

RNN과 강화 학습을 이용한 자동 문서 제목 생성

조성민* · 김우생**

Automatic Document Title Generation with RNN and Reinforcement Learning

Sung-Min Cho* · Wooseng Kim**

Abstract

Lately, a large amount of textual data have been poured out of the Internet and the technology to refine them is needed. Most of these data are long text and often have no title. Therefore, in this paper, we propose a technique to combine the sequence-to-sequence model of RNN and the REINFORCE algorithm to generate the title of the long text automatically. In addition, the TextRank algorithm was applied to extract a summarized text to minimize information loss in order to protect the shortcomings of the sequence-to-sequence model in which an information is lost when long texts are used. Through the experiment, the techniques proposed in this study are shown to be superior to the existing ones.

Keywords : Title Generation, Deep Learning, Reinforcement Learning

Received : 2019. 11. 29. Revised : 2020. 01. 15. Final Acceptance : 2020. 02. 15.

※ The present Research has been conducted by the Research Grant of Kwangwoon University in 2019.

* First Author, Graduate School of Computer Science, Kwangwoon University, e-mail : jsm9301@gmail.com

** Corresponding Author, Professor, School of Software, Kwangwoon University, 20 Kwangwoon-ro, Nowon-gu, Seoul 01897, Korea,
Tel : +82-2-940-5217, e-mail : kwsrain@gmail.com

1. 서 론

근래에는 인터넷상에 뉴스 기사, 블로그, SNS 등을 통해 수많은 텍스트 데이터들이 쏟아져 나오고 있다. 이러한 데이터들은 대부분 본문의 길이가 길거나 제목이 없는 경우가 많다. 따라서 이러한 긴 본문의 내용을 요약해 사람들의 이해를 도울 수 있는 데이터의 정제 기술도 필요해지고 있다. 특히 문서의 내용을 압축해서 대표할 수 있는 문서의 제목을 자동으로 생성할 수 있는 기술이 요구된다. 근래에는 RNN(Recurrent Neural Network, 이하 RNN)(Mikolov et al., 2010)의 sequence-to-sequence 모델(Sutskever et al., 2014)을 사용해 문서의 요약을 생성하는 연구가 활발히 이루어지고 있다. sequence-to-sequence 모델은 RNN이나 LSTM(Sundermeyer et al., 2012) 같은 셀을 길고 길게 쌓아서 복잡하고 방대한 시계열 데이터를 처리하는 데 특화된 모델이다.

본 논문에서는 RNN과 강화 학습을 결합한 제목 생성 모델을 제안한다. 특히 본 논문에서 제안하는 모델은 입력 본문의 길이가 길어졌을 때 발생하는 정보 손실 문제를 줄이기 위해 TextRank 알고리즘을 적용해 추출된 요약된 본문을 학습에 사용하였고, 적절한 보상함수를 적용해 반복된 단어가 생성되는 문제를 보완하였다.

본 논문은 다음과 같이 구성된다. 제2장에서는 기존에 문서 요약이나 제목 생성에서 사용되었던 방법들과 최근에 딥러닝의 sequence-to-sequence 모델 등을 적용해 추상적인 문서 요약을 생성하는 방법들에 대해 설명한다. 제3장에서는 본 논문에서 제안하는 RNN과 강화 학습을 결합한 제목 생성 모델에 대한 설명과 보상 함수와 보상 값을 최대화하는 혼합 손실(mixed loss) 함수(Paulus et al., 2017)를 사용해 학습하는 과정을 설명한다. 제4장, 제5장에서는 성능 지표와 실험을 통하여 제안하는 기법이 기존의 기법들 보다 우수함을 보이고, 제6장에서는 결론과 추후 연구 과제에 관하여 설명한다.

2. 관련 연구

기존 문서 요약과 제목 생성 연구에는 대표적으로 추출 요약과 추상 요약이 있다. 과거에는 TextRank

(Mihalcea and Tarau, 2004), TF-idf(Salton and McGill, 1983)와 같은 알고리즘을 이용해 문서의 중요한 문장 또는 키워드를 선정해 추출하는 형태의 추출 요약 연구가 활발하게 진행되었다. TextRank 알고리즘은 Google의 PageRank(Haveliwala, 2002) 알고리즘을 응용한 기법으로서 문서 내의 중요한 정보를 파악해 추출할 수 있는 알고리즘이다. 추출 요약은 본문의 내용을 그대로 가져오는 형태로 추상적인 요약보다 요약 결과와 문법 등 높은 정확도를 가질 수 있다는 장점이 있다. TextRank를 활용한 연구에는 트위터의 타임라인을 이루고 있는 트윗들을 그래프로 모델링한 후, 그래프 기반의 랭킹 알고리즘을 적용하여 타임라인을 요약하는 기법(An et al., 2012)도 있었다. 최근 연구에서는 문장 임베딩 기반 TextRank를 이용한 문서 요약(Jeong et al., 2019)과 같은 연구도 진행되었다. Jeong et al.(2019)의 연구에서는 TextRank를 사용했을 때 문장이 가진 의미적인 정도를 고려하기 위해 문장 임베딩을 사용한 새로운 TextRank 알고리즘을 제안해 기존의 TextRank 알고리즘의 성능을 올리는 연구 결과를 보였다.

근래에는 딥러닝의 sequence-to-sequence 모델을 이용해 문서의 내용을 함축할 수 있는 추상적인 제목을 생성하는 연구가 활발하게 진행되고 있다. sequence-to-sequence 모델은 두 개의 RNN 시퀀스를 가지고 있어 입력 부분과 출력 부분의 길이를 다르게 설정할 수 있어 문서 요약 부분에서 좋은 결과를 보여 주고 있다. Rush et al.(2015), Chopra et al.(2016)과 같이 sequence-to-sequence 모델에 attention 메커니즘(Vaswani, 2017)을 적용한 모델을 사용해 문서를 대표할 수 있는 추상적인 제목 생성 모델 연구가 활발히 진행되었으며 DUC-2004 데이터 셋을 사용한 결과에서 ROUGE(Recall-Oriented Understudy for Gisting Evaluation, 이하 ROUGE) 점수 또한 좋은 결과를 보였다. Lee et al.(2016)의 연구에서는 sequence-to-sequence 모델과 attention 메커니즘을 적용한 모델로 본문에서 주요한 키워드를 추출해 그 키워드를 포함할 수 있는 제목 생성 모델을 제안해 좋은 결과를 보였다. Shin et al.(2019)의 연구에서는 sequence-to-sequence 모델과 attention 메커니즘을 결합한 모델을 사용해 다중 문서의 추상적인 요약문을 생성하는 좋은 결과를 보였다. 최근에는 sequence-

to-sequence 모델에 성능을 올리하고자 나온 sequence-to-sequence 모델에 정책 기반 학습을 결합한 모델이 있다[Paulus et al., 2017]. Paulus et al.(2017)의 연구에서는 최적의 정책을 찾기 위해 정책 기반 학습을 이용하는데 그중 REINFORCE 알고리즘(Williams, 1992)을 사용했다. Paulus et al.(2017)의 연구에서 강화 학습에 사용한 보상함수는 ROUGE 점수를 사용해 sequence-to-sequence 모델의 성능을 향상시켰다. 데이터 셋은 CNN/Daily를 사용해 기존의 모델보다 ROUGE 점수에서 우수한 결과를 보였으나 sequence-to-sequence 모델의 문제점 중 하나인 반복된 단어 생성이 자주 발생한다. 본 논문에서 제안하는 모델은 Paulus et al.(2017)의 연구에서 사용한 ROUGE 보상함수와 본 논문에서 제안하는 보상함수를 결합한 보상함수를 이용해 기존 연구의 문제점인 반복된 단어 생성 문제와 본문의 핵심 키워드를 포함한 제목을 생성할 수 있는 모델은 제안한다.

3. RNN과 강화 학습을 결합한 제목 생성 모델

제안하는 제목 생성 모델은 사람이 제목을 작성하

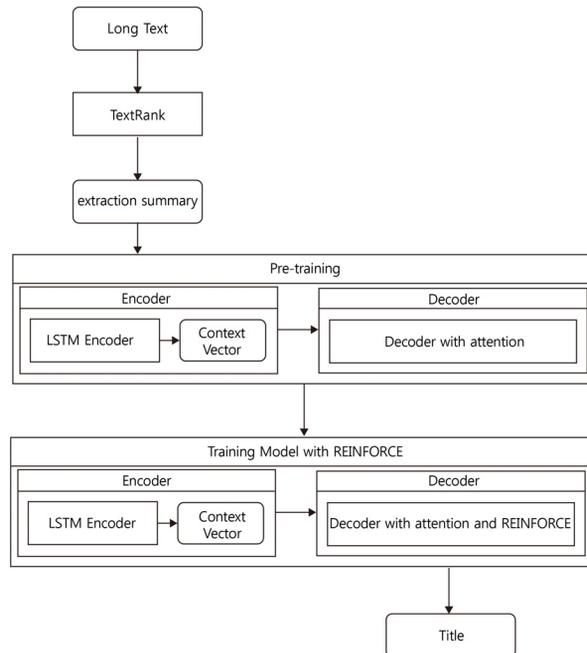
는 과정을 응용했다. 보통 사람들은 긴 본문의 내용을 대표할 수 있는 제목을 생성할 때에는 본문의 내용을 읽고 한 단계 요약하게 된다. 그 후 요약문에서 중요한 키워드를 추출해 한 문장으로 된 제목을 만든다.

〈Figure1〉은 제안하는 제목 생성 모델의 전체 흐름도로, 긴 본문으로부터 핵심 문장을 선정해 추출 요약한 후 이를 입력으로 받아 사전 학습을 거친다. 다음으로, 강화 학습을 적용한 모델로 학습을 진행해 본문의 내용을 함축한 제목을 생성한다.

3.1 추출 요약 모델

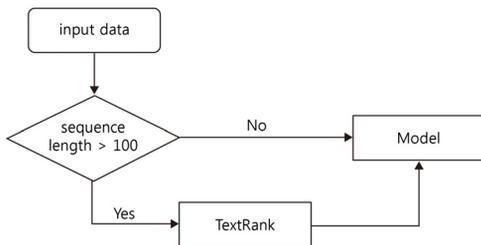
본 논문의 제목 생성 모델은 긴 본문의 내용을 입력으로 받아서 제목을 생성하는 모델이다. 하지만 기존의 sequence-to-sequence 모델은 긴 입력을 받았을 때 정보가 손실되어 모델의 성능이 떨어지는 문제가 발생한다. 이러한 정보 손실을 줄이고자 sequence-to-sequence 모델의 입력으로 들어가기 전에 추출 요약 모델을 통해 핵심 문장을 골라 추출 요약된 본문을 입력으로 넣고 학습하는 방법을 사용한다.

TextRank를 사용한 추출 요약 모델 부분은 〈Figure 2〉와 같은 흐름도를 보여준다. 추출 요약



〈Figure 1〉 Flow Chart of Model

모델은 입력 본문의 길이가 100보다 클 때 즉, 입력 본문이 100개 이상의 토큰으로 이루어져 있을 때 TextRank 알고리즘을 적용해 본문의 중요한 문장을 골라 100개의 토큰 이하로 본문을 추출 요약한다. 반면 본문의 내용이 100개의 토큰 이하면 추출 요약을 거치지 않고 바로 모델에 들어가게 된다. 추출 요약 모델은 모델이 학습하기 전 데이터의 전처리 과정과 이후 테스트 과정에서도 학습된 모델에 입력되기 전에 적용 된다.



〈Figure 2〉 Flow chart of the Extraction Summary Model

추출 요약 모델에서 사용한 TextRank 알고리즘은 문장 간 유사도를 기반으로 그래프(sentence similarity graph)를 구축한 후 많은 유입 링크를 가진 페이지를 구하는 PageRank 알고리즘을 응용한 방법으로 키워드와 핵심 문장을 선택한다. PageRank에서 각 node는 페이지, edge는 링크를 나타내며 PageRank에서 중요도는 다음 식 (1)과 같이 계산된다. 이 식에서 B_u 는 u로 가는 유입 링크 출발점들, N_v 는 각 노드 v의 링크 개수, v는 자신의 랭킹을 N_v 개로 나눠 링크로 연결된 페이지 u에 전달한다. 웹 페이지 유입의 c만큼은 링크, (1-c) 만큼은 임의적인 유입이라 가정한다.

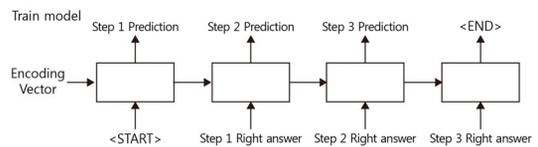
$$PR(u) = c \times \sum_{v \in B_u} \frac{PR(v)}{N_v} + (1-c) \times \frac{1}{N} \quad (1)$$

TextRank를 사용한 추출 요약 모델은 PageRank의 그래프에서 node를 문장으로 응용한 형태를 사용하여 본문 내에서 중요한 문장을 선택할 수 있도록 했다.

3.2 사전 학습

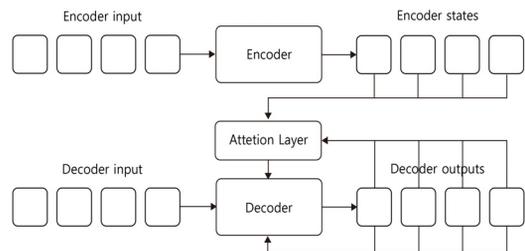
본 논문의 제목 생성 모델은 강화 학습의 효율을 올리

기 위해 사전 학습을 진행한다. 사전 학습에서 기본적으로 사용한 모델은 sequence-to-sequence 모델에 attention 메커니즘을 적용한 모델이다. 〈Figure 3〉은 기본적인 sequence-to-sequence 모델의 디코더 부분의 흐름도를 보여준다. 보통 학습 과정에서는 teacher forcing 기법[Williams and Zipser, 1989]을 사용하여 학습의 효율을 높이며, 문맥 벡터(Encoding Vector)를 초기값으로 각 디코더 스텝에서 실제 정답 단어를 입력으로 받아서 학습한다. 반면, 학습이 끝난 후 테스트 과정에서는 디코더의 이전 스텝의 값을 입력으로 받아 다음 단어를 예측해 나간다.



〈Figure 3〉 Teacher forcing.

Attention 메커니즘은 인코더에 긴 문장이 들어가 정보가 손실되는 단점을 보완할 수 있는 기법으로, 〈Figure4〉는 본 논문에서 사전 학습에 사용한 sequence-to-sequence 모델에 attention 메커니즘을 적용한 모델의 흐름도이다.



〈Figure 4〉 Attention Mechanism with model

Attention 메커니즘을 적용한 디코더는 아래와 같은 식 (2)를 통해 계산된다. e_{ij} 는 디코더가 i 번째 단어를 예측할 때 사용하는 이전 스텝의 벡터 s_{i-1} 값이 인코더의 h_j 와 유사도 값을 나타내는 값이며, α_{ij} 는 e_{ij} 값을 확률 값으로 변환한 것이며, \vec{c}_i 는 디코더가 i 번째 단어를 예측할 때 사용하는 문맥 벡터다.

$$e_{ij} = a(s_{i-1}, h_j) \quad (2)$$

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{T_x} \exp(e_{ik})}$$

$$\vec{c}_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j = F\alpha_i$$

사전 학습 알고리즘은 (Figure 5)와 같으며, 식 (3)의 MLE(Maximum Likelihood Estimation) 손실 값만 사용하여 추출 요약문을 입력으로 받아 학습한다.

$$L_{ml} = - \sum_{t=1}^n \log p(y_t^g | y_1^g, \dots, y_{t-1}^g, X) \quad (3)$$

또한 사전 학습에 적용한 attention 메커니즘은 모델의 디코더 부분에서 출력 단어를 예측하는 매 시점마다 인코더에서 전체 입력 문장을 다시 한 번 참고한다. 단, 전체 입력 문장을 전부 동일한 비율로 참고하는 것이 아닌 해당 시점에서 예측해야 할 단어와 연관이 있는 입력 단어 부분을 더 집중해서 보게 된다.

Algorithm1 Pre-training

Input: 2단계를 지난 뉴스 본문, 실제 제목
Output: 뉴스 제목
 epoch=0
 batch size=64
 $y^g = (y_1^g, y_2^g, \dots, y_n^g)$ // 실제 제목
While epoch < 20001
 for X, y^g in batch(본문, 제목)
 인코더에서 X를 입력으로 받아서 인코더의 마지막 스테이트 h_T 를 받음.
 디코더는 h_T 값과 매 스텝마다 attention layer를 지나 샘플링된 \hat{y} 출력
 L_{ml} 식을 통해 손실 값을 구해 모델을 업데이트.
 epoch += 1

(Figure 5) Algorithm of Pre-Training

3.3 강화 학습을 적용한 모델

본 논문의 제목 생성 모델은 sequence-to-sequence 모델의 성능을 올리고자 누적된 보상 값을 최대화하는 정책(optimal policy)을 찾는 REINFORCE 강화 학습을 결합하였다. 자연어 처리 분야와 같이 상태들과 동작들이 연속적인 공간에서 우수한 결과를 내는 정책에 기반한 REINFORCE 알고리즘은, 한 에피소드가 끝났을 때 한 번에 정책 신경망을 업데이트하는 방식으로 디코더

의 모든 예측이 끝난 시점에 적용할 수 있는 방법이다.

제목 생성 모델은 식 (5)와 같이 두 가지의 손실 함수를 결합한 혼합 손실 함수를 사용해 모델을 업데이트한다. 첫 번째는 식 (3)의 손실 함수를 사용해 MLE 손실 값을 구한다. MLE 손실 함수를 이용해 sequence-to-sequence 모델은 각 디코더 스텝에서 정답을 고를 수 있는 확률을 최대화하도록 학습한다. 두 번째는 식 (4)의 손실 함수는 강화 학습의 손실 함수로 각 디코더 스텝에서 정답을 고를 수 있는 확률을 최대화하면서 보상 값 또한 최대화 되게 학습을 진행한다.

$$L_{rl} = - \sum_{t=1}^n \log p(y_t^g | y_1^g, \dots, y_{t-1}^g, X) (-\log r(y^g)) \quad (4)$$

식 (5)의 혼합 손실 함수는 MLE 손실과 강화 학습의 손실 함수를 혼합한 손실 함수로서 γ 값을 조절해 두 손실 함수의 의존도를 결정한다.

$$L_{mixed} = \gamma L_{rl} + (1 - \gamma) L_{ml} \quad (5)$$

본 논문에서 혼합 손실 함수를 사용하는 이유는 식 (4)만 이용해 학습을 진행한다면 ROUGE 보상 값이 최대화되어 실제 정답과 유사한 단어를 사용해 제목을 생성할 수 있지만 불규칙한 어순으로 가독성이 떨어질 수 있다. 따라서 식 (3)과 같이 이전에 예측한 단어에 근거해 다음 단어를 예측할 수 있는 MLE 손실 함수를 혼합해 사용한다.

Algorithm2 mixed training

Input: 2단계를 지난 뉴스 본문, 실제 제목
Output: 뉴스 제목
 epoch=20000
 batch size=64
 $y^g = (y_1^g, y_2^g, \dots, y_n^g)$ # 실제 제목
 eta=0.000025
While epoch < 40001
 $\gamma = \text{eta} \times \text{epoch}$
 for X, y^g in batch(본문, 제목)
 인코더에서 X를 입력으로 받아서 인코더의 마지막 스테이트 h_T 값을 받음.
 디코더는 h_T 값과 매 스텝마다 attention layer를 지나 샘플링된 \hat{y} 를 모아 full sequence N을 출력.
 N은 L_{mixed} 식을 통해 손실 값을 구해 모델을 업데이트.
 epoch += 1

(Figure 6) Algorithm of Mixed Training

제목 생성 모델은 혼합 손실 함수를 이용해 <Figure 6>에서 보는 것과 같이 초기에는 MLE 손실 함수에 의존해서 학습을 진행한다. 하지만 epoch 값이 올라가면서 γ 값이 올라가 점점 강화 학습 손실 함수에 의존도가 높아지는 것을 볼 수 있다. 학습이 종료될 때에는 결국 강화 학습 손실 함수에 모두 의존한 형태로 모델이 학습된다. 실제 학습 과정에서는 20000부터 40000 epoch 까지 γ 값을 점점 올려 강화 학습 손실 함수에 의존도를 높이도록 했다.

3.4 보상함수

강화 학습의 목적은 최적의 정책을 찾는 것으로, 보상함수가 최대화되게 행동을 취하는 것이 최적의 정책이라고 볼 수 있다. 본 논문에서는 기존 sequence-to-sequence 모델의 단어 반복 문제를 보완하고, 본문에서 빈도수가 높은 단어가 생성된 제목에 포함되게끔 ROUGES와 again_score 두 가지 보상함수를 사용한다.

$$reward = ROUGES(y^g, y^s) \times again_score(X, y^s) \quad (6)$$

ROUGES 보상함수는 문서 요약, 챗봇에서 성능 평가 방법으로 자주 사용되는 문장 유사도 점수인 ROUGE 값을 사용해 구했다. ROUGES 함수는 정답 제목과 모델이 예측한 제목을 비교하여 정답 제목에 나타난 단어의 빈도수에 따라 점수가 올라간다. 반면, Again_score 값은 <Figure 7>과 같이 구해 ROUGES 보상함수와 곱해서 최종 보상 값인 식 (6)이 구해진다. 식 (6)은 식 (4)의 $r(y^s)$ 에 대입해 손실

값을 구하게 된다. <Figure 7>의 보상 알고리즘에서 단어의 반복만 검사하는 것이 아니라 모델이 예측한 단어가 입력 본문에서 빈도수 값(Term Frequency)을 최대화하도록 구성했다. 본문의 빈도수가 높은 단어를 제목에 포함한다면 더욱 본문의 내용을 함축한 제목이라 말할 수 있겠다.

4. 실험

실험은 tensorflow를 이용해 진행하였다. 제안하는 모델은 한글 제목 생성이 목적이기에 파이썬의 BeautifulSoup 라이브러리를 사용해 직접 크롤링한 한글 뉴스 기사 1만 건을 사용하였다. 1만 건의 데이터에서 테스트 데이터 100개를 제외한 나머지 데이터는 모두 학습에 사용하였으며 단어장은 1만 건의 뉴스 데이터를 모두 사용해 생성하였다. 수집한 데이터를 분석해 보았을 때 본문의 길이는 90에서 110 사이가 가장 분포가 높게 나왔다. 그래서 3.1절에서 소개한 것과 같이 100개 이상의 단어를 가진 본문은 추출 요약을 통해 요약문이 모델의 인코더 부분에 들어가게 된다. 데이터 분석에서 간혹 200개 이상의 단어를 가진 본문도 포함되어 있어 추출 요약 단계를 거쳐 학습이 효율적으로 진행되었다. 실험에서 사용한 뉴스 기사는 제목과 본문으로 <Table1>과 같이 나누었으며 실험 순서는 먼저 수집된 데이터를 전처리 과정을 거쳐 추출 요약 모델로 들어간다. 추출 요약된 본문은 사전 학습을 먼저 진행하게 된다. 사전 학습 후 강화 학습을 적용한 모델을 사용해 최종적으로 모델을 학습하게 된다. 수집한 뉴스 데이터의 원문은 <Table 1>과 같다.

<Table 1> The Original Text of News Article

본문	
[중시라인 11][김동환의 시선] 출연: 김동환 경제 칼럼니스트 / 경희대학교 국제지역연구원 원연구위원오늘 김동환의 시선이 머무는 곳은 '가성비'입니다. 시청자 여러분, 가격 대비 성능이 좋다고 할 때 흔히 가성비란 용어를 쓰죠? 사실 소비라는 것은 모두 가성비에 의해 결정됩니다. 가성비가 좋은 물건을 사면 내내 흐뭇하지만 (생략)	
제목	김동환의 시선 <가성비>

수집한 데이터는 본문과 제목에 각종 기호 및 특수 문자들이 포함되어 있다. 따라서 본 논문에서 전처리

Algorithm3 again score

Input: 예측 제목, 본문

Output: $0 < score < 1$ 의 보상 값

```
for 단어 in 예측 제목
  if(예측한 단어가 이미 생성한 단어라면)
    return 0
  else
    입력 본문에서 예측한 단어의 tf 값을 구해서 0.1
    과 곱해준다.
    tf_score = 예측한 단어 출현 빈도 / 본문의 길이
    score = (0.1)×tf_score
    return score
```

<Figure 7> Algorithm of Reward (Again Score)

과정에서는 <Table 1>에서 <Table 2>와 같이 특수 문자 및 각종 기호를 제거하는 작업을 진행하였다. 특수문자 등을 제거한 후에는 Konlpy의 Okt 형태소 분석기를 이용해 형태소 별로 토근화 한 단어장을 구축했다. 모델에 들어가는 단어 임베딩은 tensorflow의 내장 함수를 사용해 임베딩 했다.

<Table 2> After Preprocessing of Article

본문	
증시라인 김동환의 시선 출연 김동환 경제 칼럼니스트 경희대학교 국제지역연구원 객원연구위원오늘 김동환의 시선이 머무는 곳은 가성비입니다 시청자 여러분 가격 대비 성능이 좋냐고 할 때 흔히 가성비란 용어를 쓰죠 사실 소비라는 것은 모두 가성비에 의해 결정됩니다 가성비가 좋은 물건을 사면 내내 흐뭇하지만 (생략)	
제목	김동환의 시선 가성비

5.결과 및 성능 평가

테스트는 100개의 본문을 이용했다. 실험 결과의 예시는 <Table 3>과 같이 1번 모델은 단어의 반복이 심하게 생성되는 것을 볼 수 있다. 2번 모델은 위 결과 보다 조금 더 다양한 단어를 사용하였지만 마찬가지로 단어 반복 현상이 나타났다. 마지막으로 3번 모델은 중복된 단어가 조금 생성되었지만 기존의 두 모델보다 더다양한 단어를 사용해 제목을 생성한 결과를 볼 수 있다.

<Table 3> Result of model.

본문	
청년단체 '아판', SNS에 렌터카 때려 부수는 영상 공개...경찰 수사 착수사진은 스페인 마요르카섬에 정착한 크루즈선 [EPA=연합뉴스 자료사진] (파리=연합뉴스) 김용래 특파원 = 스페인의 세계적인 휴양지인 마요르카섬에서 대규모 관광객 유입에 반대하는 극좌 성향의 단체 회원들이 렌터카 업체를 습격해 차를 파손해 경찰이 수사에 나섰다. (생략)	
정답 제목	스페인 극좌단체, 마요르카서 "관광 반대" 렌터카 업체 습격
1번 모델	청년단체 때려 때려 때려 청년단체 때려 청년단체 렌터카 때려 때려
2번 모델	스페인 극좌단체 렌터카 렌터카 렌터카 때려 때려 때려 때려 때려
3번 모델	스페인 청년 극좌성향 청년단체 SNS에 렌터카 렌터카 때려 때려 때려

<Table 3>, <Table 4>, <Table 6>, <Table 7>, <Figure 8>에서 1번 모델은 sequence-to- sequence 모델에 attention 메커니즘을 적용한 모델[Vaswani, 2017], 2번 모델은 ROUGES 보상만 적용해 sequence-to-sequence 모델과 강화 학습을 결합한 모델[Paulus et al., 2017], 3번 모델은 본 논문의 모델로서 TextRank와 새로운 보상함수를 적용한 모델이다. <Table 4>에서 4번 모델은 TextRank에서 80개, 5번 모델은 50개 이하의 단어를 가지도록 추출 요약 과정을 거치고 보상함수는 제안한 방법과 같은 방법을 사용하였다.

<Table 4>는 문서 요약과 챗봇 분야에서 많이 사용하는 성능 지표로서 ROUGE를 사용했다. ROUGE-1은 정답 제목과 생성한 제목 사이의 uni-gram 일치도를 보여주며, ROUGE-2는 bi-gram 일치도를 보여준다. 마지막으로 ROUGE-L은 LCS(Longest Common Subsequence)를 사용한 값으로 가장 긴 동시 발생 값이다. 제안한 제목 생성 모델이 3가지 성능지표에서 모두 우수한 결과를 볼 수 있다. 추출 요약의 효율을 검증하기 위해 추출 요약 계수를 바꾸어 결과를 비교해 보았다. 추출 요약 과정에서 단어를 80과 50 이하로 요약했을 때 본문에서 제안한 100개의 결과보다 점수가 낮게 나오는 것을 볼 수 있다. 추출 요약은 앞장에서 설명과 같이 학습한 본문의 단어 분포가 90~110개로 최대한 본문의 내용을 살릴 수 있는 계수 100으로 설정했을 때 가장 좋은 결과를 보였다.

<Table 4> ROUGE Score of Model

	ROUGE-1	ROUGE-2	ROUGE-L
1번 모델	0.281	0.102	0.265
2번 모델	0.312	0.130	0.186
3번 모델	0.316	0.189	0.286
4번 모델	0.316	0.163	0.273
5번 모델	0.232	0.134	0.094

ROUGE 지표는 의미적인 정보는 반영하지 못하고 단순히 정답 제목과의 일치도를 검사하는 지표이기 때문에 사람이 직접 평가한 정성 평가 지표가 필요하다. 본 논문에서 정성 평가는 가독성과 본문과의 연관성을 중심으로 다음 <Table 5>와 같이 평가했다.

〈Table 5〉 Evaluation Indicator of Readability and Relevance

	가독성	연관성
1점	올바르지 않은 문법과 불규칙한 어순으로 읽을 수 없는 문장	본문과 상관없는 문장
2점	문법과 어순이 조금 맞지 않지만 어느 정도 읽을 수 있는 문장	본문의 등장하는 단어 등을 포함해 본문의 의미를 어느 정도 담고 있는 문장
3점	읽기 좋은 문장	본문의 의미를 함축해 본문을 대표할 수 있는 문장

평가는 20~40대 남성과 여성 10명의 사람에게 테스트 데이터 100개 중 30개의 예문을 선정해 정답 제목을 제외한 1번~3번 모델이 생성한 제목과 본문을 〈Table 6〉과 같이 보여주었고 진행하였다.

〈Table 6〉 Example of Evaluation

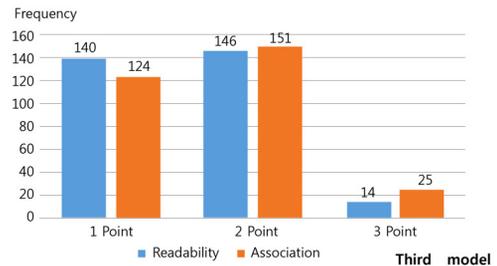
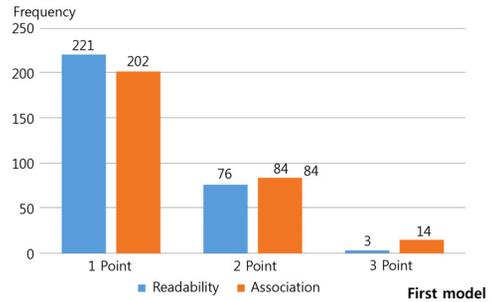
본문			
독일 드레스덴 양송이 통신헌 지난달 일부터 일까지 독일 프랑크푸르트에서는 페이퍼월드 라는 국제 박람회가 열렸다 예술가들에게 사랑받은 노트 브랜드 몰스킨 등 세계적인 기업을 포함해 여 기업체가 이 박람회에 참여했다 (생략)			
1번 모델	독일 한지 독일 독일 독일 한지 한지 한지 독일 한지	가독성	연관성
2번 모델	독일 독일 독일 독일 독일 독일 한지 한지 독일 한지	가독성	연관성
3번 모델	독일 한지 독일 프랑크푸르트 독일 드레스덴 독일 독일 페이퍼월드 이야기	가독성	연관성

평가 결과는 〈Table 7〉과 같이 가독성과 연관성 점수에서 제안한 모델이 가장 높은 점수를 받았다.

〈Table 7〉 Result of Readability and Relevance Score

	가독성	연관성
1번 모델	1.273	1.373
2번 모델	1.400	1.360
3번 모델	1.580	1.670

〈Figure 8〉에서 점수별 빈도수를 보면 3번 모델이 가독성과 연관성 점수에서 1점의 빈도수가 가장 낮은 것을 볼 수 있다. 특히 연관성 점수에서 2점의 빈도수가 가장 높게 나와 본문과 연관되는 제목을 생성할 수 있다는 것을 확인할 수 있었다.



〈Figure 8〉 Frequency According to the Score

6. 결론 및 추후 연구

본 논문에서 제안하는 제목 생성 모델은 학습 모델에 설정한 길이보다 긴 본문의 입력에도 추출 요약 모델을 통해 정보 손실을 최소화하며 모델을 학습시킬 수 있으며, sequence-to-sequence 모델의 반복된 단어 생성 문제를 보상함수를 통해 해결할 수 있음을 보였다. 또한 입력한 본문의 핵심 키워드를 제목에 포함해 본문의 내용을 함축할 수 있는 제목을 생성할 수 있었다.

본 논문의 실험 결과에서도 반복 단어 생성 문제가 발생하는 경우도 있었다. 하지만 양질의 한글 텍스트 데이터가 부족한 상황에서 본다면 동 데이터 대비 좋은 결과를 얻을 수 있겠다. 이는 추후 연구 과제로서 더 방대한 양의 신문 및 논문 데이터를 수집하고 많은 양의 단어장을 word2Vec을 통해 학습을 진행한다면 좀 더 좋은 성능을 낼 것으로 사료된다.

References

- [1] An, I. S., Kim, H. W., and Kim, H. J., "A User Timeline Summarization Technique using TextRank Algorithm", *Journal of KISS: Databases*, 2012. 8, pp. 238-245.
- [2] Chopra, S., Auli, M., and Rush, A. M., "Abstractive sentence summarization with attentive recurrent neural networks", *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2016. pp. 93-98.
- [3] Haveliwala, T. H., "Topic-sensitive pagerank", *Proceedings of the 11th international conference on World Wide Web*, ACM, 2002, pp. 517-526.
- [4] Jeong, S. W., Kim, J. T., and Kim, H. S., "Document Summarization Using TextRank Based on Sentence Embedding", *Journal of KIISE*, 2019. 3, pp. 285-289.
- [5] Lee, H. G., Lee, S. H., Kim, J. T., and Kim, H. S., "Generating End-to-End Document Title using Sequence to Sequence Model and Keyword", *Korea Information Science Society*, 2016. 12, pp. 452-454.
- [6] Mihalcea, R. and Tarau, P., "TextRank: Bringing order into text", *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, 2004, pp. 404-411.
- [7] Mikolov, T., Karafiát, M., Burget, L., Černocký, J., and Khudanpur, S., "Recurrent neural network based language model", *Eleventh Annual Conference of the International Speech Communication Association*, 2010.
- [8] Paulus, R., Xiong, C., and Socher, R., "A deep reinforced model for abstractive summarization", arXiv preprint arXiv:1705.04304, 2017.
- [9] Rush, A. M., Chopra, S., and Weston, J., "A neural attention model for abstractive sentence summarization", arXiv preprint arXiv:1509.00685, 2015.
- [10] Salton, G. and McGill, M. J., "Introduction to modern information retrieval", McGraw-Hill, 1983.
- [11] Shin, Y. M., Noh, Y. S., and Park, S. Y., "Abstractive Multi-Document Summarization via Self-Attention based Multi Document Encoder," *The Korean Institute of Information Scientists and Engineers*, 2019. 6, pp. 527-529.
- [12] Sundermeyer, M., Schlüter, R., and Ney, H., "LSTM neural networks for language modeling", *Thirteenth Annual Conference of the International Speech Communication Association*, 2012.
- [13] Sutskever, I., Vinyals, O., and Le, Q. V., "Sequence to sequence learning with neural networks", *Advances in NIPS*, 2014.
- [14] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Łukasz, K., and Polosukhin, I., "Attention is all you need", *Advances in Neural Information Processing Systems*, 2017, pp. 5998-6008.
- [15] Williams, R. J. and Zipser, D., "A learning algorithm for continually running fully recurrent neural networks", *Neural Computation*, Vol. 1, No. 2, 1989, pp. 270-280.
- [16] Williams, R. J., "Simple statistical gradient-following algorithms for connectionist reinforcement learning", *Machine Learning*, Vol. 8, No. 3-4, 1992, pp. 229-256.

■ Author Profile



김 우 생

텍사스 오스틴 주립대학교 학사, 미네소타 주립대학교 석사 및 박사 학위를 취득 하였으며, 현대전자 근무를 거쳐 현재는 광운대학교 소프트웨어 학부 교수로 재직 중이다. 주요 관심분야는 빅데이터, 딥러닝 등이다.



조 성 민

광운대 컴퓨터과학과에서 석사 학위를 2020년에 취득하였으며, 주요 관심분야는 딥러닝, 자연어 처리, 챗봇 등이다.