

Automatic Expansion of ConceptNet by Using Neural Tensor Networks

Yong Seok Choi[†] · Gyoung Ho Lee^{**} · Kong Joo Lee^{***}

ABSTRACT

ConceptNet is a common sense knowledge base which is formed in a semantic graph whose nodes represent concepts and edges show relationships between concepts. As it is difficult to make knowledge base integrity, a knowledge base often suffers from incompleteness problem. Therefore the quality of reasoning performed over such knowledge bases is sometimes unreliable. This work presents neural tensor networks which can alleviate the problem of knowledge bases incompleteness by reasoning new assertions and adding them into ConceptNet. The neural tensor networks are trained with a collection of assertions extracted from ConceptNet. The input of the networks is two concepts, and the output is the confidence score, telling how possible the connection between two concepts is under a specified relationship. The neural tensor networks can expand the usefulness of ConceptNet by increasing the degree of nodes. The accuracy of the neural tensor networks is 87.7% on testing data set. Also the neural tensor networks can predict a new assertion which does not exist in ConceptNet with an accuracy 85.01%.

Keywords : ConceptNet, Neural Tensor Networks, Recurrent Neural Networks

신경 텐서망을 이용한 컨셉넷 자동 확장

최 용 석[†] · 이 경 호^{**} · 이 공 주^{***}

요 약

컨셉넷은 일반상식을 노드(개념)와 에지(관계)로 표현해 놓은 그래프 형태의 지식 베이스이다. 완전한 지식 베이스를 구축하는 것은 매우 어려운 문제이기 때문에 지식 베이스는 미완결된 형태의 데이터를 담고 있는 경우가 많다. 불완전한 지식을 담고 있는 지식 베이스로부터의 추론 결과는 신뢰하기 어렵기 때문에 지식의 완결성을 높이기 위한 방법이 필요하다. 본 논문에서는 신경 텐서망을 이용하여 컨셉넷의 지식 미완결성 문제를 완화해 보고자 한다. 컨셉넷에서 추출한 사실주장(assertion)을 이용하여 신경 텐서망을 학습시킨다. 학습된 신경 텐서망은 두 개의 개념 정보를 입력으로 받고, 그 두 개념이 특정 관계로 연결될 수 있는지를 나타내는 점수값을 출력한다. 이와 같이 신경 텐서망은 노드들의 연결 차수(degree)를 높여, 컨셉넷의 완결성을 증대시킬 수 있다. 본 연구에서 학습시킨 신경 텐서망은 평가데이터에 대해서 약 87.7%의 정확도를 보였다. 또한 컨셉넷에 연결이 없는 노드 쌍에 대하여 85.01%의 정확도로 새로운 관계를 예측할 수 있었다.

키워드 : 컨셉넷, 신경 텐서망, 재귀 신경망

1. 서 론

컨셉넷(ConceptNet)은 “재즈는 뮤직장르이다”와 같은 Is-A 관계뿐 아니라 “색소폰은 재즈에 사용된다”와 같은 일반 상식도 함께 다루는 지식 베이스이다[1]. 컨셉넷은 자연 언어로 표현된 실세계의 지식을 노드(개념)와 노드 간의 방향이 있는 에지(관계)로 표현한 의미 그래프이며 질의-응답(question answering), 추론(reasoning)과 같은 인공지능의

여러 응용에 필수적인 지식 자원으로 사용될 수 있다.

컨셉넷은 Open Mind Common Sense(OMCS) 프로젝트의 결과물이다. 컨셉넷에는 WordNet[2]과 DBPedia[3]와 같은 기존의 지식 베이스뿐만 아니라 OMCS 사이트를 방문한 사용자들이 직접 입력한 문장, 위키피디아 텍스트, 온라인 게임 문서 등을 자동 분석하여 추출한 지식도 포함되어 있다. 이와 같이 여러 다른 형태의 지식베이스를 수집하여 만들었기 때문에 노드와 노드 사이가 연결이 불완전하게 구축되어 있는 경우가 많다.

Fig. 1은 두 노드들 사이의 연결된 노드들을 추상화한 그림이다.

(a)는 “drive car”와 “drive truck” 사이의 노드들의 일부이다. “drive car”는 3개의 노드들과 ‘HasSubevent’의 관계로 연결되어 있는 반면, “drive truck”은 ‘shift gear’와만 연

※ 이 논문은 2015년도 정부(미래창조과학부)의 재원으로 한국연구재단 - 이공분야기초연구 - 여성과학자 지원을 받아 수행된 연구임(No. 2015051685).

† 준 회 원 : 충남대학교 전자전파정보통신공학과 석사과정

** 준 회 원 : 충남대학교 전자전파정보통신공학과 박사과정

*** 종신회원 : 충남대학교 전자전파정보통신공학과 교수

Manuscript Received : October 4, 2016

Accepted : October 13, 2016

* Corresponding Author : Kong Joo Lee(kjoolee@cnu.ac.kr)

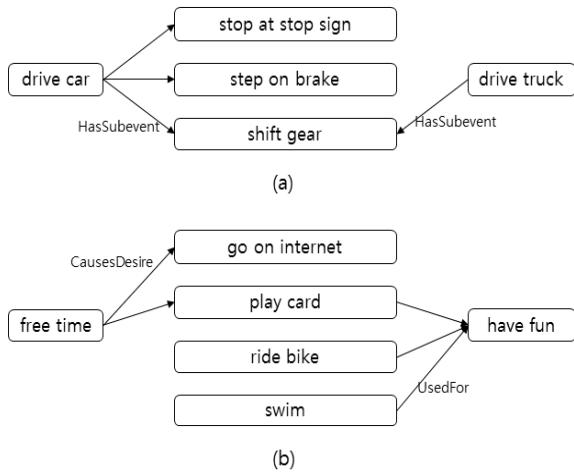


Fig. 1. Examples of Connection Between Two Nodes in ConceptNet

결되어 있음을 볼 수 있다. 그러나 “drive car”와 유사한 의미인 “drive truck”도 나머지 2개 노드와 ‘HasSubevent’와의 관계로 연결될 수 있어야 한다. (b)는 “free time”과 “have fun” 사이의 노드들의 일부이다. (b)로부터 노드 “free time”은 “go on internet”과 “play card”를 하고 싶게 만들며, “play card”를 하면 “have fun” 할 수 있다는 추론이 가능하다. 그러나 노드들 사이의 다른 연결이 완전하지 않기 때문에 “자유 시간이 있어서 자전거를 탔고 결과적으로 즐거운 시간을 가졌다”라는 추론은 가능하지 않다.

지식 베이스를 완벽하게 구축하는 것은 매우 어려운 작업이기 때문에 대다수의 지식 베이스들은 불완전하게 구축된 지식 정보를 포함한다[4]. Fig. 1의 예제에서 보았듯이 컨셉넷도 노드와 노드 간의 연결 정도가 매우 낮음을 알 수 있다. 본 논문에서는 신경망 중 하나인 신경 텐서망[5]을 이용하여 노드와 노드 간의 불완전한 연결 문제를 완화해 보고자 한다. 컨셉넷에 이미 존재하는 사실주장을 이용하여 신경 텐서망을 학습시키고 학습된 신경 텐서망을 이용하여 컨셉넷에는 연결이 없지만 연결 가능성이 높은 노드와 노드를 자동으로 연결해 보고자 한다. 이렇게 함으로써 컨셉넷 지식 베이스의 완성도를 높일 수 있고 결과적으로 컨셉넷을 지식 자원으로 사용하는 응용 시스템의 성능 향상을 기대할 수 있다.

2. 관련 연구

2.1 컨셉넷

Fig. 2는 컨셉넷의 일부를 그래프로 표현한 것이다. 컨셉넷에서 다루는 지식 단위인 사실주장(assertion)은 두 개의 노드(개념)와 방향성을 가진 에지(관계)로 표현된다. Fig. 2에서 (‘eat’, ‘HasSubevent’, ‘swallow’)가 사실주장의 예이다.

Table 1은 각각의 관계가 자연언어로 어떻게 표현될 수 있는지를 보여준다. 예를 들어, 하나의 사실주장 (‘A, CapableOf, B’)는 자연언어 ‘A can B’로 기술될 수 있다.

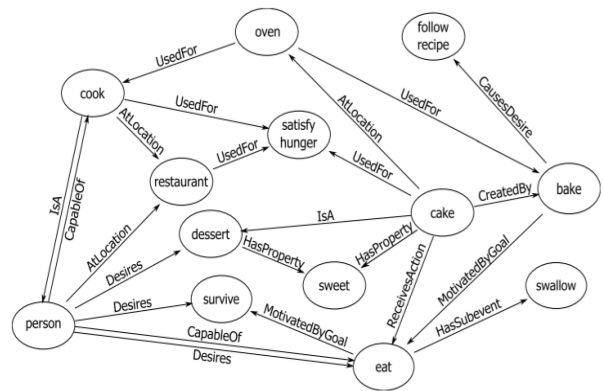


Fig. 2. Example of ConceptNet[1]

Table 1. The Relations in ConceptNet, With Example Sentence Frames[1]

Relation	Sentence pattern
IsA	NP is a kind of NP.
UsedFor	NP is used for VP.
HasA	NP has NP.
CapableOf	NP can VP.
Desires	NP wants to VP.
CreatedBy	You make NP by VP.
PartOf	NP is part of NP.
Causes	The effect of VP is NP VP.
HasFirstSubevent	The first thing you do when you VP is NP VP.
AtLocation	Somewhere NP can be is NP.
HasProperty	NP is AP.
LocatedNear	You are likely to find NP near NP.
DefinedAs	NP is defined as NP.
SymbolOf	NP represents NP.
ReceivesAction	NP can be VP.
HasPrerequisite	NP VP requires NP VP.
MotivatedByGoal	You would VP because you want VP.
CausesDesire	NP would make you want to VP.
MadeOf	NP is made of NP.
HasSubevent	One of the things you do when you VP is NP VP.
HasLastSubevent	The last thing you do when you VP is NP VP.

2.2 지식 베이스의 자동 추론

[4]의 연구에서는 지식 베이스의 완결성을 위해 자동 추론을 시도하였고 자동 추론 엔진으로는 신경 텐서망을 사용하였다. 지식 베이스 WordNet[2]과 Freebase[6]를 사용하여 신경망 모델을 학습시켰다. 학습된 신경 텐서망은 입력으로 주어진 두 개의 노드에 대하여 두 노드의 연결 가능성을 점수로 환산하여 출력한다. 개념을 나타내는 입력 노드는 개념에 쓰인 단어들의 임베딩(word embedding)값을 사용하여 표현한다. 단어 임베딩 값은 WordNet과 Freebase가 아닌 다른 코퍼스에서 학습시켰다. 실험 결과 WordNet로 학습한 모델은 86.2%, Freebase는 90.0%의 성능을 보였다.

우리의 연구는 [4]의 연구 결과를 기반으로 수행되었다. 그러나 본 연구에서 다루는 지식 베이스는 컨셉넷이다. [4]에서

대상으로 한 지식 베이스 WordNet이나 Freebase에 비해 컨셉넷에는 구나 절로 구성되는 개념들에 대한 정보가 많이 포함되어 있다. 본 연구에서는 이와 같은 컨셉넷의 특성을 파악하고 컨셉넷 지식 베이스에 적합하도록 모델을 수정하였다.

3. 컨셉넷 자동 확장을 위한 모델

3.1 신경 텐서망 모델

신경 텐서망은 텐서, 표준 선형 계층으로 구성되어있다 [5]. 두 노드는 $e_1, e_2 \in R^d$ 이다. Equation (1)은 신경 텐서망의 기본 수식이다.

$$g(e_1, X, e_2) = u_X^T f(e_1^T W_X^{[1:k]} e_2 + V_X \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} + b_X) \quad (1)$$

함수 f 는 비선형 함수인 \tanh 이고, $W^{[1:k]} \in R^{d \times d \times k}$ 이다. $h = R^k$ 는 k 개의 슬라이스로 텐서 계층을 표현한다. 이때 h 는 Equation (2)와 같다.

$$h_i = e_1^T W^{[i]} e_2 \quad (0 < i \leq k) \quad (2)$$

함수 f 안에 텐서 계층을 제외한 나머지 수식은 표준 선형 계층이다. 표준 선형 계층에 사용된 값은 다음과 같다:

$$V \in R^{k \times 2d}, U \in R^k, b \in R^k$$

Fig. 3은 $k = 2$ 일 때, Equation (1)을 그림으로 표현한 것이다.

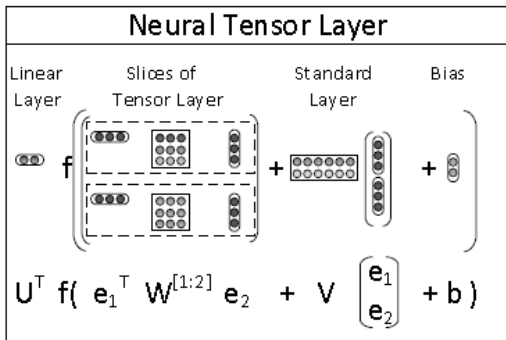


Fig. 3. Neural Tensor Networks[4]

신경 텐서망의 학습은 최대-마진(Max-margin)을 목적 함수로 사용한다. $T^{(i)} = (e_1^{(i)}, R^{(i)}, e_2^{(i)})$ 는 긍정 데이터, $T_c^{(i)} = (e_1^{(i)}, R^{(i)}, e_c)$ 는 부정 데이터를 나타내며, 목적 함수는 Equation (3)과 같고, 이 목적함수의 값이 최소가 되도록 학습한다.

$$\mathcal{J}(\Omega) = \sum_{i=1}^N \sum_{c=1}^C \max(0, 1 - g(T^{(i)}) + g(T_c^{(i)})) + \lambda \|\Omega\|_2^2 \quad (3)$$

본 연구에서는 컨셉넷의 각 관계마다 독립적인 신경 텐서망을 구축하였다. 두 노드 e_1, e_2 이 관계 X 를 학습한 신경 텐서망의 입력으로 주어지면, Equation (1)을 통해 구해진 점수가 출력으로 제시된다. 이 값이 관계 X 의 신경 텐서망의 임계값보다 크면 노드 e_1 와 e_2 는 관계 X 로 연결이 가능하다고 판단한다.

3.2 노드의 의미 표현

[4]의 연구에서 Equation (1)의 노드 e_1, e_2 의 입력값을 만들 때 노드를 표현하는 단어의 임베딩 값을 사용하였다. 그러나 노드가 구나 절과 같이 여러 단어로 구성되어 있을 때에는 여러 단어의 임베딩 값의 결합 방법을 고려해야 한다.

본 연구에서 다루는 컨셉넷의 노드는 약 80%가 단어의 구/절로 이루어져 있다. 그래서 본 연구에서는 노드를 구성하는 구/절의 의미를 결합하는 방법을 두 가지로 나누어 시도해 보고자 한다. 첫째는 구/절에 포함된 단어 임베딩의 평균값을 사용한다[4]. 둘째는 단어 임베딩을 재귀 신경망(Recurrent Neural Network)[7]의 입력으로 주어 출력 값을 사용한다. 이들 각각의 Fig. 4와 Fig. 5에 사실주장('stand_in_line', 'MotivatedByGoal', 'purchase_ticket')를 예제로 하여 도식화하였다.

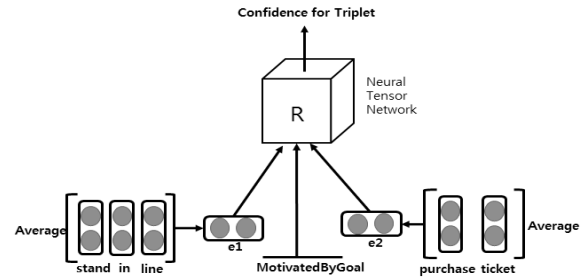


Fig. 4. Semantic Representation of the Node 1[4]

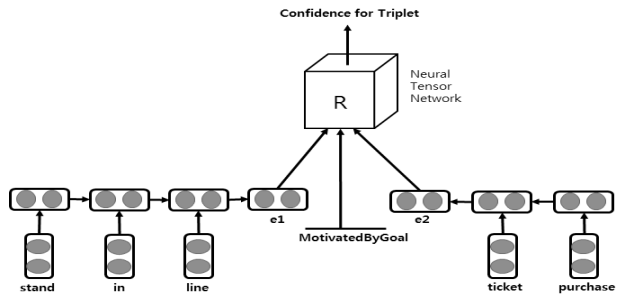


Fig. 5. Semantic Representation of the Node 2

4. 실험 및 결과

4.1 컨셉넷 학습 및 평가 데이터

본 연구에서는 컨셉넷의 관계 중, 자연언어로 기술 했을 때 동사구(VP)와 동사구(VP) 사이에서 발생할 수 있는 관계들을 실험 대상으로 다루고자 한다. 즉, Table 1의 문장

패턴에서 관계의 양쪽 끝 노드에 모두 동사구가 가능한 관계들을 실험 대상으로 삼았다. 이렇게 설정한 이유는 (동사구, 관계, 동사구)의 사실주장을 자연언어로 기술했을 때 대부분 단문이기보다는 구나 절을 포함하고 있는 복문 형태를 취하게 되고 복문 형태를 취하는 사실주장들의 수집이 단문 형태의 사실주장을 수집하는 것보다 어려울 것으로 예상되어 데이터의 완결성이 더 낮을 것으로 판단되기 때문이다. 여기에 해당되는 관계는 총 6개('Causes', 'HasPrerequisite', 'MotivatedByGoal', 'HasSubevent', 'HasFirstSubevent', 'HasLastSubevent')인데, 이 중, 'HasFirstSubevent'와 'HasLastSubevent'는 데이터 수가 부족하여 실험에서 제외하였다.

4개 관계에 대한 신경 텐서망을 구축하기 위한 학습, 검증, 평가 데이터 구성은 Table 2와 같다. 학습 데이터는 40,722개의 긍정 데이터만으로 구성되어 있고, 학습 시에 임의의 노드를 선택하여 매번 부정 데이터를 생성하였다. 검증과 평가에는 1,000개의 긍정 데이터와 임의의 노드를 선택하여 만든 1,000개의 부정 데이터를 합하여 2,000개의 데이터가 각각 사용되었다. 데이터에 포함된 노드의 개수는 24,644개이다.

Table 2. Train/Validation/Test Dataset

	Train	Validation	Test
number of data	40,722	2,000	2,000

4.2 모델 실험 결과

신경 텐서망 모델 학습을 위한 파라미터 설정은 다음과 같다. (1) regularization parameter $\lambda = 0.0001$, (2) 슬라이스 크기 $k = 4$, (3) 학습 반복횟수는 iteration=500, (4) 배치 크기는 5,000, (5) 단어 임베딩의 차원 $d = 300$ 으로 설정하였다. 두 입력 노드에 대한 관계 연결이 가능한지를 최종 판단하는 임계값은 (4.1)절의 검증 데이터를 이용하여 각 관계 신경 텐서망마다 가장 높은 정확도를 보이는 값으로 설정하였다.

단어 임베딩 초기화 방법과 노드의 의미 결합 방법에 따라 총 4개의 모델에 대하여 평가를 수행하였다. 단어 임베딩 초기화 경우, 랜덤 값으로 초기화한 경우에는 'Random', [8]로부터 학습한 값으로 초기화한 경우에는 'Pretrained'로 표시하였다. 노드의 의미 결합 방법에서 각 단어의 평균값

Table 3. Experimental Results of Model

Model Relation	Random - Mean	Pretrained - Mean	Random - RNN	Pretrained - RNN
Causes	75.94%	85.29%	87.70%	88.50%
HasPrerequisite	77.51%	85.29%	87.89%	90.66%
HasSubevent	78.73%	79.03%	85.09%	81.99%
MotivatedByGoal	79.21%	87.62%	91.09%	91.83%
Average	77.95%	83.75%	87.60%	87.7%

을 취한 경우에는 'Mean', 재귀 신경망을 이용한 경우에는 'RNN'으로 표시하였다.

Table 3은 평가 데이터에 대한 4개 모델의 관계별 정확도 실험결과이다. 전체적으로 'Pretrained-RNN' 모델의 정확도가 가장 높은 것을 볼 수 있다. 또한 노드의 의미 결합에 'RNN'을 사용한 모델이 'Mean'을 사용한 모델에 비해 좋은 성능을 보였다. 두 개의 'Mean' 모델은 단어 임베딩의 초기화 방법에 따라 성능 차이가 컸지만 'RNN' 모델에서는 큰 차이가 없었다. Table 3의 실험 결과를 살펴보면 노드의 의미 결합 방법이 모델 성능에 가장 많은 영향을 미치고 있음을 알 수 있다. 노드의 의미 결합 방법에 따라 여러 단어로 구성된 노드의 임베딩 값이 어떻게 변하는지 살펴보기 위하여 'have_rest'와 'go_to_party'와 가장 유사도가 높은 노드 10개를 추출해 보았다. Table 4에 각 방법에 따른 유사도 top-10 노드들을 제시하였다. 'Mean' 방식의 결합은 구절에 포함된 단어들의 가중치가 모두 동일하다 보니 'have_rest'의 유사도 top-10 중에 'have_asthma', 'have_skate'와 같이 'have_rest'와 다소 상관이 없는 의미의 노드들이 6개나 포함되어 있었다. 반면 'RNN' 방식의 결합에는 이런 현상이 발생하지 않았다. 'go_to_party'의 경우에도 'Mean' 방식의 결합에는 'go_to_war', 'go_to_court'와 같이 다소 관련 없는 노드가 6개나 포함되어 있었다. 'Mean' 방식의 결합은 구절을 구성하는 단어들 사이에 동일 가중치를 사용하여 전체 노드의 의미를 얻어낸다. 그러다 보니 동사구절에 많이 나타나는 'have_', 'take_', 'get_'과 같은 단어들 전체 노드의 의미를 결정하게 되는 경우들이 발생했다. 이와 같은 이유로 'RNN'을 이용한 의미 결합이 'Mean' 방식보다 좋은 성능을 보인 것으로 분석된다.

Table 4. Examples of Similar Word of the Node According to the Method of Combining Meaning the Node

'have_rest'	Mean	we_have_to_rest, rest, have_energy_after_rest, have_to_stop_for_rest, have_asthma* , have_skate* , have_dinner* , have_peace* , have_cold* , have_clue*
	RNN	we_have_to_rest, rest, feel_rest, have_energy_after_rest, more_rest, stop_to_rest, feel_more_rest, take_rest, should_rest, become_rest
'go_to_party'	Mean	you_go_to_party, go_to_dinner_party, go_to_birthday_party, that_you_go_to_party, go_to_war* , go_to_court* , go_to_theatre* , go_to_grocery* , not_to_go_to_jail* , go_to_cinema*
	RNN	you_go_to_party, that_you_go_to_party, go_to_dinner_party, go_to_birthday_party, invite_person_to_party, party, wake_up_ready_to_party, be_invite_to_party, invite_person_to_your_party, have_party

4.3 컨셉넷 자동 확장을 위한 신경 텐서망 모델의 유용성 평가

본 연구에서 구축한 신경 텐서망이 실제로 컨셉넷의 미연결된 노드 사이를 적절한 관계 정보로 연결시킬 수 있는지를 판단하기 위해 다음의 실험을 수행하였다.

컨셉넷에 연결이 존재하지 않으나 연결 가능성이 높은 두 노드 쌍을 추출하기 위해 다음과 같은 과정을 거쳤다.

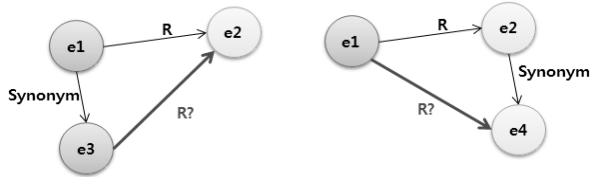


Fig. 6. Examples of Extracting the R-Linkable

Fig. 6과 같이 컨셉넷에 (e1, R, e2)가 존재하면서 (e1, Synonym, e3) 또는 (e2, Synonym, e4)이 존재하지만 (e3, R, e2), (e1, R, e4)가 존재하지 않을 경우 (e3, R, e2)와 (e1, R, e4)는 매우 가능성 높은 사실주장으로 간주할 수 있다. 이와 같은 사실주장에 포함되는 노드 쌍 집합을 컨셉넷에서 추출하고 이렇게 구축된 평가 집합을 R-Linkable이라고 하자. 즉, Fig. 6의 경우 (e3, e2)와 (e1, e4)를 R-Linkable 집합에 포함시킨다.

R 관계를 학습한 신경 텐서망 모델은 R-Linkable 집합에 포함된 노드 쌍에 대해 연결 가능성이 매우 높다고 판단할 수 있어야 한다. Table 5는 이렇게 구축된 R-Linkable 평가 집합 정보이다.

Table 5. Evaluation Set of the R-Linkable

Evaluation Set	number of data
Causes-Linkable	154
HasPrerequisite-Linkable	175
HasSubevent-Linkable	377
MotivatedByGoal-Linkable	88
Total	794

실험은 (4.2)절에서 가장 좋은 성능을 보인 ‘Pretrained-R NN’ 모델을 사용하였다. Table 6은 실험 결과이다. ‘Causes-Linkable’ 관계에서 가장 좋은 결과를 보여주었으며, 전체 정확도는 85.01%이다.

Table 6. Results of R-Linkable

Evaluation Set	Accuracy
Causes-Linkable	87.66%
HasPrerequisite-Linkable	81.14%
HasSubevent-Linkable	85.15%
MotivatedByGoal-Linkable	87.50%
Average	85.01%

Table 7에서 (a)는 R-Linkable 집합에 대해 신경 텐서망 모델이 연결 가능하다고 판단한 것들의 일부이다. (b)와 (c)는 R-Linkable 집합에 대해 연결 가능하지 않다고 예측한 것들의 일부로 (b)의 경우에는 모델의 예측이 잘못된 경우이다. 그러나 (c)는 R-Linkable 집합 구축이 잘못되었기 때문에 모델이 제시한 연결 불가능 예측이 오히려 맞는 경우이다.

(c)의 예와 같이 잘못된 R-Linkable 집합이 구축된 이유는 다음과 같다.

Causes-Linkable에 포함된 (violin, break_something) 쌍은 컨셉넷에 존재하는 (fiddle, Causes, break_something)과 (fiddle, Synonym, violin)에 의해서 추출되었다. 단어 ‘fiddle’은 ‘만지작거리다. 바이올린’의 다중 의미를 가지고 있는데 ‘Causes’ 관계에서는 ‘만지작거리다’의 의미로 사용된 반면, ‘Synonym’에서는 ‘바이올린’의 뜻으로만 사용되었기 때문에 의미 중의성이 해결되지 않은 상태에서 추출된(violin, break_something)은 ‘Causes’의 관계로 연결되지 않는 것이 옳다.

HasPrerequisite-Linkable에 포함된 (win_baseball_game, write_music) 쌍도 컨셉넷에 존재하는 (win_baseball_game, HasPrerequisite, score)과 (score, Synonym, write_music)에 의해서 추출되었으며, 이 경우 역시 단어 ‘score’의 의미 중의성이 해결되어 있지 않기 때문에 ‘HasPrerequisite’ 관계로 연결될 수 없는 쌍 (win_baseball_game, write_music)이 추출되었다.

5. 결론

본 논문은 일반상식 지식베이스의 일종인 컨셉넷이 갖고 있는 데이터 미완결성 문제를 해결하기 위하여 신경 텐서망을 이용하여 기존의 지식으로부터 부족한 지식을 자동으로 추론해 보고 이의 유용성을 살펴보았다.

본 연구에서 대상으로 한 컨셉넷의 노드 정보는 평균

Table 7. Example of the Results of the R-Linkable

(a)	(watch_film, Causes, laugh), (kill_person, Causes, prison), (relax, HasPrerequisite, lie_down_in_your_bed), (violin, HasPrerequisite, learn_how_to_play_violin), (bath, HasPrerequisite, turn_on_water), (death, HasSubevent, stop_breathe), (bath, HasSubevent, that_you_get_wet), (war, HasSubevent, kill_enemy), (bath, MotivatedByGoal, it_make_you_smell_nice), (exercise, MotivatedByGoal, maintain_muscle_strength), (join_army, MotivatedByGoal, battle)
(b)	(death, Causes, get_cold), (view_video, Causes, go_to_sleep), (relax, HasSubevent, you_have_more_energy)
(c)	(violin, Causes, break_something), (win_baseball_game, HasPrerequisite, write_music)

3.48 단어로 구성되어 있어 단일 단어보다는 구나 절로 표현되어 있는 경우가 더 많았다. 그렇기 때문에 여러 단어로 구성된 구나 절의 개념을 어떻게 하나로 결합하느냐가 신경텐서망의 성능에 중요한 요소가 될 수 있다. 본 연구에서는 구나 절을 표현하는 방법으로 재귀 신경망을 이용하여 노드의 의미를 결합하였고, 기존의 평균을 취하는 방법에 비해 약 5%의 이상의 성능 향상을 얻을 수 있었다.

본 연구의 신경 텐서망은 평가 데이터에 대해서 약 87.7%의 정확도를 얻을 수 있었다. 또한 컨셉넷에서 연결되어 있지 않은 노드 쌍에 대하여 85.01%의 정확도로 새로운 관계를 예측할 수 있었다.

References

[1] Speer, Robert and Catherine Havasi, "Representing General Relational Knowledge in ConceptNet 5," *LREC*, 2012.

[2] George A. Miller, "WordNet: a lexical database for English," *Communications of the ACM*, Vol.38, No.11, pp.39-41, 1995.

[3] Jens Lehmann, et al., "DBpedia: A large-scale, multilingual knowledge base extracted from Wikipedia," *Semantic Web*, Vol.6, No.2, pp.167-195, 2015.

[4] Richard Socher, et al., "Reasoning with neural tensor networks for knowledge base completion," *Advances in Neural Information Processing Systems*, 2013.

[5] Danqi Chen, et al., "Learning new facts from knowledge bases with neural tensor networks and semantic word vectors," *arXiv*, 2013.

[6] Kurt Bollacker, et al., "Freebase: a collaboratively created graph database for structuring human knowledge," *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, 2008.

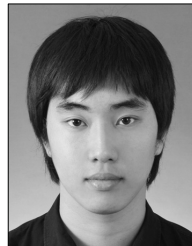
[7] Tomas Mikolov, et al., "Recurrent neural network based language model," *Interspeech*, Vol.2, 2010.

[8] Word embedding [Internet], <https://code.google.com/archive/p/word2vec/>.



최 응 석

e-mail : yongseok.choi.92@gmail.com
 2016년 충남대학교 정보통신공학과(학사)
 2016년~현 재 충남대학교 전자전파
 정보통신공학과 석사과정
 관심분야: 자연언어처리, 정보검색,
 기계학습, 인공지능



이 경 호

e-mail : gyholee@gmail.com
 2011년 충남대학교 정보통신공학과(학사)
 2013년 충남대학교 정보통신공학과(석사)
 2013년~현 재 충남대학교 전자전파
 정보통신공학과 박사과정
 관심분야: 자연언어처리, 기계학습,
 인공지능



이 공 주

e-mail : kjoolee@cnu.ac.kr
 1992년 서강대학교 전자계산학과(학사)
 1994년 한국과학기술원 전산학과
 (공학석사)
 1998년 한국과학기술원 전산학과
 (공학박사)

1998년~2003년 한국마이크로소프트(유) 연구원
 2003년 이화여자대학교 컴퓨터학과 대우전임강사
 2004년 경인여자대학 전산정보과 전임강사
 2005년~현 재 충남대학교 전파정보통신공학과 교수
 관심분야: 자연언어처리, 기계번역, 정보검색, 정보추출