텍스트 요약을 위한 BERTSUM 탐색

-6장-

BERTSUM?

- 텍스트 요약에 맞춰 파인 튜닝된 BERT 모델을 BERTSUM (BERT for summarization) 이라고 한다.

텍스트 요약

- 추출 요약

- 텍스트에서 중요한 문장만 추출해 요약

- 생성 요약

- 텍스트를 의역해 요약을 생성

- 텍스트의 의미만 지닌 다른 단어를 사용해 새로운 문장으로 만듦

BERT를 사용한 추출 요약

- 입력 문장을 토큰화
- [CLS] 문장 [SEP] [CLS] 문장 [SEP]…

  → 모든 문장의 표현이 필요하므로 각 문장 시작 부분에 [CLS]토큰 추가

- 입력 토큰을 토큰 임베딩, 위치 임베딩, 세그먼트 임베딩으로 변환
- 세그먼트 임베딩:

  → 입력 문장이 많으므로 홀수번째 문장은 E_A, 짝수번째 문장은E_B에 매핑한다.

- 모든 임베딩을 더해 BERT에 입력

분류기가 있는 BERTSUM

- 문장 표현을 이진 분류기에 공급하여 중요성 여부 판단 (요약 레이어)
- $\hat{Y}_i = \sigma(W_o R_i + b_o)$
- 이렇게 계산된 target과 정답으로 loss를 계산한다
- loss 값을 최소화하도록 BERT 모델과 분류기 레이어를 함께 학습시킨다

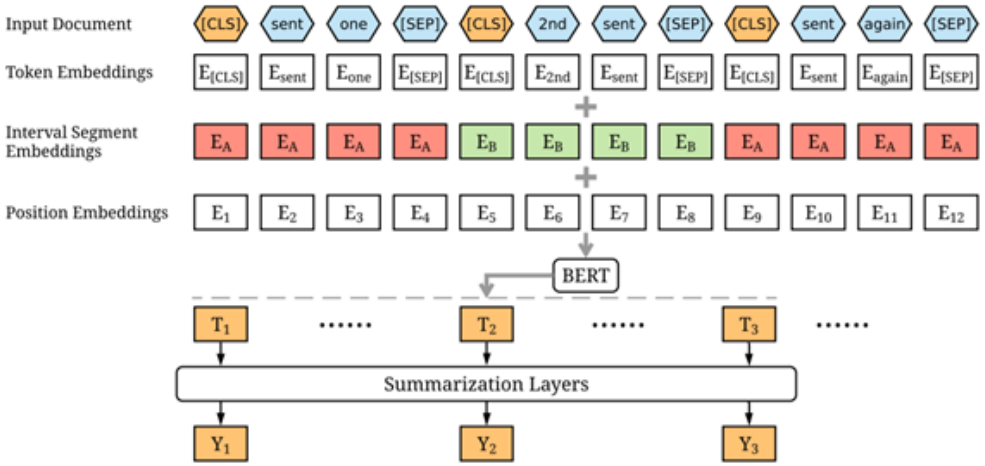

Figure 1: The overview architecture of the BERTSUM model.

Transformer와 LSTM을 활용한 BERTSUM

- 문장 간 트랜스포머를 활용한 BERTSUM
- BERT의 결과인 문장 표현 R을 트랜스포머 레이어에 입력한다
- 트랜스포머는 BERT에서 얻은 표현을 가져와 은닉 상태로 변환하는데, 이 때 도입되는 트랜스포머는 문장 간 Attention을 계산하고 문장 단위가 아닌 전체 문서 관점에서 요약 태스크를 수행한다
- BERT에서 얻은 문장 표현 R에 위치 임베딩 값을 추가하여 트랜스포머의 인코더에 입력한다(h_0=PosEmb(R)) . 인코더 l에서 서브레이어는 다음과 같이 표현한다

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

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

- 최상위 인코더에서 나온 은닉 상태를 분류기에 입력해 문장을 요약에 포함시킬지 여부의 확률을 얻는다

LSTM을 활용한 BERTSUM

- BERT에서 얻은 문장 표현을 LSTM에 입력한다
- LSTM셀이 출력한 은닉 상태를 분류기에 입력해 각 문장을 요약에 포함시킬지 여부의 확률을 반환한다

BERT를 사용한 생성 요약

- BERT는 입력한 토큰의 표현만 반환 →트랜스포머의 인코더로 BERTSUM 사용
- 인코더는 의미있는 표현 생성, 디코더는 요약 생성을 학습
- 인코더는 사전 학습된 모델이지만 디코더는 무작위로 초기화되어 있다
- 인코더는 과적합될 수 있고, 디코더는 과소적합이 발생할 수 있다.
- → Adam optimizer를 2개 사용 (인코더, 디코더용)
- 인코더는 학습률을 줄이고 보다 부드럽게 감쇠하도록 적용
- $lr_e = 2e^{-3} \min(step^{-0.5}, step \cdot 20000^{-0.5})$
- $lr_d = 0.1 \min(step^{-0.5}, step \cdot 10000^{-0.5})$

ROUGE 평가 지표

- ROUGE : Recall-Oriented Understudy for Gisting Evaluation

ROUGE-N

- 후보(예측) 요약과 참조(실제) 요약 간의 n-gram 재현율
- 재현율= $\frac{\text{서로 겹치는 } n-gram \text{수}}{\text{실제 요약의 } n-gram \text{수}}$

- 후보: Machine learning is seen as a subset of artificial intelligence
- 참조: Machine learning is a subset of artificial intelligence

ROUGE-1

- Unigram 재현율이다.
- 재현율 : 8 / 8 = 1

ROUGE-2

- Bigram 재현율이다
- 재현율 = 6 / 7 = 0.857

ROUGE-L

- 가장 긴 공통 시퀀스(LCS)기반

후보 및 참조 요약에 LCS가 있다는 건 둘이 일치한다는 것

재현율 R_{lcs} = LCS(후보, 참조) / 참조 요약의 전체 단어 수

정밀도 P_{lcs} = LCS(후보, 참조) / 후보 요약의 전체 단어 수

ROUGE-L = $F_{lcs} = \frac{(1+b^2)R_{lcs}P_{lcs}}{R_{lcs}+b^2P_{lcs}}$ (b는 R_{lcs}와 P_{lcs} 의 중요도를 조절함)

성능

CNN / DailyMail 뉴스데이터 사용 (하이라이트 기사 & 원문으로 구성)

	ROUGE-1	ROUGE-2	ROUGE-L
BERT + classifier	43.23	20.22	39.60
BERT + transformer	43.25	20.24	39.63
BERT + LSTM	43.22	20.17	39.59
BERTSUMABS	41.72	19.39	38.76