논문 2023-1-10 http://dx.doi.org/10.29056/jsav.2023.3.10

토픽 모델을 이용한 비지도 요약 기법의 연구

배현진*† , 김철연*

A Study of Unsupervised Summarization Using Topic Model

Hyunjin Bae*† , Chulyun Kim*

요 약

자동 문서 요약(Automatic Document Summarization)은 문서의 중요한 내용은 유지하면서 길이가 짧은 요약문을 만들어 내는 것을 목표로 하는 연구 분야이다. 그동안 자동 문서 요약은 대용량의 데이터셋을 이용한 지도 학습 기반의 심층 신경망 모델을 사용해왔다. 하지만 늘어나는 산업의 수요와는 달리 자동 요약을 위한 요약 데이터셋이 여전히 부족한 실정이다. 이런 데이터 부족 문제는 요약 분야뿐만 아니라 자연어 처리 전반에 걸쳐 새로운 문제점으로 대두되고 있다.

이를 해결하기 위해 Zero-Shot Learning이나 자가 지도 학습 등의 기법이 등장했다. 이들의 공통점은 기존의 데이터에 대한 좋은 표현을 만들어 모델이 기존에 보지 못한 데이터에 대해서도 잘 다룰 수 있도록 하는 것을 목표로 한다. 이에 본 논문에서는 토픽 모델을 이용한 비지도 추출 요약 모델 TES(Topic model based Extractive Summarization)에 대해 제안하고, 이에 대한 실험을 통해 기존의 비지도 추출 요약 모델들과 비슷한 성능을 기록하는 것을 확인하고, TES가 기존의 모델 대비 가지는 장점을 제시했다.

Abstract

Automatic Document Summarization is a research field that aims to create short summaries of documents while maintaining their important content. So far, document summarization has relied on supervised deep neural network models trained on large datasets. However, despite increasing demand from industry, there is still a shortage of summarization datasets. Data scarcity problem is not only a challenge for summarization but also for natural language processing in general.

To address this, techniques such as Zero-Shot Learning and Self-Supervised Learning have emerged, which aim to create good representations so that models can handle unseen data well. In this paper, we propose a topic model-based unsupervised extractive summarization model called TES(Topic model based Extractive Summarization). Through experiments, we confirm that it performs similarly to existing models while suggesting its advantages over existing models.

한글키워드 : 자동 문서 요약, 추출 요약, 토픽 모델링, 자연어 처리,딥러닝

keywords : document summarization, extractive summarization, topic modeling, NLP,Deep Learning

* 숙명여자대학교 IT공학과
† 교신저자: 배현진
(email: gloria9705@sookmyung.ac.kr)
접수일자: 2023.03.15. 심사완료: 2023.03.17.
게재확정: 2023.03.20.

1. 서 론

자동 문서 요약이란 문서의 중요한 내용은 유지하면서 길이가 짧은 요약문을 만들어 내는 기

법이다. 그동안 자동 문서 요약 연구에 있어서 특히 신경망 모델들을 사용한 기법들이 좋은 성능을 내고 있으며, 대용량의 문서-요약문 데이터셋이 공개된 이후 이를 이용한 다양한 문서 요약 기법들이 연구되고 있다[1-2]. 또한 최근 다양한 컨텐츠에 걸쳐 중요한 내용만을 요약해 서비스를 제공하는 서머리(Summary) 산업에 대한 수요가 증가하고 있으며, 이에 따라 컨텐츠를 자동으로 요약해 제공하는 자동 요약 서비스가 개발되고 있다.

하지만 이런 산업의 흐름과는 달리 여전히 자동 요약을 위한 데이터셋(Dataset)이 부족하다. 다른 데이터셋에 비해 특히 자동 요약을 위한 데이터셋은 제작 비용이 크다. 원문에 대응하는 요약문을 사람이 직접 작성해야 하며, 모든 분야에 대해 그에 맞는 문서 스타일의 요약문을 작성하는 것은 실질적으로 어려운 일이다. 이러한 요약 데이터셋의 부재와 비용 문제가 대두되며 비지도 요약 기법에 대한 연구가 다시 활발히 진행되고 있지만, 여전히 부족한 실정이다.

이러한 데이터셋의 부족 문제는 비단 자동 요약 분야 뿐만 아니라 심층 신경망을 사용하는 다양한 자연어 처리 분야에 걸쳐 대두되고 있다. 최근에는 이런 문제를 해결하고자 Zero-Shot Learning, 자가 지도 학습(Self-Supervised Learning)과 같은 다양한 기법들이 연구되고 있다. Zero-shot Learning(ZSL)은 이미 보유하고 있는 데이터셋을 외부 정보(External Knowledge)로 활용해 모델을 학습하여 모델이 접하지 못한 새로운 데이터셋에 대해서도 작업을 수행할 수 있도록 만들어 주는 것이다[3-5]. 많은 경우 Zero-shot Learning은 기존의 데이터를 잠재 공간(Latent Space)에 할당하고, 이 잠재 공간에 새로운 데이터를 입력해 정답을 얻어낸다. 이와 비슷한 기법인 Self-Supervised Learning은 다량의 정답이 없는 원 데이터로부터 데이터 간 관계를 통해 정답을 자동으로 생성하여 지도 학습에 이용하는 비지도 학습기법이다[6-8]. 자연어 처리 분야에서는 보통 문장의 토큰을 마스킹(Masking) 처리하고 이를 복원하는 Masked Language Model과 같은 Denoising Auto-Encoder 기법을 사용한다[6]. 입력에 노이즈를 추가하고, 이를 원래 입력으로 복원함으로써 정답이 없는 원 데이터로부터 좋은 잠재 표현 공간을 만들어 내는 것을 목표로 한다.

이에 본 논문에서는 토픽 모델(Topic Model)을 이용한 비지도 추출 요약 모델 TES(Topic model based Extractive Summarization)에 대해 제안하고자 한다. 토픽 모델이란 주어진 문서 집합 내에서 추상적인 주제를 찾아내기 위한 통계적 모델이다. 하나의 문서는 주제에서부터 시작해 그 주제에 해당하는 단어들로 쓰였다는 가정하에 토픽 모델링(Topic Modeling)이 진행되며, 그렇기 때문에 토픽 모델의 학습은 문서 집합 내 단어 분포를 이용해 해당 주제의 주제 분포의 추정으로 진행이 된다[9]. 마찬가지로 잠재 공간에서의 주제 분포를 추정하는 과정이기 때문에, ZSL을 위한 외부 정보로 사용할 수 있을 것이란 가정 하에 본 연구를 진행하고자 한다.

본 논문은 다음과 같은 흐름으로 전개된다: 2장에서는 토픽 모델과 추출 요약에 관련된 여러 선행 연구들을 소개하며, 3장에서는 토픽 모델을 이용한 추출 요약 모델 TES를 제안한다. 4장에서는 본 논문의 실험에 관한 세부 설정과 실험 결과에 대한 분석을 보여주고, 기존의 비지도 추출 요약 모델과 비교했을 때 가지는 장점을 제시하려고 한다.

2. 관련 연구

전통적인 토픽 모델에 더불어, 인공 신경망에

대한 연구가 활발히 진행됨에 따라 근래 들어서는 인공 신경망을 이용한 토픽 모델에 대한 관심 또한 높아지고 있다. 기존에는 단순히 단어의 빈도수에 기반한 통계적 기법으로 이를 처리했다면, 신경망이나 신경망을 이용한 변분 추론(Neural Variational Inference)을 사용한 모델들이 등장하고 있다.

추출 요약 모델의 경우도 비슷한 양상을 보인다. PageRank 알고리즘을 이용한 추출 요약 알고리즘인 TextRank이 등장한 이후, 추출 요약 모델은 통계적 기법 혹은 그래프(Graph) 기반 기법을 통해 진행되었지만, 인공 신경망에 대한 연구 이후 신경망을 이용한 다양한 구조를 가진 추출 요약 모델들이 등장했다[10].

본 장에서는 그동안 어떤 방법론들이 제시되어 왔으며, 특히 신경망을 사용한 연구로는 어떤 방법론들이 있는지를 소개하려고 한다.

2.1 토픽 모델링(Topic Modeling)

토픽 모델이란 주어진 문서 집합 내에서 추상적인 주제를 찾아내기 위한 통계적 모델이다. 가장 유명한 토픽 모델로는 2002년 발표된 LDA(Latent Dirichlet Allocation)이 있다[11].

LDA는 하나의 문서는 여러 주제가 모인 집합으로 표현할 수 있으며, 각 주제는 단어 분포로 표현할 수 있다는 가정에서 시작한다. 하지만 우리가 알고 있는 것은 문서에 대한 단어 분포뿐이기 때문에, 문서에 대한 토픽의 분포와 토픽에 대한 단어의 분포를 알아내기 위해 관측 가능한 단어들로부터 잠재 주제 분포를 디리클레 분포를 사용해 추론한다.

LDA의 등장 이후, 이를 변형한 다양한 토픽 모델들이 등장했으며, 근래 들어 인공 신경망(Neural Network)에 대한 관심이 높아지며 토픽 모델에서 또한 이런 인공 신경망 기반의 모델들이 등장하기 시작했다.

2017년에 발표된 ProdLDA는 Variational Auto-Encoder(VAE)에 기반, 디리클레 분포(Dirichlet distribution)과 가우시안 분포(Gaussian distribution)를 사용해 토픽 분포를 추정하는 모델이다[12]. Bag-of-Words(BoW)로 표현된 문서 집합을 연속 잠재 공간에 맵핑(mapping) 한 뒤, 디코더(Decoder)를 이용해 해당 잠재 공간으로부터 문서에 대한 BoW 표현을 다시 재건하는 과정을 통해 학습한다. 만약 모델이 잠재 공간으로부터 원문을 잘 복원할 수 있다면, 해당 잠재 공간은 문서를 잘 설명할 수 있는 잠재 공간이라 할 수 있으며, 기존에는 모델이 보지 못한 문서 또한 잘 설명할 수 있는 공간이 될 것이다.

2.2 추출 요약(Extractive Summarization)

추출 요약은 전체 문서에서 중요한 문장을 뽑아내 요약문을 생성하는 기법이다. 그동안 대부분의 추출 요약 모델은 그래프 기반 기법들을 사용해 문장을 추출하는 방식을 사용한다. 그래프 기반 추출 요약 기법으로 가장 유명한 알고리즘은 2004년에 발표된 TextRank가 있다[10]. 무방향 그래프 기반 모델로 문장은 노드로, 문장 간 유사도는 가중치로 표현해 동시 발생 행렬(co-occurence matrix)를 이용, 문장 간 유사도를 계산한다. 두 문장 간 유사도는 두 문장 내 동시 등장한 단어의 개수를 두 문장 내 단어 개수에 로그를 취한 값의 합으로 나눠서 구한다.

2019년에 발표된 PacSum은 이와 비슷하지만 방향 그래프를 이용했다는 점에서 차이가 있다. 또한 기존에는 tf-idf를 이용해 단어의 등장 빈도수를 이용했다면, 해당 연구에서는 자연어 처리 분야에서 좋은 성능을 보이고 있는 언어 모델인 BERT[6]의 임베딩을 이용해 문장의 유사도를 계산하는 방법을 제시했다.

기존에는 그래프 기반 추출 요약 알고리즘이

대부분이었다면, 토픽 모델의 경우와 마찬가지로 인공 신경망에 대한 관심이 높아지며 신경망 기반 추출 요약 모델들이 등장하기 시작했다.

2019년 발표된 BERTSumExt은 BERT 언어 모델을 이용해 추출 요약을 할 수 있는 구조를 고안한 모델이다[16]. 기존의 BERT의 경우, Masked Language Model로 학습되기 때문에 토큰(token) 단위로 입력을 계산하지만, 추출 요약을 위해서는 각 문장에 대한 임베딩이 필요하다. 따라서 이를 위해 문장의 처음뿐만 아니라 문장 사이에 [*CLS*] 토큰을 삽입하고, 각 문장에 대해 번갈아 가며 Segment Embedding을 부여하는 방식으로 다중 문장을 계산할 수 있는 구조를 제안했다. 이렇게 얻는 문장 벡터를 Transformer에 입력해 중요 문장 여부를 분류한다.

하지만 신경망 기반의 추출 요약 모델의 경우 지도 학습이기 때문에 요약 데이터셋에 의존적이다. 이에 최근, 대용량의 데이터로 사전 학습된 언어 모델들이 등장하며 이를 이용한 신경망 기반 비지도 추출 요약 모델 또한 고안되고 있다. STAS는 Transformer[18] 기반의 비지도 추출 요약 모델로, 계층적 모델과 Self-Supervised 기법을 사용해 정답이 매겨지지 않은 문서들을 미리 사전 학습한 다음, 해당 모델을 이용해 문장의 점수를 매기는 방식을 사용한다[17]. 하지만 STAS의 경우, 정답이 매겨지지 않은 문서들로 사전 학습을 해야 하기 때문에 상당히 많은 양의 데이터가 필요하며, Masked Sentence Prediction과 Sentence Shuffling 이라는 Self-Supervised 기법을 사용하기 때문에 모델의 구조가 매우 복잡하다. 또한 요약의 경우 미세 조정 학습 없이 논문에서 고안한 요약 문장의 랭킹식을 이용해 진행하기 때문에 End-to-End 모델이 아니다. 따라서 본 논문에서는 적은 양의 문서로 학습이 가능하며, 간단한 모델 구조로 End-to-End 비지도 추출 요약을 할 수 있는 방법을 고안하고자 한다.

3. TES: Topic Model based Extractive Summarization

본 장에서는 토픽 모델을 이용한 추출 요약 모델인 TES를 제안한다.

3.1 토픽 모델

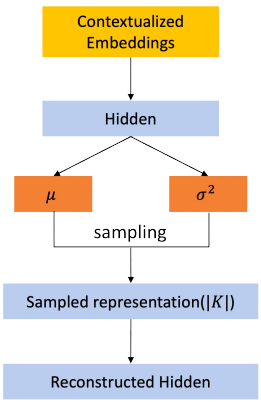

그림 1. 논문에 사용된 토픽 모델의 구조
Fig. 1. Structure of the topic model used in the paper.

본 논문에서는 VAE에 기반 한 토픽 모델인 Neural-ProdLDA[12]를 확장해 사용했다. 기본적인 구조는 ProdLDA와 동일하지만, 모델에 들어가는 입력의 형식을 수정했다. 기존의 토픽 모델들은 BoW 표현을 직접 모델에 입력한다. 하지만 논문의 토픽 모델에서는 요약 모델의 은닉 값과의 통일을 위해서 BERT의 임베딩 값을 토픽 모델의 입력 값으로 사용한다. 따라서 BoW 표현을 인코딩하고, 이를 다시 BoW로 복원하는 기존의 과정은 BERT 임베딩 값을 인코딩하고, 이를 다시 BERT의 임베딩 값으로 복원하는 과정이 된다.(그림 1)

이 과정에서 얻는 장점도 있다. BoW의 경우, 문장 내 단어의 순서는 고려하지 않고 단순히 단어의 출현 빈도만 고려하기 때문에 단어의 문맥

적 의미를 파악하지 못한다. 하지만 BERT의 경우 Self-attention[18] 기반의 구조를 사용해 양방향으로 문장을 파악하기에 해당 단어가 문장 내에서 어떤 문맥을 가졌는지를 함께 고려한다. 이렇게 문맥을 고려하게 된다면, 문서에 대해 더 정확한 토픽 분포를 얻어낼 수 있을 뿐만 아니라 학습 시 보지 못한 문서에 대해서도 합리적인 토픽 분포를 만들어 낼 수 있다.

3.2 추출 요약 모델

본 논문에서는 BERTSumExt[16]를 확장해 추출 요약을 위한 모델로 사용했다. BERTSumExt에서 사용하는 BERT가 기존의 BERT와 다른 점은, 다중 문장을 고려할 수 있도록 입력 임베딩을 수정했다는 것이다. 기존의 BERT의 경우, 문장 맨 앞에는 $[CLS]$ 토큰을, 맨 뒤에는 $[SEP]$를 삽입해 입력을 처리한다. 하지만 이 경우, 다중 문장을 고려하기 힘들다. 따라서 BERTSumExt에서는 두번째 이후의 문장 사이마다 $[CLS]$ 토큰을 삽입하고, n번째 문장의 $[CLS]$ 토큰을 해당 문장의 벡터로 사용한다. 또한 BERT에서는 Next Sentence Prediction을 위해 토크나이징을 통해 나눠준 토큰들이 첫 번째 문장에 속하는지, 두번째 문장에 속하는지 알려주는 Segmentation Embedding을 사용한다. 하지만 오직 두개의 문장에 대한 처리만 가능하기 때문에. BERTSumExt에서는 각 문장을 구분하기 위해 Interval Segment Embedding을 사용한다. 문서 d가 $[sent_1, sent_2, \ldots, sent_m]$으로 이뤄졌다고 할 때, $sent_i$에 대해 번갈아가며 E_A와 E_B를 부여한다. 즉, $[sent_1, sent_2, sent_3, sent_4, sent_5]$에 대한 Interval Segment Embedding은 $[E_A, E_B, E_A, E_B, E_A]$가 된다.

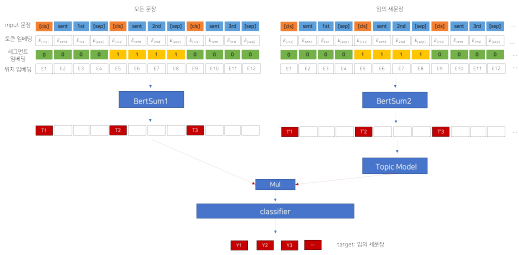

그림 2. TES의 학습 과정.
Fig. 2. Training process of TES

그림 3. (위) BERT 모델의 구조 (아래) BertSum 모델의 구조.
Fig. 3. (Top) Structure of BERT model (Bottom) Structure of BertSum model.

이렇게 얻은 문장 벡터를 Summarization 레이어에 입력해 각 문장에 대한 점수 $\hat{Y}_i$를 예측하게 된다. BERTSumExt 논문에서는 세가지 종류의 Summarization 레이어를 제시하고 있지만, 본 논문에서는 이 중 Inter-Sentence Transformer를 사용하고 있다. 기존의 Transformer를 사용하는 모델들은 토큰 단위로 입력을 받지만, 추출 요약을 위해 문장 단위의 벡터를 입력으로 받기 때문

에 Inter-Sentence Transformer가 된다. 입력받은 문장 벡터들은 다음과 같이 식 1과 식 2를 통해 계산된다.

$$\widetilde{h^l} = LN(h^{l-1} + MHAtt(h^{l-1})) \quad (1)$$

$$h^l = LN(\widetilde{h^l} + FFN(\widetilde{h^l})) \quad (2)$$

문장 벡터 T에 대해 T의 추가 위치 임베딩을 $PosEmb$라고 할 때, h^0는 $PosEmb(T)$를 뜻한다. LN은 Layer Normalization, $MHAtt$는 Multi-Head Attention을 뜻하며, l은 Transformer 레이어의 깊이를 뜻한다. h^L이 Transformer의 마지막 레이어의 은닉값이라 할 때, i번째 문장의 점수는 이를 Sigmoid Classifier에 입력해서 계산한다.

$$\hat{Y}_i = \sigma(W_o h_i^L + b_o) \quad (3)$$

BERTSumExt와 다른 점은 1) 학습시 정답으로 그래프 기반 추출 요약 모델인 PacSum[13]을 사용해 선택한 세개의 문장을 사용했다는 점과 2) 토픽 모델로 외부 정보를 포함시켜 주었다는 점이다. 기존 모델은 지도 학습을 표방하고 있기 때문에 원문-요약문 쌍을 End-to-End로 학습한다. 하지만 본 논문에서는 원문-요약문 데이터가 없이도 학습이 가능한 비지도 추출 요약 모델을 만들고자 하기 때문에 임시로 세 개의 문장을 선택해 이를 임시 정답으로 사용한다. 이렇게 추출한 세 개의 문장을 합쳐 BERT[6]에 입력한 뒤, 이를 토픽 모델에 입력해 토픽 분포를 얻는다. 그 다음 토픽 분포와 문장 벡터의 아다마르 곱(Hadamar product)를 계산해 토픽 정보가 포함된 문장 벡터를 얻는다. 따라서 원문에 대한 문장 벡터를 T, 토픽 분포 값을 TD라고 할 때, 수정된 문장 벡터 T'는 다음과 같이 계산할 수 있다.

$$T' = T \times TD \quad (4)$$

추론 시에는 토픽 모델에 세 문장을 넣지 않고 원문을 입력해 입력 문서 전체에 대한 토픽 분포를 얻어낸다. 모델은 PacSum을 통해 뽑힌 세 문장에 대한 외부 정보를 통해 해당 주제를 가진 문장을 추출하도록 학습되었다. 따라서 추론 시에 전체 문서에 대한 주제 정보를 문장 벡터에 더해주게 되면 요약 모델은 전체 문서에서 가장 주제를 잘 나타내는 문장을 추출해 낼 것이다.

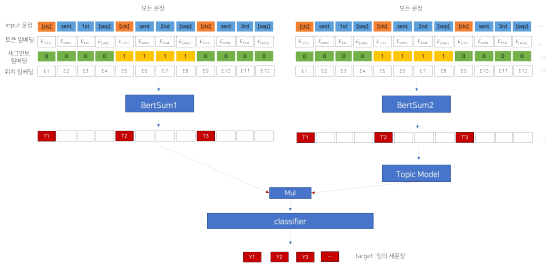

그림 4. TES의 추론 과정.
Fig. 4. Inference process of TES

4. 실험

4.1 데이터셋

본 논문에서는 CNN and Daily Mail 웹사이트에서 발췌한 뉴스 데이터로 구성된 CNN/DM 데이터셋[19]을 사용한다. CNN/DM 데이터셋은 뉴스 원문과 이에 해당하는 요약문으로 이뤄진 데이터이다. [19]에서 배포한 데이터셋 그대로 CNN의 경우 90,266개/1,220개/1,093개, DailyMail의 경우 196,961개/12,148개/10,397개의 학습/검증/평가 데이터를 사용했다. 전처리는 [16]와 동일하게 진행했다. CoreNLP[20]를 이용해 문장을 분리한 다음, [21]의 전처리 과정을 수행했다.

하지만 CNN/DM의 경우, 생성 요약을 위한

요약문으로 구성되어 있기 때문에 추출 요약 학습을 위해 [14]의 Greedy 알고리즘을 사용해 Extractive Gold Label을 생성한다. 원 데이터셋의 정답 요약문과 각 문장 간 ROUGE-2[22] 점수를 계산해 점수가 최대인 문장을 선택, 추출 요약을 위한 정답을 만들었다.

4.2 실험 설정

4.2.1 토픽 모델

PyTorch 환경에서 실험을 진행했으며, 입력 벡터를 만들기 위한 BERT 모델은 transformer 기반의 모델 학습을 위한 파이썬 라이브러리인 huggingface[27]의 'bert-based-uncased'를 사용했다. 1개의 GPU(GeForce RTX 2080 Ti)로 50 epoch 동안 학습했으며, 마지막 epoch의 모델을 사용했다. 토픽 개수는'bert-based-uncased'의 은닉층과 동일한 크기여야 하기 때문에 768개, 학습률은 0.003을 사용했으며, weight-decay로는 1e-04를 사용했다.

4.2.2 토픽 모델

추출 요약 모델의 실험 설정은 [16]의 설정을 그대로 사용했다. PyTorch 환경에서 실험을 진행했으며, OpenNMT[23]의 'bert-based-uncased' 모델을 사용했다. 2개의 GPU(GeForce RTX 2080 Ti)로 50,000 steps 동안 학습했으며, 1,000번째 step마다 체크포인트 모델을 저장했다. 이 중 가장 좋은 점수를 보인 checkpoint 모델로 결과로 기록했다. 3개의 문장을 선택하기 위해 사용한 PacSum[13]의 경우, CNN/DM 데이터셋으로 하이퍼 파라미터 튜닝을 진행했으며, 그 결과 최종적으로 $\beta = 0.4$, $\lambda_1 = 0.7$, $\lambda_2 = 0.3$을 문장 추출에 사용하였다.

예측 시, 가장 점수가 높은 3개의 문장을 추출해 요약문으로 사용했으며, 내용의 반복성을 줄이기 위해 Trigram Blocking을 사용했다[24]. 점수가 높은 순서대로 문장을 순회하며 이미 선택된 요약문 s와 현재 선택된 후보 문장 c에 대해 trigram overlapping이 존재하면 c를 건너뛰고 선택했다.

전체 문서를 인코딩하는 BERT 모델과 요약을 진행하는 Transformer는 각각 따로가 아니라 함께 미세 조정 학습(jointly finetuned)했으며, optimizer로는 Adam[25]($\beta_1 = 0.9$, $\beta_2 = 0.999$을, 손실 함수로는 Binary Cross Entropy를 사용했다. 요약 레이어로 사용되는 Transformer의 경우 2개의 레이어를 가진다. 학습 스케줄은 [18]를 따라 첫 10,000 steps 동안 식 5를 통해 warm-up 과정을 거친다.

$$lr = 2e^{-3} \cdot \min(step^{-0.5}, step \cdot warmup^{-1.5}) \quad (5)$$

4.3 실험 결과

본 절에서는 TES의 성능 평가 실험 결과에 대해 보일 것이다. 성능 평가 지표로는 요약 분야에서 가장 많이 사용되는 지표 중 하나인 ROUGE[22]를 사용한다. ROUGE는 Recall-Oriented Understudy for Gisting Evaluation의 약자로, 예측한 요약문과 정답 요약문을 자동으로 비교해주는 지표이다. 논문에서는 N-gram에 대한 재현율(Recall)인 ROUGE-N(1/2), LCS(Longest Common Subsequence)에 기반한 ROUGE-L을 사용해 다른 요약 모델들의 CNN/DM 데이터셋에 대한 결과와 비교를 진행한다. 또한 실험 결과, 요약 모델 내에서 임시 정답을 선택할 때 어떤 방식을 사용하는지에 따라서도 성능이 크게 차이가 난다는 것을 확인했다. 이에 대해 다양한 방법으로 세 문장을 선택하고 이를 임시 정답으로 사용하여 학습한 뒤, 이에 대한 결과를 분석했다.

표 1. CNN/DM 데이터셋에 대한 추출 요약 결과
Table 1. Results of extractive summarization on CNN/DM

	모델	R-1	R-2	R-L
지도 학습	REFRESH(Narayan et al., 2018)	41.30	18.40	37.50
	BERTSumExt(Liu and Lapata., 2019)	42.71	19.95	39.18
비지도 학습	TEXTRANK(BERT)	29.86	9.53	26.96
	PACSUM(Zheng and Lapata., 2019)	39.92	17.35	36.47
	STAS(Xu et al., 2020)	40.90	18.02	37.21
	TES	40.67	17.86	30.38

논문에서 사용하는 베이스라인 모델은 다음과 같다.

- REFRESH[27]: 강화 학습을 기반으로 하는 지도 추출 요약 모델로 문장의 중요도 순위를 매길 때, 강화학습을 이용해 global하게 ROUGE 점수를 최적화 하는 알고리즘을 사용한다.
- BERTSumExt[16]: BERT를 기반으로 하는 지도 추출 요약 모델로 BERT 언어 모델의 입력을 문장 단위로 계산할 수 있도록 구조를 변경하고, Transformer를 이용해 해당 문장의 중요도 점수를 계산한다.
- TEXTRANK[10]: 그래프 기반의 비지도 추출 요약 모델로 단방향 그래프를 기반으로 한다. 문장은 노드로, 문장 간 유사도는 가중치로 표현해 동시 발생 행렬(co-occurence matrix)를 이용, 문장 간 유사도를 계산한다.
- PACSUM[13]: 그래프 기반의 비지도 추출 요약 모델로 양방향 그래프를 기반으로 한다. 기존에는 문장의 중요도를 계산할 때 tf-idf를 사용했다면, PACSUM은 자연어처리 분야에서 좋은 성능을 내고 있는 BERT를 사용해 문장의 임베딩을 얻고, 이를 통해 중요도 점수를 계산한다.
- STAS[17]: Transformer 기반의 비지도 추출 요약 모델로, 계층적 모델과 Self-Supervised 기법을 사용해 정답이 매겨지지 않은 문서들을 미리 사전 학습한 다음, 해당 모델을 이용해 문장의 점수를 매기는 방식을 사용한다.

표 1은 베이스라인 모델과 TES 성능 평가 결과에 대한 표이다. 당연한 결과로 비지도 학습 계열 추출 요약 모델보다 지도 학습 계열의 추출 요약 모델의 점수가 높다. 하지만 비지도 학습 모델들과의 결과의 차이가 ROUGE 1-2인 것으로 보아 비지도 학습을 통한 추출 요약도 어느 정도 효과가 있음을 알 수 있다. STAS를 제외하고는 논문에서 발표한 결과보다 ROUGE 점수가 떨어진다. 이는 각 논문들의 학습 환경과 본 논문의 학습 환경이 다르기 때문에 발생한 결과라 보여진다. PACSUM과 TES를 비교하면 논문에서 제안한 방식으로 토픽 주제 분포를 더해주는 방식이 요약의 성능을 높힐 수 있다는 것을 확인할 수 있다. 본 논문이 기존 SOTA 모델에 비해 가지는 장점을 고려한다면, 어느 정도 고려할만한 결과라고 볼 수 있다.

첫 번째로, TES는 심층 신경망 계열의 모델과 기계 학습 계열의 모델을 함께 사용한다. 최근 심층 신경망과 관련해 가장 큰 문제점으로 '설명 불가능함'이 대두되고 있다. 모델이 예측하는 결과에 대해 설명할 수 있는 방법이 없기 때문에 이를 위해 설명 가능한 AI(Explainable AI)와 같은 분야가 각광받고 있기도 하며, 기계 학습에 대한 연구도 다시 활발히 진행되고 있다. SOTA 모델의 경우, 입력으로 들어오는 문서에 대한 인코딩은 모두 심층 신경망 모델을 이용해 이뤄진다. 하지만 TES의 경우, 토픽 모델을 이용해 주제 분포를 추출하고, 이와 유사한 분포를 가진 문장이 추출되기를 바라기 때문에 결과에 대한 기준과 근거가 명확하다.

두 번째로, SOTA 모델의 경우 Self-Supervised 기법을 사용해 정답이 매겨지지 않은 문서들에 대해 사전학습을 진행한다. 하지만 이런 사전 학습의 경우 상당히 많은 양의 데이터가 필요하며, Masked Sentence Prediction과 Sentence Shuffling 이라는 Self-Supervised 기법

을 사용하기 때문에 모델의 구조가 매우 복잡하다. 이에 비해 TES는 미리 사전 학습된 BERT를 이용해 미세 조정 학습을 할 뿐이며, 토픽 모델의 경우에도 요약에 필요한 문서로만 학습을 진행해도 충분하다. 또한 입력으로 들어오는 문서를 처리하고, 이를 Transformer에 입력해 문장의 점수를 얻어내는 구조이기 때문에 모델의 구조가 단순하다.

마지막으로 SOTA 모델의 경우 미세 조정 학습 없이 논문에서 고안한 요약 문장의 랭킹식을 이용해 추출 요약을 진행하기 때문에 End-to-End 모델이 아니다. 또한 표 의 다른 비지도 모델들 또한 그래프 기반 모델들이다. 하지만 TES의 경우, End-to-End로 추출 요약을 진행하는 비지도 심층 신경망 기반 모델을 제안한다.

5. 결론

본 논문에서는 토픽 모델을 이용한 비지도 추출 요약 모델인 TES를 제안했다. 우선 요약을 원하는 문서를 심층 신경망을 사용한 토픽 모델인 Prod-LDA를 사용해 학습했으며, 요약 모델과의 벡터값 통일을 위해 기존에는 BoW를 복원하던 모델 구조를 은닉층의 값을 복원하도록 수정했다. 이렇게 학습한 토픽 모델의 분포를 요약 모델에 입력해 외부 정보로 사용했다. 심층 신경망 기반 추출 요약 모델 중 가장 많이 사용되는 모델 중 하나인 BERTSumExt를 이용, 입력 문서의 BERT 결과값에 PacSum을 이용해 선택된 정답 요약문의 토픽 분포를 합쳐 요약 모델의 학습을 진행했다. 이를 통해 단순히 원문-요약문의 학습이 아니라 모델에게 정답 문장을 선택하는 과정을 학습시켜주는 효과를 보았다.

제안한 모델의 성능을 입증하기 위해, 동일 데이터셋에 대해 기존의 지도 혹은 비지도 추출 요약 모델과의 성능 비교 실험을 진행했다. 기존의 모델 대비 비슷한 결과 수치를 보였으며, 본 논문의 모델이 기존의 모델에 비해 가지는 장점을 함께 분석해 효과성과 효용성을 보였다.

본 논문에서 제안한 모델을 통해 데이터셋이 부족한 상황에서도 심층 신경망 기반 End-to-End 모델로 비지도 추출 요약을 효율적으로 진행할 수 있다. 또한 인공지능의 '설명 불가능함'이 대두되고 있는 현재, 이렇게 기계 학습 기반 모델과 심층 신경망 기반 모델을 함께 사용함으로써 기존의 문제를 해결하고, 앞으로도 이러한 방법을 사용해 다양한 자연어 처리 문제를 해결할 수 있을 것이다.

본 연구는 문화체육관광부 및 한국콘텐츠진흥원의 2022년도 소프트웨어 저작권 연구개발지원사업의 연구결과로 수행되었음 (R2022020041)

참 고 문 헌

[1] Linguistic Data Consortium and New York Times Company. (2008). The New York Times Annotated Corpus. Linguistic Data Consortium. DOI: 10.35111/77ba-9x74

[2] Hermann, Karl Moritz, et al. (2005). Teaching machines to read and comprehend. Advances in neural information processing systems. 28. DOI: 10.48550/arXiv.1506.03340

[3] Song, Yangqiu & Upadhyay, Shyam & Peng, Haoruo & Mayhew, Stephen & Roth, Dan. (2019). Toward any-language zero-shot topic classification of textual documents. Artificial Intelligence. 274. pp.133-150. DOI: 10.1016/j.artint.2019.02.002

[4] Ben Zhou, Daniel Khashabi, Chen-Tse

Tsai, & Dan Roth. (2018). Zero-Shot Open Entity Typing as Type-Compatible Grounding. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. pp.2065 - 2076. DOI: 10.18653/v1/D18-1231

[5] Omer Levy, Minjoon Seo, Eunsol Choi, & Luke Zettlemoyer. (2017). Zero-Shot Relation Extraction via Reading Comprehension. In Proceedings of the 21st Conference on Computational Natural Language Learning. pp.333 - 342. DOI: 10.18653/v1/K17-1034

[6] Devlin, Jacob, Chang, Ming-Wei, Lee, Kenton & Toutanova, Kristina. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Vol1, 4171-4186. DOI: 10.48550/arXiv.1810.04805

[7] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov & Luke Zettlemoyer. (2020). BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp.7871 - 7880. DOI: 10.18653/v1/2020.acl-main.703

[8] Genta Indra Winata, Andrea Madotto, Zhaojiang Lin, Rosanne Liu, Jason Yosinski & Pascale Fung. (2021). Language Models are Few-shot Multilingual Learners. In Proceedings of the 1st Workshop on Multilingual Representation Learning. pp.1 - 15. DOI: 10.18653/v1/2021.mrl-1.1

[9] David M. Blei. (2012). Probabilistic topic models. Commun. ACM 55(4). pp.77 - 84. DOI: 10.1145/2133806.2133826

[10] Rada Mihalcea & Paul Tarau. (2004). TextRank: Bringing Order into Text. In Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing. pp.404 - 411. URL: https://aclanthology.org/W04-3252

[11] David Blei, Andrew Ng & Michael Jordan. (2003). Latent Dirichlet Allocation. Journal of Machine Learning Research. 3. pp.993-1022. ISSN: 1532-4435

[12] Akash Srivastava & Charles Sutton. (2017). Autoencoding Variational Inference For Topic Models. International Conference on Learning Representations. DOI: 10.48550/arXiv.1703.01488

[13] Hao Zheng & Mirella Lapata. (2019). Sentence Centrality Revisited for Unsupervised Summarization. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp.6236 - 6247. DOI: 10.18653/v1/P19-1628

[14] Ramesh Nallapati, Feifei Zhai & Bowen Zhou. (2017). SummaRuNNer: a recurrent neural network based sequence model for extractive summarization of documents. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence. pp.3075-3081. DOI: 10.48550/arXiv.1611.04230

[15] Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau & Yoshua Bengio. (2014). On the Properties of Neural Machine Translation: Encoder - Decoder Approaches. In Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation. pp.103 - 111. DOI: 10.3115/v1/W14-4012

[16] Yang Liu & Mirella Lapata. (2019). Text Summarization with Pretrained Encoders. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. pp.3730 - 3740. DOI:

10.18653/v1/D19-1387
[17] Shusheng Xu, Xingxing Zhang, Yi Wu, Furu Wei & Ming Zhou. (2020). Unsupervised Extractive Summarization by Pre-training Hierarchical Transformers. In Findings of the Association for Computational Linguistics: EMNLP 2020. pp.1784 - 1795. DOI: 10.48550/arXiv.2010.08242
[18] Vaswani, Ashish & Shazeer, Noam & Parmar, Niki & Uszkoreit, Jakob & Jones, Llion & Gomez, Aidan N & Kaiser, Lukasz & Polosukhin, Illia. (2017). Attention is All you Need. Advances in Neural Information Processing Systems. 30. DOI: 10.48550/arXiv.1706.03762
[19] Karl Moritz Hermann, Tomáš Kočiský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman & Phil Blunsom. (2015). Teaching machines to read and comprehend. In Proceedings of the 28th International Conference on Neural Information Processing Systems. Vol.1. pp.1693 - 1701. DOI: 10.48550/arXiv.1506.03340
[20] Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard & David McClosky. (2014). The Stanford CoreNLP Natural Language Processing Toolkit. In Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations. pp.55 - 60. DOI: 10.3115/v1/P14-5010
[21] See, Abigail & Liu, Peter J. & Manning, Christopher D. (2017). Get To The Point: Summarization with Pointer-Generator Networks. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. Vol.1. pp.2073-1083. DOI: 10.48550/arXiv.1704.04368
[22] Lin, Chin-Yew. (2004). ROUGE: A Package for Automatic Evaluation of Summaries. Text Summarization Branches Out. pp.74-81. URL: https://aclanthology.org/W04-1013
[23] Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart & Alexander Rush. (2017). OpenNMT: Open-Source Toolkit for Neural Machine Translation. In Proceedings of ACL 2017, System Demonstrations. pp.67 - 72. URL: https://aclanthology.org/P17-4012
[24] Romain Paulus, Caiming Xiong & Richard Socher. (2018). A Deep Reinforced Model for Abstractive Summarization. 6th International Conference on Learning Representations, ICLR 2018. DOI: 10.48550/arXiv.1705.04304
[25] Kingma, Diederik P & Jimmy Ba. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980. DOI: 10.48550/arXiv.1706.03762
[26] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, et al.. (2020). Transformers: State-of-the-Art Natural Language Processing. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pp.38 - 45. DOI: 10.18653/v1/2020.emnlp-demos.6
[27] Shashi Narayan, Shay B. Cohen, & Mirella Lapata. (2018). Ranking Sentences for Extractive Summarization with Reinforcement Learning. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Vol.1. pp.1747 - 1759. DOI: 10.18653/v1/N18-1158

저 자 소 개

배현진(Hyunjin Bae)

2020.5 숙명여자대학교 IT공학과 졸업
2023.2 숙명여자대학교 IT공학과 석사
<주관심분야> 자동 문서 요약, 비지도 요약, 자연어 처리, 인공지능

김철연(Chulyun Kim)

1996 서울대학교 컴퓨터공학과 졸업
1998 서울대학교 인지과학협동과정 석사
2010 서울대학교 전기컴퓨터공학부 박사
1992.2 대한대학교 소프트웨어학과 박사
2010-2016 가천대학교 소프트웨어설계경영학과 조교수
2016-현재 숙명여자대학교 IT공학과 부교수
<주관심분야> 머신러닝, 데이터 마이닝, 빅데이터, 인공지능