

다중 흡 다중 작업 질문 응답을 위한 계층적 그래프 추론

이상의, 이기호, 김인철

경기대학교 컴퓨터과학과

rmrlrmrl125@kyonggi.ac.kr, kyo9250@kyonggi.ac.kr, kic@kyonggi.ac.kr

Hierarchical Graph Reasoning for Multi-hop, Multi-task Question Answering

Sangui Lee, Giho Lee, Incheol Kim

Department of Computer Science, Kyonggi University

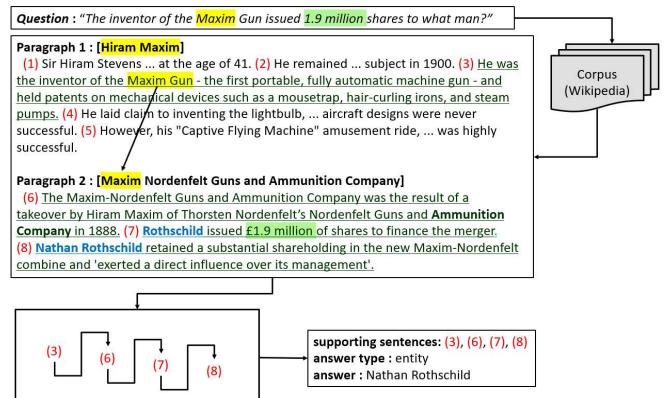
요약

최근 오픈 도메인 자연어 질문 응답 분야에서는 폭넓은 다중 문서들을 토대로 다중 흡 추론과 동시에 서로 다른 수준의 여러 문제들을 한꺼번에 해결해야 하는 다중 작업 질문 응답에 관한 관심이 높다. 본 논문에서는 이러한 다중 흡 추론과 다중 작업을 요구하는 복잡 질문들에 효과적으로 응답하기 위해, 계층적 그래프 기반의 새로운 심층 신경망 모델을 제안한다. 제안 모델에서는 계층적 그래프와 그래프 신경망을 이용해 다중 문서들로부터 서로 다른 수준의 맥락 정보를 얻어낸 후, 이를 활용하여 뒷받침 문장들, 답변 영역, 응답 유형 등을 동시에 구해야 하는 다중 작업 문제에 관한 답들을 예측해낸다. 본 논문에서는 오픈 도메인 자연어 질문 응답 데이터 집합인 HotpotQA를 이용한 실험들을 통해, 제안 모델의 궁정적 효과를 입증한다.

1. 서론

최근 오픈 도메인 자연어 질문 응답 분야에서는 답변에 도달하기까지 둘 이상의 문서나 문장들을 이용하는 다중 흡 추론(multi-hop reasoning)과 질문에 관한 답변(answer)이외에 뒷받침 문장들(supporting sentences)과 응답 유형(answer type) 등을 함께 구해야 하는 다중 작업(multi-task) 질문 응답 문제에 관한 관심이 높다[1]. (그림 1)은 이러한 오픈 도메인의 다중 흡, 다중 작업 질문 응답의 한 예를 보여준다. (그림 1)의 예와 같이 자연어 질문과 이와 연관된 다수의 문서와 문단 집합들이 주어졌을 때, 이들로부터 (3), (6), (7), (8)과 같은 뒷받침 문장을 연쇄적으로 찾은 후에야 질문에 관한 답변에 도달할 수 있으며, 답변이 외에 서로 다른 수준의 정보를 요구하는 뒷받침 문장들과 응답 유형들도 함께 구해야 한다.

다중 흡, 다중 작업 복잡 질문 응답에서 해결해야 할 중요한 문제들은 (1) 질문에 답하기 위한 뒷받침 증거(evidence)들이 여러 문서(document), 문단(paragraph), 문장(sentence)들에 흩어져 있어, 대규모 문서 집합으로부터 해당 질문에 연관성이 높은 문단과 문장들을 가려내어 하나의 자료 구조로 통합하는 일과 (2) 이들로부터 답변 예측에 필요한 서로 다른 수준의 문맥 정보(context)들을 효과적으로 추출해내는 일 등이 있다. 이러한 문제들을 해결하고자, 기존 연구 [2]에서는 순환 신경망(Recurrent Neural Network, RNN)을 이용해 문단 사이의 추론 경로를 탐색하였고, [1, 3, 4]의 연구들에서는 질문에 연관된 문서들로부터 추출된 문단, 문장, 혹은 개체들을 이용해 그래프를 생성한 뒤 그래프 신경망(Graph Neural Network, GNN)을 이용해 답변에 필요한 문맥 정보를 추출하였다. 특히 [1]에서는 문서에서 추출한 개체들로 구성된 개체 그래프(entity graph)를 이용하였고, [3]에서는 개체 그래프 외에 문장 그래프(sentence graph)도 함께 이용하였으며, [4]에서는 질문(question), 문단(paragraph), 문장(sentence), 개체(entity)들로 구성된 계층적 그래프를 이용하였다. 하지만 이들 그래프에는 서로 다른 레벨의 두 노드 간의 관계(relationship)들을 구분하여 표현할 수 있는 별도의 관계 노드(relation node)들을 포함하지 않고 있다는 한계가 있다.



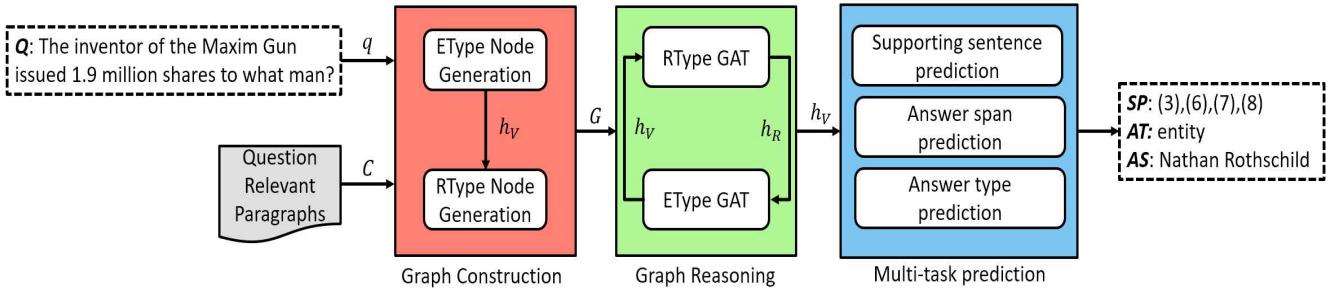
(그림 1) 다중 흡, 다중 작업 질문 응답의 예

본 논문에서는 이러한 기존 연구들의 문제점을 보완하는 새로운 다중 흡, 다중 작업 질문 응답 모델을 제안한다. 제안 모델에서는 문서 집합에서 질문과 연관성이 높은 문단들을 가려낸 뒤, 이 문단들을 기초로 서로 다른 수준의 다양한 맥락 정보를 얻기 위해 질문, 문단, 문장, 개체들로 구성된 계층적 그래프(hierarchical graph)를 생성한다. 기존의 연구들과는 달리, 이 계층적 그래프는 서로 다른 레벨의 두 노드 간의 관계들을 효과적으로 표현하기 위해 관계 노드(relation node)들을 포함한다. 그리고 이 그래프에 노드 유형별로 특화된 정보 생성 방식을 적용한 그래프 신경망(graph neural network)을 이용함으로써, 다중 작업 질문 응답에 필요한 각기 다른 수준의 맥락 정보를 얻어낸다. 본 논문에서는 제안 모델의 성능 평가를 위해, 대표적인 오픈 도메인 자연어 질문 응답 데이터 집합인 HotpotQA[5]를 이용한 실험들을 수행하고 그 결과를 소개한다.

2 다중 흡 다중 작업 질문 응답

2.1 문제 정의

본 논문에서는 문서에서 추출한 질문과 연관된 문단들로부터 뒷받침 문장들과 응답을 찾는 다중 작업 문제를 다루고 있다. 본 논문에서 자연어 질문은 질문을 구성하는



(그림 2) 제안 모델의 구성

단어들의 집합 $Q = \{w_1, w_2, \dots, w_q\}$, 문서 집합은 $D = \{d_1, d_2, \dots, d_T\}$ 로 정의한다. D 로부터 추출한 Q 와 관련 있는 문단들의 집합은 $P = \{p_1, p_2, \dots, p_l\}$ 로 정의한다. 각 문단을 구성하는 문장들의 집합은 $S = \{s_1, s_2, \dots, s_m\}$ 로 정의한다. 각 문장 내 포함된 개체들의 집합은 $E = \{e_1, e_2, \dots, e_n\}$ 로 정의한다. 이에 따라 추출한 문단들을 연결한 문맥은 $C = \{\text{concat}(p_1, p_2, \dots, p_l)\} (P \cup S \cup E \subset C)$ 로 정의된다. 여기서 concat 은 연결(concatenation)을 의미한다.

본 논문에서 다루는 문서 기반 질문 응답에서 다중 작업은 뒷받침 문장 찾기와 응답 찾기로 구성되어 있다. 뒷받침 문장 찾기는 C 에서 Q 와 관련 있는 뒷받침 문장들 $\{s_{r_1}, s_{r_2}, \dots, s_{r_z}\} \subset S$ 찾기로 정의된다. 응답 찾기는 한 뒷받침 문장 s_{r_1} 으로부터 다른 뒷받침 문장 s_{r_z} 까지의 추론 경로 $\langle s_{r_1}, s_{r_2}, \dots, s_{r_z} \rangle$ 를 탐색하는 다중 흡 추론을 수행하여 질문에 대한 응답 $e \in E$ 찾기로 정의된다.

2.2 제안 모델

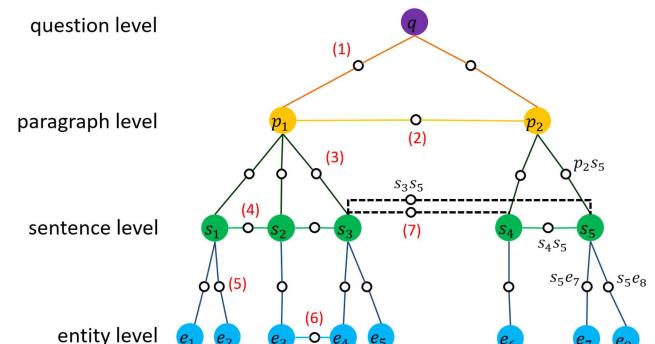
본 논문에서 제안하는 모델은 문서로부터 질문과 관련된 문단들을 추출하고, 이를 토대로 계층적 그래프를 생성하여 다중 흡 추론 및 다중 작업을 수행한다. (그림 2)는 제안하는 모델의 전체 구조도를 나타낸다. 입력으로 질문과 문서를 가지며, 출력으로 질문에 대한 응답인, 문서에 언급된 개체 또는 예/아니오를 갖는다. 문서에 자연어로 표현된 개체를 추출하기 위해서 본 논문에서는 BERT-NER를 이용한다. 질문과 관련된 문단들을 추출하기 위해 [1]처럼 BERT 기반 문단 분류기를 이용한다. 제안하는 모델은 그래프 구성(Graph Construction), 그래프 추론(Graph Reasoning), 다중 작업 예측(Multi-task Prediction)으로 구성되어 있다. 그래프 구성은 추출된 문단들과 질문을 이용해 계층적 그래프를 생성한다. 계층적 그래프는 4가지 종류의 Etype 노드와 Rtype 노드들 사이를 이어주는 6가지 종류의 Rtype 노드로 구성된다. 각 노드의 특징(feature)은 질문과 문단들을 BERT 및 Bi-Attention을 이용해 임베딩된 자연어 문장들에 대해 LSTM(Long Short-Term Memory)으로 추출한 특징을 활용하여 초기화한다. 그래프 추론은 계층적 그래프 위에서 질문에 대한 응답을 찾는 다중 흡 추론을 수행한다. 다중 흡 추론은 그래프를 이루는 노드들이 반복적으로 이웃 노드들의 맥락 정보(structured context)를 받고 특징을 갱신함으로써 이뤄진다. 다중 작업 예측은 그래프 추론이 끝난 계층적 그래프의 Etype 노드들을 이용하여 최종적으로 뒷받침 문장 예측(supporting sentences prediction), 응답 범위 예측(answer span prediction), 응답 유형 예측(answer type prediction)을 수행한다.

2.3 그래프 구성

문서는 문서를 이루는 각 문단, 문장, 단어 등의 종속 관계로 인해 계층적 그래프 형태로 표현될 수 있다. 계층

적 그래프의 각 계층이 갖는 맥락 정보는 맥락 수준에 따라 나뉘어 있어 다중 작업을 수행하는 데 개별적으로 사용될 수 있다. 본 논문에서 생성하는 계층적 그래프는 (그림 3)과 같은 구조로 표현된다. 이 그래프는 질문 및 문서를 이용해 생성되며 질문 계층, 문단 계층, 문장 계층과 개체 계층을 가진다. 노드 간의 연결 관계는 두 노드의 계층에 따라 다르게 표현된다. 이 그래프의 노드는 Etype(entity type) 노드와 Rtype(relation type) 노드로 구분된다.

Etype 노드는 질문 및 문서에 표시된 자연어를 이용하여 생성되는 노드로서 각 계층에 따라 질문 노드, 문단 노드, 문장 노드, 개체 노드로 구성된다. Rtype 노드는 Etype 노드를 이용하여 생성되는 노드로서 Etype 노드들 간의 관계를 표현한다.



(그림 3) 계층적 그래프 구조

(그림 3)에서 보듯이 Rtype 노드는 연결된 이웃 노드들의 계층에 따라 (1) 질문-문단 노드, (2) 문단-문단 노드, (3) 문단-문장 노드, (4), (7) 문장-문장 노드, (5) 문장-개체 노드, (6) 개체-개체 노드로 구성된다.

이 그래프는 다음과 같은 규칙에 따라 Etype 노드들이 서로 연결되고 각 연결은 Rtype 노드로 표현된다.

- (1) 질문 노드와 모든 문단 노드들을 연결한다.
- (2) 모든 문단 노드는 서로 완전히 연결된다.
- (3) 해당 문단 내의 문장들이면 문단 노드와 문장 노드들을 연결한다.
- (4) 문장 노드들은 문단에 등장하는 순서에 따라 앞뒤로 연결된다.
- (5) 해당 문장 내의 개체들이면 문장 노드와 개체 노드들을 연결한다.
- (6) 해당 문장 또는 다른 문장에 등장하는 개체들이 서로 같은 개체이면 서로 같은 개체를 가리키는 개체 노드들끼리 연결한다.
- (7) 문장 내 하이퍼링크가 존재하는 경우 해당 문장 노드와 하이퍼링크에 연결된 문서 내 모든 문장 노드들을 연결한다.

Etype 노드 중 질문 노드, 문단 노드, 문장 노드의 특징은 (식 2)와 같이 MeanPooling으로 자연어의 특징을 추출

한 값을 초기값으로 사용한다.

$$h_q = \text{MeanPooling}(Q) \quad (\text{식 } 2)$$

$$h_v = \text{MeanPooling}(C[v_{\text{start}} : v_{\text{end}}])$$

여기서 C 는 질문 Q 와 관련 있는 문단들 P 를 서로 앞뒤로 이은 하나의 문맥으로 만들어진 것을 의미한다. v 는 $v \in INP \cup S \cup E$ 이다. 이에 따라, p_{start} , s_{start} , e_{start} 는 C 에서의 각 문단, 문장, 개체의 시작 위치를 나타내며, p_{end} , s_{end} , e_{end} 는 각 문단, 문장, 개체의 끝 위치를 나타낸다.

Rtype 노드의 특징은 (식 3)처럼 Rtype 노드와 이웃한 두 Etype 노드들을 요소별 곱(element-wise product)으로 계산한 값을 초기값으로 사용한다.

$$h_{ab} = h_a \odot h_b \quad (\text{식 } 3)$$

이때 a , b 는 $(a,b) \in \{(q,p), (p,p), (p,s), (s,s), (s,e), (e,e)\}$ 을 만족한다.

이러한 계층적 그래프는 개체 노드 또는 문장 노드만을 이용한 그래프보다 더 다양한 수준의 문맥 정보를 활용할 수 있다. 또한, 두 Etype 노드 사이의 관계를 표현하는 Rtype 노드도 생성함으로써 서로 다른 계층의 노드들 사이에 존재할 수 있는 문맥 수준의 차이 문제를 완화할 수 있다.

2.4 그래프 추론

질문 응답에 대한 다중 흡 추론은 앞서 생성된 계층적 그래프를 효과적으로 추론에 이용할 수 있도록 그래프 신경망을 이용한다. 그래프 신경망을 통해 노드를 생성함으로써 전체적인 글의 문맥 이해뿐만 아니라, 그래프의 계층별 추론 또한 이루어진다. 본 논문에서는 그래프 신경망으로 GAT(Graph Attention Network)를 이용한다. 이 그래프 신경망은 노드 간의 연관성에 따라 각 노드의 맥락 정보가 이웃 노드에 반복적으로 전파된다. GAT는 (식 4)와 같이 나타낼 수 있다.

$$h_{i,j}^{(t)} = \text{LeakyReLU}(\text{concat}(h_i^{(t)}, h_j^{(t)}) W_i^{(t)}) \quad (\text{식 } 4)$$

$$\alpha_{i,j}^{(t)} = \text{softmax}(h_{i,j}^{(t)})$$

$$h_i^{(t+1)} = \text{GAT}(h_i^{(t)}, h_j^{(t)}) = \sum \alpha_{i,j}^{(t)} h_j^{(t)}$$

최종적으로 Etype 노드에서 응답 추론을 하므로 GAT를 이용한 그래프 추론은 Rtype 노드에서 먼저 이뤄진 다음에 Etype 노드에서 이뤄진다. Rtype 노드의 특징은 (식 5)와 같다. 예를 들어, (그림 4)에서 Rtype 노드인 $p_2 s_5$ 의 특징을 생신한다면 Etype 이웃 노드인 p_2 노드와 s_5 노드의 맥락 정보를 이용하게 된다.

$$h_r^{(t+1)} = \text{GAT}(h_r^{(t)}, h_v^{(t)}) \quad (\text{식 } 5)$$

이때, h_r 는 Rtype 노드 r 의 특징, h_v 는 Rtype 노드 r 과 이웃인 Etype 노드 v 의 특징을 뜻한다.

Etype 노드의 특징 생신은 (식 6)과 같다. Rtype 노드뿐만 아니라 Etype 노드의 맥락 정보 또한 활용하기 위해 잔차 연결(skip connection)을 이용한다. 예를 들어, (그림 4)에서 Etype 노드인 s_5 의 특징을 생신한다면 Rtype 이웃 노드인 $p_2 s_5$, $s_3 s_5$, $s_4 s_5$, $s_5 e_7$, $s_5 e_8$ 의 맥락 정보뿐만 아니라 잔차 연결에 의해 Etype 이웃 노드인 p_2 , s_3 , s_4 , e_7 , e_8 의 맥락 정보도 활용하게 된다.

$$h_v^{(t+1)} = \text{GAT}(h_v^{(t)}, h_r^{(t)}) + \text{GAT}(h_v^{(t)}, h_v^{(t)}) \quad (\text{식 } 6)$$

여기서 h_v 는 Etype 노드 v 의 잔차 연결에 의한 이웃 노드인 Etype 노드 v' 의 특징을 뜻한다.

이런 식으로 Etype 노드와 Rtype 노드 사이에 맥락 정보가 서로 전파되는 그래프 추론은 각 계층의 노드가 갖는 문맥 정보와 서로 다른 계층 사이의 맥락 정보 및 같은 계층의 노드 사이의 맥락 정보를 바탕으로 다중 흡 추론을 수행한다.

GAT를 통해 질문과 연관성이 높은 개체로의 정보 전파가 명확해진다. 또한, Rtype 노드와 Etype 노드를 순차

적으로 생성하여 서로 이웃한 Etype 노드 간의 맥락 정보를 효과적으로 공유한다.

2.5 다중 작업 예측

본 논문에서는 뒷받침 문장 예측(supporting sentence prediction), 응답 범위 예측(answer span prediction)과 응답 유형 예측(answer type prediction)으로 구성된 다중 작업을 수행한다. 각 작업에 필요한 문맥 정보는 문맥 수준에 차이가 있어 계층적 그래프의 계층별 맥락 정보가 다중 작업에 각각 이용된다. 질문 정보는 질문 응답의 모든 작업에 있어서 중요하며, 뒷받침 문장 예측에서는 문장 정보, 응답 범위 예측에서는 개체 정보, 응답 유형 예측에서는 질문과 개체 정보가 중요하다. 따라서 다중 작업에는 질문 노드가 공통으로 사용되며, 작업별로 문장 노드, 개체 노드, 개체 노드가 같이 이용된다.

- 뒷받침 문장 예측은 응답에 대한 뒷받침 문장이 되는 문장들을 예측한다. (식 8)과 같이 문맥 C , 질문 Q , 문장 노드들의 집합 S 에 대해 LSTM을 적용한다.

$$I_{\text{support}} = \text{LSTM}_{\text{support}}(\text{concat}(C, Q, S)) \quad (\text{식 } 8)$$

- 응답 범위 예측은 질문에 대한 응답의 범위를 예측한다. (식 9)과 같이 문맥 C , 질문 Q , 개체 노드들의 집합 E 와 뒷받침 문장을 예측한 정보에 대해 LSTM을 적용한다.

$$I_{\text{start}} = \text{LSTM}_{\text{start}}(\text{concat}(C, Q, E, I_{\text{support}})) \quad (\text{식 } 9)$$

$$I_{\text{end}} = \text{LSTM}_{\text{end}}(\text{concat}(C, Q, E, I_{\text{start}}))$$

여기서 I_{start} 는 응답 범위의 시작 위치를, I_{end} 는 응답 범위의 끝 위치를 나타낸다.

- 응답 유형 예측은 응답의 유형이 되는 개체, 예/아니오 중 한 가지를 예측한다. (식 10)와 같이 문맥 C , 질문 Q , 개체 노드들의 집합 E 에 대해 LSTM을 적용한다.

$$I_{\text{type}} = \text{LSTM}_{\text{type}}(\text{concat}(C, Q, E)) \quad (\text{식 } 10)$$

응답 유형 예측에서 예/아니오를 예측할 경우 응답 범위 예측에서 출력된 결과 대신, 예/아니오를 질문에 대한 응답으로 예측한다.

이처럼 모든 작업에 같은 문맥 정보만을 제공하는 것이 아니라, 각각의 작업에 따라 필요한 맥락 정보를 갖는 Etype 노드를 적재적소에 함께 사용함으로써 효과적이고 효율적인 다중 작업을 수행할 수 있도록 한다.

3. 구현 및 실험

3.1 데이터 집합과 모델 학습

본 논문의 제안 모델은 운영체제인 Ubuntu 16.04 LTS에서 Python 딥러닝 라이브러리인 PyTorch를 이용하여 구현하였다. 모델의 학습 및 평가를 위한 질문 응답 데이터 집합으로는 HotpotQA[5]를 사용하였다. HotpotQA는 Wikipedia를 이용해 생성한 오픈 도메인 질문 응답 데이터 집합이다. 이 데이터 집합은 약 11만 개의 자연어 질문과 답변들 중 훈련용(training set)은 90,564개, Distractor 환경의 검증용(validation set)은 7,405개, Fullwiki 환경의 검증용은 7,405개, 나머지는 테스트용(test set)으로 구성되어 있다. Distractor 환경은 질문마다 정답과 뒷받침 문장들을 포함한 2개의 문단들과 그렇지 않은 8개의 문단들을 제공한다. Fullwiki 환경은 Distractor 환경과 다르게, 질문과 관련 있는 문단들을 제공하지 않는다. 훈련용 데이터 집합의 약 80%는 다중 흡 추론을 요구하는 복잡 질문들(complex questions)이며, 약 20%는 단순 질문들(simple questions)이다. 검증용 및 테스트용 데이터 집합은 모두 복잡 질문들로 구성되어 있다. 모델을 학습하기 위해 제안 모델의 레이어 수(number of layers)는 3, 반복 학습주기(epoch)는 30, 배치 크기(batch size)는 20, 학습률(learning rate)은 0.0002로 설정하였다. 실험은 64GB의 메인 메모리와 Geforce RTX 2080 Ti 2개를 탑재한 컴퓨터 환경에서 수행되었다.

3.2 성능 평가 실험

본 논문에서는 제안 모델에서 채택하고 있는 계층적 그래프와 그래프 신경망 추론의 궁정적 효과를 입증하기 위한 실험을 Distractor 환경의 검증용 데이터 집합을 이용하여 수행하였다. 이 실험에서는 제안 모델의 계층적 그래프(Ours)를 개체 그래프(E), 문장 그래프(S) 등과 같은 비-계층적 그래프(NH graph)뿐만 아니라, 문장과 개체들로 구성된 계층적 그래프(SE), 문단과 문장들로 구성된 계층적 그래프(PS) 등과 같은 다른 유형의 계층적 그래프(H graph)들과 성능 비교를 수행하였다. 실험에 사용된 성능 평가 척도는 EM(Exact Matching)과 F1 score이다.

<표 1> 그래프 모델 간의 성능 비교

Model		Answer		Supporting Sentence		Joint	
		EM	F1	EM	F1	EM	F1
NH	E	30.76	40.42	4.4	42.79	2.6	21.84
	S	31.19	40.36	3.73	41.9	1.94	21.14
H	SE	31.22	40.67	4.56	43.37	2.44	21.75
	PS	30.30	39.76	3.91	42.38	1.94	20.99
	Ours	32.19	41.67	6.68	44.71	3.74	23.37

<표 1>은 이 실험의 결과를 나타낸다. 실험 결과를 살펴보면, 모든 작업에서 문단 노드, 문장 노드와 개체 노드로 구성된 Ours가 가장 높은 성능을 보였다. 계층적 그래프끼리 비교했을 때는, 문장 노드와 개체 노드로 구성된 SE 그래프를 이용한 경우가 Ours 다음으로 높은 성능을 보였다. 비-계층적 그래프끼리 비교했을 때는 개체 노드로 구성된 E 그래프를 이용한 경우가 문장 노드로 구성된 S 그래프를 이용한 경우보다 대체로 높은 성능을 보였다. 이러한 결과로 볼 때, 응답의 대상이 되는 개체 정보를 가진 개체 노드가 추론에 영향을 끼치는 것을 짐작할 수 있다. 또한, 단일 문맥 정보보다 다양한 수준의 문맥 정보를 함께 활용하는 것이 답변 추론에 더 효과적임을 알 수 있다.

다음 실험은 몇 가지 HotpotQA 사례들을 이용해, 제안 모델의 계층적 그래프와 그래프 신경망 추론의 우수성을 분석하기 위한 정성적 평가를 수행하였다. 이 실험에서는 제안 모델의 계층적 그래프(Ours)를 개체 그래프(E), 문장과 개체들로 구성된 계층적 그래프(SE)들과 비교하였다. 표에서 보듯이, 제안 모델의 계층적 그래프(Ours)가 질문에 대해 뒷받침 문장(supporting sentence)들과 응답(answer)을 정확히 예측하였다. 반면, SE 그래프를 이용한 모델은 뒷받침 문장을 정확히 예측하였으나, 응답은 잘 예측하지 못하였다. E 그래프를 이용한 모델은 응답을 잘 예측하였지만, 뒷받침 문장들은 전부 예측하지는 못하였다. 이 둘 모델들은 다중 작업 모두에 필요한 문맥 정보를 충분히 확보하지 못한 결과로 판단된다. 따라서 제안 모델의 계층적 그래프가 뒷받침 문장 예측과 응답 예측을 요구하는 다중 작업에 더 효과적인 것을 확인할 수 있다.

4. 결론

본 논문에서는 다중 흡 추론 및 다중 작업을 요구하는 복잡 질문에 대해 효과적인 추론을 수행하는 새로운 심층 신경망 모델을 제안하였다. 제안 모델은 문서로부터 생성되는 Etype 노드와 Rtype 노드로 구성된 계층적 그래프를 생성하고, 그래프 신경망을 이용해 답변에 필요한 다양한 수준의 맥락 정보들을 추출하였다. 그리고 HotpotQA 벤치마크 데이터 집합을 이용한 비교 실험을 통해, 제안 모델의 궁정적 효과를 확인할 수 있었다.

<표 2> 그래프 모델에 따른 질문 응답의 예

[Question] Who passed away first Max Ophüls and Shirley Clarke ?			
[Paragraphs]			
Paragraph 1 : [Shirley Clarke]			
(1) Shirley Clarke (October 2, 1919 - September 23, 1997) was an American experimental and independent filmmaker. (2) She was also a director and editor to many famous works.			
Paragraph 2 : [Max Ophüls]			
(3) Maximillian Openheimer (6 May 1902 - 26 March 1957), known as Max Ophüls (), was a German-born film director who worked in Germany (1931 - 1933), France (1933 - 1940 and 1950 - 1957), and the United States (1947 - 1950). (4) He made nearly 30 films, the latter ones being especially notable: "La Ronde" (1950), "Le Plaisir" (1952), "The Earrings of Madame de..." (1953) and "Lola Montès" (1955).			
Paragraph 3 : [The Exile (1947 film)]			
(5) The Exile (1947) is a adventure romantic film directed by Max Ophüls, and produced, written by, and starring Douglas Fairbanks, Jr. (6) Rita Corday (billed as "Paule Croset") played the romantic interest. (7) According to Robert Osborne, the primary host of Turner Classic Movies, María Montez had a stipulation in her contract that she had to have top billing in any film in which she appeared, so her name comes first in the opening credits, despite her secondary role. (8) The movie is based on the novel "His Majesty, the King: A Romantic Love Chase of the Seventeenth Century" by Cosmo Hamilton.			
Model			
Answer			
Supporting Sentence			
Answer Type			
Ground Truth	Max Ophüls	["Shirley Clarke", 1], ["Max Ophüls", 3]	entity
E	Max Ophüls	['Shirley Clarke', 1]	entity
SE	Shirley Clarke	['Shirley Clarke', 1], ['Max Ophüls', 3]	entity
Ours	Max Ophüls	['Shirley Clarke', 1], ['Max Ophüls', 3]	entity

감사의 글

“본 연구는 과학기술정보통신부 및 정보통신기술진흥센터의 대학ICT연구센터육성지원사업의 연구결과로 수행되었음” (IITP-2017-0-01642)

참고 문헌

- [1] Y. Xiao, Y. Qu, and L. Qiu, et al., "Dynamically Fused Graph Network for Multi-hop Reasoning," *Proc. of ACL-19*, 2019.
- [2] A. Asai, K. Hashimoto, and H. Hajishirzi, et al., "Learning to Retrieve Reasoning Paths over Wikipedia Graph for Question Answering," *Proc. of ICLR-20*, 2020.
- [3] M. Zhang, F. Li, and Y. Wang, et al, "Coarse and Fine Granularity Graph Reasoning for Interpretable Multi-Hop Question Answering," *IEEE Access*, vol 8, 56755-56765, 2020.
- [4] Y. Fang, S. Sun, and Z. Gan, et al., "Hierarchical Graph Network for Multi-hop Question Answering," *Proc. of EMNLP-20*, 2020.
- [5] Z. Yang, P. Qi, and S. Zhang, et al., "HOTPOTQA: A Dataset for Diverse, Explainable Multi-hop Question Answering," *Proc. of EMNLP-18*, 2018.
- [6] K. Nishida, K. Nishida, and M. Nagata, et al., "Answering while Summarizing: Multi-task Learning for Multi-hop QA with Evidence Extraction," *Proc. of ACL-19*, 2019.