단위 지식 간 관계 추출 모델 성능 분석서

버전 0.1

2021. 12. 16 김산 (KETI)

Revision History

수정	시작 날짜	끝난 날짜	작성자	설명
V0.1	2021.12.16	2021.12.18	김산	기술문서 초안 작성

목차

1 문서 개요

1.1 개요

본 문서는 비정형 텍스트를 학습하여 쟁점별 사실과 논리적 근거 추론이 가능한 인공지능 원천기술 과제의 기술문서이다. 본 과제는 명시적/암시적 추론 문제의 질의에 대한 논리적 응답을 위해, 비정형 텍스트에서 근거 문서를 검색하고, 근거 문서와 지식 베이스에서 단위 지식(문단, 문장, 개체 등) 간 관계(선후관계, 인과관계 등)를 추출하여 질의에 대한 사실과 논리적 근거를 추론하는 인공지능 기술 개발을 목표로 한다. 이 과제의 개발 기술들 중 하나인 단위 지식 간 관계 추출 모델은 문서내에서 두 엔티티간의 관계를 추출하는 것을 목적으로 한다. 단위 지식간 관계 추출 성능 분석서는 관계 추출 모델의 구조와 설계 내용 및 성능 분석을 포함한다.

1.2 구성 및 범위

본 문서는 다음과 같은 범위와 내용을 기술한다.

- 데이터
- ㅇㅇ
- ㅇㅇ

1.3 용어 정의

2 단위 지식 간 관계 추출 모델 구조

2.1 개요

단위 지식 간 관계 추출은 자연어 문서에서 두 엔티티 간의 관계를 추출하는 것이다.

그림1은 단위 지식 간 관계추출의 예시를 보여주고 있다.

그림1에서 'COVID'와 '박쥐'의 관계는 '기생'이라고 할 수 있다.

또한 그림 1에서 보듯이 두 엔티티 간의 관계는 1개 이상이 될 수 있다.

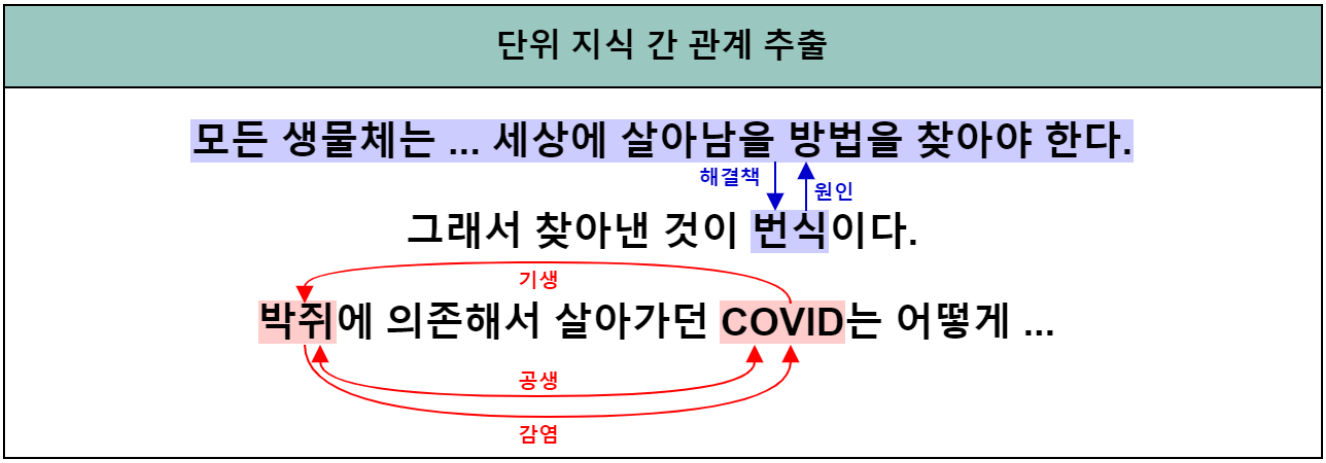

그림 1. 단위 지식 간 관계 추출 예시

문서 내 엔티티 간 관계 추출의 경우 후보 엔티티를 선정하고 엔티티 간 관계들을 분류기로 분류하여 관계 트리플을 추출해야한다.

그림2는 법률 관련 문서에서 엔티티 간 관계 추출 예시를 보여준다.

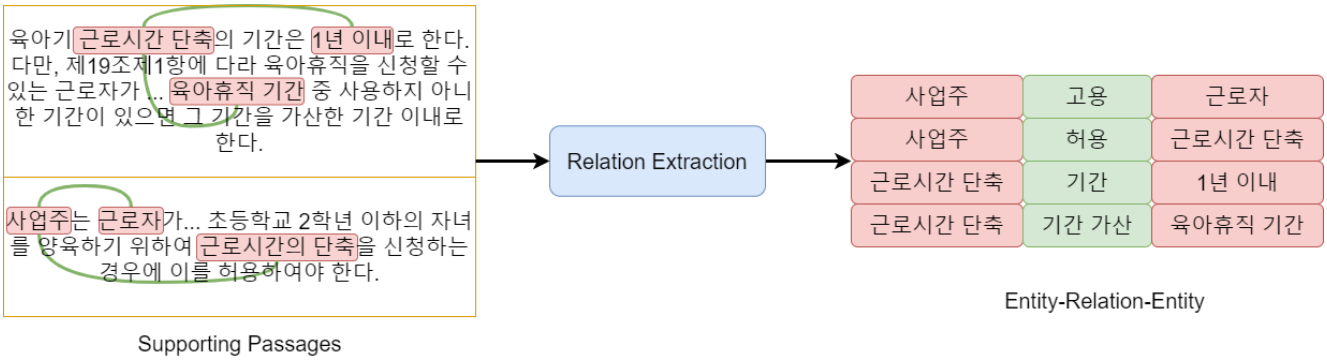

그림 2. 문서내에서 단위 지식 간 관계 추출의 예시 (법률 도메인)

2.2 모델의 구조

문서내 관계추출 모델의 구조는 그림3과 같다.

먼저 형태소 분리 및 Part of Speech 태거를 이용하여 후보 엔티티를 추출한다.

고유 명사를 후보 영역으로 선정하기 위하여 Named Entity Recognition(NER)을 통해 후보 엔티티를 추출한다.

Coreference Resolution을 이용하여 겹치는 영역들을 통합한다.

통합한 영역들 간 관계 분류기로 관계를 추출한다.

추출한 Subject-Relation-Object 트리플들을 Relation으로 정렬한다.

그림 3. 문서 내에서 단위 지식 추출 및 단위 지식 간 관계 추출 모델의 개념도

앞서 말한 방법으로 추출된 트리플들 중에서 중복된 트리플들을 그림 4와 같이 제거한다.

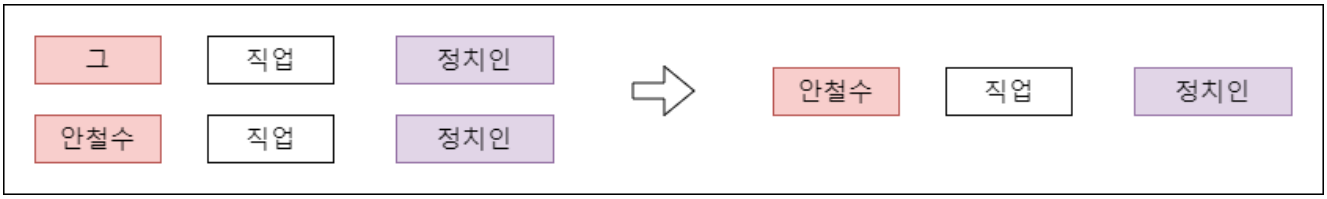

그림 4. 중복된 Subject-Relation-Object 트리플들을 제거하는 예

그림 4에서 '그-직업-정치인', '안철수-직업-정치인'은 같은 의미를 가지는 트리플이므로 이 경우 고유명사인 '안철수-직업-정치인'을 남기고 나머지 트리플들은 제거한다.

2.3 관계 분류 모델의 구조

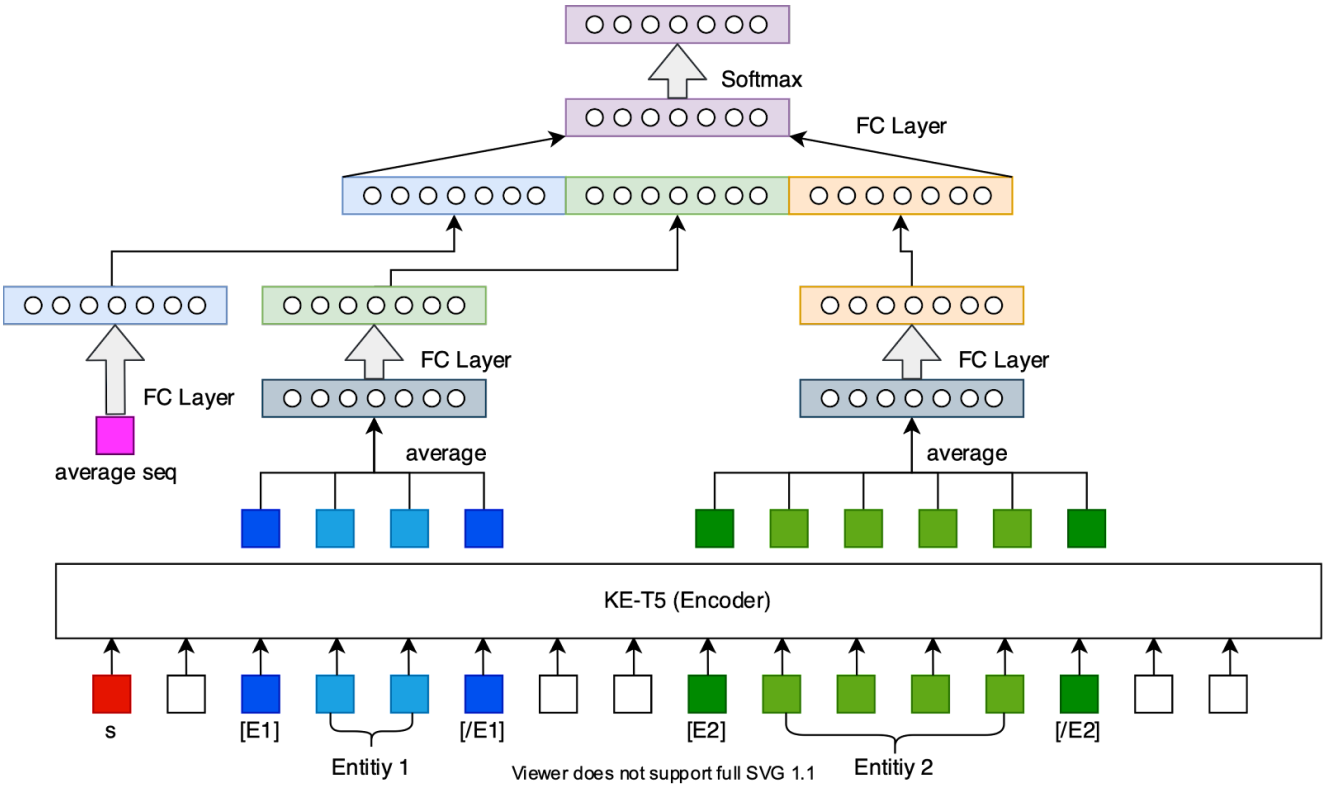

그림 5. 관계 분류 모델의 구조도

관계 분류 모델의 구조도는 그림 5와 같다. 문서를 토크나이저로 토큰화 한뒤, 관계를 추출하고자 하는 Subject Span과 Object Span 앞 뒤로 특수 토큰을 추가한다.

이를 입력으로 하여 인코더를 통과 시킨 후 Subject Span과 Object Span의 벡터들을 각각 average pooling하여 평균 벡터를 구한다. 또한 전체 시퀀스의 평균 벡터 또한 구한다. 이렇게 구한 평균 벡터들을 concatenation한 뒤 Fully Connected 레이어를 이용하여 Relation 차원의 벡터로 만든 후 softmax를 취해 분류를 진행한다.

2.4 개발된 관계 분류 모델의 성능

표 1. 개발된 관계 분류 모델의 성능

Base model	F1 Micro
RoBERTa-base	66.66
RoBERTa-large	69.59
KE-T5 base (encoder)	70.47

개발된 관계 분류 모델의 경우 기존 개발된 모델들 보다 높은 성능을 보이며, 1차년도 연구 목표인 65보다 높은 성능을 보임

KLUE RoBERTa Large의 경우 355M parameters로 109M parameters인 KE-T5 base 인코더 기반 모델보다 모델 크기가 3배 큰데도 불구하고 성능은 KE-T5 base 인코더 모델이 약 1 높음

개발된 모델이 더 적은 parameters를 사용함에도 불구하고 우수한 성능을 보임

3 모델 인터페이스

표 2. 개발된 모델의 입력 명세

Key	Value	Default Value	Explanation
Doc	Str	(required)	분석할 문서

Arg_pairs	List[Tuple[Tuple[int, int], Tuple[int, int]]]	None	관계를 알고 싶은 Entity 쌍들의 문서에서의 시작, 종료 위치. 주로 주어와 목적어 쌍. E.g. [[(3, 8), (14, 22)], ((46, 47), (57, 60)), ...].

표 3. 개발된 모델의 출력 명세

Key	Value	Explanation
Num_of_triples	Int	반환된 triple의 개수
Triples	List[Tuple[str, str, str]]	문서에서 추출된 Arg0, v, Arg1 트리플 리스트

- 입력 예시

```
{
  "doc": "문성민은 경기대학교에 입학하여 아내인 이선희와 함께 경기대학교의 전성기를 이끌면서 하계대회, 전국체전, 최강전 등 3관왕을 이룬다.",
  "arg_pairs":[
    [
      [0,2],
      [5,8]
    ],
    [
      [0,2],
      [21,23]
    ]
  ]
}
```

- 출력예시


```
{
  "result":
    [
      {
        "subject": "문성민",
        "relation": "per:schools_attended",
        "object": "경기대학"
      },
      {
        "subject": "문성민",
        "relation": "per:spouse",
        "object": "이선희"
    }
  ]
}
```


4 문서 내 단위 지식 추출 데이터 셋 명세

- 단위 지식 관계 트리플의 예 1

```
parsing ko_dbpeida_nt/mappingbased_properties_unredirected_ko.nt.bz2 # OK
0 :  (rdflib.term.URIRef('http://ko.dbpedia.org/resource/래티티아_카스타'), rdflib.term.URIRef('http://dbpedia.org/ontology/activeYearsStartYear'), rdflib.term.Literal('1993-01-01', datatype=rdflib.term.URIRef('http://www.w3.org/2001/XMLSchema#gYear')))
1 :  (rdflib.term.URIRef('http://ko.dbpedia.org/resource/디네일루릭티스'), rdflib.term.URIRef('http://xmlns.com/foaf/0.1/name'), rdflib.term.Literal('디네일루릭티스', lang='ko'))
2 :  (rdflib.term.URIRef('http://ko.dbpedia.org/resource/링_(1998년_영화)'), rdflib.term.URIRef('http://dbpedia.org/ontology/distributor'), rdflib.term.URIRef('http://ko.dbpedia.org/resource/도호'))
```

- 단위 지식 관계 트리플의 예 2

```
parsing ko_dbpeida_nt/mappingbased_properties_unredirected_ko.nt.bz2 # OK
0 :  (rdflib.term.URIRef('http://ko.dbpedia.org/resource/Forever_Memories'), rdflib.term.URIRef('http://ko.dbpedia.org/property/곡명'), rdflib.term.Literal('Forever Memories', lang='ko'))
1 :  (rdflib.term.URIRef('http://ko.dbpedia.org/resource/선림원지_삼층석탑'), rdflib.term.URIRef('http://ko.dbpedia.org/property/그림'), rdflib.term.Literal('Seollimwonji 01.JPG', lang='ko'))
2 :  (rdflib.term.URIRef('http://ko.dbpedia.org/resource/김민하'), rdflib.term.URIRef('http://ko.dbpedia.org/property/프로입단연도'), rdflib.term.Literal('2011', datatype=rdflib.term.URIRef('http://www.w3.org/2001/XMLSchema#integer')))
```

- 구축된 데이터의 관계별 분포

구축된 데이터의 개수는 133,058개의 트리플이며, 관계의 개수는 총 308개 이다.

구축된 데이터의 관계별 데이터 분포는 그림6과 같다. 관계들 중 트리플의 갯수는 'name'이 가장 많으며, 'occupation', 'address', 'birthPlace' 순이다.

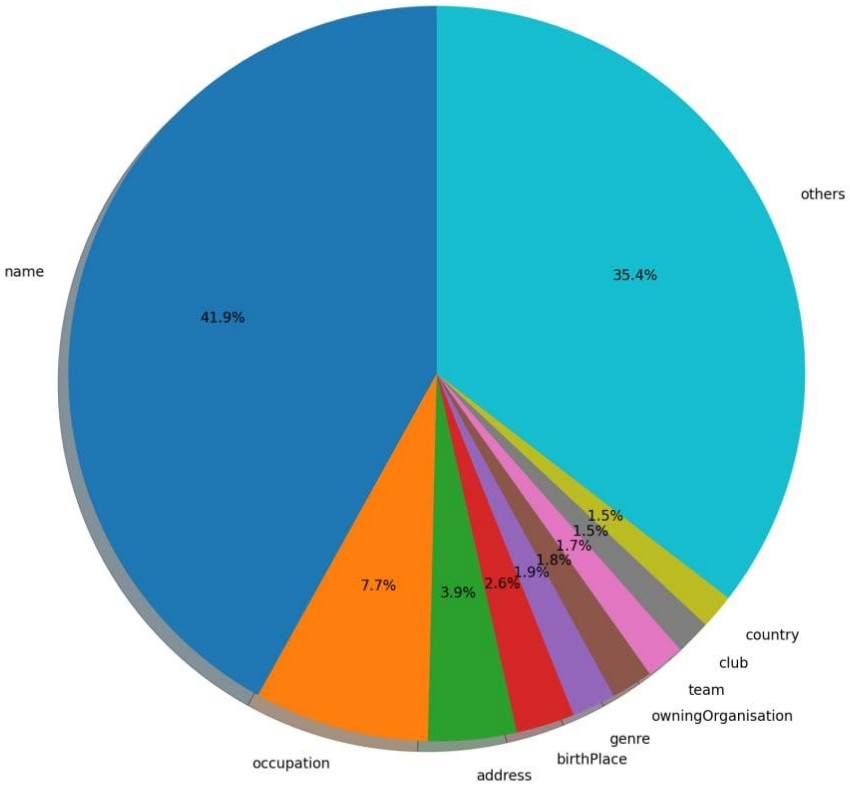

그림 6. 구축된 트리플들의 관계별 분포