중첩 분할된 양방향 LSTM 기반의 한국어 프레임넷의 프레임 분류 및 논항의 의미역 분류

함영균O, 신기연, 최기선

한국과학기술원
hahmyg@kaist.ac.kr, nuclear852@kaist.ac.kr, kschoi@kaist.ac.kr

Frame-semantics and Argument Disambiguation of Korean FrameNet using Bi-directional LSTM

Younggyun HahmO, Giyeon Shin, and Key-Sun Choi
KAIST

요 약

본 논문에서는 한국어 프레임넷 분석기를 구축하기 위하여 한국어 프레임넷 데이터를 가공하여 공개하고, 한국어 프레임 분류 및 논항의 의미역 분류 문제를 해결하기 위한 방법을 제안한다. 프레임넷은 단어 단위가 아닌 단어들의 범위로 구성된 범위에 대해 어노테이션된 코퍼스라는 점에 착안하여, 어휘 및 논항의 내부 의미 정보와 외부 의미 정보, 그리고 프레임과 각 의미역들의 임베딩을 학습한 중첩 분할된 양방향 LSTM 모델을 사용하였다. 이를 통해 한국어 프레임 분류에서 72.48%, 논항의 의미역 분류에서 84.08%의 성능을 보였다. 또한 본 연구를 통해 한국어 프레임넷 데이터의 개선 방안을 논의한다.

주제어: 자연언어처리, 프레임넷, 의미역 결정

1. 서론

프레임 시멘틱스(이하 프레임)란 텍스트에서 나타나는 의미를 표현하기 위해 정의된 구조화된 스키마이며[1,2], 프레임넷은 이러한 프레임을 문장에 어노테이션 한 어휘 데이터베이스이다. 프레임넷은 질의응답[3,4], 정보 추출[5], 대화 시스템[6] 등의 다양한 분야에서 사용되며, 이러한 활용성이 증가하면서 다국어 프레임넷을 구축하는 연구 역시 활발히 진행되고 있다[7,8,9,10,11].

프레임 파싱은 구축된 프레임넷 코퍼스를 학습데이터로 사용하여 문장에서 특정 프레임을 갖는 어휘(target)를 인식(identification)하고, 해당 어휘가 프레임넷의 어떤 어휘 유닛(lexical unit, LU)에 해당하는지 선택하고, 해당 LU가 갖는 프레임을 분류(disambiguation)한 뒤, 선택된 프레임의 논항들을 인식하고, 다시 이 논항에 대한 의미역을 분류하는 일련의 작업을 거친다[2,12]. 그림 1은 프레임 어노테이션의 예 이다. 그림 1에서, 단어 '팔았다'는 문장에서 target으로 프레임 Commerce_cell을 가지며, 해당 프레임은 문장 내부에서의 각 논항들을 Seller, Goods, 그리고 Buyer로서 의미역으로 갖는다. 이때, target을 인식하고 LU를 선택하는 것은 데이터의 일관성 문제로 휴리스틱한 방법에 의존하며[12], 문장의 LU에 대해 프레임을 분류하고 논항을 인식하며, 논항의 의미역을 분류하는 연구가 딥러닝 알고리즘을 사용하여 활발히 이루어지고 있다[13,14].

한국어 프레임넷은 [15, 16]의 연구를 통해 영어 프레임넷 및 일본어 프레임넷 코퍼스를 번역하여 프레임이 어노테이션된 코퍼스가 구축되었지만, LU를 중심으로 가공이 되어있지 않아 프레임 분석기를 구축하기에 어려움이 있어 프레임 분석기 개발 연구가 활발히 이루어지지 못하였다.

본 논문은 다음의 기여가 있다:

1) 한국어 프레임넷 데이터의 가공 및 공개(v0.8)
2) 한국어 LU에 대한 프레임 분류 문제의 첫 결과
3) 한국어 논항에 대한 의미역 분류 문제의 첫 결과

2장에서는 본 논문에서 수행한 타스크들의 문제정의에 대해 상술하고, 3장에서는 한국어 프레임넷 데이터를, 4장에서는 중첩 분할된 양방향 LSTM 알고리즘을 사용한 프레임 및 논항의 의미역 분류 방법론을 기술한다. 5장에서는 평가 결과를 논의하고 결론을 6장에 기술한다.

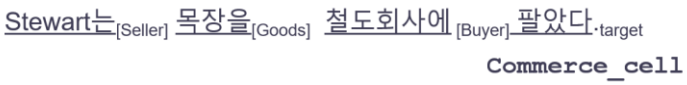

그림 1 프레임 어노테이션의 예: Commerce_cell

2. 문제정의

2.1. 한국어 프레임넷 데이터 가공 및 공개

1장에서 논의된 바와 같이, 주어진 텍스트에 대해 프레임 파싱을 하기 위하여서는 문장에서 나타난 LU에 대한 프레임 분류 및 논항의 의미역 분류 작업이 필요하다. 이를 위하여서는 프레임넷 데이터에 다음의 전제조건이

필요하다.

1) 프레임넷에 LU가 정의되어 있어야 한다.
2) 각 LU가 가질 수 있는 프레임이 정의되어 있어야 한다.
3) 프레임넷 코퍼스에는 LU, 프레임, 프레임의 논항과 범위, 그리고 각 논항의 의미역이 어노테이션 되어 있어야 한다.

본 논문에서는 한국어 프레임넷 코퍼스로부터 LU 목록을 구축하고, 이를 가공하여 의미역 어노테이션을 위해 설계된 CoNLL-09의 데이터 형태로 구축한 작업을 3.1장에서 상술한다.

2.2. 프레임 분류 문제

프레임 분류 타스크는 문장에서 주어진 LU에 대하여 프레임넷에 정의된 프레임 중 하나로 분류하는 문제이다. 프레임 분류의 입력은 다음과 같이 구성된다: n개 어절로 이루어진 문장 $w_d =< w_{d_1}, w_{d_2}, \dots, w_{d_n} >$, 각 어절의 품사 분석 결과의 시퀀스 $w_o =< w_{o_1}, w_{o_2}, \dots, w_{o_n} >$, 문장에서 target어휘의 범위 $t = < t_{start}, t_{end} >$ 와 그 어휘에 해당하는 LU ℓ 로 구성된 입력 $x = < w_d, w_o, t, \ell >$ 이다. 이때 해당 LU ℓ 이 가질 수 있는 프레임들의 집합인 F_ℓ 중에서 적합한 프레임 f를 선택하는 문제이다. 위의 표기법은 프레임 파싱 문제를 다룬 [13]의 표기를 따랐다.

2.3. 논항의 의미역 분류 문제

논항의 의미역 분류 타스크는 문장에서 주어진 LU 및 프레임에 의하여 논항들의 범위를 인식하고, 논항들의 의미역을 분류하는 문제이다. [3, 13]의 연구에서는 영어 프레임넷 코퍼스를 사용하여 논항들에 대한 후보들을 인식하고, 각 후보 논항들에 대하여 적합한 의미역을 분류하였다. 영어 프레임넷은 문장에 대해 가능한 모든 논항에 대한 의미역이 충분히 어노테이션 되어 있지만, 한국어 프레임넷은 논항에 대해 의미역이 누락되어 있는 경우가 종종 발견되었다. 이에 논항의 범위를 인식하는 알고리즘은 실험 결과를 왜곡할 경향이 크다고 판단되었다. 본 논문에서는 논항의 범위 인식 문제는 연구 범위에 포함시키지 않고 주어진 논항의 의미역 분류 문제만을 다룬다.

논항의 의미역 분류의 입력은 2.2장의 표기법을 그대로 사용하여 다음으로 구성된다: n개 어절로 이루어진 문장 w_d와, 각 어절의 품사 시퀀스 w_o, 문장에서 target 어휘의 범위 t와 그 어휘에 해당하는 LU ℓ, ℓ에 해당하는 프레임 f, 그리고 문장에서의 각 k개 논항들의 위치 정보 $a_k =< a_{1_{start,end}}, a_{2_{start,end}}, \dots, a_{k_{start,end}} >$이다. 즉 입력은 $x = < w_d, w_o, t, \ell, f, a_k >$ 로 정의할 수 있다. 논항의 의미역 분류는 입력 x 로부터 각 k개 논항들에 대해 각각의 의미역 $y_k \in Y_f$ 를 출력한다. 이때 각 의미역 y_k는 주어진 프레임 f에 대해 정의된 의미역들의 집합인 Y_f중에 하나에 해당한다.

3. 한국어 프레임넷 데이터 가공

한국어 프레임넷 데이터는 영어 프레임넷과 일본어 프레임넷 코퍼스와 어노테이션을 수작업으로 번역한 코퍼스를 바탕으로 한다. 영어 프레임넷이 LU를 정의하고, 그 LU에 해당하는 프레임을 정의한 뒤에, 이후 문장에 대해 어노테이션을 수행한 것[17]과 달리, 한국어 프레임넷은 LU 목록을 번역어휘에서 가져오는 것을 목표로 하였다. 이때 각 LU는 단어의 표제어와 품사로 구성된다. 예를 들어 영어 LU인 visit.v 의 경우 단어 visit과 온점, 그리고 동사 품사를 나타내는 v가 합쳐진 형태이다.

예를 들어 영어 어휘 visiting이 문장에서 target이 되어 영어 LU인 visit.v 이 어노테이션 되어 있고, 해당 LU가 프레임 Arrving 으로 어노테이션 되어 있는 경우가 있다. 이 경우 한국어 프레임넷은 '... 방문한 ...' 이라는 문장으로 번역되어 있고, 해당 어절인 '방문한' 이 target이 되어 프레임 Arrving이 어노테이션 되어 있다. 한국어 프레임넷의 LU 목록을 생성하기 위해 다음의 과정을 수행하였다. 1) 각 target에 대한 형태소 분석을 수행하여 어근을 선택하였다. 이때 형태소 분석 결과에서 명사가 관형형 전성 어미(예:-ㄴ하)로 쓰여 동사로서 사용되었다면, 이를 동사 어휘로서 간주하였다. 2) 해당 어근을 사용하는, 동일한 품사를 갖는 어휘의 표제어를 세종의미사전을 사용하여 생성하였다. 결과적으로 위의 예시에서 target인 '방문한' 에 대해 '방문하다.v' 의 LU를 생성하였고, 해당 LU는 프레임 Arrving을 갖도록 목록화 하였다. 목록은 LU와 온점, 그리고 프레임이 합쳐진 형태로 작성되었다. 위의 예시의 경우 '방문하다.v.Arrving'의 형태이다.

또한 한국어 프레임넷 코퍼스의 어노테이션 정보를 사용하기 위하여, 기존의 JSON방식의 어노테이션을 CoNLL-09 포맷의 형식으로 변환하는 작업을 수행하였다. 이때, 각 어노테이션 데이터의 오류로 인해 범위가 부정확한 경우와 어노테이션의 명백한 오류의 경우 일부 수정하는 작업을 거쳤다. 본 작업에서의 한국어 문장에 대한 자연언어처리는 [24]를 사용하여 진행되었다. 이를 통해 구축된 데이터의 형식은 그림 2의 예시와 같다

0	센터장은	_	_	S-Speaker
1	안전	_	_	B-Message
2	절차의	_	_	I-Message
3	철저한	_	_	I-Message
4	검토를	_	_	I-Message
5	약속하였다.	약속하다.v	Commitment	

그림 2 한국어 프레임넷 데이터 예시

각 라인은 복수개의 컬럼으로 구성되어 있으며, 각 컬럼의 구성요소는 CoNLL-09 포맷을 따랐다. 위의 예시에서 각 컬럼은 각 어절의 번호, 어절, LU, 프레임, 그리고 논항의 의미역이다. 이때 각 논항의 의미역에 대해 논항이 시작하는 경우 B- 태그가 앞에 붙으며, 이후에는

I-, 논항이 하나의 어절에만 쓰인 경우 S-, 그리고 의미역이 없는 경우 O로 태깅되어 있다. 위의 작업을 거쳐 3,892개의 LU(명사 2,340, 동사 1,276, 형용사 276) 목록이 생성되었다. 또한 총 4,527개의 고유 문장과, 각 문장에 복수개의 프레임을 어노테이션 되어 17,438개의 어노테이션을 구축하였다. 해당 데이터는 [25]를 통하여 공개되어 있다.

4. 프레임넷 프레임 및 논항의 의미역 분류

한국어 프레임넷의 논항의 의미역은 BIOS태그를 사용하여 복수개의 단어가 하나의 논항이 되고, 각 논항이 의미역 태그를 갖도록 어노테이션 되어있다. 그림 2의 예시에서, 복수개의 단어들로 구성된 "안전 절차의 철저한 검토를" 이 하나의 논항이 되고, 그 논항의 의미역은 Message이다. 한국어 propbank를 사용한 의미역 결정 데이터[18,19]는 하나의 의미역이 하나의 단어에만 태깅되어 있지만, 한국어 프레임넷에서는 복수개의 단어가 하나의 의미역으로 태깅되어 있다는 점에서 다르다. 그리고 이러한 논항의 각 단어들은 문장 전체에서의 맥락이 반영될 뿐만 아니라 논항 내부의 단어들 사이에서의 맥락이 반영되어 있다.

기존의 한국어 의미역 결정 알고리즘은 한국어 propbank 데이터를 사용하여 각 단어가 갖는 주변 맥락과, 각 단어가 내포한 임베딩 벡터를 사용하여 양방향 LSTM 알고리즘을 적용하였고, 또한 문장의 의존구조가 갖는 구문 정보가 없더라도 형태소정보와 주변 맥락의 벡터만으로도 충분한 성능을 보인 바 있다[20]. 이러한 특징과 기존 연구에 착안하여, 본 논문에서는 각 논항의 의미를 벡터 공간에 사상하기 위하여 [13]에서 사용된 중첩 분할된 양방향 LSTM 네트워크를 사용하였다. 이를 도식화 한 것은 아래 그림과 같다.

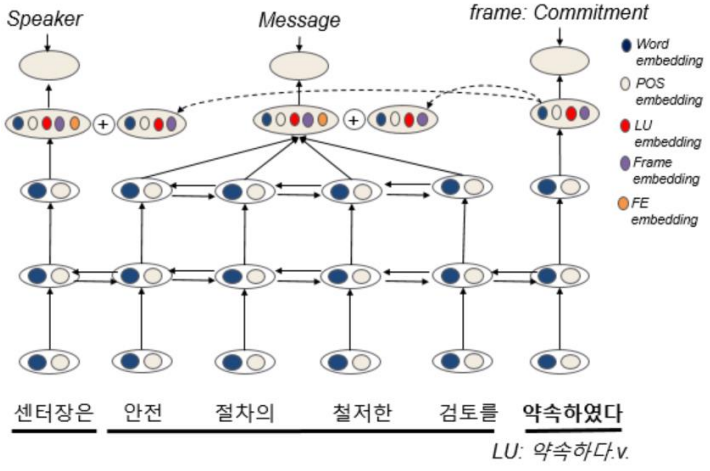

그림 3 중첩 분할된 양방향 LSTM 알고리즘 예시

그림 3에서, 각각의 단어는 하나의 LSTM 층을 거쳐 전체 문장에서의 맥락을 반영한 벡터값으로 변환되고, 또한 각각의 논항의 의미정보를 파악하기 위해 논항 내부의 단어들의 벡터값을 다른 LSTM 층을 거쳐 계산한다. 또한, 각각의 논항은 문장에서 항상 논항으로서 의미역을 갖지 않는다. 즉 논항의 의미는 문장에서의 맥락과 함께, 문장에 대해 주어진 LU와 프레임에 의존적이다. 따라서 각각의 논항의 벡터값은 주어진 LU와 프레임의 의미를 반영한 벡터값과 합쳐져(concatenate) 각각의 의미역으로 분류될 확률값을 갖는다. 자세한 알고리즘은 각 4.1 및 4.2장에서 상술한다.

4.1. 프레임 분류

2.1장에서 기술된 바와 같이, 프레임 분류 문제는 문장 w_d와 그에 상응하는 품사 시퀀스 w_o, target의 범위 t와 LU ℓ 이 주어졌을 때, 입력 $x = <w_d, w_o, t, \ell>$에 대하여 적합한 프레임 f를 선택하는 문제이다. 각각의 ℓ이 가질 수 있는 프레임들의 목록 F_ℓ은 3장에서 구축된 한국어 프레임넷 데이터에서 제공되었다.

그림 3의 첫번째 양방향 LSTM은 각 토큰에 대한 벡터값을 입력으로 받는 $\mathrm{LSTM}_{\mathrm{token}}$이다. 이때 입력 벡터값은 단어의 위치 q에 대하여, 각 단어에 대해 학습되는 임베딩값 $\mathrm{d_q}$와 품사에 대해 학습되는 임베딩 $\mathrm{o_q}$, 그리고 미리 학습된 워드임베딩의 concatenate된 벡터값 $\mathrm{token_q} = [\mathrm{d_q}; \mathrm{o_q}; \mathrm{e_q}]$이다. 각각의 $\mathrm{token_q}$에 대하여, $\mathrm{LSTM}_{\mathrm{token}}$은 그에 대응하는 은닉 계층의 벡터인 $\mathrm{hidden_q}$의 시퀀스를 출력으로 내어준다.

주어진 LU는 그 자체로 하나의 임베딩값을 학습할 수 있지만, 또한 문장 내에서 target의 어휘 및 주변단어에 의해서 벡터값으로 학습될 수 있다. 예를 들어, 그림 2에서 LU '약속하다.v'의 경우, 단어 '약속하였다' 는 물론 주변 단어인 '검토를' 과 같은 단어의 의미와 밀접한 관련이 있다. 이때 target의 벡터 u_t는 target의 범위 t 에 속하는 단어들과 주변 단어(context=1) 의 $\mathrm{LSTM}_{\mathrm{token}}$의 최종 은닉 계층의 벡터값에 해당한다. 이를 수식으로 표현하면 다음과 같다.

$$\mathrm{u_t} = \mathrm{LSTM}_{\mathrm{target}}(\mathrm{hidden}_{t_{start}-1}, \dots, \mathrm{hidden}_{t_{end}+1})$$

본 모델에서는 LU의 벡터값은 학습된 LU의 임베딩 벡터 u_ℓ와, target의 벡터 u_t의 concatenate값인 $[\mathrm{u}_t; \mathrm{u}_\ell]$로 구해진다. 네트워크의 최종 계층에서는 선형 파라미터를 거쳐 ReLU[21]가 적용되 모델에서의 $[\mathrm{u}_t; \mathrm{u}_\ell]$의 확률값을 계산한다. 이를 수식으로 표현하면 다음과 같다.

$$V(\mathrm{f}) = w_2 \mathrm{reLU}\{w_1[\mathrm{u}_t; \mathrm{u}_\ell]\}$$

이때 각각의 w_1, w_2는 모델에서 선형 파라미터를 의미한다. 주어진 $V(\mathrm{f})$에 대하여, 모델은 softmax를 통해 의미역에 대한 확률값을 계산한다. 이때, 전체 의미역 레이블에 대한 확률값을 계산하지 않고, 주어진 LU ℓ 가 가질 수 있는 프레임 후보들 F_ℓ에 대해서만 계산을 수행하였다. 이에 대한 수식은 아래와 같다.

$$p(f) = \frac{expV(f)}{\sum_{f` \in F_\ell} expV(f`)}$$

주어진 확률값에 대한 Negative log likelihood는 SGD optimizer를 사용하여 학습과정에서 최소화 되는 과정을 거쳤다. 미리 학습된 단어의 벡터값은 FastText의 300차원의 한국어 워드임베딩 데이터[23]를 사용하였다. 모델에서의 하이퍼 파라미터는 다음과 같다: 단어는 60, 품사는 4, LU는 64, 프레임은 100, 그리고 각각의 LSTM의 입출력은 64차원의 임베딩값을 갖는다. 학습률은 0.001, 드랍아웃 비율은 0.01으로 설정하였다.

이때에 계산된 벡터 $v_{f,t,\ell} = [u_t; u_\ell]$는 4.2장에서 기술되는 논항의 의미역 분류 타스크에서 다시 사용된다.

4.2. 논항의 의미역 분류

2.2장에서 기술된 바와 같이, 논항의 의미역 분류 문제는 문장에서 k개의 논항의 범위가 주어졌을 때, $x = < w_d, w_o, t, \ell, f, a_k >$를 입력으로 받아 각 k개 논항들에 대해 의미역 $y_k \in Y_f$ 를 출력한다. 4.1장의 네트워크 모델을 그대로 사용하여, 첫번째 $LSTM_{token}$에서 각각의 $token_q = [d_q; o_q; e_q]$에 대해 대응하는 은닉 계층의 벡터인 $hidden_q$의 시퀀스를 출력으로 받는다. 이후 주어진 LU ℓ 및 프레임 f에 대한 벡터값을 구하기 위하여 4.1장에서 계산된 $v_{f,t,\ell} = [u_t; u_\ell]$ 를 계산한다.

그리고 각 논항 a_k에 대하여, 범위인 $a_{k_{start,end}}$에 대응하는 $hidden_q$를 계산하기 위하여 두번째 양방향 LSTM인 $LSTM_{argument}$ 에서의 마지막 은닉 계층의 벡터값 $hidden_{start,end}$을 취한다. 이를 수식으로 표현하면 다음과 같다.

$$hidden_{start,end} = LSTM^{start,end}_{argument}(hidden_{start}, \dots, hidden_{end})$$

이는 논항의 내부 단어들에 의한 맥락 정보를 의미한다. 각 논항의 벡터값들이 의미역들의 임베딩값 $y_k \in Y_f$와, 또한 주어진 target과의 위치정보를 고려한다. 위치정보는 target보다 먼저 나왔는지, 뒤에 나왔는지, target과 중첩되는지, 혹은 target 내부에 있는지를 나타내는 one-hot 벡터인 p로 구성된다. 각각의 벡터를 concatenate하여 논항의 벡터값은 아래 수식과 같이 계산된다.

$$v_{argument} = [hidden_{start,end}; y_k; p]$$

그러나 논항의 의미역은 내부 단어들만으로 결정되는 것은 아니며 주어진 $v_{f,t,\ell}$ 에 의존적이므로, 이 둘을 concatenate하여 최종 벡터값 $[v_{argument}; v_{f,t,\ell}]$ 을 구한다. 이후 4.1장과 동일하게 최종 계층에서는 선형 파라미터를 거쳐 ReLU 함수를 사용해 논항의 확률을 구한다. 이는 아래 수식과 같다.

$$V(y_k) = w_2 \mathrm{reLU}\{w_1[v_{argument}; v_{f,t,\ell}]\}$$

이때 하이퍼 파라미터는 4.1장에서 사용된 하이퍼 파라미터와 동일하게 사용되었다.

5. 평가

5.1. 평가데이터

평가데이터는 프레임넷의 4,527 문장을 임의로 학습셋, 평가셋, 개발셋으로 나누어 구축하였다. 학습셋은 3,220, 개발셋은 183, 그리고 평가셋은 1,124 문장으로 구성하였다. 영어 프레임넷 데이터[3,13]의 학습 및 평가셋과의 유의미한 비율을 유지하는 것이 고려되었다. 아래 표는 영어 프레임넷 데이터와 한국어 프레임넷 데이터의 비교표이다.

표 1 한국어 및 영어 프레임넷 데이터 비교

	학습셋		평가셋		개발셋	
	문장수	프레임	문장수	프레임	문장수	프레임
영어	3,139	16,621	2,420	4,428	387	2,284
한국어	3,220	12,431	1,124	4,382	183	624

5.1. 프레임 분류 모델 성능

프레임 분류 모델은 주어진 문장과 LU에 대하여, 정답셋에 레이블링된 프레임을 선택하는 것으로 측정하였다. 학습셋의 고유한 LU의 개수는 3,315개이고 평가셋은 1,688개이다. 이 중 평가셋의 501개 LU는 학습셋에서 나타나지 않아 학습할 수 없기에 모델의 성능 평가를 위해 해당 LU는 평가에서 제외하였다. 전체 평가대상 LU에 대하여 프레임을 올바르게 찾은 비중의 백분률로 평가하였다.

표 2 프레임 분류 모델의 성능

모델	영어 프레임넷	한국어 프레임넷
무작위	77.39%	54.82%
본 논문	**87.00%**	**72.48%**

본 논문의 모델은 [13] 프레임 파서의 프레임 분류 알고리즘과 동일한 하이퍼 파라미터로 구현되었고 pytorch를 사용하여 재구현되었다. 영어 프레임넷 데이터 프레임 분류에 있어서 동일한 데이터에 대해 보고된 성능인 87.51%과 유사한 성능을 보였다. 그러나 한국어 프레임넷 데이터에 대하여서는 프레임 분류의 성능은 72.48%로 비교적 낮은 수치를 보였다.

이는 무작위 모델, 즉 주어진 LU에 대해 해당 LU가 가질 수 있는 프레임 중 임의의 하나를 무작위로 선택하는 모델의 성능 비교에서 미루어 알 수 있다. 영어 프레임넷의 경우에는 각 LU가 가질 수 있는 프레임의 후보가 많지 않은 것에 비해, 한국어의 경우에는 후보가 많다고 볼 수 있다. 실제로 프레임 후보가 2개 이상인 LU의 비율은 한국어는 26%, 영어는 13.26%의 비중을 차지하고 있었고, 프레임 후보가 5개 이상인 LU는 한국어는 73개인 반면 영어는 9개이다.

이는 영어 프레임넷과 한국어 프레임넷의 구축 방법의 차이에서 기인하는데, 영어 프레임넷의 경우에는 LU를 먼저 수작업으로 고르고, LU에 대한 프레임을 수작업으로 선택하는 하향식의 작업을 거치는데, 한국어 프레임넷의 경우에는 번역가에 의하여 단어를 번역하고, 해당 단어를 LU화 하는 상향식의 방법에서 기인한다. 아래 그림은 한국어 LU '확인하다.v'에 대응되는 영어 LU들의 목록이다. 즉, 영어의 경우에는 각 단어가 나타내는 세부적인 개념이 프레임에 나타나 있는 반면, 한국어 LU '확인하다.v'의 경우에는 다양한 영어 LU의 프레임들을 갖게 되어 모호성이 증대된 경우이다.

영어 LU	Frame	한국어 LU
identify.v	Categorization	확인하다.v
	Verification	
certain.v	Certainty	
ascertain.v	Coming_to_believe	
confirm.v	Evidence	
	Statement	
	Verification	
verify.v	Verification	

그림 4 한국어 LU '확인하다.v' 에 대응되는 영어 LU

또한 영어의 경우에는 가장 많은 프레임 후보를 갖는 경우 최대 8개의 후보이지만, 한국어의 경우에는 LU '하다.v' 의 경우 32개의 프레임 후보를 갖는 경우도 있었다. 그러나 이는 대부분의 경우, 한국어 어휘가 실제 의미를 충분히 반영하지 못하도록 번역되었기 때문인데, 이에 대한 예시는 아래와 같다.

영어 프레임넷 문장: "... legislation designed to punish ..."
한국어 프레임넷 문장: "... 처벌하도록 하는 법률 제정을 ..."

위 예시에서, 영어 프레임넷 코퍼스의 문장에서 어휘 'designed' 가 '하는' 으로 번역된 사례이다. 이때 '하는' 은 3장에서 논의된 방법으로 LU화 되면 '하다.v'로서 구축된다. 따라서 프레임 분류에 있어서 한국어의 언어적 특성을 반영하는 알고리즘의 개선 방안도 고려되어야 하겠지만, 한국어 프레임넷 데이터에 대한 교정 작업 역시 병행되어야 할 필요가 있다고 보여진다.

5.2. 논항의 의미역 분류 모델 성능

논항의 의미역 분류 모델은 주어진 문장과 LU, 프레임에 대응되는 논항들에 대하여 각 논항이 가질 수 있는 의미역 중 정답을 올바르게 선택한 비중의 백분률로 평가하였다.

룰베이스 방법은 학습데이터에서 논항의 마지막 단어의 조사의 빈도에 따라 의미역이 가질 수 있는 조사의 확률값을 계산한다. 또한 의존구조의 패턴을 룰에 추가하였는데, 조건은 다음과 같다: 1) target의 의존구조 레이블, 2) 논항의 품사, 의존구조 레이블 정보. 이러한 룰들을 각 LU에 대하여 학습데이터에서 추출하여 입력된 논항에 대하여 룰에 일치하는 경우 논항을 부여한다. 표 3의 무작위 모델은 주어진 프레임과 각 논항에 대하여 각 논항이 가질 수 있는 프레임 후보군 중에서 임의의 프레임을 선택하는 방법이다. 결과에서 볼 수 있듯이 논항의 의미역을 분류하는 문제는 프레임을 분류하는 문제보다 어려운 문제라고 볼 수 있다.

표 3 논항의 의미역 분류 모델의 성능

모델	성능
무작위	16.27%
룰베이스	48.34%
중첩 분할 Bi-LSTM	**84.08%**
중첩 분할 Bi-LSTM + DP	84.02%
중첩 분할 Bi-LSTM + DP + 조사	82.85%

본 논문이 제안한 중첩 분할된 양방향 LSTM 모델을 사용하여 논항이 갖는 의미를 벡터공간에 사상하는 방식을 사용함으로서, 논항의 문장의 프레임에서의 의미역 부여 성능은 84.08%의 정확도를 보였다. 중첩 분할 Bi-LSTM + DP 모델은 기존 모델에 각 단어 w_q 의 의존구조 레이블을 10차원의 임베딩 dp_q로 학습하여 입력 벡터에 추가한 $token_q = [d_q; o_q; e_q; dp_q]$ 을 적용한 모델이다. 이 경우 성능상의 유의미한 차이는 보이지 않았다. 중첩 분할 Bi-LSTM + DP + 조사 모델은, 각 논항의 마지막 단어의 조사의 경우 논항의 의미역을 분류하는데 있어 중요한 자질로 생각될 것으로 가정하여 전체 네트워크의 최종 계층에서 각 조사에 대해 학습된 20차원의 임베딩 $josa_{argument}$ 를 추가 자질로 사용하였으나 전체 성능이 하락하였다. 추가적인 평가가 필요하겠지만, 각 논항의 $LSTM_{argument}^{start,end}$ 계층에서는 최종 은닉계층의 벡터값을 취하기 때문에 이미 논항의 마지막 조사 정보가 반영되었다고 해석될 수 있으며, 또한 충분히 분석되지 않은 자질이 추가되었기 때문으로도 해석될 수 있다. 그러나 논항의 마지막 조사 정보는 일종의 격틀 정보로서 한국어 논항의 의미역을 분류하는 유의미한 자질로 사용되므로[22] 이에 대한 추가적인 연구가 수행된다면 성능 개선에 도움이 될 것으로 보여진다.

데이터의 측면에서, 한국어 프레임넷 데이터의 경우 번역된 문장을 사용하고 또한 한 문장에서의 다양한 논항들의 관계를 기술해야 하기 때문에 영어 및 일본어의 단어 구성과 비슷한 형태를 따른다는 문제가 발견되었다.

영어 프레임넷 문장: "... out of sight of the raiding [Aggregate_property] parties [Aggregate] ..."
한국어 프레임넷 문장: "... 습격하는 [Aggregate_property] 일당들의 [Aggregate] 눈에 띄지 않는..."

위의 예시에서, 두 논항, 'raiding' 과 'parties' 의 관계를 기술하기 위해 이 논항의 역할을 고려하여 번역이 수행되었음을 알 수 있다. 그러나 '습격하는 일당들의' 표현은 한국어 문장으로서는 어색한 표현으로 이러한 학습데이터는 향후 한국어 프레임 파싱을 다른 도메인에 적용할 때 문제가 될 수 있다. 따라서 논항의 의미역을 분류

하는 알고리즘의 개선과 함께 한국어 프레임넷 데이터의 문장의 재구성 및 자연스러운 의역 작업이 필요하다고 보여진다.

5. 결론

본 논문을 통하여, 기존의 한국어 프레임넷 데이터를 가공하여 공개하고(v0.8), 해당 데이터를 사용하여 한국어 프레임 분류 문제와 논항의 의미역 분류 문제를 다루었다. 프레임넷 의미역 분류 문제는 한국어 propbank 기반 의미역 분류 문제와 달리 논항이 복수개의 단어로 구성되어 있다는 점에 착안하여, 중첩 분할된 양방향 LSTM 모델을 적용, 각 논항의 의미를 벡터공간에 사상하여 의미역을 분류하였다. 본 실험을 통하여 한국어의 특성에 맞는 알고리즘 개선의 필요성과 동시에 한국어 프레임넷 데이터에 대한 교정이 필요하다는 것을 보였다. 향후 연구로서 한국어 프레임넷 데이터에 대한 LU의 교정(v.0.9) 및 논항의 문장을 재구성하는 작업(v.1.0)을 수행할 예정이다. 또한, 본 논문에서 다루지 않았던 논항의 범위 인식 문제 역시도 해결해야 할 문제로 남아있다.

사사

이 논문은 2017년도 정부(과학기술정보통신부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임 (2017-0-01780, 비디오 이해를 위한 이벤트-상황 지식체계 학습 및 이벤트인식/관계추론 기술 개발)

참고문헌

[1] C. Baker, C. J. Fillmore, and J. B. Lowe, Berkeley framenet project, In Proceedings of the 36th Annual Meeding of the Association for Computational Linguistics and 17th internatioin Conference on Computational Linguistics-Volume 1, pp. 86-90, 1998.

[2] C. Baker, M. Ellsworth, and K. Erk, SemEval'07 Task 19: Frame Semantic Structure Extraction, In Proceedings of the 4th International Workshop on Semantic Evaluations. Association for Computational Linguistics, p. 99-104, 2007.

[3] D. Shen, and M. Lapata, Using Semantic Roles to Improve Question Answering, In Emnlp-conll, pp. 12-21, 2007.

[4] Y. Hahm, S. Nam, K. S. Choi, QAF: Frame Semantics-based Question Interpretation, In Proceedings of the Open Knowledge Base and Question Answering, 2016.

[5] M. Surdeanu, S. Harabagiu, J. Williams, and P. Aarseth, Using predicate-argument structures for information extraction, In Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1, pp. 8-15, 2003.

[6] Y. Chen, W. Wang, and A. Rudnicky, Unsupervised induction and filling of semantic slots for spoken dialogue systems using frame-semantic parsing, In Automatic Speech Recognition and Understanding (ASRU), pp. 120-125, 2013.

[7] L. Borin, D. Dannélls,M. Forsberg, M. Gronostaj, and D. Kokkinakis, The past meets the present in Swedish FrameNet++, In 14th EURALEX international congress, pp. 269-281, 2010.

[8] L. You, and K. Liu, Building chinese framenet database. In Natural Language Processing and Knowledge Engineering, 2005. IEEE NLP-KE'05. Proceedings of 2005 IEEE International Conference, pp. 301-306, 2005.

[9] M. Meurs, F. Duvert, F. Béchet, F. Lefevre, and R. Demori, Semantic Frame Annotation on the French MEDIA corpus, In Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08), 2008

[10] C. Subirats, and M. Petruck, Surprise: Spanish FrameNet. In Proceedings of CIL Vol. 17, p. 188, 2003

[11] A. Burchardt, K. Erk, A. Frank, A. Kowalski, S. Padó, and M. Pinkal, The SALSA corpus: a German corpus resource for lexical semantics, In Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC-2006), pp. 969-974, 2006.

[12] D. Das, D. Chen, A. Martins, N. Schneider, and N. Smith, Frame-semantic parsing, Computational linguistics, 40.1: 9-56, 2014.

[13] S. Swayamdipta, S. Thopmson, C. Dyer, N. Smith, Frame-semantic parsing with softmax-margin segmental RNNs and a syntactic scaffold, arXiv preprint arXiv:1706.09528, 2017.

[14] B. Yang, and T. Mitchell, A joint sequential and relational model for frame-semantic parsing, In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. p. 1247-1256, 2017.

[15] J. Park, S. Nam, Y. Kim, Y. Hahm, D. Hwang, and K. S. Choi, Frame-semantic web: a case study for Korean, In Proceedings of the 2014 International Conference on Posters & Demonstrations Track-Volume, 2014.

[16] J. Kim, Y. Hahm, and K. S. Choi, Korean FrameNet Expansion Based on Projection of Japanese FrameNet, In COLING 2016, 2016

[17] J. Ruppenhofer, M. Ellsworth, M. Petruck, C. Johnson, C. Baker, J. Scheffczyk, FrameNet II: Extended theory and practice, Institut für Deutsche Sprache, Bibliothek, 2016.

[18] 이창기, 임수종, 김현기, Structural SVM 기반의 한국어 의미역 결정, Structural SVM 기반의 한국어 의미역 결정, 정보과학회논문지, 42.2: p. 220-226, 2015.

[19] 박광현, 나승훈, 문자 기반 LSTM CRF 를 이용한 한국어 의미역 결정, 한국정보과학회 학술발표논문집, p. 1817-1819, 2017.

[20] 배장성, 이창기, Stacked Bidirectional LSTM-CRFs를 이용한 한국어 의미역 결정, 정보과학회논문지 제44권 제1호, p. 36-43, 2017.

[21] V. Nair, and G. Hinton, Rectified linear units improve restricted Boltzmann machines.In Proceedings of ICML, 2010.

[22] 박태호, 차정원, CRFs 기반의 한국어 의미역 결정. 정보과학회지, 34.8: 37-41. 2016.

[23] https://github.com/facebookresearch/fastText/blob/master/pretrained-vectors.md

[24] http://aiopen.etri.re.kr/

[25] https://github.com/machinereading/koreanframenet