KIPS Trans. Softw. and Data Eng.
Vol.10, No.8 pp.319~330
ISSN: 2287-5905 (Print), ISSN: 2734-0503 (Online)

https://doi.org/10.3745/KTSDE.2021.10.8.319

Graph Reasoning and Context Fusion for Multi-Task, Multi-Hop Question Answering

Sangui Lee† · Incheol Kim††

ABSTRACT

Recently, in the field of open domain natural language question answering, multi-task, multi-hop question answering has been studied extensively. In this paper, we propose a novel deep neural network model using hierarchical graphs to answer effectively such multi-task, multi-hop questions. The proposed model extracts different levels of contextual information from multiple paragraphs using hierarchical graphs and graph neural networks, and then utilize them to predict answer type, supporting sentences and answer spans simultaneously. Conducting experiments with the HotpotQA benchmark dataset, we show high performance and positive effects of the proposed model.

Keywords : Open Domain Question Answering, Multi-hop Reasoning, Multi-task Question, Hierarchical Graph, Graph Neural Network

다중 작업, 다중 홉 질문 응답을 위한 그래프 추론 및 맥락 융합

이 상 의† · 김 인 철††

요 약

최근 오픈 도메인 자연어 질문 응답 분야에서는 다중 작업, 다중 홉 질문 응답에 관한 연구들이 활발히 진행되어 오고 있다. 본 논문에서는 이러한 다중 작업, 다중 홉 질문들에 효과적으로 응답하기 위해, 계층적 그래프 기반의 새로운 심층 신경망 모델을 제안한다. 제안 모델에서는 계층적 그래프와 그래프 신경망을 이용해 여러 문단들로부터 서로 다른 수준의 맥락 정보를 얻어낸 후, 이들을 활용하여 답변 유형, 뒷받침 문장들과 답변 영역 등을 동시에 예측해낸다. 본 논문에서는 오픈 도메인 자연어 질문 응답 데이터 집합인 HotpotQA를 이용한 실험들을 통해, 제안 모델의 높은 성능과 긍정적 효과를 입증한다.

키워드 : 오픈 도메인 질문 응답, 다중 홉 추론, 다중 작업 질문, 계층적 그래프, 그래프 신경망

1. 서 론

최근 오픈 도메인 자연어 질문 응답(open domain question answering) 분야에서는 질문에 관한 답변을 얻기 위해서는 관련 있는 여러 문서나 문단, 문장들을 거치는 추론 과정을 요구하는 다중 홉 질문 응답(multi-hop QA)과 답변 외에 뒷받침 문장들(supporting sentences)과 답변 유형(answer type)도 함께 구해야 하는 다중 작업 질문 응답(multi-task QA)에 관한 관심이 높아지고 있다[1]. Fig. 1은 이러한 오픈 도메인의 다중 작업, 다중 홉 질문 응답의 한 예를 보여준다. 이 질문의 답변을 얻기 위해서는 (1), (6)과 같은 뒷받침 문장들을 거치는 추론 과정이 필요하며, 최종적으로 “Norwood, Massachusetts”라는 답변 외에도, 답변 유형과 모든 뒷받침 문장들도 제시하여야 한다.

다중 작업, 다중 홉 질문 응답 시스템을 설계하기 위해서는 몇 가지 중요한 도전적인 문제들을 해결해야 한다. 첫째는 대용량의 문서 집합(corpus)에서 주어진 질문에 관련 있는 문단들(paragraphs)만을 효과적으로 검색해내야 하는 문단 선택(paragraph selection) 문제이다. 둘째는 검색된 문단들을 기초로 답변 예측에 필요한 다양한 맥락 정보를 임베딩해내야 하는 맥락 임베딩(context embedding) 문제이다. 마지막은 맥락 정보들을 어떤 방식으로 다중 작업 질문들에 효과적으로 활용하는가 하는 답변 예측(answer prediction) 문제이다.

※ 이 연구는 2021년도 산업통상자원부 및 산업기술평가관리원(KEIT) 연구비 지원에 의한 연구임('10077538').
† 준 회 원 : 경기대학교 컴퓨터과학과 석사과정
†† 종신회원 : 경기대학교 컴퓨터과학과 교수
Manuscript Received : April 12, 2021
Accepted : May 6, 2021
* Corresponding Author : Incheol Kim(kic@kyonggi.ac.kr)

※ This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (http://creativecommons.org/ licenses/by-nc/3.0/) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

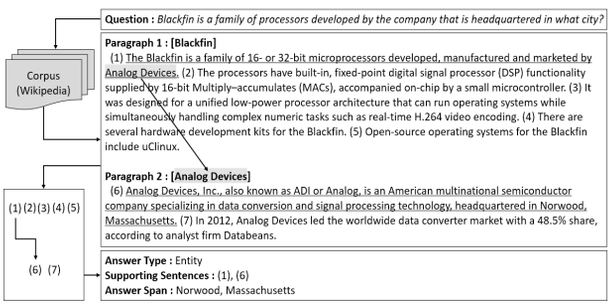

Fig. 1. Example of Multi-task, Multi-hop Question Answering

문단 선택 문제를 해결하기 위해, 기존의 선행 연구들[2, 3]에서는 대부분 사전 학습된 BERT[4] 모델과 문단 분류기 계층(paragraph classification layer)으로 구성된 문단 선택 서브 네트워크를 이용하였다. 이 문단 선택 서브네트워크는 질문과 한 문단을 입력받아 BERT로 임베딩한 후, 해당 문단이 질문에 얼마나 연관된 것인지를 나타내는 관련성 점수(relevance score)를 출력으로 내놓는다. 이 서브 네트워크를 학습하기 위해서는 적어도 하나의 뒷받침 문장을 포함하고 있는 문단에는 레이블(label)로 1을, 그렇지 않은 문단에는 0을 부여한 학습 데이터를 이용한다. 하지만 이러한 문단 선택 방법은 문단들을 연결하는 하이퍼링크(hyperlink)들이나 문단 제목(title) 등의 중요한 부가 정보들을 충분히 활용하지 못하는 한계가 있다.

맥락 임베딩 문제를 해결하기 위해서 대부분의 기존 연구들[2, 3, 5, 6]에서는 검색된 문단들로부터 개체 혹은 문장들을 추출하여 맥락 그래프를 구성하고, 그래프 추론을 통해 답변 예측에 활용할 다양한 맥락 정보를 구해내려고 하였다. [2]의 연구에서는 개체 그래프(entity graph)를, [3]의 연구에서는 개체 그래프(entity graph)와 문장 그래프(sentence graph)를, [5]의 연구에서는 문단 그래프(paragraph graph)를, [6]의 연구에서는 문단, 문장 및 개체들로 구성된 계층적 그래프(hierarchical graph)를 각각 이용하였다. 하지만 많은 경우, 문단, 문장 혹은 개체 등 문단 구성 요소들 중 하나만으로 그래프를 구성하거나, 다른 계층 혹은 다른 그래프와의 맥락 융합(context fusion)에 한계가 있었다. 또한, 기존 연구들[2, 3, 5, 6]에서는 답변 예측 문제를 해결하기 위해 대부분 맥락 그래프의 특정 계층만을 이용하거나, 답변 유형, 뒷받침 문장과 답변 영역을 구별하지 않고 같은 맥락 정보를 이용해 답을 예측함으로써, 다중 작업 질문 응답의 특성을 제대로 반영하지 못하였다.

본 논문에서는 이러한 기존 모델들의 한계점을 보완하는 새로운 다중 작업, 다중 홉 질문 응답 모델을 제안한다. 제안 모델에서는 문서 집합에서 질문과 연관성이 높은 문단들을 가려내기 위해, 문단 제목 매칭(title matching), 내용 매칭(content matching), 하이퍼링크 연결 문단 선택(hyperlinked paragraph selection) 등 다양한 의미적 검색(semantic retrieval)을 함께 수행한다. 또한, 제안 모델에서는 서로 다른 수준의 다양한 맥락 정보를 얻기 위해 질문, 문단, 문장과 개체 등 4가지 유형의 노드들로 구성된 계층적 그래프(hierarchical graph)를 생성한다. 그리고 그래프 신경망(graph neural network)을 이용해 이 계층적 그래프의 각 계층 내의 노드들과 서로 다른 계층 간의 노드들끼리 맥락 정보를 교환한다. 또 그래프의 맥락 정보와 텍스트 맥락 정보를 반복적으로 융합한다. 이렇게 얻어낸 풍부한 맥락 정보들을 다중 작업 답변 예측에 다양한 방법으로 활용한다.

본 논문에서는 제안 모델의 성능 평가를 위해, 오픈 도메인 자연어 질문 응답 데이터 집합인 HotpotQA[1]를 이용해 다양한 실험들을 수행하고, 그 결과를 소개한다. 본 논문의 2장에서는 기존의 관련 연구들을 간단히 살펴보고, 3장에서는 제안 모델의 설계에 관해 상세히 설명한다. 4장에서는 제안 모델의 구현과 성능 실험들을 소개하고, 마지막으로 5장에서는 결론과 향후 연구를 정리한다.

2. 관련 연구

2.1 오픈 도메인 질문 응답

최근 자연어 질문 응답 문제는 다양한 정보를 포함하는 문서들을 이해하고 다중 홉 추론을 요구하는 복잡 질문에 응답하는 문제로서 활발히 연구되고 있다. 실세계에서 사람들이 하는 질문처럼, 답변을 도출하기 위해 둘 이상의 사실 정보를 요구하는 복잡한 질문을 해결하기 위한 데이터 집합과 연구들이 제시됐다. 대부분 Wikipedia 같은 문서를 기반으로 질문 응답 데이터 집합이 만들어졌으며, Wikihop[7], HotpotQA 등이 제시되었다.

문서 기반 질문 응답 문제에서는 먼저 문서 집합으로부터 질문과 관련된 문서 또는 문단들을 선택해야 한다. 기존 연구들[2, 3]에서는 BERT 기반의 분류기를 이용하여 일정 점수 이상 받은 문단들을 질문과 연관성 높은 문단으로 선택하였다. [6]의 연구에서는 문서의 제목이 질문에 나타나는지 확인하는 제목 매칭과 RoBERTa[8] 기반의 분류기를 이용하여 상위 L개의 문단들을 선택하도록 하였다. 질문에 대한 답변은 이렇게 선택한 문단들로부터 찾게 된다.

선택한 문단들로 신경망을 학습하기 위해서는 자연어를 임베딩해서 벡터값으로 변환해야 한다. 과거 연구들은 Word2Vec[9], GloVe[10] 등을 주로 이용해 임베딩하였으나, 단어의 문맥적인 의미가 잘 반영되지 않는 문제가 있었다. 최근에는 이러한 문제가 개선된 Transformer[11] 계열의 사전 학습 모델인 BERT, RoBERTa 등이 제시되었다. 이런 모델들은 대규모 문서 집합을 사전 학습하기 때문에 문서 기반 질문 응답 연구들에서 활발히 적용되고 있다.

문서로부터 답변을 찾기 위한 다중 홉 추론은 대개 순환신경망 또는 그래프 신경망을 사용하여 이뤄졌다. 순환신경망

을 사용한 연구들[1,12]은 GRU나 LSTM을 여러 층 쌓아 답변을 추론할 수 있도록 하였다. 그래프 신경망을 이용하는 연구들[2, 3, 6]은 문서로부터 문장이나 개체 등을 추출하여 그래프를 생성하여 그 위에서 다중 홉 추론을 수행하였다. 최근에 제시된 HotpotQA 데이터 집합을 이용하는 연구들은 질문에 대한 답변뿐만 아니라 뒷받침 문장도 찾는 다중 작업을 수행한다. 뒷받침 문장을 중점적으로 찾는 연구인 [5, 12]는 순환신경망으로 뒷받침 문장을 추출한다. [5]의 연구에서는 질문과 연관된 문단들을 추출한 후 순환신경망을 이용해 문장들로 이뤄진 추론 경로를 여러 개 탐색하도록 하였다. 그중 점수가 가장 높은 추론 경로를 선택하여 뒷받침 문장들을 찾도록 하였다. QFE[12]는 뒷받침 문장을 찾기 위해 순환신경망을 이용하여 질문 정보와 문장들의 정보를 번갈아 참조하면서 뒷받침 문장일 것으로 예상하는 문장들을 추출하였다.

2.2 그래프 신경망

최근 그래프를 다루는 신경망인 GCN[13], GAT[14] 등이 제안되어 많은 연구에서 이러한 모델들을 적용하였다. 이에 따라 질문 응답 문제에서도 그러한 그래프 신경망을 활용하는 연구들이 활발히 진행되어왔다. 그래프 신경망을 적용한 연구들은 문서로부터 문단, 문장 또는 개체를 추출하여 그래프를 생성하고 이에 대해 그래프 신경망을 적용하여 답을 추론하도록 하였다.

[15]의 연구에서는 질문에 대한 답변 후보들, 문서 내 문장들과 문장 내에 언급된 개체들을 서로 연결한 HDE(Heterogeneous Document-Entity) 그래프를 생성하고, 그래프 신경망 GCN을 적용하였다. DFGN[2]은 문서로부터 추출한 개체들로 개체 그래프를 생성하고, 그래프 신경망 GAT를 적용하였다. 특히, 개체 그래프에 소프트마스크(softmask) 연산을 적용하여 질문과 연관 있는 개체들의 정보에 더 중점을 두었다. CFGGN[3]은 문서로부터 추출한 개체와 문장들을 이용해 각각 개체 그래프와 문장 그래프를 생성하여, 그래프 신경망 GAT를 적용하였다. HGN[6]은 문서를 문단, 문장, 개체 등 여러 계층으로 나뉘어 계층적 그래프를 생성하고 그래프 신경망 GAT를 적용하였다. SAE[16]는 뒷받침 문장을 찾기 위해 문서로부터 문장 그래프를 생성하고 [17]의 모델처럼 게이팅(gating) 기법을 쓰면서 다중 관계(multi-relational) GCN을 적용하였다.

그러나 기존 연구들에서는 문서로부터 그래프를 생성하였음에도 문서의 텍스트 맥락(textual context) 정보와 문서로부터 생성한 그래프의 맥락 정보(graph context) 간에 정보 교환이 이뤄지지 않거나 문장 또는 개체 수준의 단일 정보에서만 이뤄졌다는 단점이 있다. 본 논문에서는 그러한 문제를 해결하기 위해, 문서 내 다양한 수준의 맥락 정보를 그래프화하고, 그래프의 맥락 정보와 문서의 맥락 정보 간에 정보 교환 또한 전역적으로 이루어질 수 있도록 모델을 설계하였다.

3. 다중 작업 질문 응답

3.1 문제 정의

본 논문에서는 문서에서 추출한 질문과 연관된 문단들로부터 뒷받침 문장들과 답변을 찾는 다중 작업 문제를 다루고 있다. 본 논문에서 자연어 질문은 질문(question)을 구성하는 단어들의 집합 $Q=\{w_1, w_2, ..., w_q\}$로, 문서(document) 집합은 $D=\{d_1, d_2, ..., d_U\}$로 각각 정의한다. 또, 질문 Q와 관련하여 문서 집합 D에서 추출한 문단(paragraph)들의 집합은 $P=\{p_1, p_2, ..., p_l\}$로, 문단 집합 P를 구성하는 문장(sentence)들의 집합은 $S=\{s_1, s_2, ..., s_m\}$로, 문단 집합 P에 등장하는 개체(entity)들의 집합은 $E=\{e_1, e_2, ..., e_n\}$로 정의한다. 또한, 집합 P에 속한 문단들을 모두 연결한 전체 텍스트는 $C=concat(p_1, p_2, ..., p_l)$로 정의한다. 여기서 $concat$은 연결(concatenation)을 의미한다.

본 논문에서 다루는 문서 기반 다중 작업(multi task) 질문 응답은 하나의 질문 Q에 대해, 답변 유형 예측(answer type prediction), 뒷받침 문장 예측(supporting sentence prediction), 답변 영역 예측(answer span prediction) 등의 작업을 동시에 수행하는 것을 의미한다. 또, 다중 홉(multi hop) 질문 응답은 텍스트 C에서 하나 이상의 뒷받침 문장들 $\langle s_{r_1}, s_{r_2}, ..., s_{r_z} \rangle$ 을 연결하는 추론 과정을 통해야 질문 Q에 대한 답변을 구할 수 있는 질문과 답변을 의미한다.

3.2 제안 모델

본 논문에서 제안하는 모델은 문서 집합으로부터 질문과 관련된 문단들을 추출하고, 이를 토대로 계층적 그래프를 생성하여 다중 홉 추론 및 다중 작업 답변 예측을 수행한다. Fig. 2는 제안하는 모델의 전체 구조도를 나타낸다. 입력으로 질문과 문서를 가지며, 출력으로 질문에 대한 답변인, 문서에 언급된 개체 또는 예/아니오와 답변에 대한 뒷받침 문장들을 갖는다. 제안하는 모델은 문단 선택(Paragraph Selection), 맥락 임베딩(Context Embedding), 그래프 추론(Graph Reasoning) 및 답변 예측(Answer Prediction)으로 구성되어 있다. 문단 선택은 질문과 관련된 문단들을 추출하기 위해 [6]처럼 RoBERTa 기반 문단 분류기를 이용한다. 맥락 임베딩은 질문과 추출한 문단들에 대해 임베딩을 수행한다. 우선 질문과 문단들을 사전 학습된 ALBERT[18] 모델로 함께 임베딩한 다음, LSTM(Long Short-Term Memory)으로 인코딩한다. 그래프 추론은 질문과 추출된 문단들을 이용해 계층적 그래프를 생성하고 다중 홉 추론을 수행한다. 계층적 그래프는 질문, 문단, 문장과 개체 등 4가지 종류의 계층으로 구성된다. 다중 홉 추론은 그래프를 이루는 노드들이 반복적으로 이웃 노드들의 맥락 정보를 받고 특징을 갱신함으로써 이뤄진다. 그래프 추론이 반복적으로 이뤄지는 동안 맥락 융합은 그래프 노드의 맥락 정보를 텍스트 맥락 정보에 전파함으로써 행해진다. 답변 예측은 그래프 추론이 끝난 계층적 그래프의 맥락 정보와 텍스트 맥락 정보를 이용하여 최종적으로 뒷받침 문

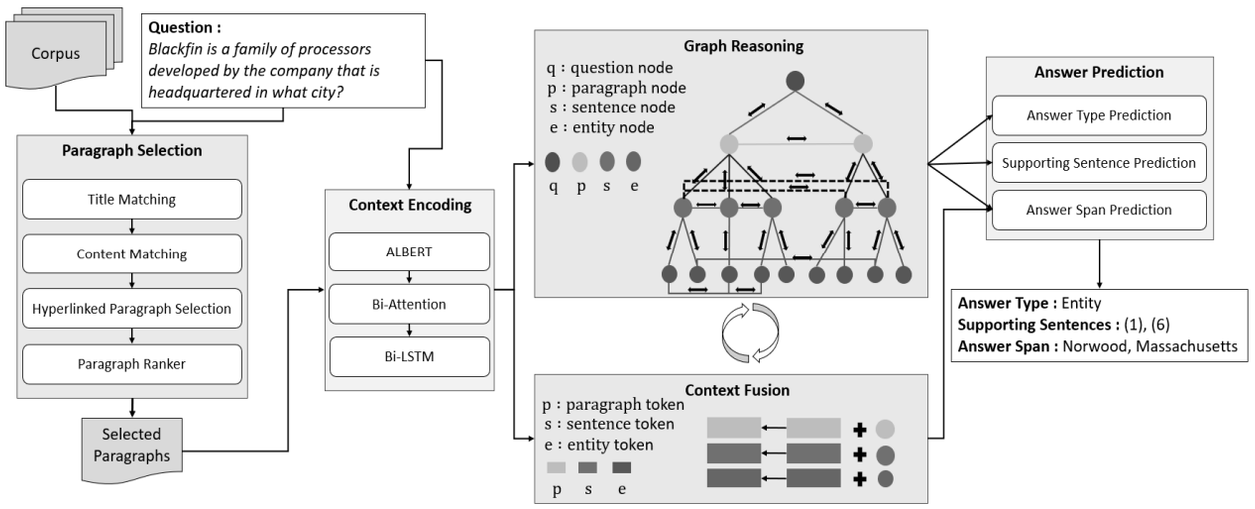

Fig. 2. Architecture of the Proposed Model

장 예측(Supporting Sentence Prediction), 답변 영역 예측(Answer Span Prediction), 답변 유형 예측(Answer Type Prediction)을 수행한다.

3.3 문단 선택 및 맥락 임베딩

대규모 문서 집합(corpus)을 이용한 질문 응답 문제를 해결하기 위하여, 가장 먼저 질문과 관련성이 높은 문단들을 선택한다. 대규모 문서 집합에는 주어진 질문과 관련성이 낮거나 무관한 문단들이 매우 많다. 따라서 실질적으로 질문에 대한 답변 추론에 도움이 되도록, 문서 집합에서 질문과 관련 있는 문단들만 선택하는 과정이 필요하다. 이러한 문단 선택(paragraph selection) 과정은 답변을 예측해내는 문서 범위를 한정시킬 수 있고, 추론 단계에서 불필요한 연산도 줄일 수 있다.

Fig. 3은 문단 선택 과정을 나타내며, 이 과정은 크게 두 개의 세부 과정들로 나뉜다. 첫 번째 과정은 추론의 시작점이 되는 문단을 선택하는 단계로, 질문과 연관성이 높은 문단을 선택한다. 두 번째 과정은 앞선 첫 번째 과정에서 선택한 문단과 하이퍼링크(hyperlink)로 연결된 문서의 문단들을 선택한다. 첫 번째 과정에서는 문서의 제목이 질문에 나타나는 문단을 찾는 제목 매칭(title matching) 단계를 거친다. 제목 매칭을 통해 문단을 찾지 못한 경우, 질문에 등장하는 개체가 문단 내에 존재하면 해당 문단을 선택하는 내용 매칭(content matching) 단계를 수행한다. 제목 매칭 단계와 내용 매칭 단계는 일반적으로 사람이 질문에 관련된 문단을 선택하는 과정을 모방하여 합리적으로 질문과 관련된 문단을 찾는 것으로 볼 수 있다. 두 번째 과정은 첫 번째 과정에서 선택한 문단들과 하이퍼링크로 연결된 문서의 문단들 중에서 문단 순위 결정기(paragraph ranker)를 이용해 K개의 문단들을 선택하는 과정이다. 이 과정은 Equation (1)과 같이 질문과 문단들을 사전 학습된 RoBERTa로 임베딩한 다음, MLP(Mutli

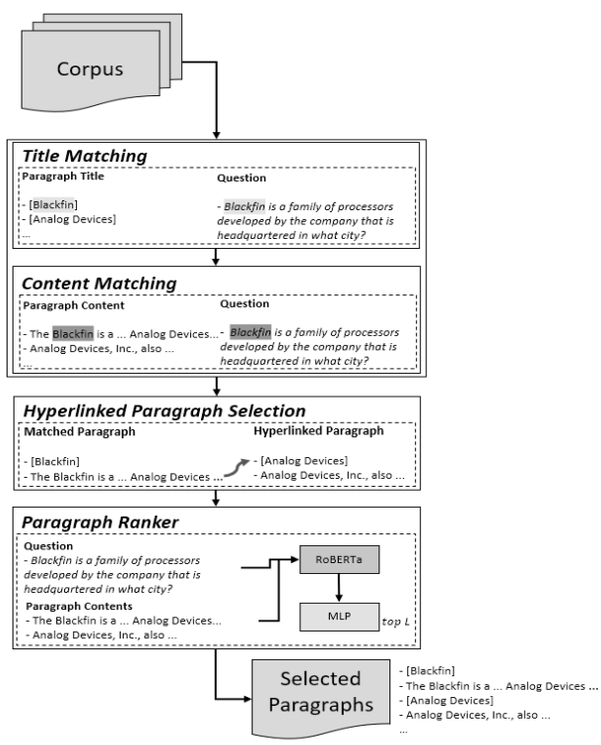

Fig. 3. Paragraph Selection

Layer Perceptron) 계층을 이용해 점수를 계산하여 상위 K개의 문단들을 선택한다.

$$score(P) = MLP(RoBERTa(concat(Q,P))[0]) \quad (1)$$

앞선 두 과정을 통해 문단을 선택하지 못하는 경우, 문단 순위 결정기를 이용해 상위 L개의 문단을 선택한다.

맥락 임베딩(context embedding) 과정에서는 질문과 선

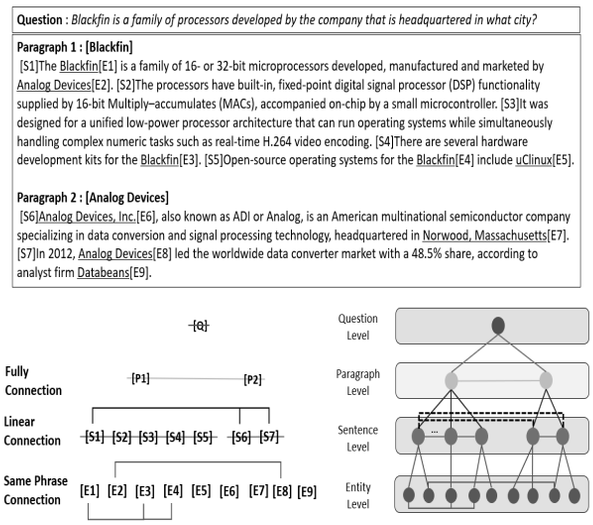

Fig. 4. Hierarchical Context Graph

택된 문단들을 하나의 텍스트 C로 결합한 다음, 사전 학습된 ALBERT 모델로 단어 임베딩을 수행한다. ALBERT에 의해 임베딩된 텍스트 벡터는 질문과 문단 간의 Bi-Attention[19]을 적용한 후, 양방향 순환 신경망인 Bi-LSTM로 인코딩된다.

3.4 그래프 추론 및 맥락 융합

문서를 구성하는 문단(paragraph)들, 각 문단을 구성하는 문장(sentence)들, 각 문장에 포함된 개체(entity)들 간의 계층적 관계는 하나의 계층적 그래프(hierarchical graph)로 표현할 수 있다. 이러한 계층적 그래프는 계층별로 서로 다른 수준의 맥락 정보를 표현할 수 있으므로, 하나의 계층적 그래프는 다중 작업 질문 응답(multi task QA)에 필요한 다양한 맥락 정보를 가질 수 있다. 본 논문에서 제안하는 계층적 그래프 G는 Fig. 4와 같이 구성된다. 이 그래프는 주어진 질문 Q와 선택된 문단들의 집합 P를 토대로 생성되며, 최상위의 질문 계층, 문단 계층, 문장 계층, 그리고 최하위의 개체 계층들로 구성된다. 따라서 각 계층은 각각 질문 노드, 문단 노드, 문장 노드, 그리고 개체 노드들로 구성된다.

Fig. 4와 같이, 이 계층적 그래프 G는 다음과 같은 규칙들에 따라 간선(edge)으로 노드(node)들을 연결한다.

(1) 질문 노드와 모든 문단 노드들을 연결한다.
(2) 모든 문단 노드는 서로 완전히 연결된다.
(3) 문단 내 포함된 문장들에 대해서는, 해당 문단 노드와 문장 노드들을 연결한다.
(4) 문장 노드들은 해당 문장이 문단에 등장하는 순서에 따라 앞뒤로 연결된다.
(5) 문장 내에 포함된 개체들에 대해서는, 해당 문장 노드와 개체 노드들을 연결한다.
(6) 각기 서로 다른 문장에 속하지만 동일한 개체를 가리키는 개체 노드들끼리는 서로 연결한다.
(7) 문장 내 하이퍼링크가 포함된 경우, 해당 문장 노드와 하이퍼링크에 연결된 다른 모든 문장 노드들을 연결한다.

이처럼 계층적 그래프 G를 구성하는 노드들과 간선들로 그래프 구조(graph structure)가 결정되고 나면, 텍스트 맥락(textural context) C를 이용해 각 그래프 노드의 초기 특징값(initial feature)들을 계산한다. 이때, Equation (2)와 같이 C로부터 각각 문단, 문장, 개체에 해당하는 영역들을 동일한 크기의 특징 벡터들로 변환하기 위한 MeanPooling 연산을 수행한다.

$$h_q = MeanPooling(Q)$$
$$h_v = MeanPooling(C[v_{start} \ : \ v_{end}]) \quad (2)$$

Equation 2에서 Q는 질문을, C는 텍스트 맥락을 각각 나타낸다. 그리고 v_{start}는 C에서 각 문단, 문장, 그리고 개체의 시작 위치(start position)를 나타내며, v_{end}는 이들의 끝 위치(end position)를 나타낸다.

다중 작업, 다중 홉 질문 응답을 효과적으로 수행하기 위해서는 답변 예측에 필요한 다양한 맥락 정보를 이끌어내는 추론 과정이 필요하다. 본 논문에서 제안하는 모델에서는 계층적 그래프 G를 이용한 그래프 추론(graph reasoning)과 그래프의 맥락 정보를 텍스트 맥락 C에 융합시키는 맥락 융합(context fusion)을 통해 이러한 맥락 추론 과정이 수행된다.

제안 모델에서는 그래프 추론을 수행하기 위해 그래프 신경망(Graph Neural Network, GNN)을 이용한다. 하나의 그래프 신경망 계층(layer)은 입력 그래프의 각 노드 특징값을 간선으로 연결된 이웃 노드들의 특징값을 반영하여 새롭게 갱신하는 역할을 한다. 따라서 다수의 그래프 신경망 계층들을 거칠 때마다 그래프상의 각 노드의 특징값은 연결된 간선들을 따라 더 먼 이웃 노드들까지 전파되는 효과가 발생한다. 제안 모델에는 계층적 그래프 G에 이러한 그래프 신경망(GNN)을 적용함으로써, 각 계층 내의 노드들 간에 정보 교환(intra-level information exchange)은 물론이고, 서로 다른 계층들에 속한 노드들 간에도 정보 교환(inter-level information exchange)이 이루어진다.

Fig. 5는 제안 모델에 적용되는 그래프 신경망 QA-GNN (Question-Attentional Graph Neural Network)의 동작 방식의 예시이며, 구체적인 그래프 신경망 QA-GNN의 연산식은 Equation (3)과 같다.

$$cs_j^{(t)} = Cosine\,Similarity(h_q^{(t)}, h_j^{(t)})$$
$$\alpha_j^{(t)} = \frac{cs_j^{(t)}}{\Sigma_{k \in \widetilde{N}_i} cs_k^{(t)}}$$
$$h'^{(t)}_i = \Sigma_{j \in \widetilde{N}_i} \alpha_j^{(t)} h_j^{(t)}$$
$$g = sigmoid(h'^{(t)}_i \odot h'^{(t)}_q)$$
$$h_i^{(t+1)} = g * h'^{(t)}_i + (1-g) * h_i^{(t)} \quad (3)$$

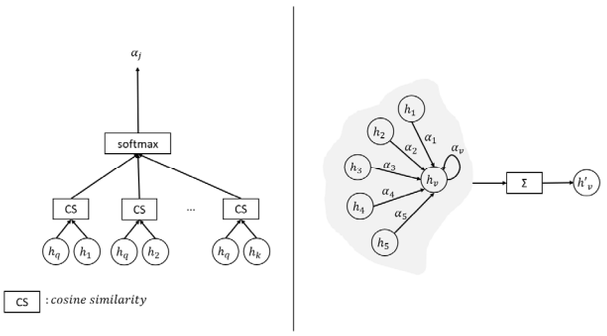

Fig. 5. Question-Attentional Graph Neural Network (QA-GNN)

Equation (3)에서 $h_q^{(t)}$는 질문 노드 q의 특징값을, $h_i^{(t)}$는 계층적 그래프 상의 한 노드 i의 특징값을 각각 나타낸다. 또 $\widetilde{N}_i$는 노드 i의 이웃 노드(neighbor node)들의 집합을, h_j는 한 이웃 노드 j의 특징값을 각각 나타낸다. 이웃 노드 j의 가중치 α_j는 질문 노드 q와 노드 j의 코사인 유사도(cosine similarity)에 따라 결정되며, 이러한 가중치에 따라 이웃 노드들의 특징값들을 결합하여($\sum_{j\in\widetilde{N}_i}\alpha_j^{(t)}h_j^{(t)}$) 각 노드 i의 새로운 특징값 $h_i^{\prime(t)}$를 계산한다. 그리고 기존의 특징값 $h_i^{(t)}$와 이웃 노드들로부터 새로 계산한 특징값 $h_i^{\prime(t)}$를 g의 비율로 혼합하여($g*h_i^{\prime(t)}+(1-g)*h_i^{(t)}$) 노드 i의 특징값 $h_i^{(t+1)}$을 갱신한다. 이때 혼합 비율 g는 노드 i의 새로운 특징값 $h_i^{\prime(t)}$과 질문 노드 q의 특징값 $h_q^{\prime(t)}$을 요소별 곱(element-wise product)한 다음, sigmoid 함수를 적용하여 결정되며, 0~1 사이의 값이 된다. 이와 같이 그래프 신경망 QA-GNN에 따라 계층적 그래프의 각 노드 특징값을 갱신함으로써, 질문 정보에 더 초점을 맞춰서 추론이 수행될 수 있도록 설계하였다.

텍스트 맥락 C의 갱신은 Equation (4)와 같이 그래프 신경망 QA-GNN에 의해 한 차례 갱신된 새로운 그래프 맥락 정보(graph context)를 텍스트 맥락 정보(textual context)에 융합함으로써 이루어진다.

$$C^{(t+1)} = concat(C^{(t)},\ G^{(t+1)}) \tag{4}$$

Equation (4)에서 $G^{(t+1)}$는 그래프 신경망에 의해 이전 그래프 $G^{(t)}$의 각 노드의 특징값들이 한 차례 갱신된 새로운 계층적 그래프를 나타낸다. 이렇게 갱신된 텍스트 맥락 정보 $C^{(t+1)}$는 Equation (2)에 따라 다시 새로운 계층적 그래프 G의 각 노드를 초기화하는 데 사용된다.

이런 방식으로 그래프 맥락 정보 $G^{(t)}$와 텍스트 맥락 정보 $C^{(t)}$사이에 일어나는 상호작용은 두 맥락 정보 간의 정보 괴리를 최소화하며 다중 홉 추론을 수행한다.

3.5 답변 예측

제안 모델에서는 하나의 질문에 대해 답변 유형 예측(answer type prediction), 뒷받침 문장 예측(supporting sentence prediction), 답변 영역 예측(answer span prediction) 등 세 가지 작업을 동시에 수행하는 다중 작업 답변 예측(multi-task answer prediction)을 수행한다. 그리고 이러한 다중 작업 답변 예측에는 그래프 추론과 맥락 융합을 통해 얻은 그래프 맥락 정보(graph context)와 텍스트 맥락 정보(textual context)를 함께 이용한다. 특히 그래프 맥락 정보는 그래프 신경망에 의해 다중 홉 추론이 이뤄진 정보이기 때문에, 세 가지 종류의 답변 예측에 공통적으로 이용된다. 반면에 텍스트 맥락 정보는 답변 영역 예측에 국한되어 이용된다. Fig. 6은 세 가지 답변 예측 각각에 이용되는 그래프 맥락 정보 및 텍스트 맥락 정보를 나타낸다.

먼저, 답변 유형 예측은 답변이 하나의 개체(entity)인지, 아니면 예/아니오(yes/no) 인지를 결정하는 것이다. 이러한 답변 유형은 질문에 따라 결정되기 때문에, Equation (5)와 같이 질문 노드 q의 특징값에 MLP를 적용하여 답변 유형을 결정한다.

$$I_{type} = MLP_1(q) \tag{5}$$

만약 답변 유형 예측 결과가 예/아니오인 경우, 답변 영역 예측 결과 대신, 예/아니오를 해당 질문의 답변으로 출력한다. 뒷받침 문장 예측은 주어진 질문에서 시작해서 답변에 도달하는 경로를 구성하는 모든 뒷받침 문장들을 결정한다. 제

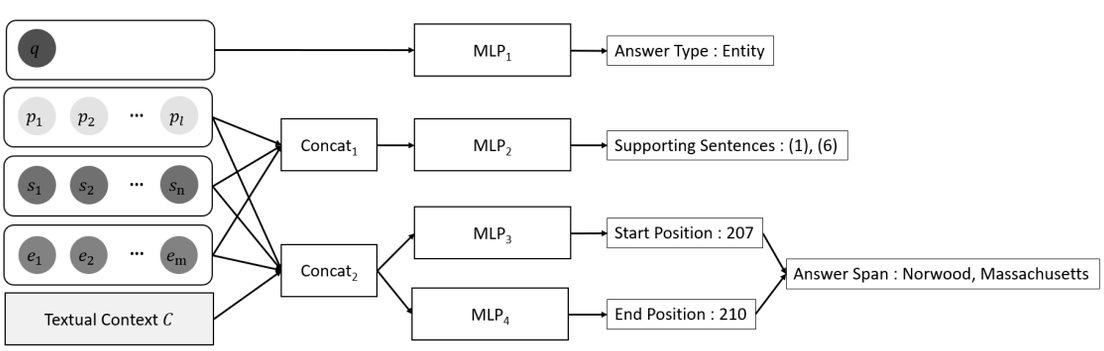

Fig. 6. Multi-task Answer Prediction

안 모델에서는 이러한 뒷받침 문장 예측을 위해 질문 노드 q를 제외한 모든 계층의 그래프 맥락 정보(graph context)를 함께 이용하는 것이 바람직하다고 판단하여, Equation (6)과 같이 계층적 그래프상의 문단 노드들(P), 문장 노드들(S), 개체 노드들(E)의 특징값들을 모두 결합한 후 MLP를 적용한다.

$$I_{support} = MLP_2(concat(P, S, E)) \quad (6)$$

한편, 답변 영역 예측에서는 텍스트 맥락(C)에서 답변이 시작되는 위치(I_{start})와 끝나는 위치(I_{end})를 각각 결정해야 한다. 제안 모델에서는 이러한 답변 영역 예측에는 텍스트 맥락 정보 외에 그래프 맥락 정보도 함께 활용하도록, Equation (7)과 같이 텍스트 맥락 정보 C와 그래프 맥락 정보인 문단 노드들(P), 문장 노드들(S), 개체 노드들(E)의 특징값들을 모두 결합한 후, MLP를 적용한다.

$$I_{start} = MLP_3(concat(C, P, S, E)) \quad (7)$$
$$I_{end} = MLP_4(concat(C, P, S, E))$$

여기서 I_{start}는 답변 영역의 시작 위치를, I_{end}는 답변 영역의 끝 위치를 나타낸다. 이처럼 제안 모델에서는 그래프 맥락 정보와 텍스트 맥락 정보를 세 가지 작업 각각의 특성에 따라 적절히 함께 사용함으로써, 효과적이고 효율적인 다중 작업 답변 예측을 수행하도록 설계하였다.

4. 구현 및 실험

4.1 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 HotpotQA를 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 답변과 뒷받침 문장들을 포함한 2개의 문단들과 그렇지 않은 8개의 문단들을 제공한다. Fullwiki 환경은 Distractor 환경과 다르게, 질문과 관련 있는 문단들을 온전히 제공하지 않는다. 훈련용 데이터 집합의 약 80%는 다중 홉 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 학습 주기(epoch)는 20, 배치 크기(batch size)는 16, 학습률(learning rate)은 0.0007로 설정하였다. 성능 평가 실험들은 64GB의 메인 메모리와 Geforce RTX 2080 Ti 2개를 탑재한 컴퓨터 환경에서 수행되었다.

Table 1. Results using Different Context Graphs

Graph	Supporting Sentence		Answer Prediction		Joint Prediction	
	EM	F1	EM	F1	EM	F1
(a) No graph	54.08	84.54	57.03	70.58	33.86	61.62
(b) QE	54.25	84.81	57.27	70.59	34.33	62.05
(c) QSE	55.19	85.04	57.08	70.73	35.22	62.54
(d) QPSE(Ours)	56.83	85.23	57.60	71.28	36.37	63.06

4.2 성능 평가 실험

본 논문에서는 앞서 설명한 HotpotQA 데이터 집합을 이용하여 제안 모델의 성능 평가 실험들을 수행하였다. 실험에서 사용한 성능 평가 항목들은 뒷받침 문장 예측(Supporting Sentence Prediction), 답변 예측(Answer Prediction), 공동 예측(Joint Prediction) 등이다. 답변 유형 예측(Answer Type Prediction)과 답변 영역 예측(Answer Span Prediction)의 결과는 최종 답변(Answer)을 결정하는데 함께 사용되기 때문에, 실험 결과에는 별도로 표시하지 않는다. 실험에 사용된 성능 평가 척도들은 EM(Exact Matching)과 F1 score이다.

첫 번째 실험은 제안 모델에서 사용하는 계층적 그래프와 그래프 신경망 추론의 효과를 분석하기 위한 실험이다. 이 실험에서는 그래프 추론을 하지 않는 경우(No graph), 질문과 개체 노드들로만 구성된 그래프를 이용한 경우(QE), 질문, 문장, 그리고 개체 노드들로 구성된 그래프를 이용한 경우(QSE), 그리고 제안 모델과 같이 질문, 문단, 문장, 개체 노드들로 구성된 계층적 그래프를 이용한 경우(QPSE)를 서로 비교하였다.

Table 1은 이 실험의 결과를 나타낸다. 실험 결과를 살펴보면, 모든 작업, 모든 척도에서 제안 모델과 같이 질문, 문단, 문장, 개체 등 모든 계층들을 포함하는 계층적 그래프를 이용한 (d) QPSE(Ours)의 경우가 가장 높은 성능을 보였다. 답변 (Answer) 예측에서는 (d)를 제외한 (a), (b), (c) 모두 대체로 비슷한 성능을 보였으나, 뒷받침 문장(Supporting Sentence) 예측의 EM 척도 면에서는 (b)가 (a)보다 0.17%, (c)가 (b)보다 0.94%, (d)가 (c)보다 1.64% 높은 성능을 보였다. 즉, 개체 계층, 문장 계층, 문단 계층 순으로 계층을 쌓아 계층적 그래프를 생성하여 사용했을 때 성능이 점진적으로 증가하였다. 이러한 실험 결과를 토대로, 다양한 수준의 맥락 정보들을 함께 활용하는 것이 성능 개선에 효과적임을 알 수 있다.

두 번째 실험은 제안 모델에서 채택한 그래프 맥락 정보와 텍스트 맥락 정보의 융합 방식의 효과를 분석하기 위한 실험이다. 이 실험에서는 Fig. 7의 (a), (b), (c), (d)에 표현된 서

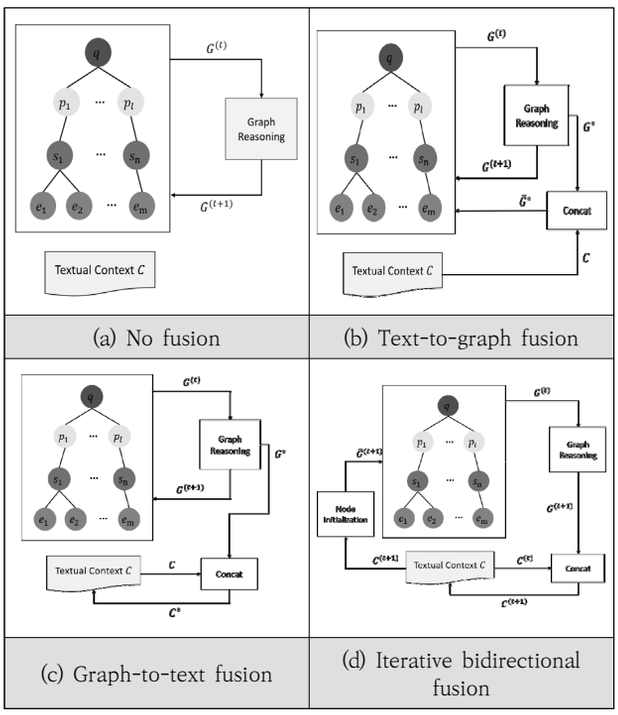

Fig. 7. Different Context Fusion Methods

로 다른 4가지 융합 방식들을 각각 구현하여 비교 실험하였다. (a)는 그래프 맥락 정보와 텍스트 맥락 정보 간의 융합을 수행하지 않는 경우(no fusion)를 나타내며, (b)는 그래프 신경망을 통해 갱신이 완료된 그래프 맥락 정보 G^*에 텍스트 맥락 정보 C를 결합해 최종적인 그래프 맥락 정보 $\widetilde{G}^*$를 구하는 융합(text-to-graph fusion) 방식을 나타낸다. 반면에 (c)는 그래프 신경망을 통해 갱신이 완료된 그래프 맥락 정보 G^*를 텍스트 맥락 정보 C에 결합하여 최종 텍스트 맥락 정보 C^*를 구하는 융합(graph-to-text fusion) 방식을 나타내며, (d)는 제안 모델과 같이, 그래프 신경망 계층을 통해 한차례 새롭게 갱신된 그래프 맥락 정보 $G^{(t+1)}$를 텍스트 맥락 정보 $C^{(t)}$에 결합하여 새로운 텍스트 맥락 정보 $C^{(t+1)}$를 구할 뿐만 아니라, 다음 단계의 그래프 추론을 위해 그래프 맥락 정보 $\widetilde{G}^{(t+1)}$를 텍스트 맥락 정보 $C^{(t+1)}$로 다시 초기화하는 반복적 양방향 융합(iterative bidirectional fusion) 방식을 나타낸다.

Table 2는 이 실험의 결과를 나타낸다. 실험 결과를 봤을 때, 모든 평가 항목 대부분의 성능 척도에서 (d) (Ours) 융합 방식이 가장 높은 성능을 보였다. 그리고 (a)와 (b)의 융합 방식을 서로 비교했을 때, 뒷받침 문장 예측(Supporting Sentence Prediction))의 EM은 (a)가 (b)보다 0.41% 더 높고 F1은 (b)가 (a)보다 0.42% 더 높았으며, 답변 예측(Answer Prediction))의 EM과 F1에서는 (a)가 (b)보다 각각 0.26%, 0.44% 더 높았다. 이러한 결과로 볼 때 뒷받침 문장 예측에서는 그래프 맥락 정보에 텍스트 맥락 정보를 융합

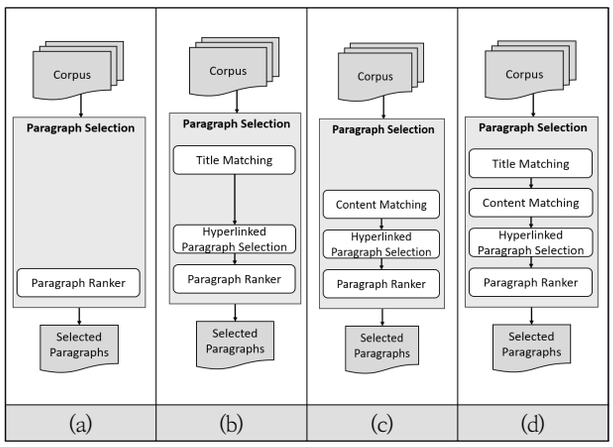

Fig. 8. Different Paragraph Selection Methods

Table 2. Results using Different Context Fusion Methods

Method	Supporting Sentence		Answer Prediction		Joint Prediction	
	EM	F1	EM	F1	EM	F1
(a)	56.74	85.40	56.77	70.54	35.18	62.25
(b)	56.33	85.82	56.51	70.10	34.98	62.22
(c)	54.85	84.94	57.18	70.74	34.30	62.01
(d) (Ours)	56.83	85.23	57.60	71.28	36.37	63.06

해 사용하는 것은 크게 부각될만큼 큰 효과가 없는 것으로 생각된다. 한편, (b)와 (c)의 융합 방식들을 서로 비교했을 때, 뒷받침 문장 예측의 EM과 F1에서는 (b)가 (c)보다 각각 1.48%, 0.88% 더 높은 성능을 보였다. 반면에, 답변 예측의 EM과 F1에서는 (c)가 (b)보다 각각 0.67%, 0.64% 더 높은 성능을 보였다. 또한, (d)와 (c)의 융합 방식을 서로 비교했을 때는 뒷받침 문장 예측의 EM과 F1에서 (d)가 (c)보다 각각 1.98%, 0.29% 더 높았으며, 답변 예측의 EM과 F1에서 (d)가 (c)보다 각각 0.42%, 0.54% 더 높은 성능을 보였다. 답변 예측에서 (d)와 (c)의 융합 방식은 그래프 맥락 정보를 텍스트 맥락 정보에 융합해 사용했기 때문에, (a)와 (b)의 융합 방식들보다 상대적으로 성능이 높은 것을 확인할 수 있다. 따라서 이러한 실험 결과를 볼 때, 그래프 맥락 정보와 텍스트 맥락 정보 사이의 상호 교환이 더 많을수록 질문 응답에 더 효과적임을 알 수 있다.

세 번째 실험은 제안 모델에서 사용하는 문단 선택(paragraph selection) 방식이 질문 응답 성능에 미치는 효과를 분석하는 실험이다. 이 실험에서는 Fig. 8의 (a), (b), (c), (d)에 표현된 서로 다른 4가지 문단 선택 방식들의 성능을 서로 비교하였다. 문단 선택을 위해, (a)는 문단 순위 결정기(PR)만을 이용하는 방식이며, (b)는 제목 매칭(TM), 하이퍼링크 연결 문단 선택(HP), 문단 순위 결정기(PR)를 이용하는 방식이다. 또 (c)는 내용 매칭(CM), 하이퍼링크 연결 문단 선택(HP), 문

Table 3. Results using Different Paragraph Selection Methods

Method	Precision	Recall
(a) PR	49.72	99.13
(b) TM+HP+PR	49.77	99.23
(c) CM+HP+PR	49.81	99.30
(d) TM+CM+HP+PR (Ours)	49.81	99.31

Table 4. Results using Different Word Embeddings

Method	Supporting Sentence		Answer Prediction		Joint Prediction	
	EM	F1	EM	F1	EM	F1
BERT	55.31	85.48	56.86	70.64	34.72	62.37
RoBERTa	55.57	85.13	57.0	70.34	34.23	61.73
ALBERT	56.83	85.23	57.60	71.28	36.37	63.06

단 순위 결정기(PR)를 이용하는 방식이며, (d)는 제안 모델과 같이, 제목 매칭(TM), 내용 매칭(CM), 하이퍼링크 연결 문단 선택(HP)과 문단 순위 결정기(PR)를 모두 이용하는 방식이다. 이 실험에서는 성능 척도로 정확도(precision)와 재현율(recall)을 이용하였다.

Table 3은 이 실험의 결과를 나타낸다. 이 실험 결과에서 알 수 있듯이, 제안 모델과 같이 문단 선택을 위해 제목 매칭(TM), 내용 매칭(CM), 하이퍼링크 연결 문단 선택(HP)과 문단 순위 결정기(PR)를 모두 이용하는(d) (Ours) 방식이 가장 높은 성능을 보였다. 반면에 문단 순위 결정기만 이용하는 (a)의 경우가 가장 낮은 성능을 보였다. 한편, (a)와 (b) 방식을 비교했을 때, (a) 방식에 제목 매칭(TM)을 추가로 사용하는 (b)가 (a)보다 정확도(precision)에서 0.05%, 재현율(recall)에서 0.1% 더 높은 성능을 보였다. 또, (b)와 (c) 방식을 서로 비교했을 때는, 내용 매칭(CM)을 이용하는 (c)가 제목 매칭(TM)을 이용하는 (b)보다 정확도(precision)에서 0.04%, 재현율(recall)에서 0.07% 더 높은 성능을 보였다. (d)와 (c)를 비교했을 때는, 정확도(precision)에서 차이가 없었으며 재현율(recall)에서 (d)가 (c)보다 0.01% 정도로 아주 미미하게 더 높은 성능을 보였다. 이러한 실험 결과를 볼 때, 질문에서 문서의 제목이 등장하는지 확인하는 제목 매칭(TM)보다는 실질적으로 내용에서 개체 정보를 다루는 내용 매칭(CM)이 질문과 연관된 문단을 선택하는데 더 효과적임을 알 수 있다.

네 번째 실험은 제안 모델에서 사용하는 ALBERT 기반의 단어 임베딩(word embedding) 방식의 효과를 분석하기 위한 실험이다. 이 실험에서는 BERT, RoBERTa, ALBERT 등 대표적인 사전 학습된 단어 임베딩 모델(pretrained word embedding model)들을 서로 비교하였다. 이 실험에서 BERT와 RoBERTa는 large 버전을 사용하였으며, ALBERT는 xxlarge 버전을 사용하였다. Table 4는 이 실험의 결과를 나타낸다. 실험 결과를 살펴보면, 제안 모델에서 단어 임베딩을 위해 이용하는 ALBERT 모델이 BERT와 RoBERTa 모델들에 비해 전반적으로 가장 높은 성능을 보였다. 하지만, BERT와 RoBERTa 모델을 서로 비교했을 때, EM 척도에서는 RoBERTa가 각각 0.26%와 0.14% 더 높은 성능을 보였으나, F1 척도에서는 오히려 BERT가 0.35%와 0.3%로 더 높은 성능을 보이는 등 서로 우열을 가리기 어려웠다.

Table 5. Comparison with State-of-the-art Models on HotpotQA Dataset

Model	Supporting Sentence		Answer Prediction		Joint Prediction	
	EM	F1	EM	F1	EM	F1
Baseline[1]	21.95	66.66	44.44	58.28	11.56	40.86
QFE[12]	58.80	84.70	53.70	68.70	35.40	60.60
DFGN[2]	53.10	82.24	55.66	69.34	33.68	59.86
CFGGN[3]	52.73	82.41	56.35	70.14	34.45	60.67
HGN*[6]	56.08	85.23	57.08	70.90	34.84	62.26
Ours	56.83	85.23	57.60	71.28	36.37	63.06

다섯 번째 실험은 기존 모델들과의 비교를 통해 제안 모델의 우수성을 입증하기 위한 실험이다. 이 실험에서는 제안 모델(Ours)을 기존 모델들인 Baseline[1], QFE[12], DFGN[2], CFGGN[3], HGN*[6] 등과 비교하였다. 이들중 HGN*은 HGN[6] 모델을 제안 모델과 동일한 컴퓨터 환경에서 재구성하여 학습한 모델이다.

Table 5는 이 실험의 결과를 나타낸다. 실험 결과를 보면 뒷받침 문장 예측, 답변 예측 등 모든 성능 평가 항목에서 제안 모델(Ours)이 전체적으로 가장 높은 성능을 보였다. 가장 최근 모델인 HGN*과 비교했을 때도 제안 모델(Ours)은 뒷받침 문장 예측의 EM에서 0.8% 더 높은 성능을, F1에서 서로 같은 성능을 보였다. 또 답변 예측의 EM과 F1에서도 제안 모델(Ours)이 HGN*보다 각각 0.52%, 0.38% 더 높은 성능을 보였다. 하지만 뒷받침 문장 예측의 EM 성능 척도에서는 예외적으로 기존의 QFE 모델이 제안 모델(Ours)보다 1.97% 더 높은 성능을 보였다. QFE 모델은 RNN을 이용해 뒷받침 문장들을 추출할 때 질문 정보와 문장들의 정보를 집중적으로 번갈아 참조한다. 따라서 이러한 QFE 모델의 특성 때문에 상대적으로 성능이 제안 모델을 포함해 다른 모델들보다 좀 더 성능이 높게 나온 것으로 생각된다. Table 5의 실험 결과들을 종합해보건대, 그래프 맥락 정보와 텍스트 맥락 정보를 효과적으로 활용할 수 있는 제안 모델의 높은 성능을 확인할 수 있었다.

마지막 실험은 HotpotQA 벤치마크 데이터 집합을 이용해, 제안 모델의 질문 응답 능력을 정성적으로 평가하는 실험이다. 본 논문에서는 Table 6, 7, 8과 같이 총 3가지 대표적

Table 6. The First Example Resulted by Proposed Model

[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 ...

Paragraph 2 : [Max Ophüls]
(3) Maximillian Oppenheimer (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, ...

Model	Supporting Sentence	Answer Prediction
Ground Truth	['Shirley Clarke', 1], ['Max Ophüls', 3]	Max Ophüls
Ours	['Shirley Clarke', 1], ['Max Ophüls', 3]	Max Ophüls

Table 7. The Second Example Resulted by Proposed Model

[Question] When was the female character that on June 9, 2015 was promoted to a series regular first introduced on "Once Upon a Time"?

[Paragraphs]
Paragraph 1 : [Once Upon a Time (season 5)]
(1) The fifth season of the American ABC fantasy-drama "Once Upon a Time" was ordered on ... (2) It began airing on September 27, 2015, and ended on ... (3) On June 9, 2015, the promotion of Rebecca Mader and Sean Maguire to series regulars was announced for the fifth season, portraying their characters Zelena / Wicked Witch of the West and Robin Hood, respectively, while a few days later, Michael Socha was confirmed to not be returning as a series regular as Will Scarlet / Knave of Hearts. (4) The fifth season also saw the series reach its 100th episode, which ...

Paragraph 2 : [Zelena (Once Upon a Time)]
(5) Zelena, also known as the Wicked Witch of the West, is a fictional character ... (6) She is portrayed by Rebecca Mader and was first introduced in the second half of the third season, serving as the new main antagonist. (7) After making recurring appearances in both the third and fourth seasons, Mader was promoted to series regular ...

Model	Supporting Sentence	Answer Prediction
Ground Truth	['Once Upon a Time (season 5)', 3], ['Zelena (Once Upon a Time)', 6]	the second half of the third season
Ours	['Once Upon a Time (season 5)', 3], ['Zelena (Once Upon a Time)', 6]	the second half of the third season

Table 8. The Third Example Resulted by Proposed Model

[Question] What was the last date the creator of the NOI was seen by Elijah Muhammad?

[Paragraphs]
Paragraph 1 : [Nation of Islam]
(1) The Nation of Islam, abbreviated as NOI, is an African American political and religious movement, founded in Detroit, Michigan, United States, by Wallace D. Fard Muhammad on July 4, 1930. (2) Its stated goals are to improve the spiritual, ... (3) Critics have described the organization ... (4) The Southern Poverty Law Center tracks the NOI as ... (5) Its official newspaper is ... (6) In 2007, the core membership was estimated to be ...

Paragraph 2 : [Wallace Fard Muhammad]
(7) Wallace D. Fard aka Wallace Fard Muhammad (born February 26, 1877) was a co-founder of ... (8) He arrived in Detroit in 1930 with an obscure background and ... (9) He was also known as being a seller of ... (10) Fard was last seen in 1933 by Elijah Muhammad, when Fard took off in an airplane from the Detroit airport.

Model	Supporting Sentence	Answer Prediction
Ground Truth	['Nation of Islam', 1], ['Wallace Fard Muhammad', 10]	last seen in 1933
Ours	['Nation of Islam', 1], ['Wallace Fard Muhammad', 10]	July 4, 1930

인 사례들을 중심으로 제안 모델의 질문 응답 결과를 분석하였다. 먼저 Table 6의 경우는 제안 모델이 질문에 대해 뒷받침 문장들(1번째 문장과 3번째 문장)과 답변(Max Ophüls)을 정확히 예측한 사례이다. 따라서 이 사례에서 제안 모델은 계층적 그래프를 이용한 다중 홉 추론과 다중 작업 답변 예측을 정확히 수행한 것으로 판단한다. Table 7의 경우도 제안 모델이 질문에 대한 뒷받침 문장들과 답변을 정확히 예측한 사례이다. 이 질문의 경우, 답변은 하나의 개체(entity)가 아니라 시간대(time)가 되어야 한다. 따라서, 문서에서 추출한 개체 정보를 담고 있는 개체 그래프만 이용하였다면, 답변을 제대로 예측하지 못했을 수 있다. 하지만 제안 모델은 개체뿐만 아니라 개체들을 포함하고 있는 문장과 문단 계층까지 포함하는 계층적 그래프와 본래 문서의 맥락 정보를 담고 있는 텍스트 맥락 정보도 함께 활용하기 때문에, 개체가 아닌 답변도 잘 예측해낸 것으로 판단한다.

한편, Table 8의 경우는 제안 모델이 답변 예측을 올바르게 하지 못한 사례이다. 해당 질문의 경우, 제안 모델은 답변 예측을 위한 뒷받침 문장들(1번째 문장과 10번째 문장)은 정확히 찾아내었다. 하지만, 정작 답변이 될 수 있는 날짜는 정답을 포함하고 있는 10번째 문장이 아니라, 1번째 문장에서 잘못 추출해냈다. 이와 같은 오류는 뒷받침 문장들을 따라 답

변에 도달하는 올바른 추론 경로를 찾지 못했기 때문인 것으로 판단한다. 따라서 현재의 제안 모델은 뒷받침 문장들을 활용해 보다 정확한 추론과 답변 예측이 이루어지도록 좀 더 보완되어야 할 것으로 보인다.

5. 결 론

본 논문에서는 다중 홉 추론 및 다중 작업을 요구하는 복잡 질문에 대해 효과적인 추론을 수행하는 새로운 심층 신경망 모델을 제안하였다. 제안 모델은 제목 매칭, 내용 매칭 등 의미적 검색을 수반한 문단 선택을 이용해 문서 집합으로부터 질문과 연관성이 높은 문단들을 선택하였다. 이 문단들로부터 다양한 수준의 맥락 정보를 갖는 계층적 그래프를 생성하고, 그래프 추론 및 맥락 융합을 통해 답변에 필요한 다양한 수준의 맥락 정보들을 추출하였다. 그리고 HotpotQA 벤치마크 데이터 집합을 이용한 비교 실험을 통해, 제안 모델의 긍정적 효과를 확인할 수 있었다. 현재의 제안 모델은 앞서 정성적 평가 실험에서도 보았듯이, 여러 문장들과 문단들을 포함하는 긴 텍스트에서는 다소 부정확한 뒷받침 문장 예측과 답변 예측 능력을 보여주고 있다. 향후 연구에서는 이러한 제한점을 보완할 수 있도록 추론과 답변 예측 기능을 추가 개선해볼 계획이며, 또 HotpotQA 이외에 보다 다양한 벤치마크 데이터 집합들을 이용해 성능 검증을 수행해볼 계획이다.

References

[1] Z. Yang, P. Qi, S. Zhang, Y. Bengio, W. Cohen, R. Salakhutdinov, and C. D. Manning, "HotpotQA: A dataset for diverse, explainable multi-hop question answering," In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium, pp.2369-2380, 2018.

[2] L. Qiu, Y. Xiao, Y. Qu, H. Zhou, L. Li, W. Zhang, and Y. Yu, "Dynamically fused graph network for multi-hop reasoning," In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence*, Italy, pp.6140-6150, 2019.

[3] M. Zhang, F. Li, Y. Wang, Z. Zhang, Y. Zhou, and X. Li, "Coarse and fine granularity graph reasoning for interpretable multi-hop question answering," *IEEE Access*, Vol.8, pp.56755-56765, 2020.

[4] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Minneapolis, Minnesota, pp.4171-4186, 2019.

[5] A. Asai, K. Hashimoto, H. Hajishirzi, R. Socher, and C. Xiong, "Learning to retrieve reasoning paths over Wikipedia graph for question answering," In *Proceedings of International Conference on Learning Representation*, 2020.

[6] Y. Fang, S. Sun, Z. Gan, R. Pillai, S. Wang, and J. Liu, "Hierarchical graph network for multi-hop question answering," In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pp.8823-8838, 2020.

[7] J. Welbl, P. Stenetorp, and S. Riedel, "Constructing datasets for multi-hop reading comprehension across documents," *Transactions of the Association for Computational Linguistics*, Vol.6, pp.287-302, 2018.

[8] Y. Liu, et al., "RoBERTa: A robustly optimized BERT pretraining approach," arxiv.org/abs/1907.11692, 2019.

[9] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," In *Proceedings of International Conference on Learning Representation*, 2013.

[10] J. Pennington, R. Socher, and C. Manning, "GloVe: Global vectors for word representation," In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, Doha, Qatar, pp.1532-1543, 2014.

[11] A. Vaswani, et al., "Attention is all you need," In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, Long Beach, California, pp.6000-6010, 2017.

[12] K. Nishida, K. Nishida, M. Nagata, A. Otsuka, I. Saito, H. Asano, and J. Tomita, "Answering while summarizing: Multi-task learning for multi-hop QA with evidence extraction," In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy, pp.2335-2345, 2019.

[13] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," In *Proceedings of the 5th International Conference on Learning Representations*, Toulon, France, 2017.

[14] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," In *Proceedings of the 6th International Conference on Learning Representations*, Vancouver, Canada, 2018.

[15] M. Tu, G. Wang, J. Huang, Y. Tang, X. He, and B. Zhou, "Multi-hop reading comprehension across multiple documents by reasoning over heterogeneous graphs," In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, pp.2704-2713, 2019.

[16] M. Tu, K. Huang, G. Wang, J. Huang, X. He, and B. Zhou, "Select, answer and explain: Interpretable multi-hop reading comprehension over multiple documents," In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence*, New York, USA, pp.9073-9080, 2020.

[17] N. D. Cao, W. Aziz, and I. Titov, "Question answering by reasoning across documents with graph convolutional networks," In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis*, Minnesota, pp.2306-2317, 2019.

[18] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "ALBERT: A lite BERT for self-supervised learning of language representations," In *Proceedings of International Conference on Learning Representation*, 2020.

[19] M. Seo, A. Kembhavi, A. Farhadi, and H. Hajishirzi, "Bidirectional attention flow for machine comprehension," In *Proceedings of International Conference on Learning Representation*, 2017.

이 상 의

https://orcid.org/0000-0001-9072-8094
e-mail : rmrlrmrl125@kyonggi.ac.kr
2019년 경기대학교 컴퓨터과학과(학사)
2020년 ~ 현 재 경기대학교 컴퓨터과학과 석사과정
관심분야 : 인공지능, 자연어처리, 로봇지능

김 인 철

https://orcid.org/0000-0002-5754-133X
e-mail : kic@kyonggi.ac.kr
1985년 서울대학교 수학과(이학사)
1987년 서울대학교 전산과학과(이학석사)
1995년 서울대학교 전산과학과(이학박사)
1996년 ~ 현 재 경기대학교 컴퓨터과학과 교수
관심분야 : 인공지능, 기계학습, 로봇지능