정답 분리 인코더와 복사 메커니즘을 이용한 한국어 질문 생성

김건영○, 이창기

강원대학교
uhi7074@gmail.com leeck@kangwon.ac.kr

Korean Question Generation Using Answer-Separated Encoder And Copying Mechanism

Geon-Yeong Kim○, Chang-Ki Lee
Kangwon National University

요 약

질문과 그에 대한 근거가 있는 문서를 읽고 정답을 예측하는 기계 독해 연구가 최근 활발하게 연구되고 있다. 기계 독해 문제를 위해 주로 사용되는 방법은 다층의 신경망으로 구성된 딥러닝 모델로 좋은 성능을 위해서는 양질의 대용량 학습 데이터가 필요하다. 그러나 질과 양을 동시에 만족하는 학습 데이터를 구축하는 작업에는 많은 경제적 비용이 소모된다. 이러한 문제를 해결하기 위해, 본 논문에서는 정답 분리 인코더와 복사 메커니즘을 이용한 단답 기반 한국어 질문 자동 생성 모델을 제안한다.

주제어: 기계 독해, 질문 자동 생성, 정답 분리 인코더, CopyNet

1. 서론

질문과 그에 대한 근거가 있는 문서를 읽고 정답을 예측하는 기계 독해 연구가 최근 활발하게 연구되고 있으며, Stanford Question Answering Dataset(SQuAD)[1,2], Natural Questions (NQ)[3] 등 기계 독해를 위한 다양한 데이터가 만들어졌다. 근래에는 한국어 데이터인 Korean Question Answering Dataset (KorQuAD)[4]도 공개되었다.

기계 독해를 위해 주로 사용되는 방법은 다층의 신경망으로 구성된 딥러닝 모델을 양질의 대용량 학습 데이터로 학습하는 방법이다. 그러나 기계 독해를 위한 데이터 구축은 쉽지 않은 일로, 문서에 등장한 정답과 이를 도출할 수 있는 질문은 사람이 수작업으로 만들어야 하며, 여러 사람이 데이터를 구축할 경우 일관성의 문제가 발생할 수 있다. 이러한 문제를 해결하기 위해 최근 질문 자동 생성 연구가 활발히 연구되고 있다.

질문 생성은 문서와 질문을 보고 정답을 예측하는 기계 독해와 반대로, 문서와 정답을 보고 그에 따른 질문을 생성하는 분야이다. 딥러닝을 통한 질문 생성의 자동화는 기계 독해 데이터 구축 문제의 한 가지 방안이 될 수 있다. 본 논문에서는 정답 분리 인코더와 복사 메커니즘을 이용한 단답 기반 한국어 질문 자동 생성 모델을 제안한다.

실험을 위해 한국어 기계 독해 데이터인 KorQuAD 데이터를 정제하여 문장, 정답, 질문으로 이루어진 데이터를 만들어 사용하였으며, baseline 모델로 seq2seq 모델을 사용한다.

인코더가 1개인 단순 seq2seq 모델은 입력에 정답이 들어가 있어 정답이 생성 질문에 자주 등장하는 문제점이 존재한다. 이를 해결하기 위해 [5]에서 제안한 정답 분리 seq2seq 모델은 2개의 인코더를 사용하여 정답이 마스킹된 문장과, 정답을 각각 따로 인코딩하여 디코딩시 실제 정답이 생성 질문에 등장하지 않도록 억제한다. 그러나 정답을 마스킹하는 메타 문자가 실제 정답인 부분임을 나타내기에 부족할 수 있다. 본 논문에서는 [5]에서 제안한 정답 분리 seq2seq 모델에 정답임을 표시하는 정답 위치 자질을 추가하고, 생성할 질문의 많은 단어들이 입력 문장에 포함된다고 가정하고, 이를 잘 모델링 할 수 있는 복사 메커니즘을 추가하였다.

실험 결과, 기존 seq2seq모델보다 BLEU에서는 성능이 낮았지만 정성평가와 정답이 생성 질문에 등장하는지 여부에 대해서는 정답 분리 seq2seq가 더 우수한 성능을 보였다.

2. 질문 생성을 위한 데이터

문장	로 널드/NNP_ 레이건/NNP_ 대통령/NNG_ 밑/NNG_ 에서/JKB_ **<a>** 을/JKO_ 지내/VV_ 었/EP_ 으며/EC_ ,/SP_ 리처 드/NNP_ 닉 슨/NNP_ 과/JC_ 제 럴 드/NNP_ 포드/NNP_ 대통령/NNG_ 밑/NNG_ 에서/JKB_ 백악관/NNP_ 비서실장/NNG_ 을/JKO_ 지내/VV_ 었/EP_ 다/EF_ ./SF_
정답	**국무/NNG_ 장관/NNG_**
질문	알 렉 산 더/NNP_ 헤 이 그/NNP_ 가/JKS_ 로 널드/NNP_ 레이건/NNP_ 대통령/NNG_ 밑/NNG_ 에서/JKB_ 맡/VV_ 은/ETM_ 직 책/NNG_ 은/JX_ 무엇/NP_ 이/VCP_ 었/EP_ 나/EF_ ?/SF_

표 1. KorQuAD 질문 생성 데이터 예제

https://github.com/Hagazzusa/korquad_qg_data

[표 1]은 학습에 사용한 데이터 예제이다. 본 논문에

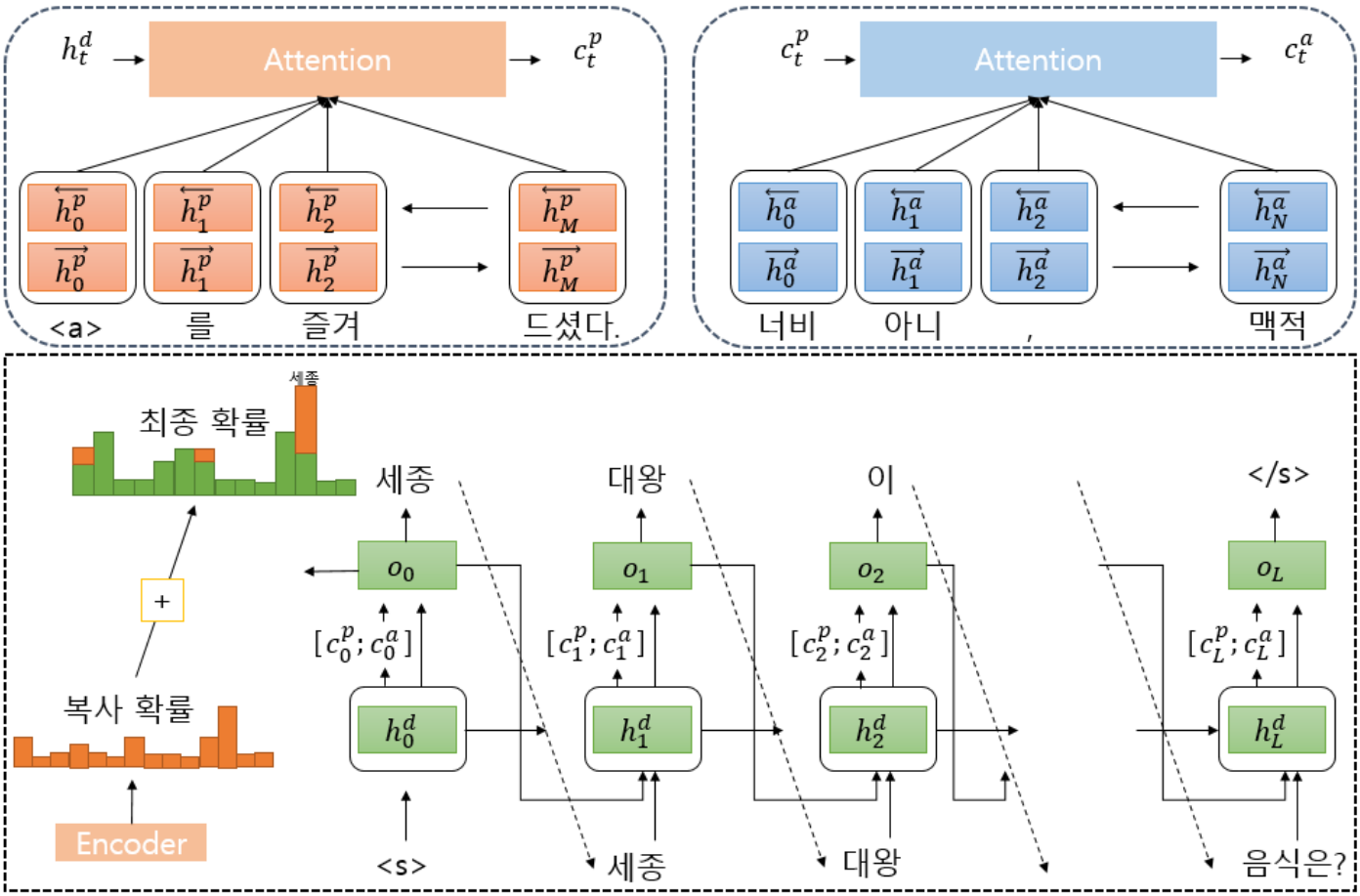

그림 1. 정답 분리 seq2seq 모델

(왼쪽 위: 문장 인코더, 오른쪽 위: 정답 인코더, 아래: 디코더)

서는 KorQuAD데이터를 가공하여 문장, 정답, 질문으로 이루어진 데이터를 만들었으며, 이때 문서에서 정답을 포함하는 문장만 추출하였다. 문장이나 질문이 250단어를 넘으면 데이터에서 제외하였고, 형태소 태깅과 Byte Pair Encoding (BPE)[6]을 적용하였다.

공개된 KorQuAD는 학습, 개발데이터만 제공하므로 학습데이터의 90%는 학습, 10%는 개발, KorQuAD 개발데이터는 평가데이터로 사용하였다. 최종 학습데이터는 54,154개, 개발데이터는 6,021개, 평가데이터는 5,774개이다.

문장에서 <a>는 정답을 의미하는 메타 문자이며, 각 단어 별로 정답 유무에 따라 자질을 가진다. [표 1]로 예를 들자면 <a>와 국무 장관은 같은 정답이므로 같은 자질을 가진다. 이외에 나머지 단어들은 <a>, 국무총리와 다른 자질을 가진다.

3. 정답 분리, 복사 메커니즘 seq2seq 모델

[그림 1]은 질문 생성을 위해 사용한 정답 분리 seq2seq 모델이다. 전체적인 흐름은 다음과 같다. 정답이 마스킹된 문장이 문장 인코더로 입력되고 실제 정답은 정답인코더로 입력된다. 이후 매 디코딩 시간마다 문장인코더에 주의 집중 구조(Attention Mechanism)[8]를 사용하여 새로운 문맥 표현을 만들고, 이 문맥 표현과 정답 인코더 사이에 주의 집중을 하여 정답 문맥 표현을 만든다. 최종 출력층에선 현재 디코더의 은닉층, 문장, 정답을 고려하여 디코딩한다.

$$(1)\ \overrightarrow{h^p_{n+1}} = GRU\left([w_{n+1}; f_{n+1}], \overrightarrow{h^p_t}\right)$$
$$(2)\ \overleftarrow{h^p_{n-1}} = GRU\left([w_{n-1}; f_{n+1}], \overleftarrow{h^p_t}\right)$$
$$(3)\ h^p_n = \overrightarrow{h^p_n} + \overleftarrow{h^p_n}$$
$$(4)\ context = \overleftarrow{h^p_0} + \overrightarrow{h^p_M}$$
$$(5)\ score^p_{t,n} = \frac{h^{p^T}_n h^d_t}{\sqrt{dim}}$$
$$(6)\ a^p_{t,n} = \frac{e^{score^p_{t,n}}}{\sum_{\forall_k} e^{score^p_{t,k}}}$$
$$(7)\ c^p_t = \sum_{\forall_k} a^p_{t,k} h^p_k$$

문장 인코더는 다음과 같이 나타낼 수 있다. 우선 문장이 $passage = \{w_0, w_1, \ldots, w_M\}$ 이고 w가 각 단어를 나타낸다. f는 자질이며 w가 <a>이면 f^1 아니면 f^0이다. 각 단어들은 Gated Recurrent Unit(GRU)[10]를 통해 (1-2)처럼 정, 역방향으로 인코딩된다. 인코딩된 값들은 (3)처럼 더하여 사용한다. (4)는 디코더의 0번째 은닉층이자 매 디코더의 입력으로 들어간다. (5-7)은 주의 집중 구조

이다. 디코더의 은닉층과 문장 인코더, 두 벡터의 내적을 벡터의 크기의 제곱근으로 나눠준 값을 스코어로 사용하였다.

$$(8)\ \overrightarrow{h_{n+1}^{a}} = GRU\left([w_{n+1}, f^{1}], \overrightarrow{h_{t}^{a}}\right)$$
$$(9)\ \overleftarrow{h_{n-1}^{a}} = GRU\left([w_{n-1}, f^{1}], \overleftarrow{h_{t}^{a}}\right)$$
$$(10)\ h_{n}^{a} = \overrightarrow{h_{n}^{a}} + \overleftarrow{h_{n}^{a}}$$
$$(11)\ score_{t,n}^{a} = \frac{h_{n}^{a\,T} c_{t}^{p}}{\sqrt{dim}}$$
$$(12)\ a_{t,n}^{a} = \frac{e^{score_{t,n}^{a}}}{\sum_{\forall_k} e^{score_{t,k}^{a}}}$$
$$(13)\ c_{t}^{a} = \sum_{\forall_k} a_{t,k}^{a}\, h_{k}^{a}$$

정답 인코더도 문장 인코더와 비슷하다. 정답을 $\text{answer} = \{w_0, w_1, \ldots, w_N\}$ 처럼 표현한다. 다만 항상 정답이므로 자질은 f^1만 사용된다. (8-9)를 통해 정답이 인코딩 되며 (10)처럼 정방향, 역방향을 더하여 사용한다. (11-13)은 주의 집중 구조이며 문장 인코더와 다르게 디코더의 은닉층과 스코어 계산을 하지 않고 문장 문맥 표현인 (7)과 스코어 계산을 한다. 이렇게 하면 단순히 디코더만 보는게 아니라 문장도 같이 고려할 수 있다.

$$(14)\ h_{n+1}^{d} = GRU(w_n, h_{n}^{d}, o_n, context)$$
$$(15)\ o_n = LeakyReLU\left(W^{o}[h_{n}^{d}; c_{0}^{p}; c_{0}^{a}]\right)$$
$$(16)\ outScore_{y,n} = W^{t} O_n$$
$$(17)\ P_{out}(y) = \frac{outScore_{y,n}}{\sum_{\forall_k} outScore_{k,n}}$$

$$(18)\ P_{copy}(y) = sigmoid(W^{copy} o_n)$$

$$(19)\ copyScore_{n,m} = \frac{h_{m}^{p\,T} o_n}{\sqrt{dim}}$$

$$(20)\ P_{src}(y) = \frac{copyScore_{n,m}}{\sum_{\forall_k} copyScore_{n,k}}$$
$$(21)\ P(y|y_{0:n-1}) = P_{copy}(y) P_{src}(y) + \left(1 - P_{copy}(y)\right) P_{out}(y)$$
$$(22)\ y_n = argmax_y\, P(y|y_{0:n-1})$$

(14-17)은 디코더가 확률을 계산하는 방법을 나타낸다. (18-21)은 문장 인코더에 등장하는 단어들을 출력 분포에 더해주는 복사 메커니즘[7]이다.

(14)에서 디코더는 이전 시간 출력과 은닉층, 주의 집중 결과와 주의 집중 구조를 거치지 않은 문장의 문맥 표현으로부터 현재 시간 은닉층인 h_{n+1}^{d}를 만든다. 이후 순서대로 주의 집중이 가미된 문장, 정답 문맥포현을 만들고 (15)에서 한번의 비선형 변환을 거친다. (15)는 다음 시간 디코더의 입력으로 들어가는데 이는 [9]에서 제안한 input-feeding이다.

복사 구조는 주의 집중 구조와 비슷하다. (19, 20)을 통해 스코어와 확률값을 계산하고 (18, 21)을 통해 기존 디코더의 생성 확률인 (17)과 가중치 합을 한다. 이렇게 복사 확률이 더해진 $P(y|y_{0:n-1})$에서 (22)처럼 가장 높은 확률을 가지는 출력 단어를 선택한다.

4. 실험 및 결과

모델	BLEU1	BLEU2	BLEU3	BLEU
seq2seq	40.58	31.84	26.30	22.28
정답분리 +복사	38.61	30.58	25.18	21.12

표 2. BLEU

실험을 위해 768차원의 사전 학습된 워드 임베딩과 32차원의 자질 임베딩을 입력으로 사용했으며 모든 GRU의 은닉 차원은 800차원으로 설정하였다. 실험 결과 정답분리 인코더와 복사 메커니즘을 이용한 모델이 seq2seq 모델에 비해 BLEU에서 점수가 떨어졌으나 정성 평가 결과, 정답을 도출 할 수 있는 질문을 생성하는 것은 정답분리 seq2seq 모델이 생성한 질문임을 알 수 있었다.

문장	<a> 여의도 농민 폭력 시위를 주도한 혐의(폭력 행위 등 처벌에 관한 법률 위반)으로 지명 수배되었다.
정답	1989년 2월 15일
실제 질문	임종석이 여의도 농민 폭력 시위를 주도한 명의로 지명 수배된 날은?
seq2seq	1989년 2월 15일 여의도 농민 폭력 시위를 주도한 혐의로 지명 수배된 곳은?
정답분리 +복사	여의도 농민 폭력 시위를 주도한 혐의를 언제 지명 수배되었는가?

표 3. 생성 예제 1

문장	성 베드로 광장 양편에 각각 네줄로 늘어선 토스카나식 기둥 <a>와 벽에서 돌출된 기둥 88개로 이루어진 베르니니의 회랑은 1656년에 공사를 시작해 1667년에 완공되었다
정답	284개
실제 질문	성 베드로 광장 양편에 네줄로 늘어선 토스카나식 기둥의 수는?
seq2seq	베르니니의 회랑이 완공된 해는?
정답분리 +복사	성 베드로 광장 양편에 늘어선 토스카나식 기둥 갯수는?

표 4. 생성 예제 2

문장	2005년 독립해 <a>를 설립하고 대표 이사로 재직하며 임정희, 에이트, 2AM, 방탄소년단등의 음반을 프로듀스 하고 있다.
정답	빅히트 엔터테인먼트
실제 질문	방시혁이 설립한 회사의 이름은?
seq2seq	빅히트 엔터테인먼트를 설립한 것은 언제인가?
정답 분리 +복사	밥 딜런이 설립한 곳은 어디 입니까?

표 5. 생성 예제 3

문장	그는 하루 <a> 정도 피우던 담배와 밤새 마시던 술을 이 무렵 끊었다.
정답	6갑
실제 질문	당시 김영삼은 하루에 담배 몇 갑 정도를 피웠는가?
seq2seq	코벨레스키가 하루 6갑 정도 피우던 담배와 밤새 마시던 술을 끊었다. 이 사람은 누구인가?
정답 분리 +복사	비욘세의 후통은 몇 석에 속했나?

표 6. 생성 예제 4

[표 3-6]은 실제 생성된 질문 예제를 보여주고 있다. 단순 Seq2seq는 문장에서 정답이 마스킹 되지 않고 입력되기 때문에 대부분의 경우 실제 정답이 생성 질문에 등장하며 5,774개인 테스트 데이터를 확인한 결과 1,712질문에 정답이 등장하였다. 반면에 정답 분리 seq2seq의 경우, 단 3개의 질문에서만 정답이 질문에 등장하였다.

[그림 2]는 주의 집중 메커니즘의 시각화 예제이다. 왼쪽은 생성된 질문이고 위는 문장이다. 밝은 색일수록 주의 집중 가중치가 높음을 의미하며, 메커니즘이 잘 동작한 것을 확인할 수 있다.

5. 결론 및 향후 연구

본 논문에서는 정답 분리 인코더와 복사 메커니즘을 이용한 한국어 질문 자동 생성 모델을 제안하였다. 실험 결과, 단순한 seq2seq 모델을 이용할 경우 정답이 질문에 포함되는 문제가 발생하였으나, 정답 분리 seq2seq 모델은 이러한 문제를 해결함을 볼 수 있었다.

향후 연구로는 정답이 포함된 문장보다 더 넓은 문맥을 이해할 수 있도록 모델을 개선할 예정이다.

감사의 글

이 논문은 2019년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No.2013-2-00131, 휴먼 지식증강 서비스를 위한 지능진화형 Wise QA 플랫폼 기술 개발)

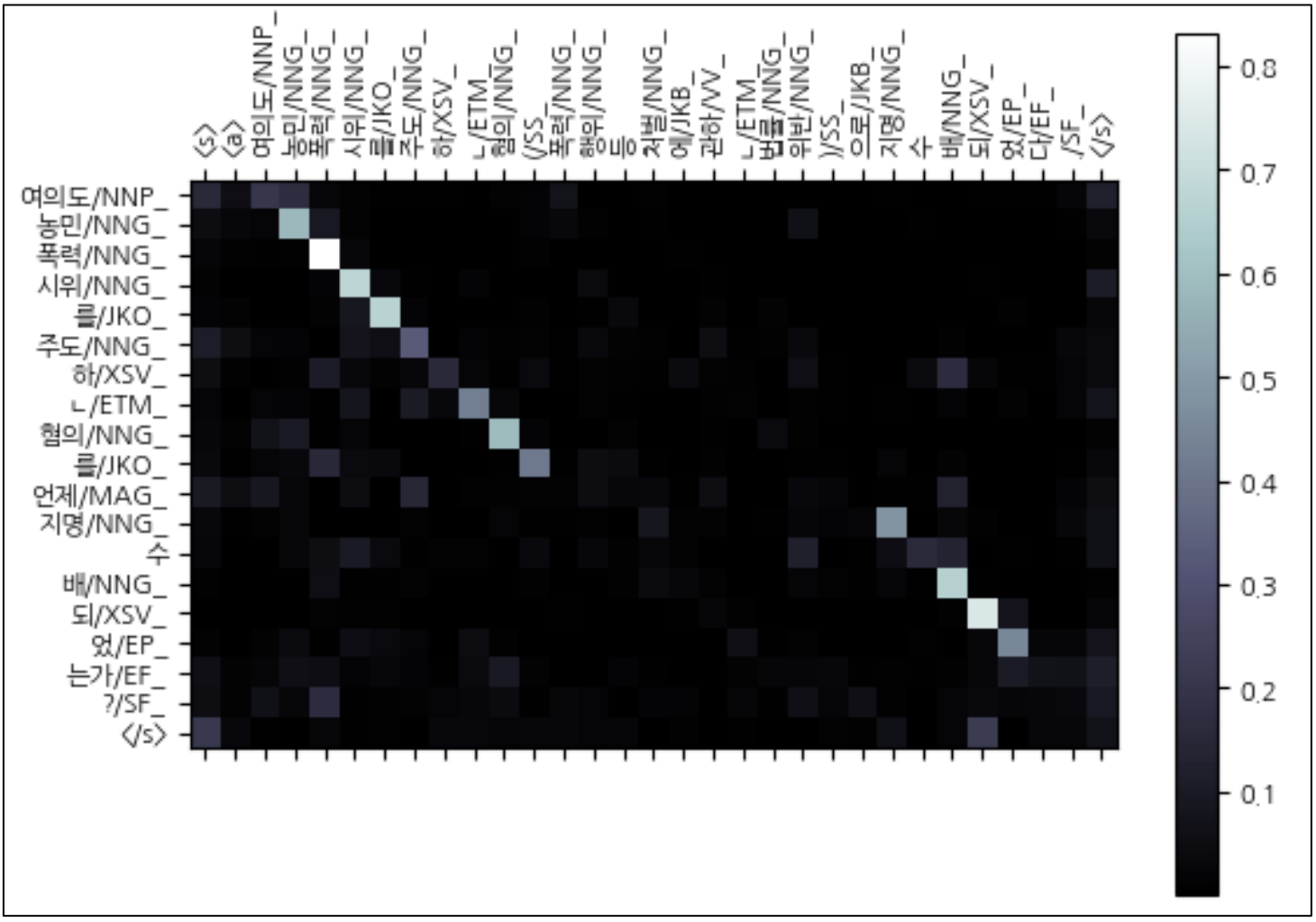

그림 2. 주의 집중 구조 시각화

참고문헌

[1] Rajpurkar Pranav, Zhang Jian, Lopyrev Konstantin, Liang Percy. “SQuAD: 100,000+ Questions for Machine Comprehension of Text.”, Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp.2383-2392, 2016.

[2] Rajpurkar, Pranav, Robin Jia, Percy Liang. "Know What You Don't Know: Unanswerable Questions for SQuAD." Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, pp.784-789, 2018.

[3] Tom Kwiatkowski, et al. “Natural Questions: a Benchmark for Question Answering Research”, Transactions of the Association of Computational Linguistics, 2019.

[4] 임승영, 김명지, 이주열. "KorQuAD: 기계독해를 위한 한국어 질의응답 데이터셋". 한국정보과학회 학술발표논문집, pp.539-541, 2018.

[5] Kim, Yanghoon, et al. "Improving neural question generation using answer separation." Proceedings of the AAAI Conference on Artificial Intelligence, pp.6602-6609, 2019.

[6] Sennrich, Rico, Barry Haddow, and Alexandra Birch. "Neural Machine Translation of Rare Words with Subword Units.", Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, pp.1715-1725, 2016.

[7] See, Abigail, Peter J. Liu, and Christopher D. Manning. "Get To The Point: Summarization with Pointer-Generator Networks.", Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, pp.1073-1083, 2017.

[8] Cho, Kyunghyun, et al. "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation.", Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp.1724-1734, 2014.

[9] Luong, Thang, Hieu Pham, and Christopher D. Manning. "Effective Approaches to Attention-based Neural Machine Translation." Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp.1412-1421, 2015.