

# 생성 요약 향상을 위한 가중치 문맥 벡터 기반 BART 모델

조성국\*, 온병원\*\*

## Weighted Context Vector-based BART Model for Improving Abstractive Summarization

Sung-Guk Jo\*, Byung-Won On\*\*

---

이 논문은 2019년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임  
(NRF-2019R1F1A1060752)

---

### 요 약

생성 요약은 원본 텍스트의 중요한 정보만을 포함한 짧은 텍스트이다. 본 논문의 제안 방안은 먼저 주어진 원본 텍스트의 문장들을 추출기를 통해 중요한 문장과 그렇지않은 문장으로 분류한다. 그런 다음, BART의 파인 튜닝 과정에서 중요한 문장에 해당하는 인코더 문맥 벡터의 크기를 조절한다. 그렇게 함으로써, 중요한 문장과 그렇지 않은 문장이라는 두 그룹 사이의 마진을 늘려 클러스터링 효과를 줄 수 있다. 결과적으로 제안 방안인 수정된 BART 모델은 이와 같은 방식으로 클러스터링 패턴을 학습하여 기존의 파인 튜닝 방식보다 높은 성능을 보여줄 수 있다. 실험 결과, 제안 방안으로 파인 튜닝한 BART가 ROUGES score에서 최대 3%, BERT score에서 0.2% 향상된 성능을 보인다.

### Abstract

An abstractive summary is a short text containing only salient information from the original text. In our work, given an original text, the Extractor first classifies whether each sentence is salient or not. Then, in fine-tuning BART, by a weight, scaling the context vectors of the encoder corresponding to the salient sentences can give the effect of clustering by increasing the margin between two clusters - One contains salient sentences and the other non-salient ones. Consequently, our modified BART model can show high performance by learning these clustering patterns. Our experimental results show that BART using the proposed method improves up to 3% in ROUGE score and 0.2% in BERT score.

### Keywords

abstractive summarization, sequence-to-sequence, pre-trained model, weighted context vector

---

\* 군산대학교 소프트웨어융합공학과 학부생  
- ORCID: <http://orcid.org/0000-0002-7389-9109>  
\*\* 군산대학교 소프트웨어융합공학과 교수(교신저자)  
- ORCID: <http://orcid.org/0000-0001-6929-1388>

• Received: Sep. 30, 2021, Revised: Oct. 19, 2021, Accepted: Oct. 22, 2021  
• Corresponding Author: Byung-Won On  
Dept. of Software Convergence Engineering, Kunsan National University, 558,  
Daehak-ro, Gunsan, Jeollabuk-do, Korea  
Tel.: +82-63-469-8913, Email: [bwon@kunsan.ac.kr](mailto:bwon@kunsan.ac.kr)

## 1. 서론

인터넷과 소셜미디어의 발전으로 매일 텍스트 데이터가 폭발적으로 생성되고 있으며, 이를 효율적으로 활용하기 위한 연구가 활발히 진행되고 있다. 그 중 자동 요약 시스템은 정보 검색의 효율성을 증진시킬 수 있는 연구이다. 예를 들어, 짧은 키워드를 통해 검색된 문서가 그에 대한 요약문을 제공한다면 사람들이 원하는 정보를 찾는 데 걸리는 시간을 크게 단축할 수 있을 것이다. 또한, 자동 요약 시스템은 문서 검색 이외에도 온라인 서점의 상품이나 뉴스 기사 등의 짧은 설명이 필요한 분야에 활용될 수 있다[1].

자동 텍스트 요약은 크게 추출 요약(Extractive summarization)과 생성(Abstractive) 요약으로 나눌 수 있다. 추출 요약은 원본 텍스트에서 중요한 부분을 그대로 가져와 조합하여 요약으로 사용하는 방법이다. 반면에, 생성 요약은 원본 텍스트를 그대로 가져다 쓰는 게 아니라 중요한 정보를 포함하는 짧은 텍스트로 재생성하는 방법이다. 원본 텍스트의 의미를 이해해야 할 뿐만 아니라 문장의 자연스러운 생성까지 고려해야 하는 생성 요약은 추출 요약보다 상대적으로 어려운 작업이기에 과거에는 통계 기반의 추출 요약에 관한 연구가 주를 이루었다. 하지만 딥러닝 기술의 발전으로 인공지능망을 이용한 생성 요약에 대한 도전이 시작되었고, RNN(Recurrent Neural Network) 기반의 시퀀스 투 시퀀스(Sequence-to-Sequence) 아키텍처를 활용한 연구[2]에서 출발하여, 현재는 트랜스포머 기반의 사전학습 모델을 이용한 연구가 활발히 이루어지고 있다.

자연어처리 분야에서 BERT(Bidirectional Encoder Representations from Transformers)[3]와 GPT(Generative Pre-trained Transformer)[4]의 등장 이후 트랜스포머 기반의 사전학습 모델을 이용한 연구가 주류를 이루고 있다. 요약 분야에서 BERT와 같은 인코더 단독모델은 원본 텍스트의 일부 문장을 선택하는 추출 요약에서 우수한 성능을 보이지만, 텍스트를 생성해야 하는 생성 요약에는 적합하지 않다. 반면, GPT와 같은 디코더 단독모델은 텍스트 생성에는 적합하지만, 학습 시 단일방향의 정보만을 이용하는 단점이 있다. 이러한 이유로 생성 요약에 관한 연구

는 인코더와 디코더를 활용한 사전학습된 시퀀스 투 시퀀스 모델을 사용한다.

생성 요약을 위한 시퀀스 투 시퀀스 모델의 파인 튜닝과정은 다음과 같다. 인코더(Encoder)에 원본 텍스트를 입력하고 디코더(Decoder)에 요약문을 입력한다. 인코더는 입력받은 원본 텍스트에 대한 정보를 인코딩하여 디코더로 전달한다. 디코더에서는 인코더로부터 전달받은 정보를 이용하여 언어모델의 목적함수를 이용해 학습한다. 요약문은 원본 텍스트의 중요한 정보를 중심으로 생성된 상대적으로 짧은 텍스트이다. 그러기 때문에 요약문을 생성하는데 원본 텍스트의 모든 정보가 필요한 것이 아니며, 이때 필요하지 않은 원본 텍스트의 일부를 노이즈(Noise)로 볼 수 있다. 이 노이즈를 학습 과정에서 어떻게 처리하느냐가 생성 요약 향상을 데에 있어 중요한 관점이 될 수 있다.

본 논문에서는 위 절에서 설명하고 있는 생성 요약을 위한 시퀀스 투 시퀀스 모델의 학습 과정에서 발생하는 특징에 주목하여, 생성 요약 향상을 위한 새로운 학습 방안을 제안한다. 그림 1은 본 연구의 제안 방안을 도식화한 것이다. 먼저 문장  $s_1, s_2, s_3, s_4, s_5$ 로 구성된 원본 텍스트를 Extractor에 입력한다. Extractor는 원본 텍스트를 중요한(Salient) 문장과 중요하지 않은(Unimportant) 문장으로 구분한다. 다음으로 Extractor를 통해 얻은 문장 구분 정보를 원본 텍스트, 정답 요약문과 함께 Summarizer에 입력한다. Summarizer는 인코더-디코더 모델이며 인코더는 원본 텍스트를 인코딩하여 각 문장에 대응하는 벡터  $V=[v_1, v_2, v_3, v_4, v_5]$ 로 변환한다. 그리고 중요한 문장  $s_1, s_3$ 에 대응되는 벡터  $v_1, v_3$ 에 가중치  $w$ 를 곱한다. 이를 통해 중요한 문장과 중요하지 않은 문장이 구별된 인코딩 정보  $w \cdot V$ 를 디코더에 전달하게 된다. 디코더에서는 이를 이용하여 정답 요약문을 예측하는 학습을 진행한다.

본 연구에서는 Extractor로 BERTSUM[5]기반의 추출 요약 모델을 사용하고 Summarizer로 사전학습된 BART[6]를 사용한다. 실험 데이터로는 생성 요약 벤치마크 데이터 세트인 XSum(Extreme Summarization)[7]을 사용하고 평가를 위한 지표로는 ROUGE score와 BERT score[8]를 사용한다.

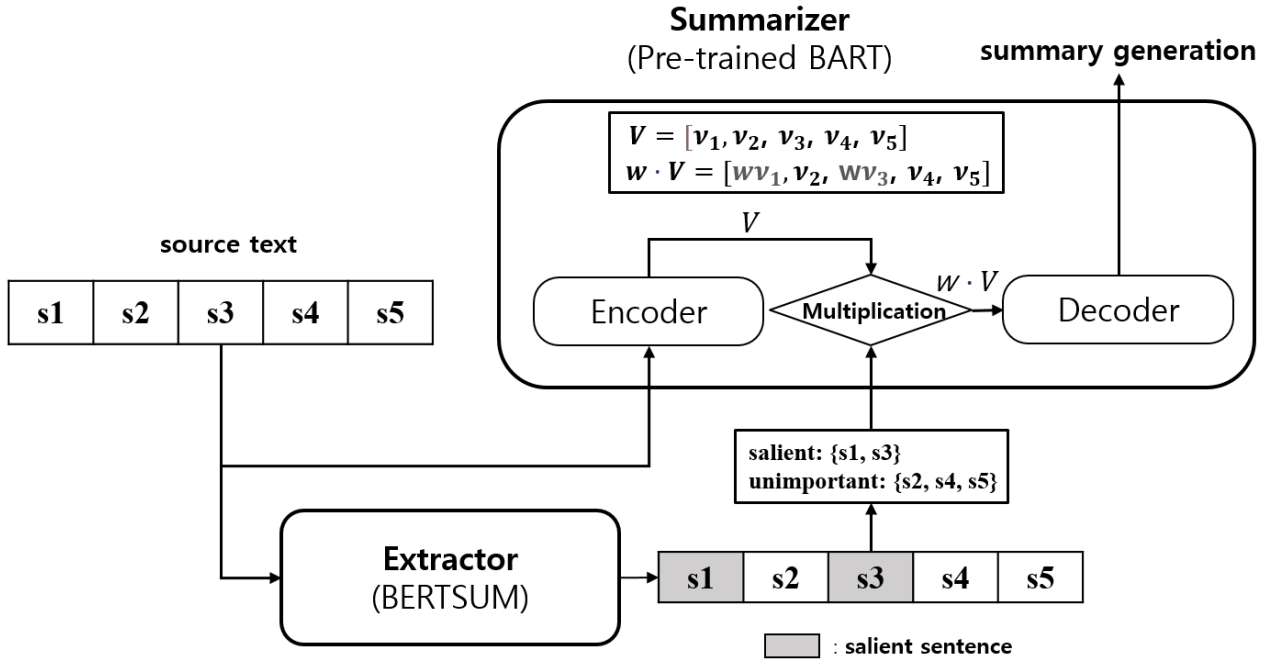


그림 1. 가중치 문맥 벡터 기반 BART 모델  
Fig. 1. Weighted context vector-based BART model

요약 모델의 성능 평가에 주로 사용되는 ROUGE score는 어휘 유사도 관점의 측정 방법이므로 의미 유사도의 평가를 위해 BERT score를 추가하였다. 사전학습된 BART를 기존의 방식대로 파인 튜닝한 모델을 기존 방안이라 하고 제안 방안의 모델을 최적으로 학습시킨 결과와 비교했을 때, 제안 방안이 ROUGE score와 BERT score에서 모두 기존 방안보다 높은 성능을 보였다.

본 연구의 기여도는 다음과 같다.

- 생성 요약 향상을 위해 인코딩 정보에서 중요한 정보와 중요하지 않은 정보를 구분하기 위해 문맥 벡터에 가중치를 곱한 가중치 문맥 벡터 기반의 BART 모델을 제안한다.
- 본 연구를 통해 제안 방안인 가중치 곱셈을 통한 노이즈 클러스터링(Clustering) 방식의 디노이징(Denoising)기법의 성능을 입증하고, 이를 노이즈 제거가 필요한 다양한 자연어처리 분야에 적용할 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 BART의 사전학습에 관한 자세한 설명을 통해 생성 요약에서 BART가 우수한 성능을 보이는 이유에 관해

기술한다. 3장에서는 본 논문에서 제안하는 방안에 대하여 자세하게 기술한다. 4장에서는 본 논문의 실험 환경과 결과를 기술하고, 마지막 5장에서 결론 및 향후 연구 방향에 관해 기술한다.

## II. 관련 연구

[6]은 디노이징 오토인코더(Denoising autoEncoder) 방식으로 사전학습된 시퀀스 투 시퀀스 모델이다. 그림 2는 BART의 구조와 사전학습 방법을 보여준다. BART는 인코더-디코더구조로 인코더로 기존의 BERT와 같은 양방향 트랜스포머 인코더를 사용하고 디코더로는 GPT와 같은 단방향 트랜스포머 디코더를 사용한다.

BART의 사전학습 과정은 다음과 같다. 먼저 인코더의 원본 텍스트를 여러 노이즈 방법을 적용하여 변형시킨 뒤 입력하고, 디코더에는 원본 텍스트를 입력한다. 인코더는 노이즈가 포함된 원본 텍스트의 정보를 디코더로 전달하고 디코더는 원본 텍스트를 예측하는 학습을 진행한다. 이를 통해 BART 모델은 손상된 텍스트를 원본 텍스트로 재구성하는 방법을 배우게 된다.

BART의 텍스트 변형을 위한 노이즈 방법은 다음과 같다.

Token Masking 입력 토큰들에서 임의로 샘플링된 토큰들을 [MASK]로 대체하고 [MASK] 토큰이 원래 무엇이었던지 예측한다.

Token Deletion 입력 토큰들에서 임의로 토큰들을 제거한다. token masking 방식과 다르게 제거된 토큰이 무엇이었던지와 함께 토큰의 위치를 찾아야 한다.

Text Infilling 입력 토큰들에서 span length에 해당하는 토큰들을 하나의 [MASK] 토큰으로 대체하고 [MASK]로 대체된 토큰들의 개수를 예측한다.

Sentence Permutation 문장의 순서를 임의로 섞어 원래 문장 순서를 찾으려 한다.

Document Rotation 하나의 토큰을 뽑아, 그 토큰을 시작점으로 회전시켜 모델이 원래 시작 위치를 찾으려 한다.

생성 요약은 원본 텍스트에서 중요하지 않은 부분을 제거하고 단어를 바꾸는 등의 텍스트 조작을 통해 요약문을 생성한다. 즉, 생성 요약은 노이즈가 포함된 시퀀스 데이터를 노이즈 없이 짧은 시퀀스 데이터로 변환하는 작업이라고 볼 수 있다.

이는 위에서 설명한 BART의 디노이징 사전학습 방식과 밀접한 관련이 있다. 즉, BART가 다양한 방식으로 텍스트를 재구성하는 방법을 배운 사전학습 모델이기 때문에 생성 요약에서 우수한 성능을 내는 것으로 볼 수 있다. 생성 요약을 위한 BART 모델은 파인 튜닝과정에서 원본 텍스트의 중요 정보를 찾아내고 재구성하여 요약문을 생성하는 방법을 학습해야 한다. 하지만 원본 텍스트를 인코딩한 정보에는 노이즈가 많아 중요한 정보를 찾기가 어렵다.

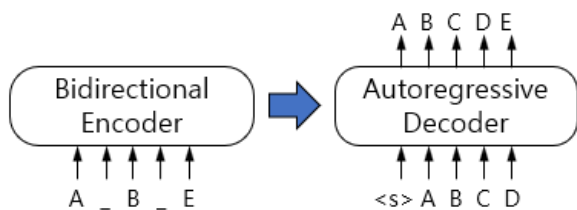


그림 2. BART의 사전학습[6]  
Fig. 2. Pre-training of BART[6]

### III. 제안 방안

본 논문의 제안 방안은 요약문을 생성하는 데에 있어, 원본 텍스트를 구성하는 문장들이 가지고 있는 정보의 중요도가 다르다는 점에 주목한다. 원본 텍스트는 중요한 문장과 중요하지 않은 문장으로 나눌 수 있으며, 생성 요약 모델이 중요하지 않은 문장보다 중요한 문장에 집중하여 요약문을 생성하는 방법을 학습할 수 있다면 더 좋은 요약문을 생성할 수 있을 것이다.

그림 3은 제안 방안에 대한 기본적인 개념을 보여준다. 원본 텍스트 전체를 한 번에 입력받는 인코더의 특성상 인코더로부터 출력된 문맥 벡터들은 벡터 공간상에서 비슷한 위치에 존재하게 된다. 여기서 오른쪽 그림과 같이 문맥 벡터에 가중치를 곱해 중요한 문장과 중요하지 않은 문장이 벡터 공간상에서 구분되게 된다.

중요한 문장은 추출 요약 모델인 BERTSUM을 이용하여 선정한다. 그리고 구분된 중요한 문장과 중요하지 않은 문장의 원본 텍스트에서 위치 정보를 BART로 전달한다. BART는 이를 이용하여 인코더의 출력인 문맥 벡터들에 가중치를 곱해준다.

본 논문에서 실험을 위해 XSum 벤치마크 데이터 세트를 이용한다. XSum은 “기사가 무엇에 관한 것인가?”라는 질문에 답하기 위한 한 문장 요약 데이터 세트이며 모든 요약은 일반적으로 문서 작성자에 의해 전문적으로 작성되었다. 표 1은 Xsum 데이터의 예시이다. 요약문에서 문서와 겹치는 단어를 찾기 힘들 정도로 문장을 추상적으로 재생성한 것을 확인할 수 있다.

#### 3.1 추출 요약 모델을 이용한 중요 문장 선정

BERTSUM은 사전학습된 BERT를 기반으로 한 요약 모델이다. 그림 4는 기존 BERT 위에 분류를 위한 2개의 계층으로 구성된 트랜스포머 인코더를 추가한 BERTSUM의 구조를 보여준다. BERTSUM은 입력 텍스트를 문장 단위로 나누고 [CLS] 토큰을 각 문장의 시작 위치에 추가한다.

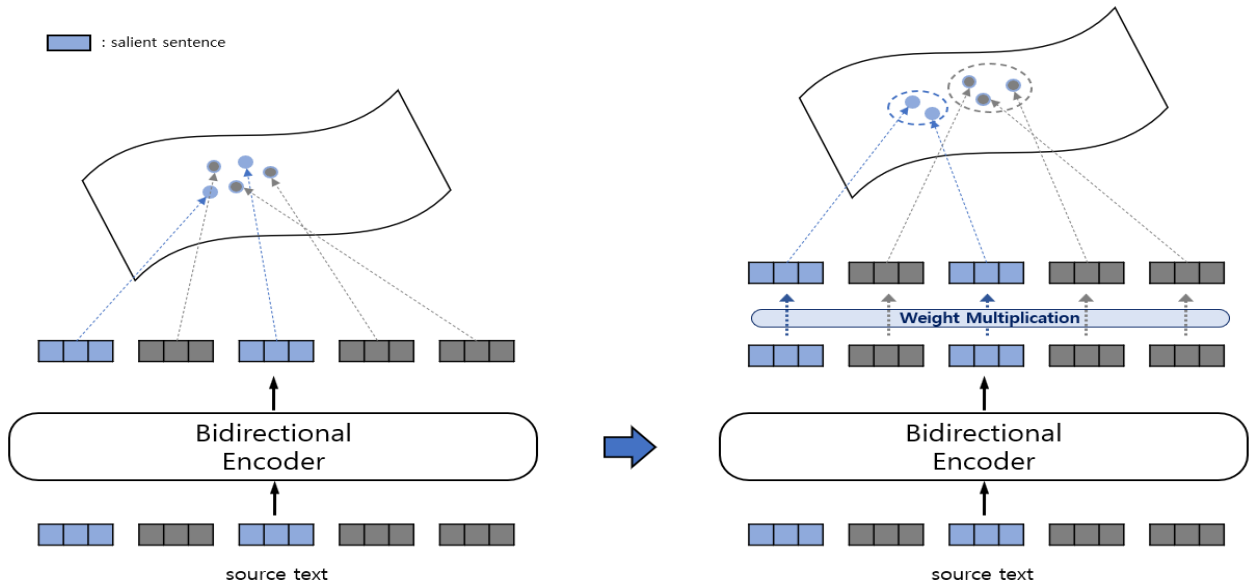


그림 3. 제안 모델의 기본 개념  
Fig. 3. Basic concept of the proposed model

표 1. Xsum 데이터 예시

Table 1. Examples of XSum data set

Document
<p>Army explosives experts were called out to deal with a suspect package at the offices on the Newtownards Road on Friday night.</p> <p>Roads were sealed off and traffic diverted as a controlled explosion was carried out.</p> <p>The premises, used by East Belfast MP Naomi Long, have been targeted a number of times.</p> <p>Most recently, petrol bomb attacks were carried out on the offices on consecutive nights in April and May.</p> <p>The attacks began following a Belfast City Council vote in December 2012 restricting the flying of the union flag at the City Hall.</p> <p>Condemning the latest hoax, Alliance MLA Chris Lyttle said: "It is a serious incident for the local area, it causes serious disruption, it puts people's lives at risk, it can prevent emergency services reaching the area. "Ultimately we need people with information to share that with the police in order for them to do their job and bring these people to justice."</p>
Summary
<p>A suspicious package left outside an Alliance Party office in east Belfast has been declared a hoax.</p>

그리고 BERT를 통과해 출력된 토큰 벡터 중 [CLS] 토큰 벡터들만 선택하여 2층의 트랜스포머 인코더로 전달하고 각 문장의 표현인 [CLS] 토큰 벡터들은 최종적으로 0~1의 값으로 출력된다.

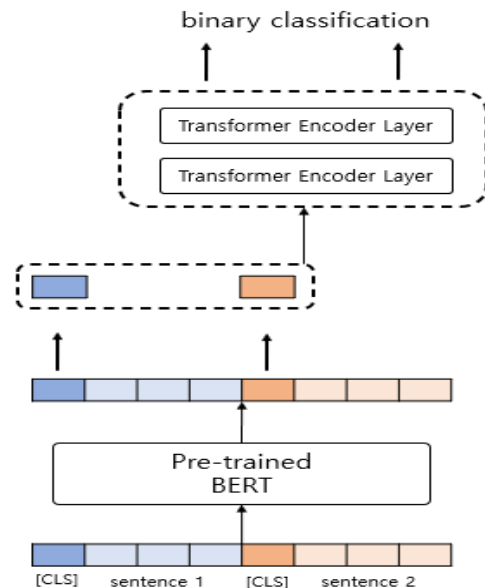


그림 4. BERTSUM의 구조  
Fig. 4. Network architecture of BERTSUM

출력값이 높을수록 추출 요약으로 적합한 문장이며 문서의 중요한 정보를 가지고 있는 문장으로 볼 수 있다.

본 연구의 제안 방안에서는 위에서 설명한 BERTSUM을 이용하여 중요한 문장과 그렇지 않은 문장을 일정 비율로 나누어 구분한다. XSum의 문서 데이터를 BERTSUM에 입력하고 점수가 높은 문장들을 중요한 문장으로 선정한다.

### 3.2 가중치 문맥 벡터

이번 절에서는 추출 요약을 통해 구분한 문장들을 BART에서 어떻게 이용하는지에 대해 그림과 수식으로 설명한다. 그림 5는 원본 텍스트의 중요한 문장과 그렇지 않은 문장의 클러스터링 아이디어를 2차원 잠재 벡터 공간상에서 나타낸다. 왼쪽 그림은 BART의 인코더에서 출력된 벡터들을 2차원 평면에 나타낸다.  $\vec{v}_1, \vec{v}_3$ 을 중요한 문장에 해당하는 벡터, 나머지를 중요하지 않은 벡터라고 할 때, 오른쪽 그림과 같이 중요한 벡터에 가중치를 곱하는 것으로 벡터의 크기를 조절해 중요한 문장과 그렇지 않은 문장을 구분할 수 있는 효과를 낼 수 있다.

Algorithm 1은 중요 문장인지 아닌지에 따라 문장 벡터에 가중치를 곱하는 과정을 설명하는 알고리즘이다.

---

Algorithm1: 가중치 기반 문맥 벡터 생성 알고리즘

---

```

1. encoder_out = encoder(input)
2. weighted_id = dict(), weight = dict()
3. for row in weighted_id['s']:
4.     encoder_out[row] *= weight['s']
5. for row in weighted_id['u']:
6.     encoder_out[row] *= weight['u']
    
```

---

Line 1은 입력 데이터에 대한 인코더의 출력 벡

터이다. Line 2는 중요한 문장과 중요하지 않은 문장의 위치와 가중치 값이다. Line 3~4는 중요한 문장에 해당하는 인코더의 출력 벡터에 가중치를 곱해주고 Line 5~6은 중요하지 않은 문장에 해당하는 출력 벡터에 가중치를 곱해준다.

## IV. 실험

### 4.1 실험 환경

이번 절에서는 본 논문의 제안 방안에서 진행한 실험 환경을 자세히 설명한다.

BERTSUM을 이용한 문장의 추출은 기본적으로 문장 개수로 설정되어있다. 문서의 길이에 따라 추출하는 문장의 개수를 달리하기 위해서 predict 함수를 수정해야 한다. 사전학습된 BART 모델과 BART 토큰라이저는 Huggingface의 transformers 모듈을 사용한다. 사용한 사전학습 모델명은 'facebook/bart-base'이다. XSum은 Huggingface의 datasets 모듈을 통해 불러온다. 그리고 BART의 최대 입력 크기인 1024에 맞춰 토큰화된 Xsum 문서의 길이가 1024를 넘으면 앞에서부터 끊어서 사용한다.

BART 인코더의 출력 벡터에 가중치를 곱해주기 위해서 BART 모델을 수정한다. 모델의 forward 함수의 인자로 중요한 문장과 그렇지 않은 문장의 위치와 곱해줄 가중치를 추가한다. 그리고 가중치를 인자로 받았을 경우 Algorithm 1을 수행하도록 한다.

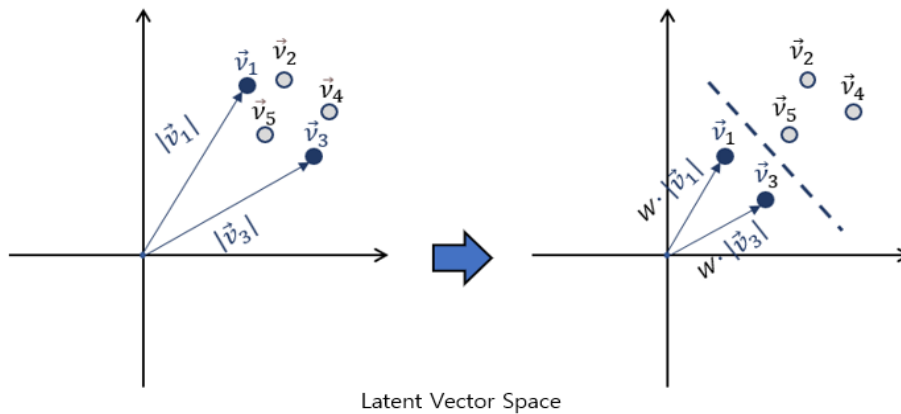


그림 5. 문맥 벡터 vs 가중치 문맥 벡터  
Fig. 5. Context vector vs weighted context vector

표 2는 BART의 파인 튜닝에서 사용한 하이퍼 파라미터이다. 기존 방안과 제안 방안은 같은 값을 사용한다. 표 3은 transformers 모델의 generate 함수를 사용한 요약문 생성에서 사용한 옵션 값이다. 표 4는 실험에 사용한 컴퓨터 하드웨어 사양을 보여준다.

#### 4.2 성능 평가

성능 평가로는 ROUGE-1, ROUGE-2, ROUGE-L, score와 BERT score의 F-Score를 사용한다. ROUGE-N은 n-gram 방식으로 토큰화된 정답 요약문과 생성된 요약문 사이의 겹치는 토큰의 개수를 기반으로 한 방식이고, ROUGE-L은 둘 사이의 가장 긴 문자열의 길이를 기반으로 계산한다.

BERT score는 정답 요약문과 생성 요약문을 각각 BERT에 입력하여 나온 임베딩 벡터의 코사인 유사도를 기반으로 계산한다. 표 5는 기존 방안과 제안 방안을 비교 평가한 것이다. 제안 방안이 기존 방안보다 ROUGE score는 약 3%, BERT score는 약 0.2% 증가한 것을 확인할 수 있다.

표 2. BART 학습 하이퍼 파라미터  
Table 2. Hyper parameters for BART

Hyper-parameters	Value
Dropout rate	0.1
Batch size	1
Learning rate	3e-5
Gradient accumulation	2
Gradient clipping	0.1

표 3. 요약 생성 하이퍼 파라미터  
Table 3. Hyper parameters for summary generation

Generation options	Value
Number of beams	6
Max length	62
Max repeat n-gram	3
Early stopping	True

표 4. 컴퓨터 하드웨어 사양  
Table 4. Computer hardware specifications

Type	SPEC	Count	Size
CPU	Intel Xeon Scalable Silver 4414 (2.20 GHz, core: 40)	2	-
GPU	GEFORCE RTX 2080 Ti D6 11GB BLOWER (core: 4,352)	4	11GB
SSD	Samsung 970 PRO M.2 2,280	1	1TB

BERT score는 BERT의 임베딩 벡터 사이의 코사인 유사도를 가운데 최댓값을 이용하기 때문에 ROUGE score보다 증가율이 떨어진다.

표 6은 중요한 문장에 곱하는 가중치 값을 조절 해가며 진행한 실험의 결과이다. 가중치의 크기보다는 방향이 더 큰 영향을 주는 것을 확인할 수 있다.

그림 6은 중요 문장의 비율에 따른 실험 결과이다. 중요 문장과 중요하지 않은 문장의 비율을 50%로 했을 때가 30%로 했을 때보다 좋은 결과를 보인다. 이는 문장 추출 비율이 너무 낮을 경우, 추출된 문장들이 원본 텍스트의 중요한 정보들을 모두 포함하고 있지 못한 것으로 분석할 수 있다.

표 5. 기존 방안과 제안 방안의 ROUGE score와 BERT score 비교

Table 5. Comparison of ROUGE and BERT score for BART as baseline and BART using the proposed method

Model \ Score	Baseline	Proposed method
ROUGE-1	36.48	37.43
ROUGE-2	14.36	15.17
ROUGE-L	29.18	29.93
ROUGE-avg	26.67	27.51
BERT Score	90.06	90.27

표 6. 가중치 변화에 따른 ROUGE score와 BERT score  
Table 6. ROUGE and BERT score according to different weight values

Weight	ROUGE-avg	Bert Score
-2.0	27.38	90.24
-1.5	27.43	90.26
-1.0	27.51	90.27
1.5	27.05	90.16
2.0	26.98	90.1

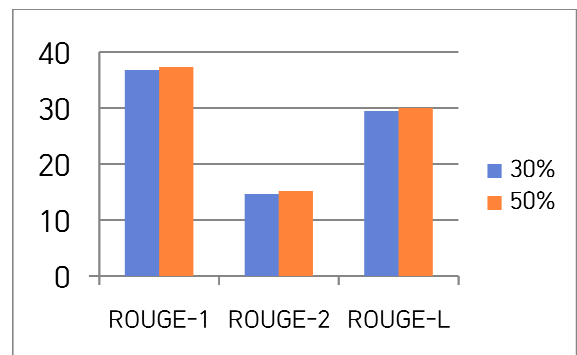


그림 6. 중요 문장 비율에 따른 ROUGE Score  
Fig. 6. ROUGE score according to different number of salient sentences

표 7. 기존 방안과 제안 방안으로 생성된 요약문 비교  
Table 7. Comparison of the summaries generated by BART and BART using the proposed method

Document	
<p>A spokesman for Palm Beach Gardens police in Florida confirmed to the BBC they were investigating a fatal crash involving the Grand Slam champion.</p> <p>A man was taken to hospital after the accident on 9 June and died two weeks later from his injuries, he said.</p> <p>According to TMZ, which broke the story, police believe the seven-time Grand Slam champion was at fault. But a lawyer for Williams said it was an "unfortunate accident".</p> <p>The man who died, Jerome Barson, was travelling with his wife who was driving their vehicle through an intersection when the accident happened.</p> <p>Williams' car suddenly darted into their path and was unable to clear the junction in time due to traffic jams, according to witness statements in a police report obtained by US media.</p> <p>Mrs Barson was also taken to hospital but survived. "[Williams] is at fault for violating the right of way of [the other driver]," the report said, adding that there were no other factors like drugs, alcohol or mobile phone distractions.</p> <p>The 37-year-old tennis star reportedly told police she did not see the couple's car and she was driving slowly. Police spokesman Major Paul Rogers said police were investigating whether the incident was connected to Mr Barson's death.</p> <p>Williams' lawyer Malcolm Cunningham told CNN in a statement: "Ms Williams entered the intersection on a green light. The police report estimates that Ms Williams was travelling at 5mph when Mrs Barson crashed into her.</p> <p>"Authorities did not issue Ms Williams with any citations or traffic violations. This is an unfortunate accident and Venus expresses her deepest condolences to the family who lost a loved one."</p> <p>Next week, Williams is due to play at Wimbledon in London, where she is seeded 10th.</p>	
Reference summary	
US tennis star Venus Williams has been involved in a car accident that led to the death of a 78-year-old man.	
Generated summary	
Baseline	Proposed method
US tennis star <u>Serena Williams</u> is at fault in the death of a man who died in a crash, police say.	<u>Venus Williams</u> has been arrested on suspicion of causing the death of a man who was killed in a car crash.

표 7은 파인 튜닝한 BART 모델을 이용해 요약문을 생성한 예시이다. 두 요약문 모두 사망 사고에 연루되었다는 중심 내용을 확인할 수 있다. 차이점으로는 기존 방안의 요약문은 US tennis star라는 정보를 놓치지 않은 대신에 이름을 잘못 생성하였고, 반면에 제안 방안은 이름에 집중하고 이름 앞에 정보를 버린 것을 볼 수 있다. 마지막 부분에서 기존 방안의 요약문은 어떤 종류의 충돌 사고였는지에 대한 정보 대신 '경찰이 발표했다'라는 중요도가 다소 떨어지는 내용으로 끝이 나는 반면 제안 방안은 차량 충돌 사고라는 구체적인 정보를 포함하고 있다. 이를 통해, 제안 방안이 중요 내용에 집중하여 요약문을 생성하고 있다는 것을 확인할 수 있다.

### V. 결론 및 향후 과제

최근 데이터 활용의 중요성이 커짐에 따라 데이터를 효율적으로 처리할 수 있는 기술들에 관한 연구가 활발히 진행되고 있다. 그중, 자동 요약 시스템은 문서와 같은 긴 텍스트를 짧은 요약문으로 변환시켜 원하는 정보를 찾는 데 걸리는 시간을 크게 단축할 수 있다.

본 논문에서는 생성 요약에 대표적인 사전학습 모델 BART의 학습 과정에서 인코더 문맥 벡터에 가중치를 곱하여 중요 문장과 중요하지 않은 문장의 패턴을 학습하는 방안을 제안한다. 해당 방안에 의해 중요한 정보에 집중하는 방법을 학습한 BART가 기존의 학습 방안보다 좋은 요약문을 생성하는 것을 확인할 수 있었다.

향후 연구로는 XSum이외의 요약 데이터셋을 이용한 실험을 진행하고, 최적의 가중치와 중요 문장 비율을 자동으로 튜닝(Tuning)할 수 있도록 새로운 모델을 연구할 예정이다.

### References

[1] <https://towardsdatascience.com/deep-learning-models-for-automatic-summarization-4c2b89f2a9ea>. [accessed: Sep. 25, 2021]

[2] A. M. Rush, S. Chopra, and J. Weston, "A neural attention model for abstractive sentence



- summarization", EMNLP, The Association for Computational Linguistics, pp. 379-389, Sep. 2015. <http://dx.doi.org/10.18653/v1/D15-1044>.
- [3] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding", NAACL, Minneapolis, Minnesota, pp. 4171-4186, Jun. 2019. <http://dx.doi.org/10.18653/v1/N19-1423>.
- [4] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving Language Understanding by Generative Pre-Training", Technical report, OpenAI, 2018.
- [5] Y. Liu, and M. Lapata, "Text Summarization with Pretrained Encoders", EMNLP/IJCNLP, Hong Kong, China, pp. 3728-3738, Nov. 2019. <https://aclanthology.org/D19-1387>.
- [6] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "BART: Denoising Sequence-to-Sequence Pre-Training for Natural Language Generation, Translation, and Comprehension", ACL, Online, pp. 7871-7880, Jul. 2020. <https://aclanthology.org/2020.acl-main.703>.
- [7] S. Narayan, S. Cohen, and M. Lapata, "Don't Give Me the Details, Just the Summary! Topic-Aware Convolutional Neural Networks for Extreme Summarization", EMNLP, Association for Computational Linguistics, Brussels, Belgium, pp. 1797-1807, Nov. 2018. <http://dx.doi.org/10.18653/v1/D18-1206>.
- [8] T. Zhang, V. Kishore, F. Wu, K. Weinberger, and Y. Artzi, "BERTScore: Evaluating Text Generation with BERT", ICLR, Mar. 2020.

## 저자소개

### 조 성 국 (Sung-Guk Jo)



2018년 3월 ~ 현재 : 군산대학교  
소프트웨어융합공학과 학부생  
관심 분야 : 자연어처리, 인공지능

### 온 병 원 (Byung-Won On)



2007년 : 미국  
펜실베이니아주립대학교  
컴퓨터공학과 박사, 캐나다  
브리티시 컬럼비아 대학교 박사  
후 연구원

2010년 : 미국 일리노이 대학교  
ADSC센터 선임연구원,

서울대학교 차세대융합기술연구원 연구교수  
2011년 : 차세대융합기술연구원 공공데이터 연구센터  
센터장  
2014년 ~ 현재 : 군산대학교 소프트웨어융합공학과 교수  
관심 분야 : 데이터 마이닝, 자연어처리, 빅데이터,  
인공지능