Fine-tune BERT for Extractive Summarization

Yang Liu
Institute for Language, Cognition and Computation
School of Informatics, University of Edinburgh
10 Crichton Street, Edinburgh EH8 9AB

발제자: 안제준

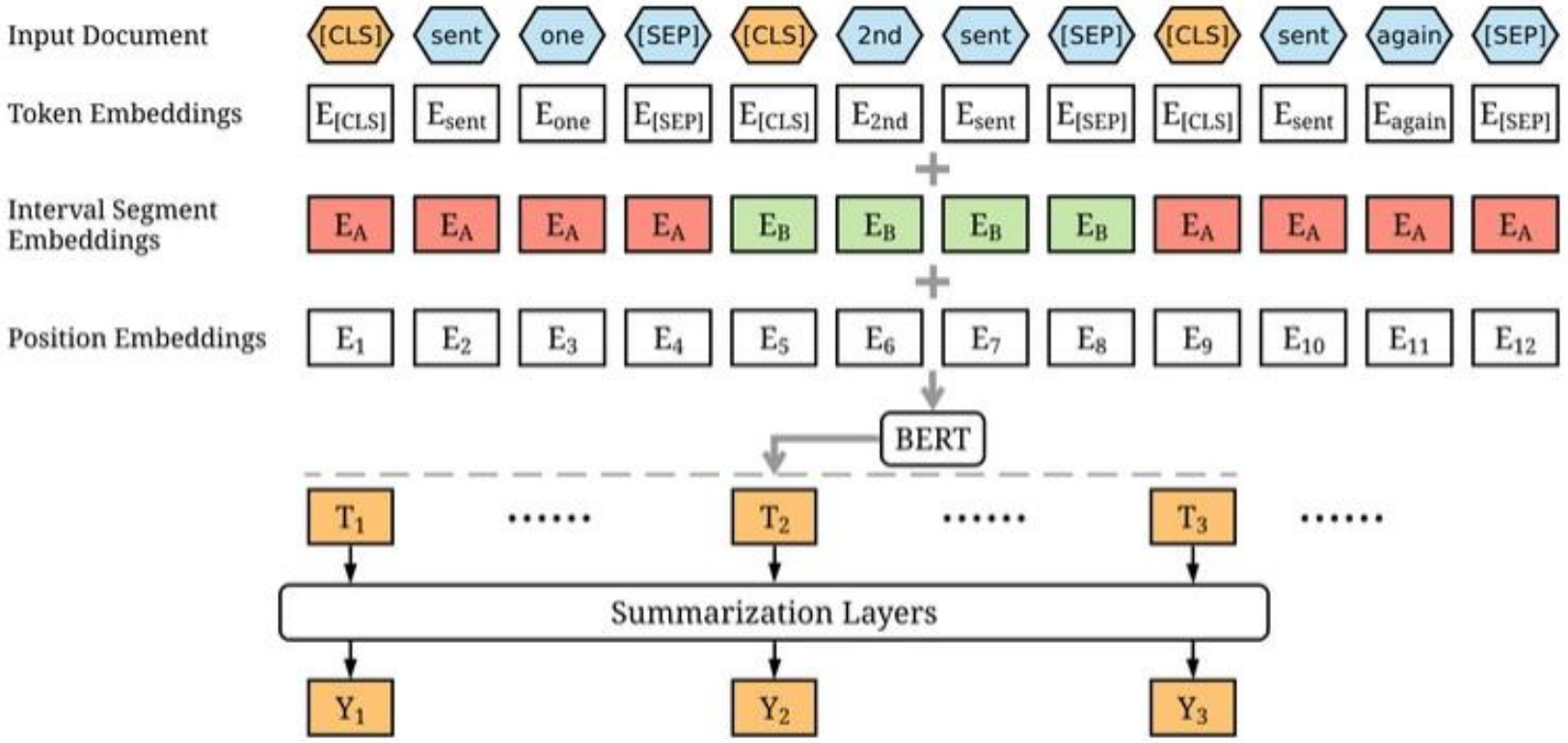
목차

Introduction

- 문서요약
  - 생성 요약 : 주어진 텍스트를 의역하여 문장 생성 (즉, 요약문을 생성할 때 본문에 없는 단어나 내용을 사용할 수 있음 : BART)
  - 추출 요약 : 본문에서 중요한 문장만을 추려내는 방법 (BERTSUM)
- 해당 논문은 추출 요약에 집중하여 연구를 진행
- BERT가 거대한 데이터 세트에 대한 사전 훈련과 복잡한 기능을 학습하기 위한 강력한 아키텍처이므로 복잡한 문제인 추출 요약의 성능을 더욱 향상시킬 수 있다는 가정
- 추출 요약 작업에서 BERT를 사용하고 CNN/Dailymail, NYT 데이터 세트에서 결과를 보여주는 다양한 변형을 설계하는 데 중점을 둠

모델 구조

- d (document) : $[sent_1, sent_2, \ldots, sent_m]$ 으로 구성된 문서
- 추출 요약은 $sent_i$ 마다 $Y_i \in \{0,1\}$ 로 summary에 포함시킬지 말지 판단



Methodology-Extractive summary

Encoding Multiple Sentences

- 기존 BERT에서는 각 문장들 사이에 [SEP] 토큰만을 사용하여 문장들을 구분

- 차이점 : 해당 논문은 각 문장의 앞에 [CLS] 토큰을 추가하여 각 문장들의 특징을 해당 토큰에 담을 수 있도록 수정

Methodology-Extractive summary

Interval Segment Embeddings

- 기존 BERT에서는 통상 두 개의 문장이 A 문장과 B 문장으로 구분되어 입력

- 차이점 : 추출 요약 태스크에서는 두 개 이상의 문장을 입력
  - Interval segment Embedding을 통해 두 개 이상의 문장에 대해서도 세그먼트 임베딩을 진행
  - Ex) 문장 1, 문장 2, 문장 3, 문장 4가 주어졌다면, 세그먼트 임베딩은 A, B, A, B 식으로 번갈아가며 문장을 구분

Methodology- Fine-tuning with Summarization Layers

- BERTSUM 출력값 상단에 Summarization layer를 추가하여 문서 요약에 필요한 특징을 추출
- 이를 통해 각 문장별로 요약 정보에 포함할지 여부를 판단

- Summarization layer를 구성하는 방법에는 3가지가 있습니다.
  1. 단순 분류 레이어 (FFN + Sigmoid), Simple Classifier
  2. 문장간 트랜스포머, Inter-sentence Transformer
  3. LSTM, Recurrent Neural Network

Methodology- Fine-tuning with Summarization Layers

Simple Classifier

- Linear layer 하나를 사용
- 단순 Linear layer 1개를 이용
- Layer를 통과한 output에 활성화 함수 Sigmoid를 취한 값을 target으로 사용
- target을 정답 label과 Binary Classification Entropy을 통해 loss를 계산

$$\hat{Y}_i = \sigma(W_o T_i + b_o)$$

Methodology- Fine-tuning with Summarization Layers

Inter-sentence Transformer

- Transformer 구조를 활용
- BERT에서도 Transformer 구조가 존재하지만, 이때 Attention은 문장 간이 아니라 토큰 간에 작용
- 따라서 문장 간의 관계를 파악하기 위해 Inter-sentence Transformer를 도입하여 Summarization Layer로 사용
- 이러한 Transformer layer은 Summarization을 위해 문서 수준의 특징을 뽑아내고 오직 문장들 간에 Transformer를 적용
- 그 후 위와 같은 방식으로 Sigmoid를 취한 단순 분류 Linear layer를 통과하여 정답 label과 비교

$$\tilde{h}^l = \mathrm{LN}(h^{l-1} + \mathrm{MHAtt}(h^{l-1})) \qquad (2)$$

$$h^l = \mathrm{LN}(\tilde{h}^l + \mathrm{FFN}(\tilde{h}^l)) \qquad (3)$$

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

Methodology- Fine-tuning with Summarization Layers

Recurrent Neural Network

- LSTM을 활용
- BERT의 last hidden layer에 RNN을 활용하면, 성능이 좋을 수 있다는 논문이 있기에 실험을 진행함
- 다만, 훈련과정을 안정화시키기 위해 단순 LSTM이 아닌 pergate layer normalization을 사용

Experiments

- Implementation Details
  - 모든 모델은 50,000 steps on 3 GPUs를 사용하여 train했고 1000step마다 model checkpoint 저장
  - 중복 문장을 줄이기 위해서 요약문 내의 후보 문장끼리 겹치는 trigram이 존재하면 해당 후보를 스킵하는 방식인 Trigram Blocking 방식을 사용
  - End-of-sequence token이 나올 때까지 decoding & Trigram Blocking

- Summarization Datasets
  - CNN/ Daily - 뉴스 본문과 associated highlights 포함
  - New York Times Annotated Corpus - 기사와 abstractive summaries 포함

Experimental Results

Model	ROUGE-1	ROUGE-2	ROUGE-L
PGN*	39.53	17.28	37.98
DCA*	41.69	19.47	37.92
LEAD	40.42	17.62	36.67
ORACLE	52.59	31.24	48.87
REFRESH*	41.0	18.8	37.7
NEUSUM*	41.59	19.01	37.98
Transformer	40.90	18.02	37.17
BERTSUM+Classifier	43.23	20.22	39.60
BERTSUM+Transformer	**43.25**	**20.24**	**39.63**
BERTSUM+LSTM	43.22	20.17	39.59

표 1

Model	R-1	R-2	R-L
BERTSUM+Classifier	43.23	20.22	39.60
-interval segments	43.21	20.17	39.57
-trigram blocking	42.57	19.96	39.04

표 2

Conclusion

- pretrained BERT는 텍스트 요약에 유용하게 활용될 수 있음을 보여줌.
- Extractive summarization을 위한 프레임워크를 제안함.