가사-제목 예측모델 설계 및 구현을 통한 text summarization model 이해

Korea University COSE461 Final Project

김상엽
Department of Computer Science
Team 18
2018320212

Abstract

노래의 제목은, 노래가 전달하고자 하는 메세지인 가사로부터 정해진다. 한마디로, 제목은 가사의 요약이라고 표현할 수 있을 것이다. 본 프로젝트에서는 자연어처리 task 중 하나인 text summarization를 노래에 접목하여, 가사로부터 제목을 예측하는 모델을 구현해보고자 한다. 가사-제목 데이터셋으로 부터, text summarization model들을 설계해보고 이해하는 것이 이 프로젝트의 목적이다. 더불어, 구현 과정에서의 NLP task 일련의 과정을 한층 깊게 이해해보고 관련 라이브러리와 패키지 사용에 익숙해지고자 한다.

1 Introduction

1.1 Motivation

노래의 제목은 그 노래가 전달하고자 하는 메세지를 함축적으로 나타낸다. 그리고 그런 노래의 메세지의 대부분은 가사를 통해 전달된다. 따라서 노래의 제목은 가사를 반영하고 있으며, 가사의 요약이라고 할 수 있다. 본 프로젝트는 이런 노래의 가사와 제목과의 연관성이, NLP task 중 하나인 text summarization을 통해 설명이 될 수 있을 것이라는 직관에서 동기를 얻어 시작하게 되었다. 가사로부터 제목을 예측할 수 있는 기계학습 모델들을 구현하해 학습해보고 그 결과를 비교함으로써 text summarization model의 작동방식을 이해해보고자 한다.

1.2 Problems

이 프로젝트를 통해 이루고자 하는 목표는 두 가지다. 첫 번째는 모델의 구현과 학습을 통한, 결과 확인 및 모델 간 성능의 비교이다. 이 프로젝트에서는 두 가지 가사-제목 예측모델을 구현해보고, 같은 dataset으로 학습 및 테스트를 진행하였다. LSTM 기반의 seq2seq 모델과 transformer 기반의 pretrained t5-base 모델을 각각 구현해 학습을 진행해 비교해 볼 수 있었다.

두 번째는 이를 진행하는 과정에서 자연어처리 summarization task의 일련의 pipeline을 이해하고 체험해보는 것이다. task 선정부터, 데이터 수집, 데이터 전처리, 모델 build, training, evaultaing, test 등등의 과정을 직접 코드로 작성해보며 이해해 보고자 한다. LSTM 기반의 seq2seq 모델을 직접 구현해보며, 이 전체적인 과정을 직관적으로 이해 할 수 있었다.

1.3 Challenges

- dataset : 당초 어려울 것이라고 생각했었던 데이터 수집에는 어려움이 없었으나, 사실 노래의 가사와 제목은 연관이 있을 뿐 완벽한 요약이라고 볼 수 없다. 따라서 기존 문서 요약과는 결이 다른 데이터이며, 완벽히 문서 요약에 걸맞은 데이터가 아니기 때문에, summarization이 제대로 수행되지 않을 수도 있을 것이다.

- 제한된 학습 환경 : 별도의 GPU없이 Google Colab GPU로 학습환경이 제한되어, 더 좋은 학습환경에서 더 큰 모델을 학습을 할 수 없었다. 이에 맞춰, 적당한 크기라고 생각되었던 LSTM 모델과 t5-base모델을 학습하였고 GPU 및 메모리크기에 맞추어 epoch, batch size 등의 hyperparameter를 수정하여 학습을 진행할 수 있었다.

2 Related Work

- Neural Abstractive Text Summarization with Sequence-to-Sequence Models [1] [1]
  본 프로젝트의 baseline 모델인 seq2seq 모델 구현을 위해 읽었던 논문이다. 그 동안 text summarization을 위해 만들어진 seq2seq 모델들의 리뷰, 발전과정과 성능 비교에 대한 이야기를 하며 이를 통해 task에 대한 전반적인 이해를 할 수 있었다. 특히, abstractive 요약 모델과 lstm 기반의 seq2seq 모델과, transformer 기반의 t5 모델을 만들어 비교해보려는 이번 프로젝트의 목적 수립에 큰 영향을 미쳤다.

- Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer [2] [2]
  본 프로젝트의 baseline인 LSTM 기반 seq2seq 모델과는 달리, 비교 모델로 사용한 T5 모델은 multi-head self-attention layer과 feed forward network를 사용한 Transformer encoder-decoder 구조의 모델이다. baseline 모델의 성능이 생각했던 것보다 좋지 않아, transformer 기반의 모델을 깊게 이해해보고, 실제로 구현 및 학습을 해보기 위해 참고하였다.

- Text Summarization with Pretrained Encoders [3] [3]
  본 프로젝트의 motivation이 되었던 논문으로 NLP 분야의 sota라고 할 수 있는 BERT 기반의 summarization 모델에 대한 논문이다. 학습환경이 갖추어지지 않아, 실제 학습을 시켜보진 못했지만 summarization이라는 task에 대한 이해와, NLP 학습환경 구축 등에 있어서 큰 도움이 되었다.

3 Approach

3.1 abstractive summarization

text summarization에는 두 가지 방식이 있는데, input text에서 가장 중요하다고 생각되는 문장을 뽑아내는 extractive summarization과 input text를 그대로 인용하지 않고 새로운 문장을 만들어 내는 abstractive summarizaion이 있다. 노래의 제목은 가사로부터 유의미한 추론을 통해 예측을 해야할 것으로 생각해 abstractive summarization model를 설계하기로 방향을 잡았다.

3.2 Basline model : LSTM

baseline으로는 원시모델이라고 할 수 있는 LSTM을 이용한 seq2seq 모델으로 진행하였다. 제안된지 오래된 고전적인 모델인 만큼, 직접 구현해보기도 쉽고 분명히 최근의 모델들에 비해서는 결과가 좋지 않을 것이기 때문에 결과분석에도 용이할 것이라고 생각했다. 확실한 비교를 위해 attention layer는 사용하지 않을 것이다.

3.3 T5

LSTM 기반 seq2seq모델 학습을 통해 NLP task의 전체적인 과정을 이해하고 어떻게 진행되는지 이해할 수 있었다. 하지만, 예상했던 성능보다 훨씬 좋지 않은 결과를 확인하였다. 어떠한 점에서 성능이 좋지 않았는 지를 알아보고, 다른 모델과의 비교를 위해 새로운 모델에 데이터셋을 학습시켜 비교해보기로 하였다. LSTM 기반 seq2seq 모델에서는 attention layer를 사용하지 않았으며, NLP분야의 획기적인 발전이 transformer 모델 이후로 나왔다는 점을 고려하여 T5을 사용하기로 하였다. BERT와는 달리, text-to-text이기 때문에, summarization task에 더 적합하다고 생각했다. 모델의 설계와 구현 단계에서는 huggingface에 잘 정리된 T5 라이브러리와 package를 사용하여 수월한 학습을 할 수 있었다.

[1] https://arxiv.org/pdf/1812.02303.pdf
[2] https://arxiv.org/abs/1910.10683
[3] https://arxiv.org/abs/1908.08345

3.4 두 모델 간의 비교

baseline인 LSTM기반의 seq2seq모델과 T5 두 모델간의 차이는 분명히 존재한다. summarization 분야에서 T5 모델의 성능이 월등하게 좋다는 것은 이미 자명한 사실이기 때문에, 어떠한 점에서 그런 성능차이가 비롯되었는지, 이로부터 LSTM의 성능을 개선하기 위해서 어떤 것을 할 수 있을지 에 대해 논의해보고자 한다.

4 Experiments

4.1 Data

kaggle에 오픈소스로 공개된 'Music Dataset : 1950 to 2019' [4] 으로부터의 데이터를 사용하였으며, 총 23689 개의 unique한 노래 제목과 가사, metadata로 이루어진 데이터셋이다. 이 중 track name 과 lyrics column을 추출하여 학습에 사용하였다. 데이터 전처리 과정에서 알 수 있었던 데이터셋에 대한 성질은 다음과 같았으며, 이는 max_title_len, epoch 등등의 hyperparameter를 정하는데 영향을 미쳤다.

- 총 23689개의 unique lyrics-title pair로 구성
- lyrics의 최대 길이 : 199
- title의 최대 길이 : 17
- 데이터셋의 lyrics의 약 90% 정도가 길이가 150 정도 이내었으며, title의 약 99%가 10 이내의 길이를 가지고 있었으며, 이에 따라 hyperparameter를 정의하였다.

4.2 Evaluation method

두 가지 evaluation method를 사용하였다. 첫 번째는, summarization 모델 뿐만 아니라 자연어처리 분야 모델 전반의 성능평가에 사용되는 ROGUE 지표를 통한 evaluation이다. 노래 제목의 길이는 대부분 10자 이내이며, 각 단어가 유기적으로 연관이 있기보다는 노래를 대표하는 몇 개의 단어로 이루어져 있는 경우가 대부분이기 때문에, ROUGE-1 score를 사용하여 평가하였다.

하지만, ROGUE score로만 설명하기는 어려운 부분이 있었다. 원래 제목과 다른 단어를 사용하여도, 분명한 의미가 전달되어 노래를 적절히 나타내고 있는 경우도 있었으며, LSTM 기반의 seq2seq 모델에서는 ROUGE 지표를 사용하는게 무의미할 정도로 prediction이 제대로 되지 않았다. 따라서 두 번째 evaluation 방법으로, 전체 testset에서 일부분을 추출하여 예측한 제목이 원래 제목의 의미나 가사를 반영하고 있는지를 정성적으로 평가하기로 하였다. t5 모델에서도 마찬가지로 정성적인 evaluation을 통해 LSTM과 어떤 차이가 발생하는 지를 알 수 있었다.

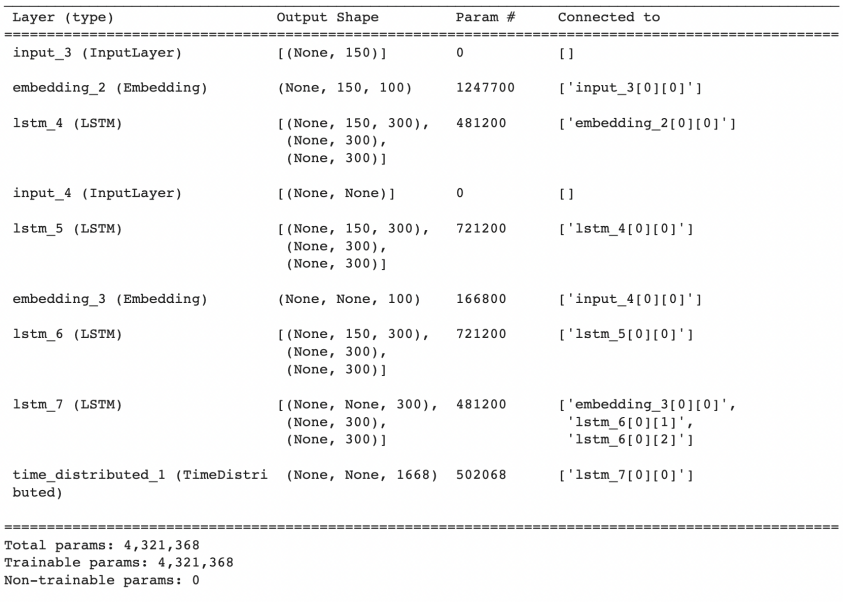
4.3 Experimental details

두 가지 모델에 대하여, 각 모델의 특징과 training hyperparameter은 다음과 같다.

1. LSTM 기반 seq2seq 모델 [5]
    - model layer : encoder로는 3개의 LSTM 층을 decoder로는 하나의 LSTM을 사용하여, encoder의 ouput이 decoder로 연결되는 구조이다. 총 parameter 수는 4,321,368 개였다.
    - learning rate : 0.001
    - optimizer : rmsdrop optimizer
    - loss function : sparse categorical cross entrophy
    - epoch : 50
    - batch size : 256
    - 총 학습 시간 : 1시간 42분 32초
    - 각 hyperparameter는 학습시간과 학습환경(google colab)을 고려하여 정하였다.
2. pretrained T5

[4] https://www.kaggle.com/datasets/saurabhshahane/music-dataset-1950-to-2019

[5] 코드의 구현 및 설계는 https://github.com/sujanshirol/Test-Summarization-LSTMs를 참고하였다.



```
Layer (type)                 Output Shape              Param #   Connected to
==================================================================================================
input_3 (InputLayer)         [(None, 150)]             0         []

embedding_2 (Embedding)      (None, 150, 100)          1247700   ['input_3[0][0]']

lstm_4 (LSTM)                [(None, 150, 300),        481200    ['embedding_2[0][0]']
                              (None, 300),
                              (None, 300)]

input_4 (InputLayer)         [(None, None)]            0         []

lstm_5 (LSTM)                [(None, 150, 300),        721200    ['lstm_4[0][0]']
                              (None, 300),
                              (None, 300)]

embedding_3 (Embedding)      (None, None, 100)         166800    ['input_4[0][0]']

lstm_6 (LSTM)                [(None, 150, 300),        721200    ['lstm_5[0][0]']
                              (None, 300),
                              (None, 300)]

lstm_7 (LSTM)                [(None, None, 300),       481200    ['embedding_3[0][0]',
                              (None, 300),                       'lstm_6[0][1]',
                              (None, 300)]                       'lstm_6[0][2]']

time_distributed_1 (TimeDistri  (None, None, 1668)     502068    ['lstm_7[0][0]']
buted)

==================================================================================================
Total params: 4,321,368
Trainable params: 4,321,368
Non-trainable params: 0
```

Figure 1: LSTM model layer

- model : T5-base
- batch size : 16
- learning rate : 1e-4
- 총 학습 시간 : 4시간 3분 57초
- 각 hyperparameter는 학습시간과 학습환경(google colab)을 고려하여 정하였다.

4.4 Results

- 모델 별 prediction의 결과는 Figure 2와 같다.
- 각 모델 별 ROGUE score는 Figure 3와 같았으며, 실제로 LSTM seq2seq에 비해 T5가 더 높은 ROUGE score를 기록한 것을 확인할 수 있었다.

5 Analysis

5.1 LSTM기반 seq2seq 모델

예상은 했었지만, 생각보다 훨씬 학습 결과가 좋지 않았다. 특히나 모든 prediction의 결과가 'love' 라는 한 단어로 예측해서 ROGUE 점수와 같은 evaluation이 의미가 없을 정도였다. 또 학습이

amed: 0	predicted titles	original titles
0	love	guess
1	love	roads
2	love	lonely gone
3	love	never really wanted
4	love	naked sunday
...	...	...
1733	love	running back
1734	love	big enough

Jnnamed: 0	Generated Text	Actual Text
0	i'll cry	finito la mouzika
1	i'm the moon	carolina moon
2	the stranded road	kanugona galano
3	blue skies	blue skies
4	a love song	hum tujhse mohabbat kar
...	...	...
5669	body is rockin'	sex, love & money
5670	i'll get better	life is better
5671	hip hop thugster	eazy-duz-it

Figure 2: 각 모델별 prediction 예시, 왼쪽부터 LSTM, T5

	f1	precision	recall
t5	0.209504	0.227486	0.223716
lstm	0.042482	0.062716	0.034465

Figure 3: 각 모델 별 ROUGE score

끝나갈때 쯤에는, test set과 training set의 loss값이 오르는 등 제대로 된 학습이 되었다고는 할 수 없었다. 왜 이런 학습결과가 나왔는지에 대한 이유는 다음과 같이 정리했다.

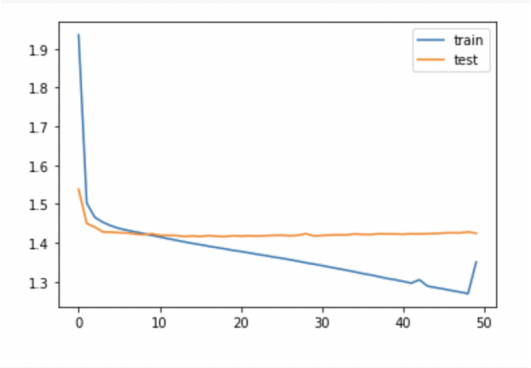


Figure 4: LSTM model plot

1. 원시 모델 : LSTM 기반의 seq2seq은 지금은 잘 쓰이지 않는 모델로, 모델의 성능을 월등히 높여주는 attention layer나 pretrained/fine tuning 등의 기법을 사용하지 않았기 때문에, 좋은 성능을 기대할 수 없는 모델이다. 실제로 대조군으로 학습했던 T5 모델에서는 baseline인 LSTM기반의 seq2seq 모델 보다는 훨씬 나은 evaluation 결과를 보여주었다.

2. 짧은 학습량 : 첫 학습은 epoch를 10으로 세팅하고 학습을 진행하였고, epoch 50으로 학습했을 때와 결과는 마찬가지로 모든 prediction이 'love'라는 하나의 단어였다. epoch를 100으로 설정하고 진행한 학습에서도 마찬가지였다. 학습환경의 제한과 학습시간 때문에, 그 이상의 epoch에서 학습을 진행해보지는 못했지만 짧은 학습량으로 인해 이런한 결과가 나왔을 가능성도 충분하며 충분한 시간이 주어진다면 최소한 하나의 단어로 모든 prediction이 이루어지지는 않을 것이라고 생각한다.

3. 데이터셋의 특징 : 학습에 사용했던 데이터셋은 1950년도부터 2019년까지의 빌보트 차트의 가요에 대한 가사와 제목에 대한 데이터이다. 가요의 특성 상, 사랑이라는 주제를 가진 노래가 많을 것이며 이런 데이터셋의 특징으로 인해 학습과정에서 'love'라는 단어의 토큰에 대해 오버피팅 되었을 가능성도 있다.

5.2 T5

baseline인 LSTM기반의 seq2seq모델 과는 다르게, T5 모델에서는 정상적인 형태의 학습과 추론이 이루어졌다. 실제 노래의 제목을 정확하게 예측해내는 경우도 많았으며, 노래의 제목과는 겹치는 글자가 없어 ROGUE 점수는 낮았지만 의미적으로 원래의 노래제목과 비슷한 경우도 있었다. 이를 통해, 단순히 단어 에 대한 학습이 이루어진 것이 아니라 노래와 제목의 관계, 노래가 전하고자 하는 메세지 등에 대한 학습 또한 이루어졌음을 알 수 있었다.

그렇다면 어떤 점이 T5와 baseline 모델에 유의미한 차이를 이끌어 냈을까? 앞서 말했듯이, LSTM은 많이 발전된 다른 모델들로 인해 잘 쓰이지 않는 원시모델이며 짧은 학습량으로 인해 학습이 잘 되지 않았다고 생각한다. T5 모델에서는 이를, self-attention 및 feed forward network를 포함한 transformer 모델을 사용하여 모델의 성능을 높일 수 있었으며 부족했던 학습량 역시 pretrain / fine-tuning을 이용해 보완할 수 있었다.

	Unnamed: 0	Generated Text	Actual Text
275	275	the duchess of cambridge	duchess
2223	2223	last night and goodbye	tonight's not the night
1846	1846	a whisper of the north	whispers of the north
3883	3883	i fall in love with you	when i fall in love
856	856	my heart never quit	quit playing games (with my heart)
2495	2495	lonley	love me or leave me
2945	2945	ain't got no way	night calls
417	417	i'm with you all night	one more night
4090	4090	pontoon	bewitched
3498	3498	i'm not fallin' anymore	fugitive life

Figure 5: T5 model에서 random sampling한 10개의 prediction : "i fall in love with you"와 "when i fall in love" 과 같이 글자 그대로 actual test와 유사한 단어도 있으며, "i'm not fallin' anymore"과 "fugitive life"와 같이 단어 자체는 다르지만 예측한 제목과 실제 제목 사이의 의미가 매우 유사한 단어 또한 존재함을 확인할 수 있다.

6 Conclusion

6.1 프로젝트 한계

본 프로젝트에서는, 노래의 가사와 제목의 연관성과 text summarization에 접목하여 노래의 가사-제목 예측모델을 설계 및 구현하였다. 그 과정에서 시간 및 자원 등의 부족으로 인한 프로젝트의 한계, 시도하지 못했던 것들 및 아쉬웠던 점들이 존재하였다. 우선, 데이터의 부족이다. NLP task에 있어, 어느정도의 데이터를 학습시켜야 할 지에 대한 명확한 기준이 정해져 있지 않기 때문에, 시행착오를 겪어가며 데이터의 양을 정할 수 밖에 없었다. kaggle의 무료 데이터셋의 전부(약 2만3천 쌍의 가사-제목)를 학습시켰지만 더 풍부한 데이터를 사용했으면, 더 직관적인 결과를 얻을 수 있었을 것이라고 생각한다. 두 번째는, 프로젝트 구현 및 학습환경에 대한 아쉬움이다. 프로젝트를 진행하면서 자연어처리와 관련한 다양한 라이브러리와 패키지, colab 학습환경 등 아직 낯설고 익숙하지 않았던 학습환경으로 인해 프로젝트의 진행에 있어 더딘 면이 있었다. 마지막으로, proposal에서 제시하였던 가사 이외의 feature를 추가하여 성능 향상 기대에 대한 시도는 시간 및 자원의 한계로 인해 시도하지 못했던 것이 개인적으로 아쉬웠다.

6.2 결론 및 의의

그럼에도 NLP task 중 핵심 task라고 할 수 있는 text summarization에 대해 처음부터 끝까지의 과정을 직접 설계해보고 구현하는 과정에서 자연어처리 및 머신러닝 task에 대한 insight를 확고히 할 수 있었다. 데이터 수집 및 전처리 과정, 모델 설계 및 구현, evaluation 등 각 단계에서 어떠한 작업이 수행되는지에 대해 이해할 수 있었으며 무엇보다도, 이 모든 일련의 과정 끝에 나오는 결과를 확인해볼 수 있었다는 것에 큰 의의가 있다. 원래 목표로 했던 seq2seq와 같은 고전적인 모델에 대한 학습 이외에도 더 나아가 비교적 최신 모델이라 할 수 있는 T5 모델을 사용하였고, 관련 라이브러리와 패키지 등을 사용하여 학습을 진행했던 것은 좋은 경험이었다. 또한, 평소에 자주 즐겨듣던 노래로부터 동기를 얻어 내가 공부하고 있는 자연어처리라는 분야에 접목해 흥미롭게 프로젝트를 진행할 수 있었다. 앞으로도 이처럼 NLP task에 대해 더 많이 관심을 가지고 프로젝트를 진행할 것이며, 이번 프로젝트를 통해 얻은 자연어처리 task에 대한 insight가 적극 활용할 수 있기를 기대한다.

References

[1] Tian Shi, Yaser Keneshloo, Naren Ramakrishnan, and Chandan K. Reddy. Neural abstractive text summarization with sequence-to-sequence models. *CoRR*, abs/1812.02303, 2018.

[2] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *CoRR*, abs/1910.10683, 2019.

[3] Yang Liu and Mirella Lapata. Text summarization with pretrained encoders. *CoRR*, abs/1908.08345, 2019.