형용사구에서의 관계추출 개선을 위한 의존구문트리의 최소공동조상 (LCA) 변경[1)]

이대석○ 맹성현
한국과학기술원
daeseoklee0317@gmail.com myaeng@kaist.ac.kr

Altering LCA of dependency parse trees for improving relation extraction from adjective clauses

Dae-Seok Lee○ Sung-Hyon Myaeng
KAIST

요 약

본 논문에서는 텍스트에서 개체(entity) 간 관계(relation) 추출 문제에서 의존구문트리를 이용하여 자질을 추출할 때 형용사구 내에 관계가 나타나는 경우의 성능을 향상시키는 방법을 제안한다. 일률적으로 의존구문트리의 최소공동조상(LCA: Least Common Ancestor)을 이용하는 일반적인 방법보다 형용사구가 나타날 때는 형용사구의 술어를 대신 이용하는 것이 더 좋은 자질이 된다는 것을 제안하고 로지스틱 회귀분석, SVM(linear), SVM(exponential kernel)을 이용한 실험들을 통해 그 효과를 확인하였다. 이는 트리커널을 이용한 것과 같이 의존구문트리의 최소공동조상이 주요한 역할을 하는 관계추출 모델들의 성능을 높일 수 있음을 보여 준다. 수행한 실험 과정을 통해 관계추출 데이터 셋에서 형용사구 내 관계를 포함하는 문장이 전체에서 차지하는 비율이 낮을 경우 생길 수 있는 문제를 추가적으로 얻을 수 있었다.

주제어: relation extraction, dependency parsing, adjective clause, tree kernel, least common ancestor

1. 서론

관계추출(Relation Extraction) 문제는 문장 내의 주어진 개체 혹은 개념들(대개 각각 한 단어로 이뤄져 있지만 연속된 여러 단어로 구성된 경우도 있음) 간의 의미적 관계를 설정하는 것으로 주어진 관계 목록에서 가장 잘 나타내는 것을 선택하는 분류의 문제로 볼 수 있다. 관계추출 문제는 지난 10여년가 많은 연구가 이루어져있는 중요한 문제로 이에 대한 다양한 기계학습 접근 방법은 [1]와 [2]에 요약되어 있다.

근래에는 워드임베딩을 이용하여 명시적 자질(feature) 추출 없이 신경망을 사용하는 기술이 많이 개발되었으나 성능 향상에 대한 설명이 쉽지 않다는 단점을 가지고 있다. 본 연구에서는 언어적 자질이 정보추출에 미치는 영향을 파악하기 위하여 문장의 구문분석 결과를 자질로 사용하는 패턴 분석 방법을 고려하였다.

의존 구문 분석(Dependency Parsing)은 주어진 문장을 구성하는 단어들을 의존 문법(Dependency Grammar)의 규칙에 따라 위계적인 방향트리 형태로 배열하는 과정이다. 이 때 각각의 부모-자식 간에는 그들의 문법적인 관계에 따라 의존 타입(Dependency Type)이라 불리는 라벨이 붙게 된다. 의존 문법 중 대표적인 것으로는 Stanford Typed Dependencies([3])와 Universal Dependencies([4])가 있는데, 본 연구에서는 후자를 이용하였다. Universal Dependencies는 한국어를 포함한 다양한 언어에 적용할 수 있도록 만들어졌고 또 그 언어들 각각에 대한 데이터를 제공하고 있다는 장점이 있다.

관계추출 알고리즘들, 특히 문장의 단어 패턴을 사용하는 방법들 중에는 의존 구문 분석을 기반으로 하는 것이 많다 ([5], [6], [7]). 그러나 이러한 알고리즘들은 대게 의존 타입을 무시하고 의존 구문 분석의 결과를 단지 트리로서만 볼 뿐이다. 하지만 본 연구에서는 의존 문법을 검토한 결과, 이러한 방법이 관계추출 과정에서 문제를 야기 시킬 수 있는 특수한 상황을 발견하여 그 해결 방법을 제시한다.

먼저 우리는 의존 구문 분석을 기반으로 한 몇 가지 관계 추출 알고리즘들을 분석하는 과정에서 의존 트리를 구성하는 두 개체들의 최소공동조상(공동조상이면서 자식들 중에는 공동조상이 없는 유일한 노드, 이하 LCA (Least Common Ancestor))이 중요한 역할을 함을 유추해 낼 수 있었다 (상세 내용은 다음 절에 설명).

이것이 관계추출에 좋은 자질이 될 것이라는 사실은 의존 문법에서 술어가 다뤄지는 방법을 보면 알 수 있다. 의존 구문 트리 상에서 주어와 목적어는 대개의 경

1) 본 연구는 과학기술정보통신부 및 정보통신기술진흥센터의 SW중심대학사업의 연구결과로 수행되었음(2016-0-00018)

우 술어를 중심으로 분기되며, 또한 어떤 문장 속에서 두 개체 사이의 관계는 그것들을 주어와 목적어로 갖는 술어에서 얻어낼 수 있을 경우가 많다. 따라서 두 개체의 관계를 알아내기 위해 그것들이 어디서부터 분기되었는가를 추적하는 것은 당연한 절차가 될 것이다. 하지만 우리는 의존문법에서의 술어의 이러한 특징에서 예외를 발견 하였는데, 이는 술어와 주어/목적어를 포함한 형용사구가 사용된 경우이다.

예를 들어 문장 "I love the girl you love."의 의존 구문 트리는 [그림 1]과 같이 만들어진다. 먼저 이 트리를 바탕으로 'I'와 'girl' 사이의 관계를 찾고 싶다고 하자. 이 두 노드의 LCA인 첫 번째 'love'는 'I'와 'girl'을 각각 주어와 목적어로 가지는 술어로서 관계를 찾기 위한 좋은 힌트가 된다.

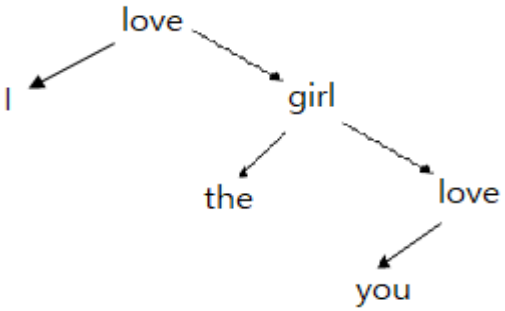


그림 1. 형용사구 내 관계와 비형용사구 관계의 차이를 보여주는 예시

반면 'girl'과 'you' 사이의 관계를 알아내려고 할 때는 같은 방법을 이용할 수 없다. 기계적으로 LCA를 찾는 경우 두 노드의 LCA가 'girl'이 되기 때문이다. 여기서 'girl'은 관계에 의해 엮일 개체이기 때문에 두 번째 'love'가 관계를 찾기 위한 더 좋은 힌트가 될 것이다. 이러한 현상은 앞서 언급한 예외적인 경우, 즉 형용사구(여기서 'you love')가 관여될 때마다 나타나게 된다. 본 논문에서는 이러한 경우를 두고 '형용사구 내 관계'라고 지칭하며, 형용사구 내 관계를 추출해야 하는 경우에 한해서 원래의 LCA 대신 술어에 해당하는 노드(위 예시에서의 두 번째 'love')를 사용하는 것이 일률적으로 LCA를 사용하는 것보다 더 나은 자질이 된다는 것을 실험적으로 확인했다.

이는 LCA를 자질로서 이용하는 기존의 의존 구문 기반 관계추출 방법들에서 LCA를 명시적으로 이용하거나 또는 암묵적으로 LCA가 중요한 역할을 하는 경우 그 알고리즘의 변형이 필요함을 시사한다. 즉 형용사구 내 관계를 따로 고려해서 LCA 대신 형용사구의 술어가 이용되도록 수정하면 성능이 더 높아질 수 있다.

2. 관련 연구

의존 구문 분석을 기반으로 하는 관계 추출 방법의 예시로는 트리커널을 이용하는 방법([5]), 최소경로를 이용하는 방법([6], [7]) 등이 있는데 본 연구와 관련이 깊은 것은 첫 번째 방법이다. 트리 커널은 두 트리의 유사도 척도인데, 몇 가지 변형이 있지만 그 중 [1]에서 소개된 것은 다음과 같이 계산된다.

- 노드 u, v에 대해 $m(u,v)$는 매칭함수의 값으로 0 또는 1을 갖는다.
- 노드 u, v에 대해 $s(u,v)$는 유사도함수의 값으로 양의 실수 값을 갖는다.
- 트리 T_1, T_2간의 커널 $k(T_1,T_2)$는 함수 $K(\cdot,\cdot)$ 사용하여 $k(T_1,T_2)=K(T_1.LCA,T_2.LCA)$ 으로 정의한다. 여기서 LCA는 관계를 추출하고 싶은 개체들에 해당하는 노드들의 최소공동조상을 뜻한다.

$$K(u,v)=\begin{cases}0 & \text{if } m(u,v)=0\\ s(u,v)+K_c(u,v) & otherwise\end{cases}$$

$$K_c(u,v)=\sum_{l(a)=l(b)}\lambda^{width(a)+width(b)}\prod_{i=1}^{l(a)}K(a_i,b_i)$$

여기서 $a=a_1a_2\cdots a_{l(a)}$와 $b=b_1b_2\cdots b_{l(b)}$는 각각 u, v의 자식들의 수열의 부분수열이며, $width(a)$는 maxindex-minindex+1 로 정의된다. $0<\lambda<1$은 감쇠상수(decay factor)이다.

요컨대 LCA들의 매칭함수 값이 0이라면 커널 값도 0이 되며, 그게 아니라면 두 LCA의 유사도 함수 값에 감쇠상수가 최소 제곱만큼 붙은(위 식에서 $width(a)\geq 1$, $width(b)\geq 1$ 이므로) 적절히 재귀적으로 계산된 항들을 더한 것이 커널 값이 된다.

[1]에 따르면 여기서 감쇠상수의 값을 0.1, 0.5, 0.9로 시험했을 때 SVM을 이용한 방법에서 성능에 별다른 차이가 없었다고 하는데, 우리는 이 중 0.1을 썼을 때 LCA의 유사도 함수 값이 대부분의 경우 다른 모든 항들을 합한 것보다 크다는 것을 간단한 실험을 통해 확인했다. 이는 LCA의 정보가 이 모델에서 상당히 중요한 역할을 한다는 사실을 말해준다.

Open IE는 주어진 문장에서 가능한 개념 쌍들과 그 개념들 사이의 관계를 나타내주는 문장 내의 표현으로 이루어진 (head, relation, tail) 형태의 triple을 모두 찾아내는 문제이다. 예를 들어 "Born in a small town, she took the midnight train going anywhere" ([8]) 에서는 (she, born_in, town)와 (she, took, midnight train)을 찾아낼 수 있다. [8]에서는 문장을 더 작은 clause들로 나누는 방법을 기반으로 한 Open IE 시스템을 제안하였다. 문장이 clause로 쪼개진 후에는 'Inter-Clause Open IE'와 'Intra-Clause Open IE'를 시행해서 triple들을 탐색하게 된다. 앞 예시에서의 첫 번째 triple은 전자에서, 두 번째 triple은 후자에서 나온 경우이다. 우리의 관심 대상인 형용사구 내 관계 추출은 이 'Inter-Clause IE'의 한 예시로 볼 수 있다. 하지만 이 논문에서 수행한 Open IE는 관계를 주어

진 목록들 중의 하나로 분류하는 것이 아니라 관계에 해당하는 문장 내에서의 표현만을 골라내는 것이 목표이므로 우리의 연구와는 거리가 있다.

3. 제안 방법

Universal Dependencies의 의존타입 중에는 'acl'(adjective clause)과 'acl:relcl'(adjective clause : relative clause)이 있는데 이것들은 형용사구의 술부와 그것이 수식하는 명사 사이에 나타나게 된다. 의존 구문 분석이 정확하게 되었다고 가정하였을 때, 서론에서 설명한 '형용사구 내 관계'는 다음의 조건들(이하 '조건1~3')이 모두 만족 되는지를 확인함으로써 선별해 낼 수 있다.

- 조건1 : 관심의 대상인 $e_1 = [x_{i_1}, \cdots, x_{j_1}]$ 과 $e_2 = [x_{i_2}, \cdots, x_{j_2}]$ 두 개체($x_{i_k}, \cdots, x_{j_k}$는 e_k를 구성하는 노드들이다.)의 LCA가 $x_{i_1}, \cdots, x_{j_1}, x_{i_2}, \cdots, x_{j_2}$ 중 하나이다. (그것을 x라고 하고, 편의상 이것이 e_1에 속해 있다고 하자.)
- 조건2 : x의 자식노드 중 의존타입이 'acl' 또는 'acl:relcl'인 것이 있다. (이를 y라고 하자.)
- 조건3 : y의 자식노드 중 e_2에 속해 있는 것이 있다.

예를 들어 그림 1에서 e1=['you'], e2=['girl']일 때는 x= 'girl' , y= 'love' 가 조건들을 만족시키지만 e1=['I'], e2=['girl']일 때는 조건들을 만족시키는 (x,y)가 존재하지 않는다.

제안하는 방법의 핵심은 이러한 조건을 만족하는 경우 변형된 방법을 쓰는 것으로, 원래의 LCA인 x 대신 y를 쓰는 것이다.

4. 실험 및 결과

4.1. 실험 개요

실험의 가설은 독립적으로 관계를 예측해 내는 능력을 기준으로 봤을 때 LCA를 그대로 사용하는 것보다 조건1~3이 만족되는 경우에는 그 형용사구의 술어(y)를 대신 사용하는 것이 관계추출에 대한 더 좋은 자질이 된다는 것이다. 이를 확인하기 위해 해당 자질들에 대응하는 단어 벡터만을 이용해서 자주 통용되는 기계학습 알고리즘들을 적용하여 정확도를 비교하였다. 기계학습 알고리즘들로는 로지스틱 회귀분석, SVM(linear), SVM(exponential kernel)을 이용하였다. 또한 더 세밀한 분석을 위해 가능한 학습 시나리오 3가지를 사용하였다.

1)조건1~3을 만족하는 문장만을 통해 학습한 경우
2)그 외의 문장만을 통해 학습한 경우
3)모든 문장을 통해 학습한 경우

또한 아래와 같이 두 가지 가능한 평가 방법들을 사용하여 실험을 수행하였다.

1)조건1~3을 만족하는 문장들로 평가하는 경우
2)그 외의 문장들로 평가하는 경우

의존 구문 분석은 Stanford CoreNLP 모듈의 Universal Dependencies를 이용하였고, 로지스틱 회귀분석, SVM(linear), SVM(exponential kernel)은 scikit-learn 모듈의 기본 옵션들을 이용하였다. 단어 벡터로는 GloVe([9])의 300차원짜리 벡터들을 이용하였다. GloVe는 비지도 학습을 이용하는 단어 임베딩 모델 중 하나로, [9]에 따르면 흔히 사용되는 또 다른 모델인 word2vec([10])보다 성능이 뛰어나다.

4.2. 데이터셋

관계추출의 데이터셋으로 SemEval2010 Task8([11]) 을 이용하였다. 학습데이터의 문장은 8000개이고 시험데이터의 문장은 2717개이다. 아래와 같이 총 10개의 관계가 있다.

Other
Cause-Effect
Instrument-Agency
Product-Producer
Content-Container
Entity-Origin
Entity-Destination
Component-Whole
Member-Collection
Message-Topic

Other를 제외한 각 관계는 개체의 순서가 Cause-Effect(e1,e2), Cause-Effect(e2,e1)와 같이 (Cause-Effect(e1,e2)는 e1이 Cause, e2가 Effect인 경우이고 Cause-Effect(e2,e1)은 그 반대의 경우이다.) 구분되어 있어서, 총 19개의 라벨이 있는 것으로 볼 수 있다. 본 실험에서는 개체의 순서를 구분하지 않은 10개의 관계들만을 분류하는 문제를 고려하였다. LCA만으로 관계에서 개체의 순서까지 구별해 낼 수 있기를 기대하기는 힘들기 때문이다.

4.3. 결과

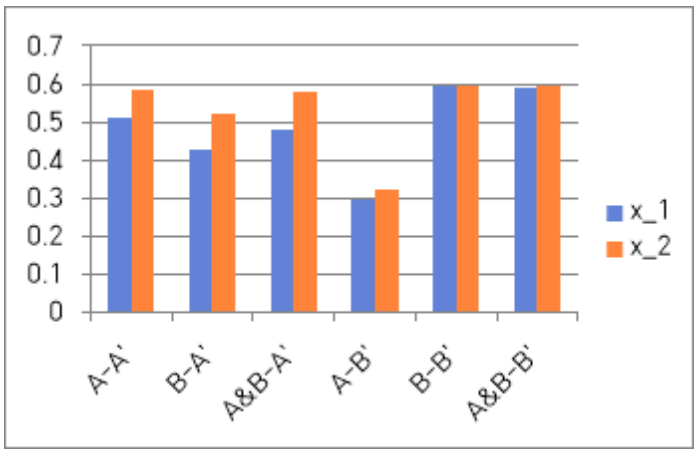
[그림 2], [그림 3], [그림 4]에 나타난 실험결과들에서 각각의 문자는 다음을 나타낸다.

- A: 학습 데이터의 문장들 중 조건1~3을 만족하는 것들의 집합
- A': 시험 데이터의 문장들 중 조건1~3을 만족하는

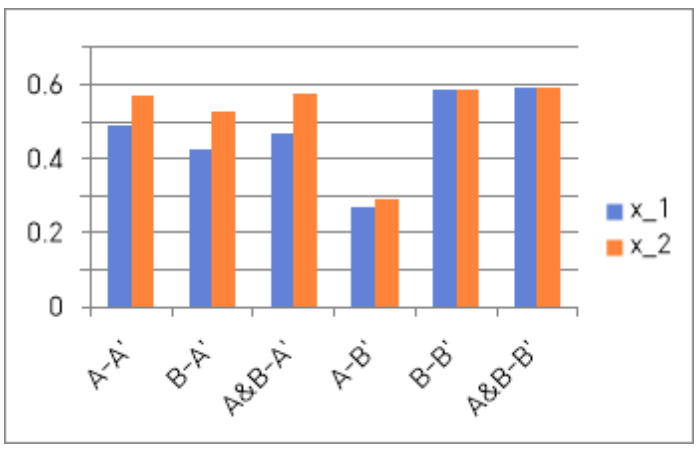
것들의 집합

- B: 학습 데이터의 문장들 중 조건1~3을 만족하지 않는 것들의 집합
- B': 학습 데이터의 문장들 중 조건1~3을 만족하지 않는 것들의 집합
- $x_1 : A \cup A' \cup B \cup B' \rightarrow \mathrm{R}^{\mathrm{d}}$: 주어진 문장의 LCA에 해당하는 단어벡터
- $x_2 : A \cup B \cup A' \cup B' \rightarrow R^d$: 주어진 문장이 조건1~3을 만족하는 경우에는 앞서 설명한 방법으로 새로 찾은 단어, 그렇지 않은 경우에는 LCA에 해당하는 단어벡터

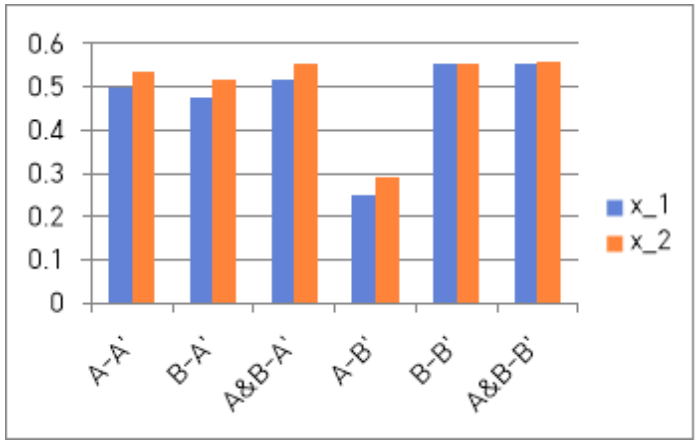
(즉 $x_2|_{B \cup B'} = x_1|_{B \cup B'}$ 이지만 $x_1|_{A \cup A'} \neq x_2|_{A \cup A'}$이다.)



[그림 2] 로지스틱 회귀분석을 이용해서 학습 방법과 평가 방법의 가능한 모든 쌍에 대해 기존의 자질과 새로운 자질을 평가한 결과 비교



[그림 3] SVM(linear) 을 이용해서 학습 방법과 평가 방법의 가능한 모든 쌍에 대해 기존의 자질과 새로운 자질을 평가한 결과 비교



[그림 4] SVM(exponential kernel) 을 이용해서 학습 방법과 평가 방법의 가능한 모든 쌍에 대해 기존의 자질과 새로운 자질을 평가한 결과 비교

실험 결과 자명하게 같은 정확도가 나오는 경우(B-B')를 제외하고는 모든 알고리즘과 모든 학습데이터, 평가데이터 쌍에서 새로운 자질을 이용했을 때 정확도가 조금씩은 상승하였음을 알 수 있다. 비형용사구 관계 추출(B')에서는 그 차이가 크지 않은 반면 형용사구 내 관계 추출(A')에서는 모든 알고리즘과 모든 학습 시나리오(A, B, A&B)에서 차이가 뚜렷하였다. 비형용사구 관계 추출의 경우에 차이가 크지 않은 것은 그 경우의 처리를 다르게 하지는 않으므로 이해할만한 결과이다. 여기서 차이가 0이 아닌 이유는 형용사구 내 관계(A)에서 뽑은 자질들이 비형용사구에서 뽑은 자질들과 더 호환되는 방향으로 바뀌었기 때문이라고 할 수 있다.

또 관찰할 수 있는 점은 형용사구 내 관계 추출(A')이 비형용사구 관계 추출(B')에 비해 형용사구 내 관계(A)만을 이용해 학습한 경우를 제외하고는 성능이 낮았으며 새로운 방법을 썼을 때는 그 정도가 덜했다는 것이다. 이는 A, A' 각각이 학습 데이터/시험 데이터에서 약 10% 정도의 크기밖에 되지 않는다는 것, 즉 데이터의 불균형 때문으로 보이며, 형용사구 내 관계 추출과 비형용사구 관계 추출이 상이한 성질을 가지며 새로운 방법이 그 차이를 줄여 더 일관적인 자질을 만들어 냄을 보여준다.

5. 결론

본 연구에서는 형용사구 내 관계추출의 경우에 원래의 LCA 대신 형용사구의 술어를 쓰는 것이 비형용사구 문장에서의 LCA와의 호환성을 고려하였을 때 자질로서 더 적합하다는 것을 실험을 통해 확인하였다. 이 결과는 일반적으로 의존 구문 기반 관계 추출 알고리즘을 설계할 때 형용사구 내 관계의 경우를 따로 고려하는 것이 중요함을 시사한다. 또한 앞 절의 마지막 문단에서 언급한 내용과 같이 관계 추출 데이터셋을 만들 때 형용사 내 관계를 추출해야 하는 경우가 전체 데이터에서 차지하는 비율이 얼마나 되는지도 고려하는 것이 바람직함을 알 수 있다.

6. 향후 연구

이번 연구에서는 LCA를 수정하는 것이 더 좋은 자질이 된다는 것만을 확인했을 뿐 그 관찰에 따른 더 나은 관계추출 모델을 제안하지는 않았다. 기존의 여러 모델들에 대해 이 점에서 착안한 변형을 고려할 수 있을 것인데 그것이 가능한 경우와 가능하지 않은 경우를 정리한다.

- 가능하지 않은 경우: 트리를 무방향 그래프(undirected graph)로 보는 경우(예를 들어 대칭인 인접행렬(adjacency matrix)을 이용하는 경우), 방향-불변인 특징들을 이용하는 경우(예를 들어 개체들 간의 최소경로)
- 가능한 경우: 트리 커널을 이용하는 경우, 최소공동조상 자체를 자질로 이용하는 경우, 방향을 고려한 인접행렬을 이용하는 경우

참고문헌

[1] Bach, Nguyen, and Sameer Badaskar. A review of relation extraction. Literature review for Language and Statistics II 2, 2007.

[2] Kumar, Shantanu. A Survey of Deep Learning Methods for Relation Extraction. arXiv preprint arXiv:1705.03645, 2017.

[3] De Marneffe, Marie-Catherine, and Christopher D. Manning, Stanford typed dependencies manual, Technical report, Stanford University, 2008, pp.338-345

[4] Nivre, Joakim, et al., Universal Dependencies v1: A Multilingual Treebank Collection, LREC, 2016.

[5] Aron Culotta and Jeffrey Sorensen, Dependency Tree Kernels for Relation Extraction, Proceedings of the 42nd annual meeting on association for computational linguistics, p.423, 2004.

[6] Razvan C. Bunescu and Raymond J. Mooney, A Shortest Path Dependency Kernel for Relation Extraction, Proceedings of the conference on human language technology and empirical methods in natural language processing, pp.724-731, 2005.

[7]Yan Xu, Lili Mou, Ge Li, Yunchuan Chen, Hao Peng and Zhi Jin, Classifying Relations via Long Short Term Memory Networks along Shortest Dependency Paths, In Proceedings of the 2015 conference on empirical methods in natural language processing, pp.1785-1794, 2015.

[8] Gabor Angeli, Melvin Johnson Premkumar amd Christopher D. Manning, Leveraging Liguistic Structure For Open Domain Information Extraction, In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, vol. 1, pp.344-354, 2015.

[9] Pennington, Jeffrey, Richard Socher, and Christopher Manning, Glove: Global vectors for word representation, Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), 2014.

[10] Mikolov, Tomas, et al., Distributed representations of words and phrases and their compositionality, Advances in neural information processing systems, 2013.

[11] Hendrickx, Iris, et al., Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals, Proceedings of the Workshop on Semantic Evaluations: Recent Achievements and Future Directions, Association for Computational Linguistics, 2009.