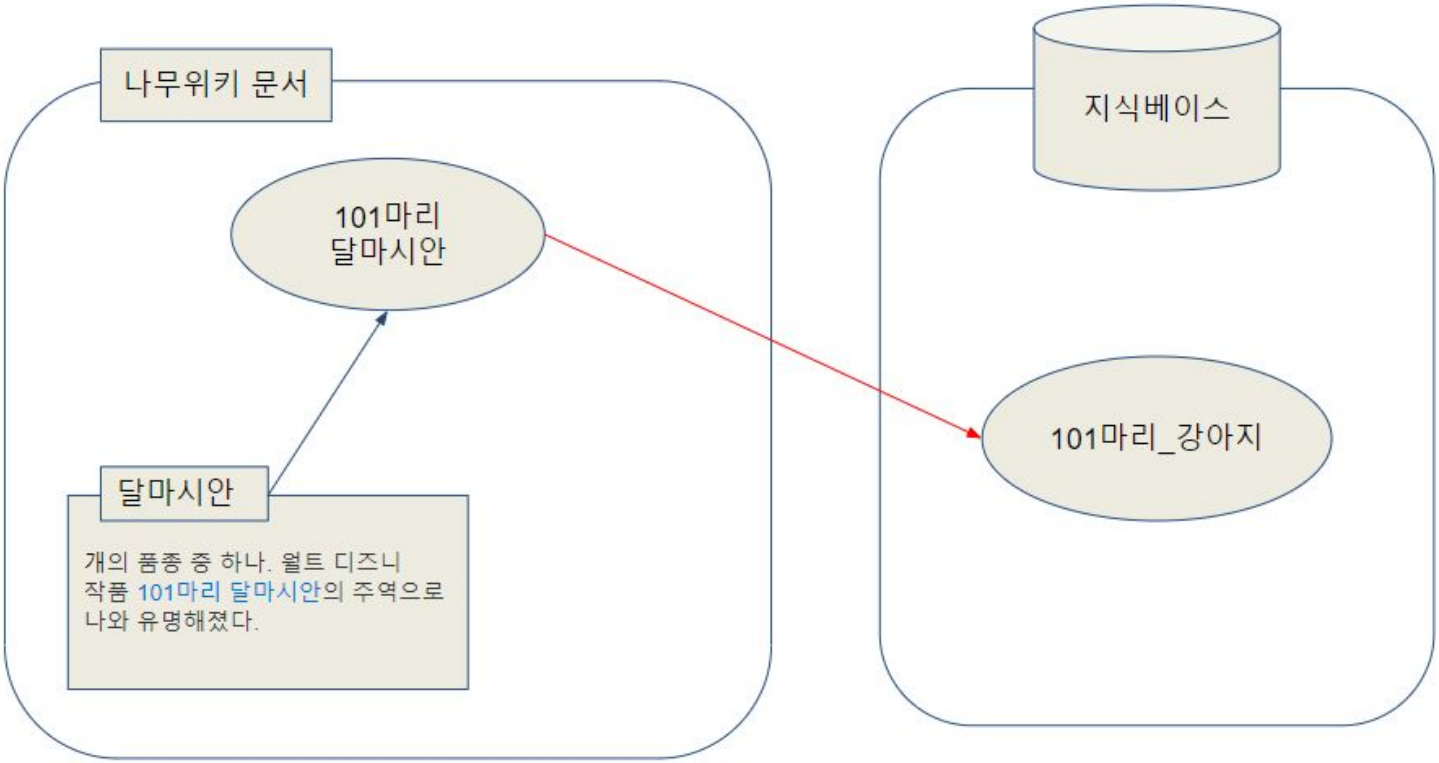
4. 캐시 지식베이스 연결/등록

- 정확도가 높은 방법(표면형의 포함 관계)을 우선 사용하여 1차 연결 시도
- 임베딩 기반 비교를 통해 코사인 유사도와 유클리드 거리를 사용하여 유사도가 일정 이상인 개체로 연결 시도
  - 연결 성공 시 개체 임베딩 수정
- 연결 실패 시 캐시 지식베이스에 새로운 URI, 개체명, 임베딩 등록



실험 설계

- 데이터셋
  - 나무위키 문서 제목을 지식베이스 내의 URI로 매핑
  - 나무위키 문서 제목을 가리키는 링크들을 수집하여 데이터셋으로 활용



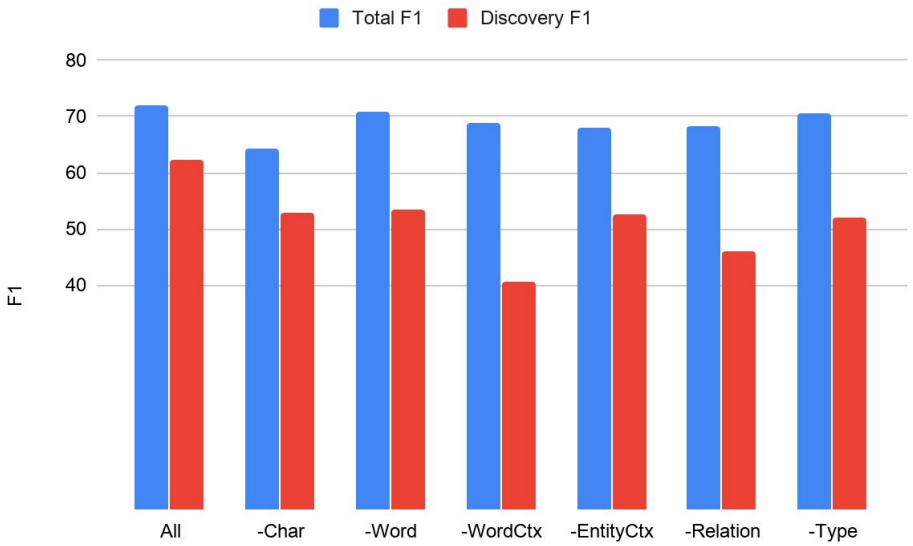
실험 설계

- 평가 방법
  - KB내에 있는 개체와 KB에 없는 개체를 잘 구분하는가 (Discovery F1)
  - Discovery를 함께 수행했을 때 기존 지식베이스에 있는 개체를 여전히 잘 연결하는가(In-KB F1)
  - 새로운 개체를 발견한 이후 잘 연결하는가(Out-KB F1)
  - 전체적으로 연결 성능이 얼마나 오르는가(Total F1)

실험 결과

	Discovery			In-KB			Out-KB			Total			
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	ARI
Model	69.02	56.69	62.25	82.71	77.06	79.78	45.53	45.53	45.53	73.91	70	71.9	90.87
Baseline	-	-	-	90.18	80.28	84.94	-	-	-	83.05	58.39	68.57	-

Discovery를 수행했을 때 precision이 감소하지만 recall이 크게 늘어 전체적 관점에서 성능이 오른다.



Feature별 Ablation study를 통해 자모 임베딩과 문맥(Word context, Entity context, Relation)이 Total F1에 큰 영향을 미친다는 것을 알 수 있다.

Discovery F1과 Total F1의 성능이 정비례하지 않는다
→ Linking 과정이 성능에 더 큰 영향을 미친다.

결론

- 지식베이스에 없는 개체도 등록하는 방식으로 개체 연결의 범위를 확장하였다.
- 기존 지식베이스와 캐시 지식베이스의 구분을 통해 서로 다른 방식의 연결 방법을 고안하였다.
- 개체 연결의 범위를 확장하여 전체적인 성능이 올랐음을 보였다.
- 보완사항
  - 개체 정보와 관계 정보를 활용하는 것은 본 모델을 수행하기 이전에 별도의 개체 연결과 관계 추출 과정을 거쳐야 한다는 것을 의미한다. 이를 잠재적 변수로 하는 모델을 제작해야 자연스러운 순서가 될 수 있다.
  - 캐시 KB 내의 개체들을 메인 KB로 옮기는 기준과 절차가 필요하다.

감사합니다.

Appendix

실험 데이터 통계

분류	지식베이스에 존재하는 개체	지식베이스에 존재하지 않는 개체	합계
인명	229	189	418
지명	141	35	176
기관명	113	43	156
기타	121	108	229
합계	604	375	979

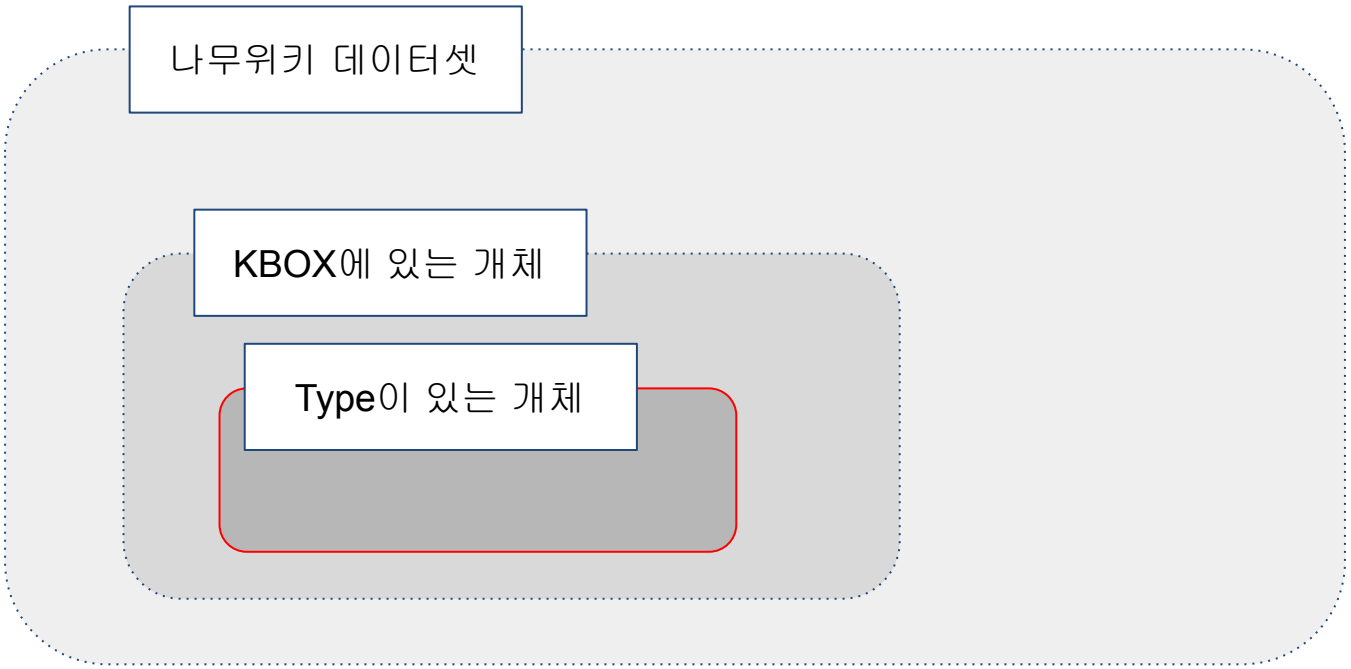
나무위키 문서 제목과 지식베이스간 매핑 결과

	학습	Dev	Test	합
지식베이스에 존재하는 개체	33,536	2,368	3,288	39,192
지식베이스에 존재하지 않는 개체	11,062	2,092	1,179	14,333
합	44,598	4,460	4,467	53,525

수집한 개체 수

Type 부여: 실험 설계

- 데이터셋
  - 나무위키 데이터셋 중 KBox에 타입이 등록된 개체



Type 부여: 실험 설계

- 실험 Flag
  - D/R 필터 기반 타입 제한(Ontology, inst, inst-ms)
    - Relation의 Domain 또는 Range에 오는 타입 제한
  - Union/Intersect
    - Union: 공집합에서 시작해서 가능한 타입을 추가하는 방식
    - Intersect: 전체 타입 집합에서 시작해서 불가능한 타입을 제거하는 방식
  - NE type 기반 타입 부여
  - 계층적 타입 부여

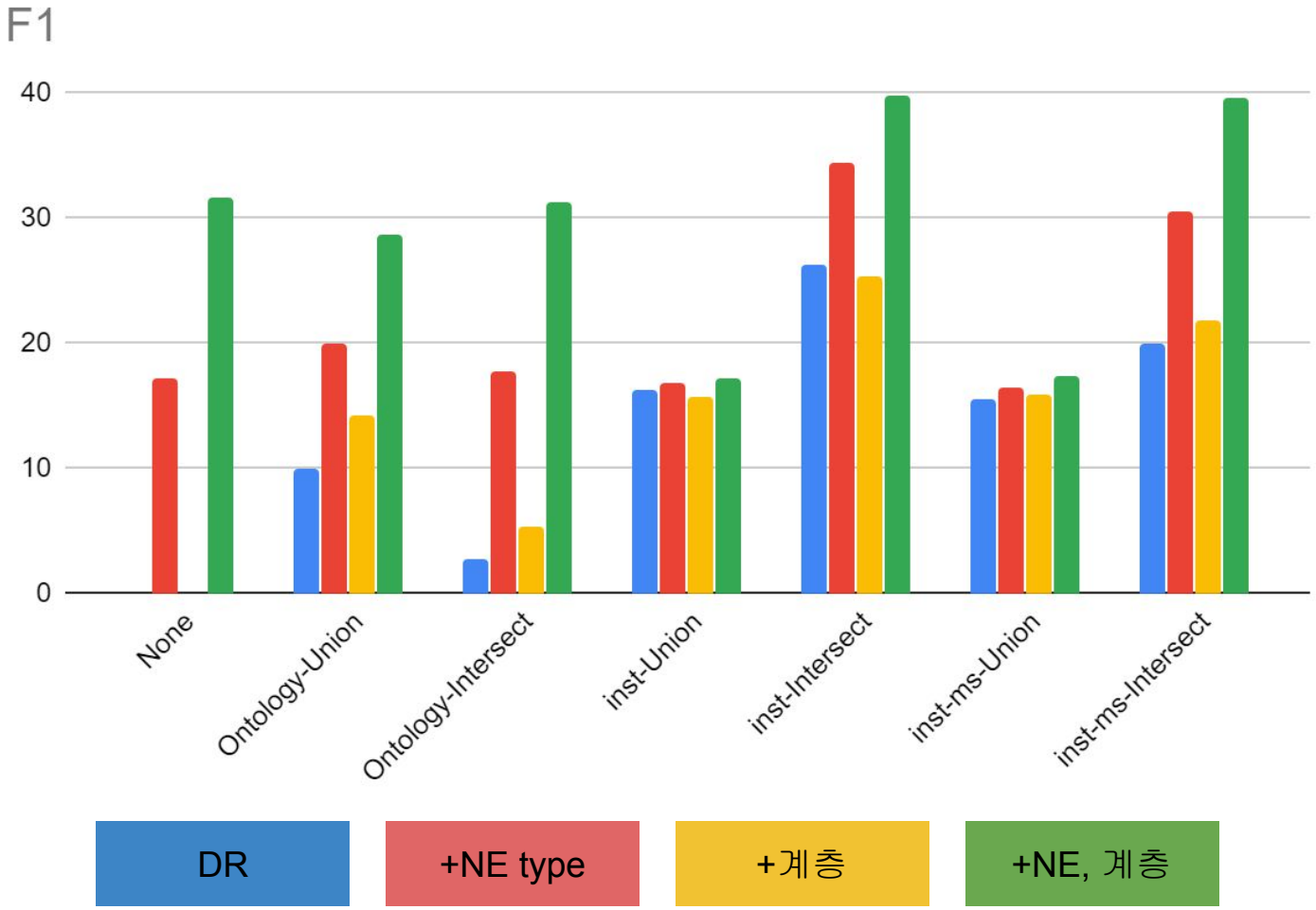
Type 부여: 실험 설계

- Note
  - 새로 부여된 타입 정보는 아직 사용하지 않음
  - D+R필터 사용하지 않음
    - Relation마다 Domain type에 specific한 Range type 정의
  - NE type은 Gold label을 수동으로 부여함
    - ETRI NER의 결과가 아닌 나무위키 링크 텍스트이기 때문에 NE type이 따로 존재하지 않음

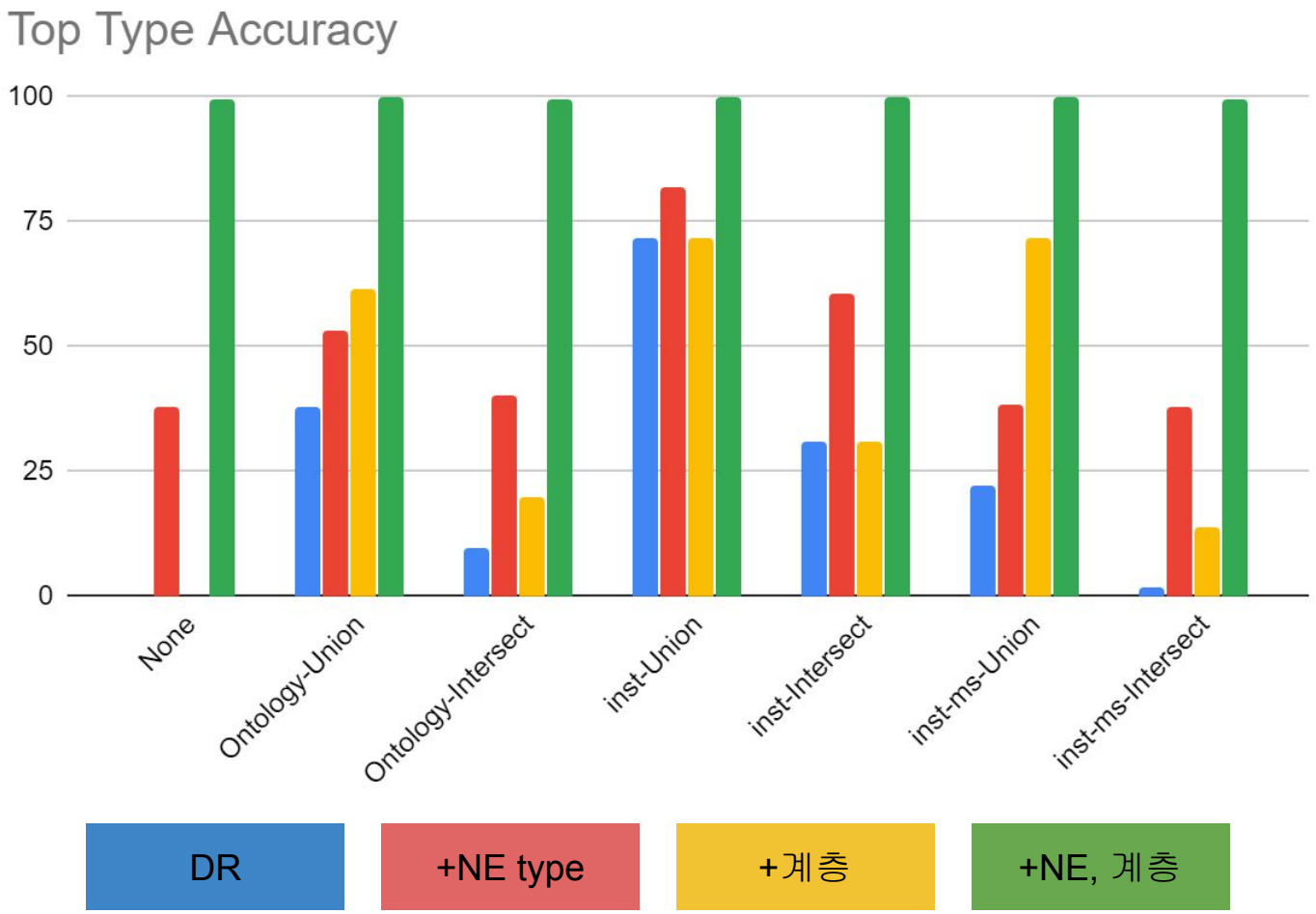
Type 부여: 실험 설계

- Metric
  - P/R/F1: 정의된 Type 전체에서의 precision / recall/ F1
  - Accuracy: Top hierarchy type을 맞춘 빈도

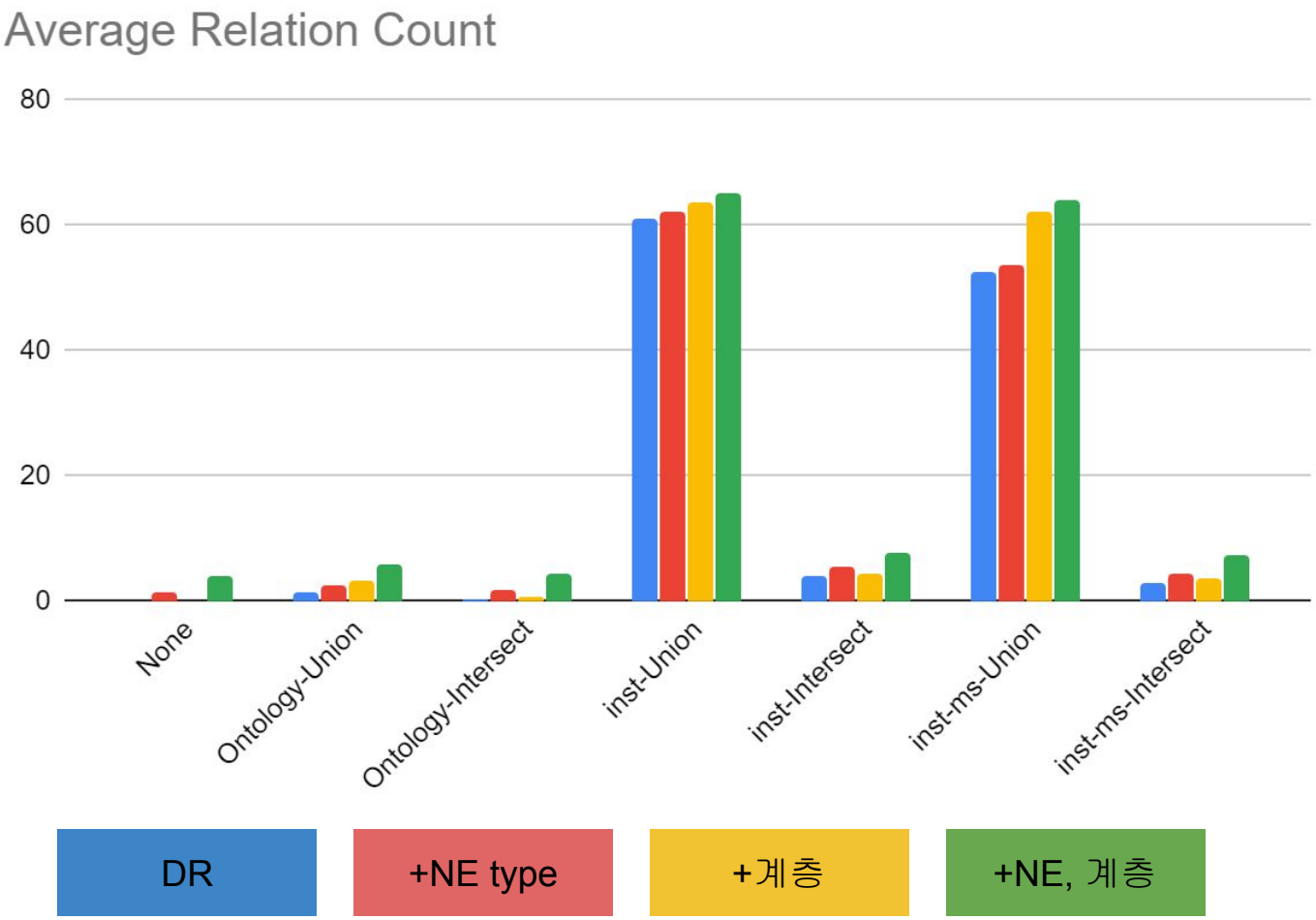
Type 부여: 실험 결과


F1
40
30
20
10
0
None
Ontology-Union
Ontology-Intersect
inst-Union
inst-Intersect
inst-ms-Union
inst-ms-Intersect
DR
+NE type
+계층
+NE, 계층

Type 부여: 실험 결과


Top Type Accuracy
100
75
50
25
0
None
Ontology-Union
Ontology-Intersect
inst-Union
inst-Intersect
inst-ms-Union
inst-ms-Intersect
DR
+NE type
+계층
+NE, 계층

Type 부여: 실험 결과


Average Relation Count
80
60
40
20
0
None
Ontology-Union
Ontology-Intersect
inst-Union
inst-Intersect
inst-ms-Union
inst-ms-Intersect
DR
+NE type
+계층
+NE, 계층

Type 부여: 분석

- NE type과 계층 타입 부여만으로 성능이 꽤 잘 나온다.
  - 하지만 정답 NE type을 부여한 상태이므로, 실제 추출에서는 어떻게 될지 미지수
- NE type을 고려하지 않을 경우, 계층 정보를 사용하지 않는 것이 대체로 유리
  - 오답뿐만 아니라 오답의 higher type이 함께 등록되게 되므로 오답이 늘어남
- Union보다는 Intersect를 사용했을 때 성능이 비교적 높음
  - Ontology 기반 타입 부여는 수가 너무 적어서 성능이 낮은 경향이 있음

Case study

* [프로듀스 101]과 그 **[시즌 2]**는 참가한 [연습생] 수가 101명이었지만 [프로듀스 48]은 참가한 연습생 수가 48명이 아니라 96명이다.

Entity: 프로듀스 101 시즌 2
Relation: subsequentWork, artist

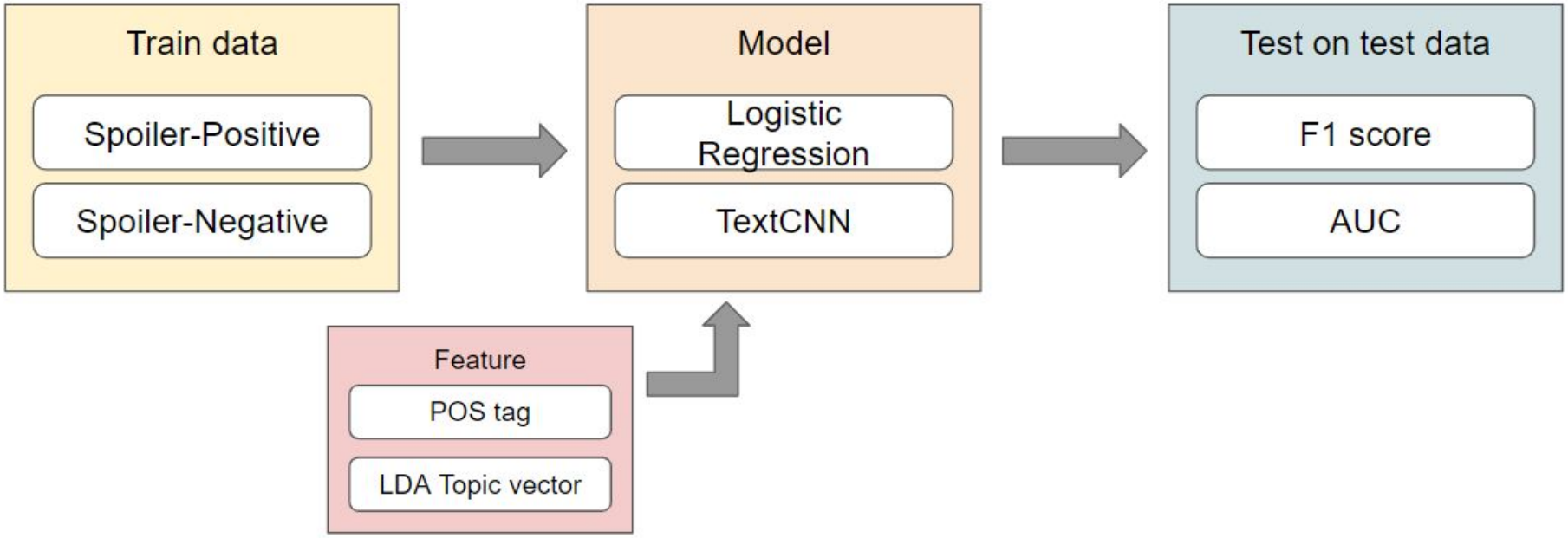
inst-intersect	MusicAlbum, MusicRecording, Work, Song, MusicalWork, Album, Single
+NE	TelevisionShow, Artifacts
+Hierarchy	InformationEntity

기타 프로젝트

연구실 외 텀 프로젝트

스포일러 감지 프로젝트

- 18년 봄 진행
- 영화 리뷰에서 스포일러가 포함된 리뷰 탐지



스포일러 감지 프로젝트

- 데이터셋
  - 영화 리뷰: 네이버 및 다음 영화 리뷰
- 사용한 방법
  - Bag of words / POS Tagging을 사용한 Document-Term Matrix
  - LDA Topic modeling
  - CNN

스포일러 감지 프로젝트

- 실험 결과

Movie	#. of topics	BOW	BOWPOS	Cosine	Topic	BOW&Topic	CNN	CNNPOS
A	2	0.682	0.894	0.236	0.229	0.868	0.567	0.722
	10			0.186	0.255	0.886		
	50			0.195	0.352	0.904		
	100			0.148	0.391	**0.915**		
T	2	0.455	0.839	0.446	0.545	0.830	0.641	0.741
	10			0.482	0.655	0.837		
	50			0.478	0.661	0.849		
	100			0.515	0.677	**0.864**		
S	2	0.552	0.841	0.482	0.578	0.833	0.747	0.764
	10			0.346	0.615	0.841		
	50			0.434	0.648	0.851		
	100			0.443	0.670	**0.865**		
I	2	0.494	0.832	0.203	0.438	0.826	0.725	0.779
	10			0.313	0.555	0.836		
	50			0.163	0.593	0.846		
	100			0.175	0.642	**0.849**		

Table 2: Experiment results in F1 score. (A=Along with the gods, T=Taken, S=Snow piercer, I=Interstella)

Rich Context Competition

- 18년 가을 진행

*"**The goal of this competition is to automate the discovery of research datasets and the associated research methods and fields in social science research publications**. Participants should use any combination of machine learning and data analysis methods to identify the datasets used in a corpus of social science publications and infer both the scientific methods and fields used in the analysis and the research fields."*

We have to do:

- Task1) Find mentions that may indicate specific dataset
- Task2) For each mention, classify what dataset it is associated to
- Task3) Find research method that the paper used
- Task4) Know what study field this paper is related to

Rich Context Competition

- 18년 가을 진행

*"**The goal of this competition is to automate the discovery of research datasets and the associated research methods and fields in social science research publications**. Participants should use any combination of machine learning and data analysis methods to identify the datasets used in a corpus of social science publications and infer both the scientific methods and fields used in the analysis and the research fields."*

We have to do:

- Task1) **Find mentions that may indicate specific dataset =** 개체명 인식
- Task2) **For each mention, classify what dataset it is associated to =** 개체 연결
- Task3) Find research method that the paper used
- Task4) Know what study field this paper is related to

Task1) Dataset Mention Detection Module

Goal & Approach

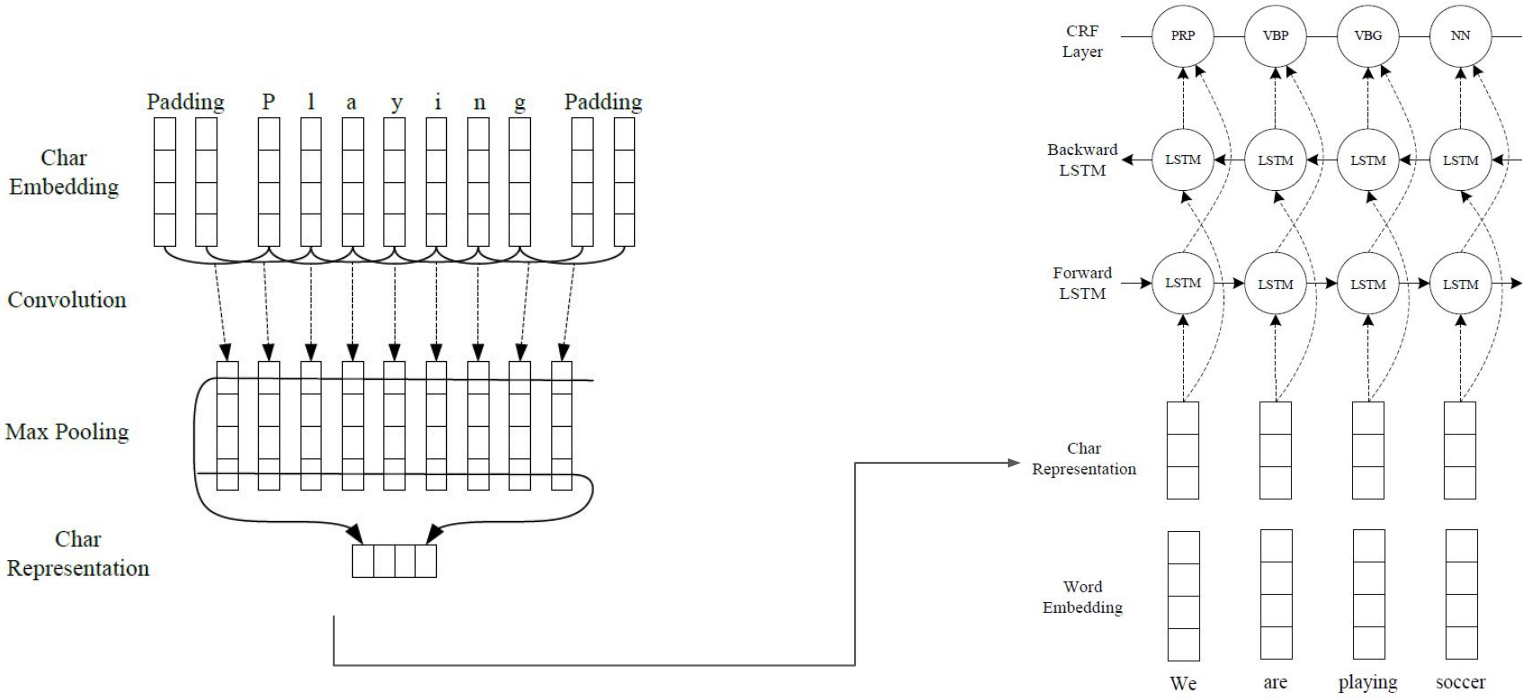
- **Goal: Find mentions that may indicate dataset**
- Approach
  - Used Named Entity Recognition(NER) technique
  - Only aims to find Dataset Mentions

Model

- Deep learning model using CNN, BiLSTM, CRF [1]
- Showing State-of-the-art performance on NER and POS Tagging
- Standard DL Model of sequence labeling and easy to modify

[1] Ma and Hovy, End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF, ACL, 2016

Model



Apply CNN at character level to get morphological feature

Apply BiLSTM and CRF to get NER tags

Result

We analyzed data from the Early Childhood Longitudinal Study-Kindergarten Class (ECLS-K).[1] (For a description of the ECLS-K, see http://nces.ed.gov/ecls/pdf/essaysmisc/


```
{
    "publication_id": 1067,
    "mention": "the Early Childhood Longitudinal Study",
    "score": 1
},
{
    "publication_id": 1067,
    "mention": "ECLS-K",
    "score": 1
},
```

Result

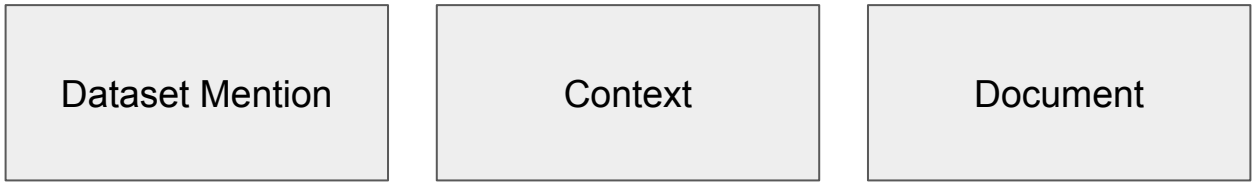
- Only trained on mention-positive sentences
- Training:Dev = 9:1
- Dev set score
  - Precision 81.4 / Recall 77.3
  - F1-score 79.3

Task2) Dataset Classification Module

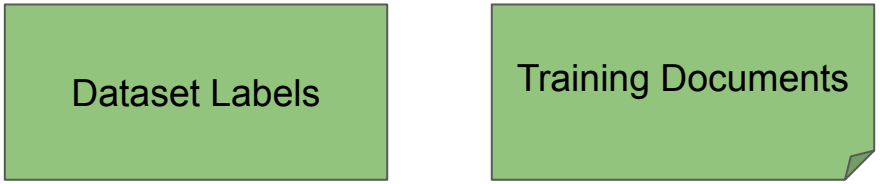
Goal & Approach

- **Goal: For each mention, classify what dataset it is associated to**
- Approach
  - Similarity based selection - Doc2Vec, Word embedding(GloVe)

Model

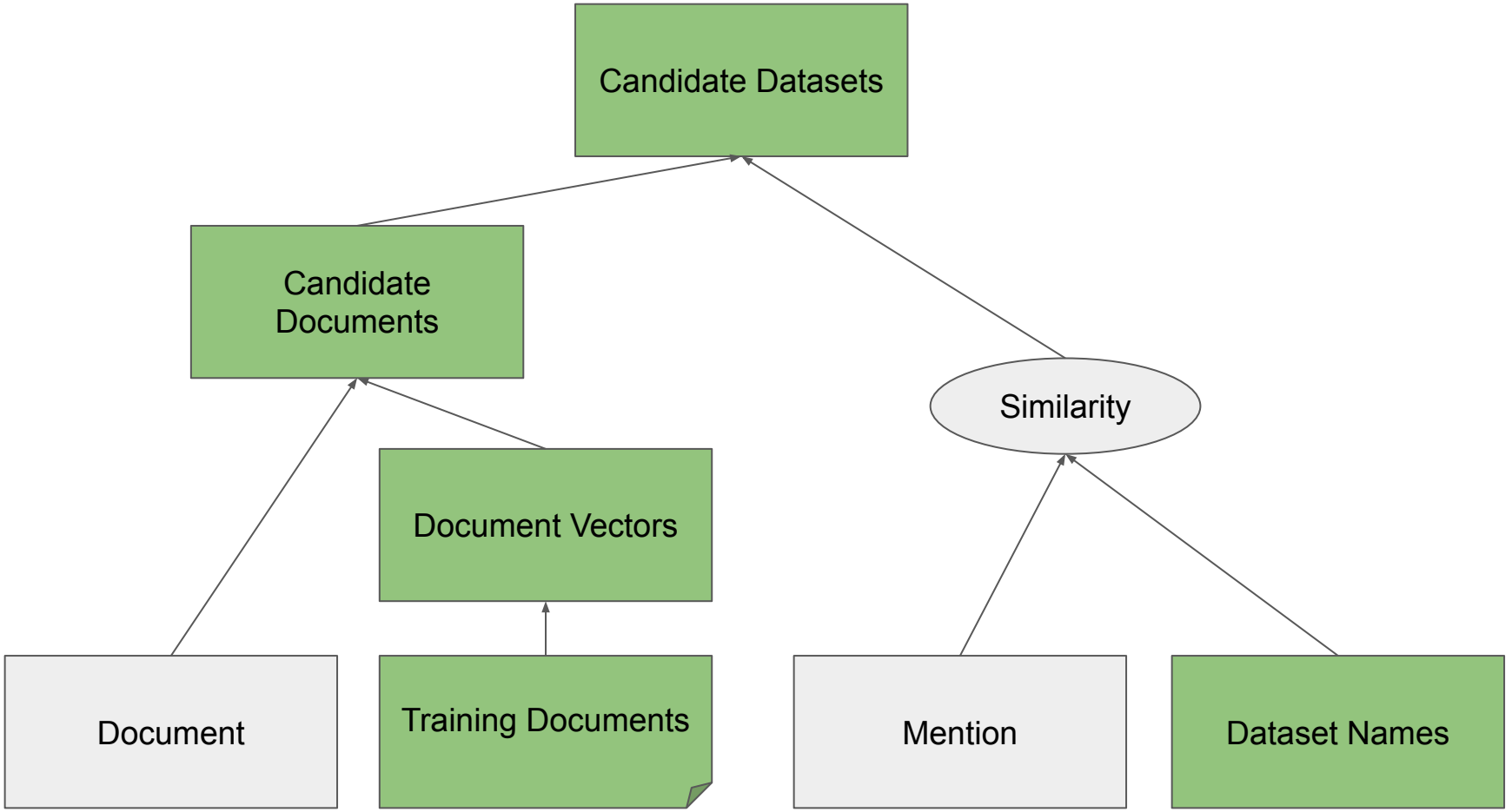


Extracted from Mention Detection Module

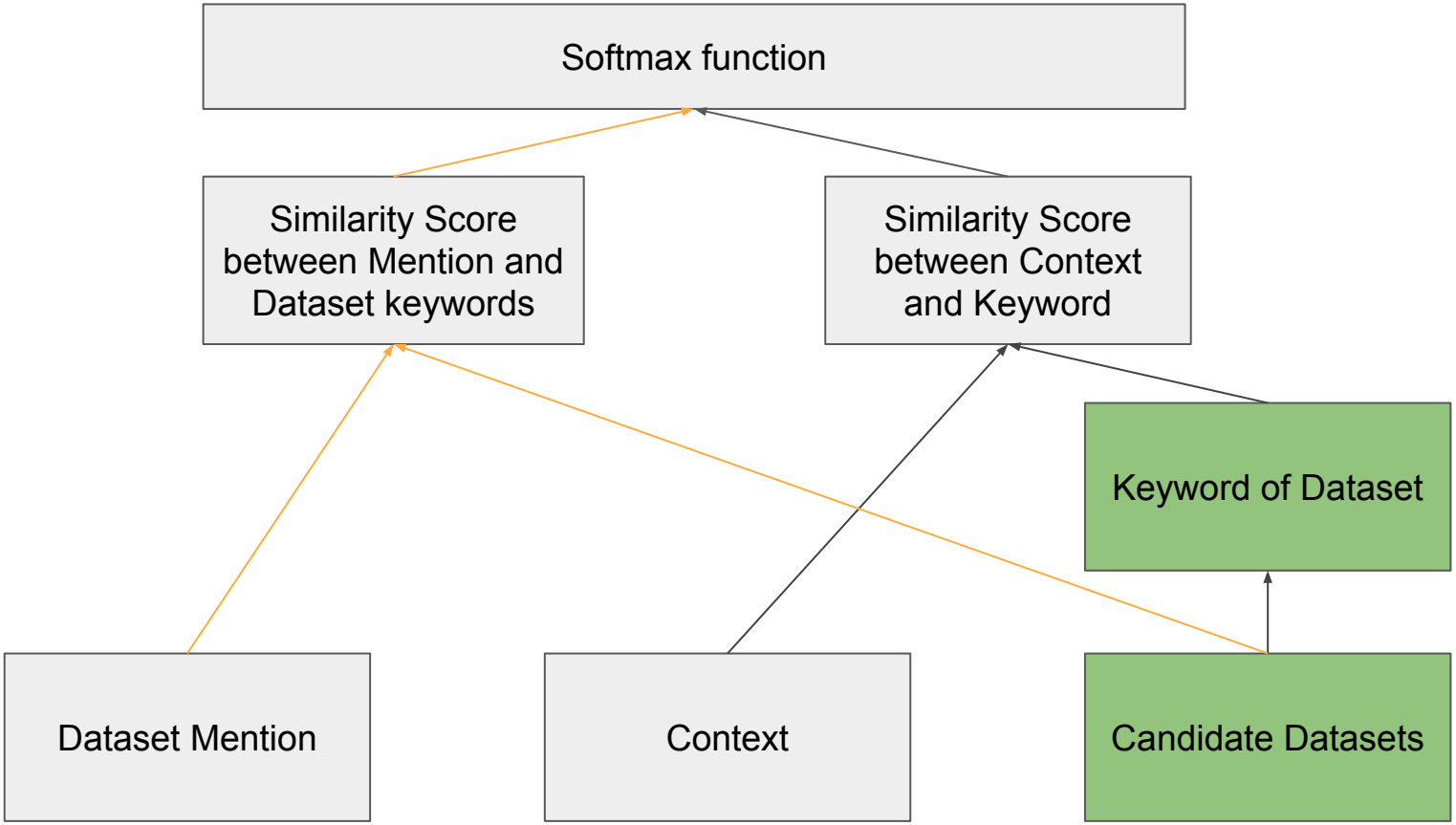


Given Dataset

Model


Candidate Datasets
Candidate Documents
Similarity
Document Vectors
Document
Training Documents
Mention
Dataset Names

Model


Softmax function
Similarity Score between Mention and Dataset keywords
Similarity Score between Context and Keyword
Keyword of Dataset
Dataset Mention
Context
Candidate Datasets

Result

Prediction Example

Dataset Description


```
{
    "confidence": 0.8645003455307727,
    "publication_id": 1935,
    "mentions": [
        "the 19992002 National Health and Nutrition Examination Survey",
        "NHANES",
        "National Health and Nutrition Examination Survey",
        "The Third National Health and Nutrition Examination Survey",
        "the National Health and Nutrition Examination Survey",
        "NHANES III"
    ],
    "dataset_id": 481
},
```



```
{
    "data_set_id": 481,
    "unique_identifier": "10.3886/ICPSR25501",
    "title": "National Health and Nutrition Examination Survey (NHANES), 1999-2000",
    "name": "National Health and Nutrition Examination Survey (NHANES), 1999-2000",
    "description": "The National Health and Nutrition Examination Surveys (NHANES) is a program of studies designed to assess the health and nutritional status of adults a
    "date": "2012-02-22 00:00:00+00:00",
    "coverages": "",
    "subjects": "acculturation,aging,alcohol consumption,allergies,anxiety,cardiovascular disease,cognitive functioning,consumer behavior,demographic characteristics,depre
    "methodology": "",
    "citation": "",
    "additional_keywords": "plos_oa",
    "family_identifier": "354",
    "mention_list": [
```