

복사-메커니즘과 추론 단계의 페널티를 이용한 Copy-Transformer 기반 문서 생성 요약

전동현⁰, 강인호

네이버

{donghyeon.jeon, once.ihkang}@navercorp.com

Copy-Transformer model using Copy-Mechanism and Inference Penalty for Document Abstractive Summarization

Donghyeon-Jeon, In-Ho Kang
Naver Corporation

요약

문서 생성 요약은 최근 딥러닝을 이용한 end-to-end 시스템을 통해 유망한 결과들을 보여주고 있어 연구가 활발히 진행되고 있는 자연어 처리 분야 중 하나이다. 하지만 문서 생성 요약 모델을 구성하기 위해서는 대량의 본문과 요약문 쌍의 데이터 셋이 필요한데, 이를 구축하기가 쉽지 않다. 따라서 본 논문에서는 정교한 뉴스 기사 요약 데이터 셋을 기계적으로 구축하는 방법을 제안한다. 또한 딥러닝 기반의 생성 요약은 입력 문서와 다른 정보를 생성하거나, 또는 같은 단어를 반복하여 생성하는 문제점들이 존재한다. 이를 해결하기 위해 요약문을 생성할 때 입력 문서의 내용을 인용하는 복사-메커니즘과, 추론 단계에서 단어 반복을 직접적으로 제어하는 페널티를 사용하면 상대적으로 안정적인 문장이 생성될 수 있다. 그리고 Transformer 모델은 순환 신경망 모델보다 요약문 생성 과정에서 시퀀스 길이가 긴 본문의 정보를 적절히 인코딩하여 줄 수 있는 모델이다. 따라서 본 논문에서는 복사-메커니즘과 추론 단계의 페널티를 이용한 Copy-Transformer 모델을 한국어 문서 생성 요약 데이터에 적용하였다. 네이버 지식iN 질문 요약 데이터 셋과 뉴스 기사 요약 데이터 셋 상에서 실험한 결과, 제안한 모델을 이용한 생성 요약이 비교 모델들 대비 가장 좋은 성능을 보이고 양질의 요약을 생성하는 것을 확인하였다.

주제어: 문서 요약, 생성 요약, 복사 메커니즘, Copy-Transformer

1. 서론

문서 요약 시스템은 긴 텍스트 문서로부터 정보를 압축하여 핵심이 되는 짧은 문장의 요약문을 생성하는 것을 목표로 한다[1]. 문서 요약의 접근법은 크게 추출 요약과 생성요약으로 나눌 수 있다. 추출 요약은 본문에서 중요한 문장을 선택하는 방법이며, 생성 요약은 본문의 내용을 대표할 수 있는 새로운 문장을 만들어 내는 방법이다. 추출 요약은 본문의 문장 중에서 선택하는 방법이기에 때문에 완전한 문장을 요약문으로 만들 수 있는 장점이 있지만 본문 내에 핵심이 될 문장이 없거나 지시 대명사가 들어간 문장이 선택될 경우는 문제가 될 수 있다. 이와 달리 생성 요약은 본문과 다른 내용을 요약문으로 생성할 위험성이 있고 자연스러운 문장을 만들어 내야하는 어려움이 있지만, 추출 요약에 비해 간결하고 정보를 밀도 있는 문장으로 표현할 수 있다는 장점이 있다.

생성 요약은 기계번역 모델과 유사한 딥러닝 기반의 end-to-end 시스템을 통해 가능성 있는 결과들을 보여주고 있어 연구가 활발히 진행되고 있다[2]. 생성 요약 시스템은 원본 문서를 인코딩한 다음 추상적인 짧은 요약으로 디코딩하는 Attentional Sequence-to-Sequence 모델 구조를 기본으로 한다[3]. 하지만 아직까지 딥러닝 기반의 생성 요약은 입력 문서와 다른 정보를 생성하거나 같

은 단어를 반복하여 생성하는 문제점이 존재한다. 이러한 문제에 대해 Pointer-Generator Network[4]는 Bi-LSTM 기반의 Attentional Sequence-to-Sequence 모델에 입력 문서에서 단어를 복사하여 가져올지, 단어 사전에서 생성할지를 결정하는 복사-메커니즘(Copy-mechanism)과 단어의 반복을 피하고자 이전 단어들의 분포를 고려하여 단어를 생성하는 커버리지-메커니즘(Coverage-mechanism)을 사용함으로써 일정 수준의 문제를 해결하였고 좋은 성능을 보여주었다.

Copy-Transformer 모델은 여러 자연어 처리 분야에서 좋은 성능을 보여주고 있는 Transformer[5] 모델에 복사-메커니즘을 적용한 모델이다. 본 논문에서는 생성 요약을 위해 Copy-Transformer 모델에 추가적으로 추론 단계 페널티를 이용한 방법을 제안한다. 그리고 제안하는 모델이 한국어 문서 생성 요약 데이터에서 기존의 요약 모델들과 비교하여 좋은 성능을 얻을 수 있음을 보인다. 제안하는 모델의 일반적인 성능을 검증하기 위해 입력 문서 길이가 짧고 구어체가 많은 네이버의 커뮤니티 기반 질의응답 시스템인 지식iN[6] 질문 데이터와 입력 문서 길이가 길고 문어체로 되어 있는 뉴스 기사 데이터를 이용한다. 본 논문에서 뉴스 기사 데이터의 생성 요약을 위한 학습 데이터 셋을 구축할 때 필요한 필터링 방식을 제안하고, 구축된 대량의 지식iN 질문 요약 데이터 셋과

뉴스 기사 요약 데이터 셋 상에서 실험한 결과, 입력 데이터의 길이나 문체에 상관없이 일반적으로 성능이 우수함을 확인할 수 있었다.

2. 관련 연구

생성 요약은 Attentional Sequence-to-Sequence 모델을 기본 틀로 연구하고 있다. 기본적인 Attentional Sequence-to-Sequence 모델은 인코더와 Attention mechanism 을 이용해 본문의 주의 분포(Attention Distribution)를 구하고 문맥 벡터(Context vector)를 생성한다. 생성된 문맥 벡터는 디코더의 은닉 계층과 결합하여 현재 단어를 생성하기 때문에 본문의 정보를 담은 요약문을 생성할 수 있다. Pointer-Generator Network는 Bi-LSTM 기반의 Attentional Sequence-to-Sequence 모델의 주의 분포와 단어 분포(Vocab Distribution)를 현재 단어와 문맥 벡터, 디코더의 은닉 계층을 통해 생성 확률을 계산하고, 본문의 단어를 복사할지 다른 단어를 생성할지 결정한다. 이 모델이 생성 요약에서 좋은 성능을 보여주었기에 [7, 8]은 Pointer-Generator Network에 강화 학습을 적용하여 좀더 정밀하게 문제를 해결하려는 시도를 하였다. 하지만 강화 학습의 적용은 성능 수치적으로는 약간 높지만, 실제 사람이 평가했을 때 다소 부자연스럽거나 간결하지 못한 문장을 생성하는 경향이 있다[9]. 최근 Attentional Sequence-to-Sequence 구조는 Transformer가 기계 번역 분야에서 최고의 성능과 학습 효율성을 보여주면서 여러 분야에서 Transformer를 중심으로 연구가 활발해졌고, 생성 요약에서도 이를 활용하는 방법이 연구되고 있다[10].

본 논문에서는 새로운 생성 요약 방법으로 Transformer 모델에 복사-메커니즘과 추론 단계에서 페널티를 이용한 Copy-Transformer 모델을 제안하고 이를 한국어 요약 데이터 셋에 적용한다.

3. Copy-Transformer 기반 생성 요약 모델

본 논문에서는 네이버 지식iN 질문 데이터와 네이버 뉴스 요약봇의 뉴스 기사 데이터를 이용하여 생성 요약을 수행하였다. 생성 요약에 필요한 학습 데이터는 각각의 필터링 방법을 통해 구축하고, 데이터의 종류에 상관없이 우수하게 본문을 요약할 수 있는 모델을 구성하였다.

3.1. 학습 데이터

문서 요약 학습 모델을 구성하기 위해서는 대량의 본문과 요약문 쌍의 데이터 셋이 필요하다. 하지만 실제 사람이 요약을 수행하여 데이터를 구축하는 데에는 상당한 시간과 비용이 소모된다. 본 논문에서는 [11]에서와 같이 네이버 지식iN의 질문 데이터에서 제목을 본문을 대표하는 요약 데이터로 간주하였다. 이때 네이버 지식iN의 질문 데이터에서 학습 데이터 필터링 알고리즘은

동일하게 적용하였다.

또다른 학습 데이터로 네이버 뉴스 요약봇 품질 개선용 데이터 셋을 활용하였다. 뉴스 기사의 경우는 본문의 첫 문장이 요약문인 경우가 많기 때문에 뉴스의 첫 문장을 요약문으로 간주하되, 본문의 첫 문장을 제외한 내용과 제목을 입력문으로 정의하였다. 그리고 이때 뉴스의 첫 문장이 기사의 내용을 대표하지 못하는 경우의 데이터는 노이즈가 되기 때문에 필터링하였다. 보통 뉴스의 제목은 본문을 대표하고 요약한다고 볼 수 있기 때문에 이들을 본문의 첫 문장인 요약문과 ROUGE (Recall-Oriented Understudy for Gisting Evaluation)[12] 점수 계산을 하여 일정 임계치 이상의 데이터만 필터링하여 사용하였다. ROUGE는 참조 문장과 후보 문장 간의 재현율을 바탕으로 평가한다. 따라서 본 논문에서는 뉴스 제목과 뉴스 본문의 첫 문장 사이의 ROUGE-1, ROUGE-2, ROUGE-L 세 점수 평균값을 구했을 때 0.2보다 작은 것은 요약문으로 간주할 수 없는 노이즈로 판단하고 학습 데이터에서 제외하였다.

3.2. Transformer model

본 논문에서는 Pointer-Generator Network의 Bi-LSTM 모델을 Transformer 모델로 대체하여 사용한다. Transformer는 순환 신경망보다 긴 시퀀스에 대해 강건하고, 대량의 학습데이터에 대한 모델 capacity가 큰 것이 장점으로 순환 신경망을 대체할 수 있는 모델이다. 최근 많은 자연어 처리 분야에서 최고 수준을 보여주고 있는 BERT, RoBERTa, ERNIE 2.0 등의 대용량 사전 학습 언어 모델들 역시 Transformer에 기반하고 있다.

Transformer 구조는 기본적으로 기존 Sequence-to-Sequence 모델과 같이 인코더-디코더 구조를 가지고 있다. 인코더는 N개의 동일한 계층이 반복되는 구조로 되어있고, 디코더도 인코더와 마찬가지로 동일한 계층이 N개 반복되는 형태이다. 인코더에서 각 계층은 두개의 sub-layer로 구성된다. 첫 sub-layer는 Multi-head self-attention이고 두번째 sub-layer는 Point-wise Feed-forward Network이다. 디코더는 총 3개의 sub-layer로 구성되어 있는데, 2개는 인코더의 sub-layer와 동일하고 나머지 하나는 인코더의 출력에 대해 Multi-head attention을 계산하는 Encoder-Decoder Multi-head attention layer가 추가된다. 여기서 디코더는 인코더와 달리 순차적으로 결과를 만들어내야 하기 때문에 자기 집중(Self-Attention) 시 현재 타임스텝보다 뒤에 있는 단어는 자기 집중을 못하도록 masking을 한다. 그리고 모델 전체적으로 각 sub-layer에 residual connection을 사용했다. Transformer 모델에서 핵심이 되는 Multi-head self-attention은 식 (1)과 같이 계산한다.

$$\text{Multihead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_n)W^o \quad (1)$$

$$\text{where } \text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$

식 (1)에서 Q는 query, K는 key, V는 value 이며, $head_i$ 는 Q, K, V의 벡터를 n개로 나눈 후 각각의 가중치를 곱하여 Scaled-Dot Product를 통해 자기 집중을 진행한다. 인코더와 디코더에서 Multi-head self-attention layer의 Q, K, V는 같은 입력 시퀀스이며, Encoder-Decoder Multi-head attention layer의 K와 V는 인코더에서 오는 시퀀스, Q는 이전 디코더의 마지막 계층에서 오는 시퀀스이다. 자기 집중은 서로 다른 문장 간의 관계를 계산하는 양방향 상호 집중과 달리, 같은 문장을 집중하여 문장에 나타나는 단어와 같은 문장에 나타나는 다른 단어와의 관계를 계산하게 된다. Q, K, V에 대한 자기 집중은 아래의 식 (2)와 같이 표현된다.

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \quad (2)$$

인코더의 Multi-head self-attention의 결과와 디코더의 최종 Multi-head self-attention 결과는 각각 식 (3)의 Position-wise Feed-forward Network를 통과한다.

$$FFN(x) = max(0, xW_1 + b_1)W_2 + b_2 \quad (3)$$

식 (3)과 같이 Feed-forward Network는 두 개의 Linear Transformations로 구성되어 있으며, 그 사이에 ReLU 활성화 함수를 적용했다. 그리고 아래의 식 (4)와 같이 입력 단어 시퀀스 $\{x_1, x_2, \dots, x_m\}$ 에 대한 인코더의 출력 h 와 출력 단어 전 시퀀스 $y_{<t} = \{y_1, y_2, \dots, y_{t-1}\}$ 가 디코더의 입력으로 들어가서 단어의 분포 $p_t^{vocab}(y)$ 가 계산된다.

$$h = f_{enc}(x_{1, \dots, m})$$

$$p_t^{vocab}(y) = softmax(V'(f_{dec}(y_{<t}, h) + b)) \quad (4)$$

식 (4)에서 V, b는 각각 Transformer의 마지막 선형 층의 가중치와 편향 값이다. 생성 요약에서 모델의 손실 함수는 Negative log-likelihood를 사용하며 y_t^* 는 현재 t 시간의 정답 단어일 때, 식 (5)로 정의된다.

$$Loss = \sum_{t=1}^{|y|} -logP(y = y_t^* | y_{<t}, h) \quad (5)$$

3.3 Copy-Transformer model

본 논문에서 사용한 복사-메커니즘[13]은 일반적으로 문서 요약에서 디코딩 과정에서 요약문을 생성할 때 필요한 어휘가 출력 사전에 없어 발생하는 OOV (Out-of-Vocabulary) 문제를 해결하기 위해 제안된 방법으로, 출력에 필요한 어휘를 입력 시퀀스에서 찾아 출력에 복사하는 방법이다. Copy-Transformer 모델은 Transformer 모델에 복사-메커니즘을 적용한 모델이다. 그림 1은 본 논문에서 사용하는 Copy-Transformer 모델의 구조를 나타

낸다.

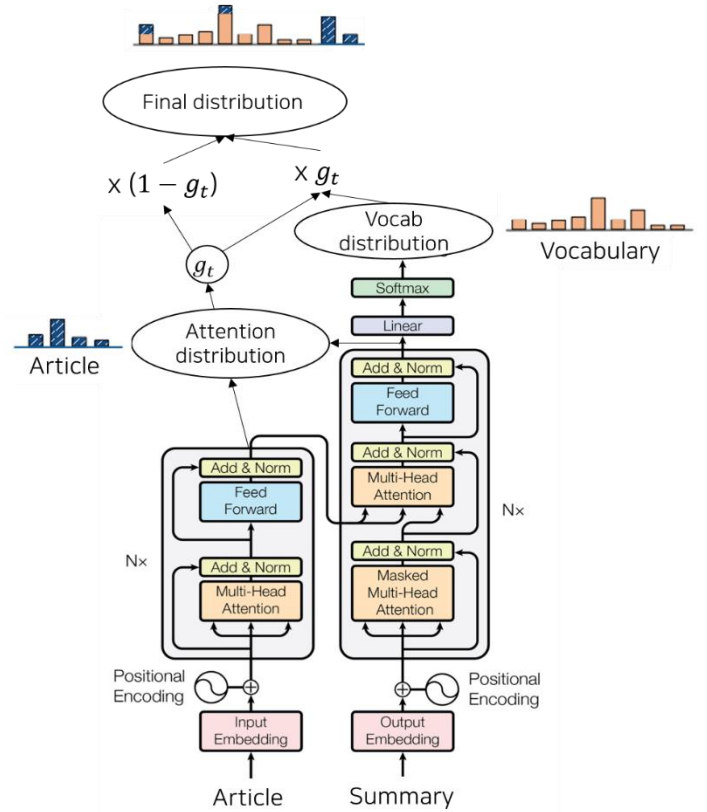


그림 1. Copy-Transformer 모델의 구조

그림 1의 주의 분포는 요약문을 한 단어씩 디코딩 할 때 본문의 어느 부분에 주의를 해야하는지를 구하는 확률 분포이고, 단어 분포는 식 (4)에서 구한 사전 단어 분포 값이다. 주의 분포를 구하기 위해서는 디코더에서 타임 스텝 t일 때, 디코더 출력 o_t 와 인코더 출력 h 로부터 식 (6)과 같이 주의 확률 a_t^i 을 계산한다.

$$u_t^i = o_t W_g h_i$$

$$a_t^i = \frac{\exp(u_t^i)}{\sum_{k=1}^N \exp(u_t^k)} \quad (6)$$

복사 게이트인 생성 확률 $g_t \in [0,1]$ 는 디코더의 출력과 인코더의 출력이 선형 계층을 통과하여 아래의 식 (7)과 같이 계산한다.

$$g_t = \sigma(W_g \cdot [o_t, h] + b_g) \quad (7)$$

식 (7)에서 σ 는 sigmoid 함수이며, W_g 와 b_g 는 선형 계층의 학습 파라미터이다. 이렇게 계산된 생성 확률을 이용해 본문에서 단어를 $(1-g_t)$ 의 확률로 그대로 인용하거나 g_t 의 확률로 단어 분포를 참고하여 새로운 단어를 생성할 수 있다. 이렇게 계산된 g_t 와 단어 분포 $p_t^{vocab}(y)$, 주의 분포 a_t^i 를 이용해 식(8)와 같이 최종적인 단어 분포 (Final Distribution)인 $P_t(y)$ 를 구한다.

$$P_t(y) = g_t P_t^{vocab}(y) + (1 - g_t) \sum_{i:w_i=w} a_t^i \quad (8)$$

따라서 최종적으로 계산된 $P_t(y)$ 에 따라 가장 높은 확률을 가진 단어가 모델에서 다음 출력 단어로 정해진다.

3.4. 추론 단계 페널티

본 논문의 모델에서는 추론 단계에서 Coverage 페널티, Length 페널티, n-gram 반복 제어 알고리즘을 수행한다. 생성 요약 모델들에서 자주 발생하는 문제 중 하나인 단어 반복을 피하기 위해서 기존의 Pointer-Generator Network는 학습 단계에서 커버리지-메커니즘을 사용한다. 하지만 커버리지-메커니즘은 손실 함수에 반영되는 요소이기 때문에 완벽하게 단어 반복 제어가 반영되지 못하는 부분이 존재한다. 따라서 본 논문에서는 커버리지-메커니즘을 사용하지 않는 대신, 추론 단계에서 더 명확하게 단어 반복 문제를 처리하기 위해 [10]에서 사용한 Coverage 페널티와 Length 페널티를 사용한다.

또한 추론 단계에서 디코더의 출력은 Beam search 알고리즘에 의해 top-k개의 beam이 생성되는데, 이때 n-gram 반복을 방지하기 위해 이전 타임 스텝의 출력 단어들을 참고하여 동일한 n-gram을 가진 빔은 $P_t(y) = 0$ 으로 설정함으로써 n-gram 반복 제어 알고리즘이 수행된다. 추론 시 Coverage 페널티, Length 페널티, n-gram 반복 제어 알고리즘은 요약 생성을 수행할 때 기존 요약 모델의 복잡도를 높이지 않고도 단어 반복을 직접적으로 제어할 수 있으므로, Pointer-Generator와 Copy-Transformer 모델에 모두 적용하였다.

4. 실험 및 평가

4.1. 실험 준비

본 논문에서는 입력 문서와 요약문의 길이가 짧은 데이터와 긴 데이터, 구어체와 문어체에 대한 다양한 실험을 통해 제안하는 모델의 우수성을 검증하였다. 길이가 상대적으로 짧고 구어체인 데이터는 [11]에서 사용한 cQA 데이터로써 지식인 질문 생성 요약을 위한 데이터이다. 길이가 상대적으로 길고 문어체인 데이터는 뉴스 기사 데이터로 실험을 위해 3.1에서 제시한 데이터 필터링 과정을 거쳐 데이터 셋을 구축하였다. 입력 문서와 요약문은 형태소 분석 결과 단위로 구성하였으며, 단어 사전 크기는 50,000을 사용하였다. 아래 표 1과 같이 두 데이터에 대해 데이터 셋의 크기와 형태소 분석 후의 데이터 길이를 분석하였다. 데이터 길이에 따라 모델에서 인코더의 입력 최대 길이와 디코더의 출력 최대 길이 파라미터를 조정하였고, 모델의 다른 하이퍼파라미터는 동일하게 사용하였다.

표 1. 데이터 별 기본 통계

	지식인 질문 데이터	뉴스 데이터
--	------------	--------

학습 셋 개수	94만	280만
개발 셋 개수	12만	5만
평가 셋 개수	12만	5만
본문 평균 길이	91	520
요약문 평균 길이	9	38

4.2. 실험 평가

Copy-Transformer 모델의 하이퍼파라미터는 다음과 같다. Transformer 블록 계층 수는 6, 은닉 크기는 512, self-attention heads 수는 8, feed-forward의 필터 크기는 2048를 사용하였으며, 총 파라미터의 수는 96,181,589개이다. 학습에 사용된 optimizer는 Adam을 사용했고, Adam optimizer에서 파라미터로는 $\beta_1 = 0.9$, $\beta_2 = 0.98$, $\epsilon = 10^{-9}$ 를 사용했다. 학습률(learning rate)의 경우 학습 경과에 따라서 아래의 공식으로 학습률을 계산하여 변화하도록 만들었다.

$$lr = d_{model}^{-0.5} \cdot \min(\text{step_num}^{-0.5}, \text{step_num} \cdot \text{warmup_steps}^{-1.5}) \quad (9)$$

식 (9)에서 d_{model} 은 은닉 크기이고, $\text{warmup_steps} = 10000$, 초기학습률은 1.0을 사용했다. 또한 학습 시의 batch 크기는 64, dropout 값과 label smoothing 값은 모두 0.1로 설정했다. 최종 추론 단계에서 디코더의 Beam size는 4로 사용하였다.

그림 2는 지식인 질문 데이터에서 모델 별 요약 성능을 비교한 그래프이다. TextRank[14], Seq2Seq+attn, Pointer-Generator의 성능은 [11]을 참고하였고, 본 논문에서 제안한 PG-IP(Pointer-Generator Inference Penalty)와 Copy-Transformer의 ROUGE-F1 성능을 정리하였다. PG-IP는 Pointer-Generator 모델에서 본 논문의 추론 단계 후처리 및 Beam search 디코딩을 적용한 모델이다.

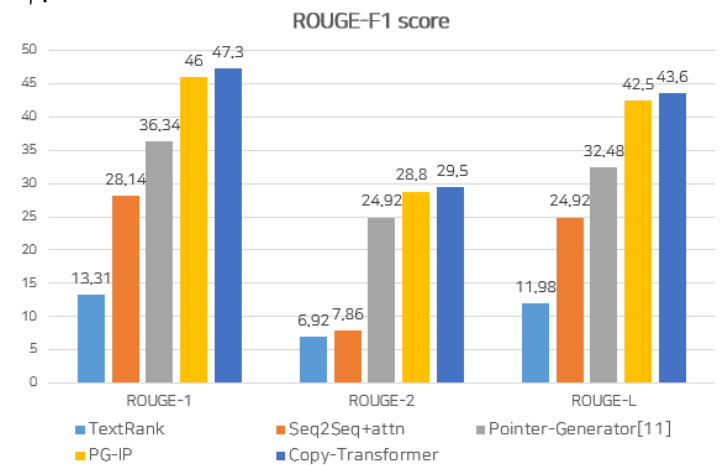


그림 2. 지식인 질문 데이터의 모델 별 요약 성능

그림 2에서 Pointer-Generator와 PG-IP 사이에 큰 성능 향상이 있는 것으로 보아 추론 단계의 페널티 및 Beam search 디코딩의 영향이 큰 것을 알 수 있다. 또한

Copy-Transformer 모델이 가장 우수한 성능을 보여주며, ROUGE-1 기준으로 기존 [11]에서 제안한 Pointer-Generator 모델 대비 약 26%의 성능이 향상되었고, PG-IP 모델 대비 약 3%의 성능 향상을 보였다.

표2는 뉴스 데이터에 대한 본 논문의 모델 별 생성 요약 성능을 ROUGE-F1 점수로 비교한 것이다. 본 논문에서 제안한 PG-IP와 Copy-Transformer 모델 모두 높은 ROUGE 점수를 보여주지만 Copy-Transformer는 대체로 좀더 정확한 문장을 생성하는 경향을 확인할 수 있었고, 따라서 ROUGE-1은 4.21, ROUGE-2는 3.51, ROUGE-L은 4.04만큼 더 높은 성능을 보여준다. 표 1과 같이 뉴스 데이터는 지식iN 질문 데이터 보다 본문 길이가 훨씬 긴 데이터이다. 지식iN 질문 데이터에서는 ROUGE-1 기준 1.3의 성능 향상을 보여준 반면 뉴스 데이터에서는 ROUGE-1 기준 4.21의 성능 향상을 보여준다. 이를 통해 Bi-LSTM 기반의 Pointer-Generator 모델보다 Transformer 기반의 Copy-Transformer 모델이 긴 시퀀스 데이터에 대해 생성 요약을 보다 잘 수행한다는 것을 확인하였다.

모델	ROUGE-1	ROUGE-2	ROUGE-L
PG-IP	44.15	23.47	37.08
Copy-Transformer	48.36	26.98	41.12

표 3과 4는 각각 지식iN 질문 데이터에서 모델 별 샘플 비교 결과와 뉴스 데이터에서 모델 별 샘플 비교 결과이다. 표 3에서 제목이 실제 사용자가 작성한 정답인데, PG-IP 모델과 Copy-Transformer 모델 모두 실제 제목과 비슷하게 본문을 잘 요약한 문장을 생성하였다. 하지만 PG-IP 보다 Copy-Transformer가 '헤라클레스 장수 풍뎡이' 라는 더 구체적이고 정확한 문장을 생성하였다. 표 4에서는 PG-IP 모델과 Copy-Transformer 모델 모두 정답 요약에서의 '지난달'을 '2월'로 생성하여 보다 더 구체적인 요약문을 완성하였지만, PG-IP 모델은 '수출 증가율이 3년만에 최저치'라는 본문과 반대되는 내용을 생성한 반면 Copy-Transformer 모델은 정확한 요약문을 생성하였다.

표 2 뉴스 데이터의 모델 별 요약 성능

표 3 모델의 지식iN 질문 요약 생성 예

입력 문서	제가 지금 헤라클레스 장수 풍뎡이나 다른 외국 곤충을 기르고 싶어서 이렇게 올리는데요 제발 좀 알려 주세요 매일 매일 파는 곳 찾아보느라 인터넷 엄청 뒤졌어요 헤라클레스 장수 풍뎡이 파는 곳 좀 알려 주세요
제목	헤라클레스 장수 풍뎡이 파는 곳
PG-IP	장수 풍뎡이 파는 곳 좀 알려 주세요
Copy-Transformer	헤라클레스 장수 풍뎡이 파는 곳 좀 알려 주세요

표 4 모델의 뉴스 요약 생성 예

입력 문서 (제목 + 본문)	중국 2월 수출 44.5 % 급증 ... 3년 만에 최대폭 증가 8일 블룸버그 통신에 따르면 2월 중국의 달러화 기준 수출은 전년 동기 대비 44.5 % 급증했다. 수출 증가율이 2015년 2월 48.2 % 이후 최대폭을 기록했다. 시장 전문가 예상치 11 %의 4배를 웃도는 수준으로, 전월 증가율 11.2 %보다도 크게 높았다. 지난달 수입은 작년 같은 기간보다 6.3 % 증가했다. 예상치 8 %를 밑돌았지만 전월치인 36.8 %보다는 크게 줄어든 수치다. 이에 따라 지난달 무역 흑자 규모는 337 억 4천만달러로 전월치 203 억 5천만달러를 상회했다. 블룸버그가 집계한 시장 전망치 57 억달러 적자와는 큰 차이를 보였다. 위안화 기준 무역 흑자도 2천 248 억 8천만위안으로 전월치 1천 373 억 3천만위안보다 많았다. 예상치는 689 억위안 적자였다. 위안화 기준 수출은 전년 같은 기간 대비 36.2 % 증가했다. 지난달 중국의 수출이 예상보다 많이 증가한 것은 기저 효과와 위안화 약세가 맞물린 데 따른 것으로 분석된다. 기저 효과란 기준 시점과 비교 시점의 상대적인 수치에 따라 그 결과가 크게 차이 나는 것으로, 작년 2월 수출 증가율이 많이 낮은 탓에 지난달 수출 증가율이 두드러졌다는 의미다. 블룸버그는 작년 같은 기간의 낮은 수치에 위안화 약세까지 겹치면서 수출 증가율이 커졌지만 미국과의 무역 전쟁 발발 시 다시 줄어들 가능성이 크다고 전했다.
정답 요약	지난달 중국 수출이 위안화 약세 등 영향으로 3년 만에 최대폭으로 늘었다.
PG-IP	중국의 2월 수출 증가율이 3년 만에 최저치를 기록한 것으로 나타났다.
Copy-Transformer	중국의 2월 수출 증가율이 3년 만에 최대폭으로 증가한 것으로 나타났다.

5. 결론

본 논문에서는 생성 요약 모델링을 위한 대량의 학습 데이터를 구축하고 복사-메커니즘과 추론 단계 페널티를 이용한 Copy-Transformer 기반 생성 요약 모델을 제안하였다. 제안한 모델을 평가하기 위해 한국어 요약 데이터셋인 지식iN 질문 데이터와 뉴스 기사 데이터를 이용하여 성능을 비교하였다. 실험 결과, 기존의 Pointer-Generator Network 보다 정확한 요약문을 생성하였고 특히 긴 입력 데이터에 대해서는 더 큰 성능 향상을 확인할 수 있었다.

향후 연구로는 생성 요약의 결과를 정밀하게 평가할 수 있는 성능 지표를 구하고자 한다. 또한 대량의 데이터로 학습된 자연어 생성용 대용량 사전 학습 언어 모델을 활용한다면 더 자연스러운 문장을 생성할 수 있을 것으로 기대되고, 생성 요약을 위한 다른 도메인의 학습 데이터가 부족한 경우에도 좋은 성능을 보여줄 수 있을 것이다.

참고문헌

- [1] J. Clarke and M. Lapata. Discourse constraints for document compression. *Computational Linguistics*, 36(3):411-441, 2010.
- [2] R. Nallapati, B. Zhou, C. Nogueira dos santos, C. Gulcehre and B. Xiang. Abstractive text summarization using sequence-to-sequence rnns and beyond. In *CoNLL*, 2016.
- [3] A. M. Rush, S. Chopra and J. Weston. A neural attention model for abstractive sentence summarization. *CoRR*, abs/1509.00685, 2015.
- [4] A. See, P. J. Liu and C. D. Manning. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073-1083. 2017.
- [5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 6000-6010, 2017.
- [6] <http://kin.naver.com>
- [7] R. Paulus, C. Xiong, R. Socher, and P. Alto. A deep reinforced model for abstractive summarization. *ICLR*, pages 1-13, 2018.
- [8] A. Celikyilmaz, A. Bosselut, X. He, and Y. Choi. Deep Communicating Agents for Abstractive Summarization. *arXiv preprint arXiv:1803.10357*, 2018.
- [9] W. Hsu, C. Lin, M. Lee, K. Min, J. Tang, and M. Sun. A unified model for extractive and abstractive summarization using inconsistency loss. *arXiv preprint arXiv:1805.06266*, 2018.
- [10] S. Gehrmann, Y. Deng, and A. Rush. 2018. Bottom-up abstractive summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4098-4109, 2018.
- [11] 김원우, 김선훈, 장현석, 강인호, 박광현. Pointer-Generator Networks를 이용한 cQA 시스템 질문 요약. 제30회 한글 및 한국어 정보처리 학술대회 논문집, pp. 126-131, 2017.
- [12] Lin, Chin-Yew. ROUGE: a Package for Automatic Evaluation of Summaries. *Proc. of the Workshop on Text Summarization Branches Out (WAS 2004)*, 2004.
- [13] J. Gu, Z. Lu, H. Li, and Victor O. K. Li. Incorporating Copying Mechanism in Sequence-to-Sequence Learning. In *ACL*, 2016.
- [14] R. Mihalcea and P. Tarau. TexRank: Bringing Order into Texts. In *Proceedings of EMNLP*, 85:404-411, 2004.