

# 뉴럴-심볼릭 구조 기반의 관계 추출

오진영<sup>0</sup>, 차정원

창원대학교

[psycheojy@cwnu.ac.kr](mailto:psycheojy@cwnu.ac.kr), [jcha@cwnu.ac.kr](mailto:jcha@cwnu.ac.kr)

## Relation Extraction based on Neural-Symbolic Structure

Jinyoung Oh<sup>0</sup>, Jeong-Won Cha  
Changwon National University

### 요약

딥러닝은 자연어처리 분야에서 우수한 성능을 보이고 있다. 하지만 우수한 성능을 달성하려면 많은 학습 데이터와 오랜 학습 시간이 필요하다. 우리는 딥러닝과 기호 규칙을 함께 사용하는 뉴럴-심볼릭 방법을 이용하여 딥러닝만으로 학습한 모델의 성능을 능가하는 방법을 제안한다. 딥러닝의 한계를 극복하기 위해서 관계추출에서 규칙 결과와 딥러닝 결과와의 불일치도를 추가한 구조를 설계하였다. 제안한 구조는 한국어 데이터에 대해서 우수한 성능을 보였으며, 빠른 성능 수렴이 이루어지는 것을 확인하였다.

주제어: 뉴럴-심볼릭, 관계추출, 딥러닝

### 1. 서론

딥러닝 학습은 목표(Target)의 사전 정보, 기준/조건, 유용한 자질 등을 정의하지 않고 데이터를 통해 추상화하고, 특징을 추출하여 학습한다. 이 중 학습 데이터의 크기는 성능을 높이는데 영향을 미치는 항목으로 대용량을 활용할수록 성능은 높아지지만 파라미터가 늘어나면서 학습 시간과 계산 비용이 많이 요구된다. 사전 학습 모델 중 많이 알려진 BERT[1]는 3300백만 단어를 학습하였고, [2]는 750기가 데이터를 학습하여 멀티도메인에 적용하여 높은 성능을 보여주었다. 그 외 [3, 4]는 번역 모델에서 정답 학습 쌍 이외에 역번역(back-translation)을 통하여 얻어진 실버 문장 쌍을 추가함으로써 성능을 높였다. 이와 같이 많은 연구들이 목표 분야로 구축된 데이터를 대용량으로 사용함으로써 성능 향상을 달성하였다.

이와 비교하여 딥러닝 연구 이전에서는 이산적인 기호 및 표현으로 구축된 지식을 활용한 연구가 많았다. 예를 들어 단어의 상하위 관계와 같은 사전을 사용하거나 학습자가 판단한 자질로 규칙을 구성하는 등의 방법을 활용한다. 이와 같은 지식은 간결하게 표현이 가능하고, 성능에 대한 영향도가 높으며 직관적이다. 하지만 많은 지식으로 인해 발생하는 충돌을 처리해야하며 학습자의 통찰에 따른 성능차이가 발생한다.

본 논문은 딥러닝의 우수한 학습 구조에 심볼릭 지식을 활용한 통합 구조를 제안하고자 한다. 즉, 데이터를 통하여 은닉된 특징을 추출하고 일반화 하는 딥러닝 학습에 지식을 활용함으로써 두 방법의 장점을 살린다면 적은 데이터로 대용량의 데이터를 활용한 만큼의 성능을 보일 수 있을 것이다. [5]는 WordNet과 Freebase를 임베딩에 추가하여 딥러닝 모델에 적용한 연구로 관계 추출의 성능을 높였다. [6]은 규칙을 사용하여 감성 분석에

서 문장 내 감성 자질이 표현된 부분을 찾는 규칙을 정의하고 규칙에 부합한 구간을 추출하여 학습에 추가하는 방법으로 성능을 높였다. [7]은 관계 추출에서 필요한 규칙을 1차 논리(First-order logic)로 정의하고 딥러닝 결과와 논리의 손실 값을 비교하여 얻은 불일치 손실 값으로 성능을 높였다. 하지만 논리에 일치(Positive correlation)하는 손실 값만을 구하여 초기 학습에서는 논리의 손실 값이 거의 적고, 학습 진행 중에는 딥러닝 모델 성능이 높아지므로 학습 전반에 논리 손실 값의 영향이 미미하다.

지식을 통합하는 방법은 여러가지가 있다. [5]와 같이 딥러닝 모델 분석에 필요한 자질 또는 어휘에 해당하는 정보를 가져와 흡수하는 방법, [6]과 같이 입력 값을 조절하는 방법, [7]과 같이 학습자가 구성한 논리 구조의 지식을 이용한 통합 등이 있다. 딥러닝과 지식 통합을 시도하는 연구의 대부분은 은닉층으로 생성된 딥러닝 모델에 지식을 통합하는 방법과 손실 함수에 대해 고려한다. 하지만 일반적으로 적용 가능한 학습 방법을 정의하기가 어려워 실험 도메인 분야에 맞춰진 것이 많다.

본 논문은 관계 추출 분야에서 트랜스포머(Transformer)를 활용하여 개체명 간 관계를 추출하고 트랜스포머 결과를 심볼릭 지식을 이용하여 제한하는 방법을 설명한다. 심볼릭 지식은 트랜스포머의 결과에 대해 제한/지도를 주고 올바른 방향으로 진행하도록 도와 성능을 높이는 효과를 준다. 즉, 트랜스포머의 결과에 재현율을 높이기 위한 비어휘(delexical) 심볼릭 지식을 사용하여 딥러닝 모델에 비교하여 관계 추출의 성능이 향상되고, 성능의 수렴 속도가 빨리 이루어지는 것을 확인하였다. 세부 구조는 2장에서 설명한다.

### 2. 제안 방법

제안 구조는 그림 1과 같다. 크게 딥뉴럴 네트워크

(Transformer)와 1차 논리(Logic) 모듈로 구성된다. 딥뉴럴 네트워크는 문장을 입력 받아 개체명과 관계를 예측한다. 논리 모듈은 심볼릭 지식을 1차 논리로 정의하고, 딥뉴럴 네트워크의 예측 결과를 논리 모듈에 입력받아 정의된 논리 규칙의 결과와 비교한다. 논리 규칙의 결과와 딥뉴럴 네트워크의 예측 결과가 일치하지 않으면 딥뉴럴 네트워크의 손실 값( $L_{NN}$ ) 이외에 논리 모듈의 손실 값( $L_{logic}$ )을 많이 반영하는 구조이다.

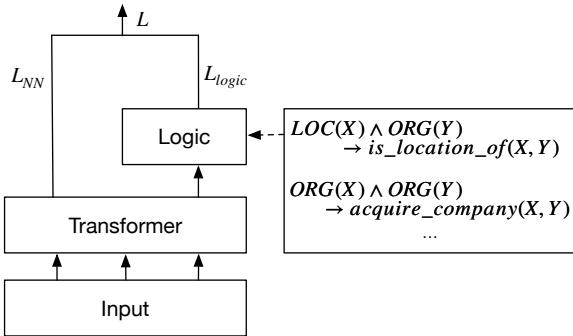


그림 1 딥뉴럴-심볼릭 구조 기반의 관계 추출

## 2.1 1차 논리

심볼릭 지식은 1차 논리 규칙( $\Phi$ )인 (1)과 같이 구성한다. 집합  $\Phi$ 에 포함되는 조건은 해당 조건을 만족하는 조건( $B, C, E$ )과 제외할 조건( $\neg F$ )으로 표현 가능하다.

$$\Phi = \{A \leftarrow B \wedge C, D \leftarrow E \wedge \neg F\} \quad (1)$$

본 논문에서는 관계 추출에 필요한 심볼릭 지식으로  $\Phi$ 를 각 관계에 필요한 개체명 분류로서 제한할 수 있는 논리 규칙을 정의한다. (2)는 논리 규칙의 예이다.

$$LOC(X) \wedge ORG(Y) \rightarrow is\_location\_of(X, Y) \quad (2)$$

화살표를 기준으로 왼쪽 두 항목(LOC, ORG)은 조건으로 개체명 분류가 되고, 오른쪽은 관계(is\_location\_of)를 나타낸다. 딥뉴럴 예측 결과는 문장에서 추출한 개체명과 개체명 간 관계가 성립되면 관계를 예측하여 나타낸다. 예를 들어 (2)는 딥뉴럴 결과 중 관계가 is\_location\_of이고 관계를 구성하는 두 개의 개체명 분류가 LOC, ORG의 조건과 일치하지 않은 경우 추가 손실 값을 부여한다.

## 2.2 딥뉴럴 네트워크 모듈

딥러닝 모델에서 대용량 데이터를 사용한다는 것은 다양한 문맥과 자질을 학습하기 위함이다. 다양한 문맥으로 학습할 수 없는 경우 새로운 단어 또는 문맥(Out of Vocabulary)에 대한 성능이 떨어질 수 밖에 없다.

본 논문에서 제안하는 논리 규칙은 개체명의 어휘에서 오는 자질을 통해 얻어지는 기존의 딥뉴럴 네트워크 예측 결과를 개체명의 어휘가 아닌 분류(ORG, LOC, ...)를 기준으로 손실 값( $L_{logic}$ )을 계산한다. 어휘의 상위 카테고리 개념을 통하여 넓은 범위를 통합한 손실 값을 계산하고자 함이다. 이는 관계의 재현을 성능을 높이는데

기여한다.

마지막으로 두 모듈에서 발생한 손실 값( $L_{NN}$ ,  $L_{logic}$ )을 통합한 손실 값  $L$ 의 정의는 다음과 같다.  $L_{NN}$ 은 트랜스포머의 결과 값과 정답을 비교한 손실 값을 사용하고, 논리 모듈에서 구하는 손실 값은 식 (3)과 같이 계산한다.

$$L_{logic} = \frac{1}{K} \sum_{\phi \in \Phi} \beta_{\phi} D(T(x), LM(T(x), \phi)) \quad (3)$$

$T(x)$ 는 입력 문장  $x$ 의 개체명과 개체명 간에 관계를 포함한 트랜스포머 결과이고,  $LM(T(x), \phi)$ 는 트랜스포머의 개체명, 관계의 결과를 논리 규칙  $\phi$ 에 적용한 결과이다.  $D$ 는 둘의 차이를 구하는 함수이고 평균 제곱 오차(Mean-squared Error)를 사용한다. 즉, 트랜스포머 결과가 논리 규칙과 일치하는지를 확인한다. 예를 들어 입력 문장이  $x = \{w_1, w_2, \dots, w_n\}$ 이라면  $T(x)$ 를 통해서 각 요소( $w_i$ )들은 개체명 범주 별로 값을 가지게 되며 최고의 값이 요소의 개체명 범주가 된다.  $w_1$ 이 LOC,  $w_4$ 가 ORG이며 둘 개체명의 관계를 acquire\_company로 예측했다고 가정하자.  $LM(T(x), \phi)$ 에서는 이 결과를 입력으로 받아 논리규칙을 탐색한다. 관계 acquire\_company에서는  $w_1$ 가 ORG가 논리 규칙에 해당하므로  $w_1$ 의 ORG 범주 값과  $T(x)$ 이 예측한 LOC의 차이를  $D$ 함수가 계산하는 것이다. 그 외  $\beta_{\phi}$ 는 각 논리 규칙에 정의된 가중치이고  $K$ 는 입력  $x$ 에 매칭된 논리 규칙의 수를 의미하는 것으로 전체 규칙의 평균을 구한다. 이후 식 (4)와 같이 트랜스포머의 손실 값과 계산하여 최종 손실 값을 구한다.

$$L = L_{NN} + (1 - \alpha)L_{logic} \quad (4)$$

논리 규칙에서 얻어진 손실 값을  $\alpha$ 비율에 적용하는 것으로 비율에 따라 성능 차이가 있다. 본 논문에서는 0.5로 설정한다. 즉 논리 규칙의 손실 값의 절반을 반영한다는 의미이다.

## 3. 실험

본 논문에서는 뉴럴-심볼릭 구조의 효과를 확인하기 위하여 관계 추출 실험을 구성하였다.

### 3.1 실험 설정

실험에는 다음과 같은 조건으로 구축한 코퍼스를 사용한다. 먼저 코퍼스는 신문에서 추출한 문장이며, 각 문장에는 개체명과 개체명 간 관계가 정의 되어있다. 실험에는 1,933의 문장(학습 1740문장, 평가 193문장)을 사용하였으며 모든 문장에는 1개 이상의 관계를 포함한다. 뉴럴-심볼릭 구조의 효과를 위하여 2개의 관계와, 연관된 2개의 개체명을 선정하여 실험하였으며, 정보는 표 1과 같다.

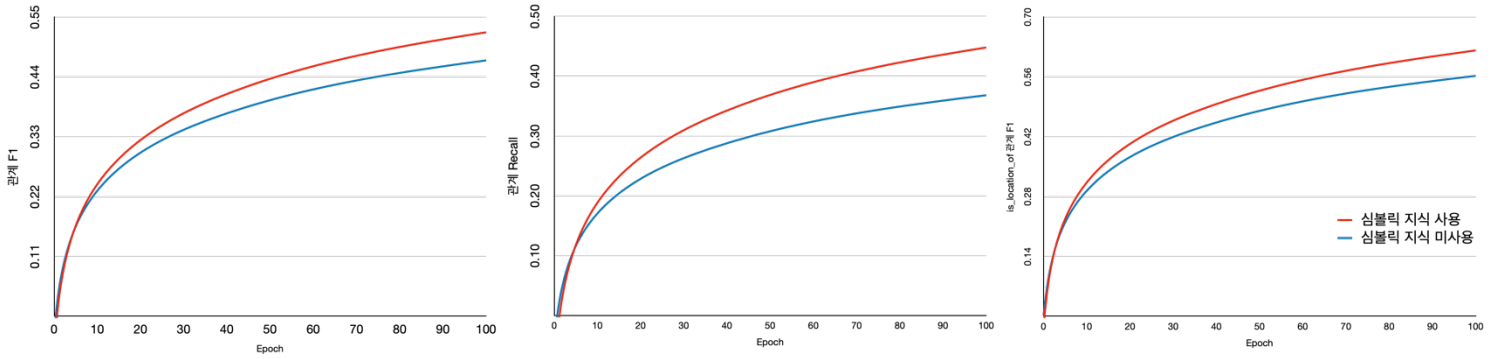


그림 2. 심볼릭 지식 사용-미사용에 따른 관계 성능(F1, Recall). 왼쪽은 관계의 F1 성능, 중간은 관계의 재현율, 오른쪽은 is\_location\_of의 F1 성능을 표시한다.

표 1. 개체명, 관계 정의

개체명	LOC	지역/장소와 지형/지리 명칭 예) 한국, 미국, 서울, 창원
	ORG	기관/단체 명칭 예) ~호텔, ~회사
관계	is_location_of	조직, 행사 등의 위치 문장 예) 경상남도청은 창원에 위치한다.
	acquire_company	인수한 조직 문장 예) 롯데가 충청북도 소주업체 '충북소주'를 인수한다.

관계는 표 1에 포함된 문장 예와 같이 밑줄로 표시된 개체명( $N_s$ ,  $N_e$ )에 연관됨을 의미한다.  $N_s$ 와  $N_e$ 는 대상 관계에 따라 순서가 존재하며 역은 성립하지 않는다. 단, 관계 중 acquire\_company는 구성하는 개체명의 조건이 조직(ORG)에만 해당하므로 1차 논리로 표현하면 역의 표현도 동일하다.

본 논문에서는 관계 추출에 심볼릭 지식 제한을 추가하여 성능을 향상하고자 한다. 심볼릭 지식 제한은 트랜스포머의 관계 결과를 확인하고 해당 관계와 관련된 개체명의 구성이 각 관계의 조건과 다를 경우 손실 값을 추가로 부여한다.

### 3.2 심볼릭 지식

2.1에서 표현한 것과 같이 각 관계에 대하여 1차 논리를 정의한다.

$$LOC(X) \wedge \neg PER(Y) \rightarrow is\_location\_of(X, Y) \quad (5)$$

(2)와 같이 일치하는 조건 뿐 아니라 (5)와 같은 반대(Negative)를 추가함으로써 각 조건에 따라 손실 값을 추가할 수 있다. 관계에 따라 가중치를 달리 설정함으로써 심각한 오류의 개체명 구성을 출력한 경우 더 많은 손실 값을 부여할 수 있도록 구성하였다.

이러한 심볼릭 지식을 추가함으로써 같은 실험 환경에서 심볼릭 지식을 사용한 경우 한 세대에서 기존보다 많은 손실 값을 전달하게 되므로 수렴 속도와 성능 향상 효과를 볼 수 있다.

### 3.3 실험 결과

표 2는 한글 관계 추출 실험에 대한 결과이다. 심볼릭 지식을 사용한 것과 미사용한 것에 대하여 개체명 및 관계 성능을 표시하였다.

학습은 100세대까지 수행하였고, 심볼릭 지식 미사용은 트랜스포머만 학습한 결과이다. 비교하였을 때 심볼릭 지식을 사용한 관계 성능은 4.4%가 향상 됨을 확인할 수 있다. 심볼릭 지식은 관계에 관한 것이므로 개체명과는 연관이 없으나 손실 값을 계산하여 트랜스포머에 그라디언트(gradient)를 전달 함으로써 개체명의 성능 또한 향상 된 것으로 추측한다.

표 2 개체명, 관계추출 성능

		심볼릭 지식 사용	심볼릭 지식 미사용
관계	Precision	82.70	82.82
	Recall	79.45	77.13
	F1	81.04	79.88
개체명	Precision	59.52	62.88
	Recall	43.86	36.40
	F1	50.51	46.11

그림 2는 심볼릭 지식을 사용하거나 미사용에 따른 관계 성능을 표시한다. 빨간색은 심볼릭 지식을 사용한 결과이고 파란색은 미사용 결과이다. 첫번째 그림은 관계의 F1 성능을 표시한 것으로 세대 10부터 성능의 차이가 벌어지는 모습을 확인하였고, 전체적으로 다른 성능 들도 수렴 속도가 심볼릭 지식을 사용하면 높아지는 것을 알 수 있다. 두번째 그림은 관계의 재현율을 세대별로 출력한 것이다. 심볼릭 지식을 개체명 카테고리 구성하여 재현율을 높이려는 의도를 반영하듯 성능 차이가 크게 난다. 마지막 세번째 그림은 관계 중 is\_location\_of에 대한 성능을 표시하였고 이 또한 심볼릭 지식을 사용하였을 때 성능의 기울기와 최고점이 높다.

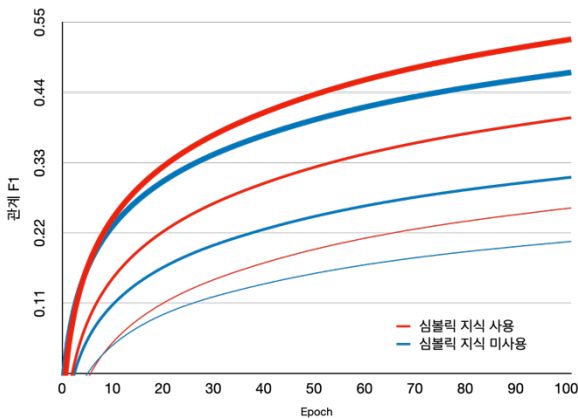


그림 3 학습 데이터 크기에 따른 관계 성능

그림 3은 학습 데이터 크기에 따른 관계의 F1 성능을 나타낸다.

그래프에 표시된 선의 굵기에 따라 데이터 사용량을 표시한다. 평가 말뚝치는 동일하고 가장 얇은 선은 500문장, 중간 굵기의 선은 1000문장, 그리고 가장 굵은 선은 전체 문장(1740문장)을 학습한다.

학습데이터에 따라 성능은 떨어지지만 심볼릭 지식을 사용함으로써 기존의 성능보다 높아짐을 확인 할 수 있다. 이것은 대용량 말뚝치가 아닌 적은 코퍼스에서도 심볼릭 지식을 사용하여 성능 향상을 할 수 있다는 것을 확인 할 수 있다.

#### 4. 결 론

본 논문에서는 뉴럴-심볼릭 방법을 이용하여 적은 학습 데이터에서 기존의 딥러닝 성능을 능가하는 방법을 제안한다. 관계 추출 분야에서 개체명 인식과 이들 사이의 관계 추출 과정에서 논리 규칙 추론을 적용하였다. 한국어 데이터를 사용하였으며 일반적인 딥러닝 방법과 심볼릭 지식을 추가한 결과를 비교하였을 때 4.4%의 성능 향상을 확인하였다. 또한 코퍼스의 크기와 상관없이 심볼릭 지식은 성능 향상에 도움을 주는 것을 실험으로 알 수 있다.

향후 연구로 심볼릭 지식을 세대 적용 시 반영할 비율을 자동으로 계산할 수 있는 방법과 초반 성능 향상에 대한 연구를 진행하고자 한다.

#### Acknowledgement

이 논문은 2019년도 정부 (과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (2019-0-00004, 준지도학습형 언어지능 원천기술 및 이에 기반한 외국인 지원용 한국어 튜터링 서비스 개발)

#### 참고문헌

[1] Jacob Devlin et al., "BERT:Pre-training of Deep Bidirectional Transformers for Language Understanding," arXiv:1810.04805, 2018.  
 [2] Colin Raffel et al., "Exploring the Limits of Transfer Learning with a Unified Text-to-Text

Transformer," arXiv:1910.10683, 2019.  
 [3] Rich Sennrich et al., "Improving Neural machine Translation Models with Monolingual Data," in ACL, 2016.  
 [4] Alina K arakanta et al., "Neural Machine translation for low-resource languages without parallel corpora," Machine Translation, pp 1-23, 2017.  
 [5] Antoine Bordes et al., "Learning Structured Embeddings of Knowledge Bases," In AAAI, 2011.  
 [6] Zhiting Hu et al., "Harnessing Deep Neural Networks with Logic Rules," in ACL, 2016.  
 [7] Wenya Wang et al., "Integrating Deep Learning with Logic Fusion for Information Extraction," in AAAI, 2020.