ALBERT: A LITE BERT FOR SELF-SUPERVISED LEARNING OF LANGUAGE REPRESENTATIONS

Zhenzhong Lan et al,.

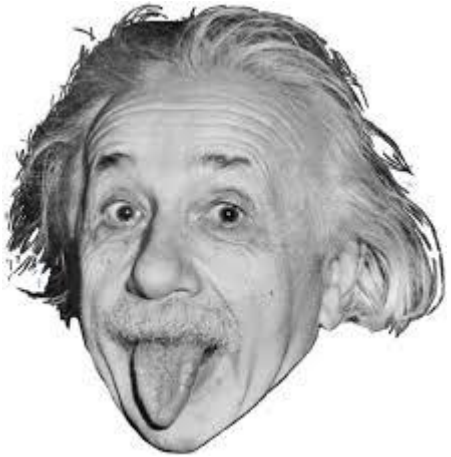
Google Research, Toyota Technological Institute at Chicago

AILAB 송지수

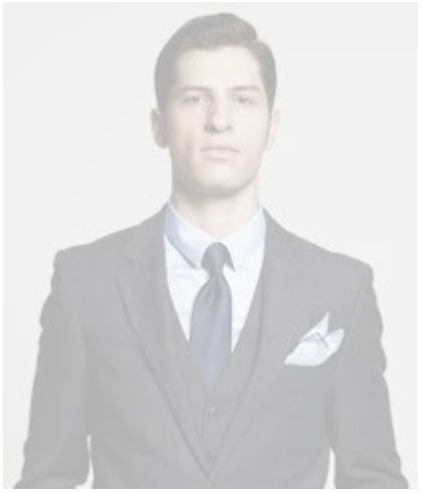
Naming?

ELMo
(Matthew E. Peters et al.)

BERT
(Jacob Devlin et al.)



ALBERT
(Zhenzhong Lan et al.)



ALBERTO
(?)

Introduction

- RACE (Reading Comprehension from Examination) test
  - 이 dataset이 처음 나왔을 2017년 당시 SotA는 44.1%
  - 현재 SotA는 RoBERTa (Liu et al., 2019) 모델이 달성한 83.2%

- 단적인 예지만, 이렇듯 NLP 분야에서 새로운 모델이 개발되며 각종 Dataset에 대한 성능이 급격하게 좋아지고 있는 추세

Introduction

- 성능이 좋은 새로운 모델은 이전의 모델들보다 parameter 수가 증가하는 경향을 보였음

- BERT-large의 경우 token당 hidden representation 차원이 1024, 총 parameter 개수는 334M

So what?

Model	Hidden Size	Parameters	RACE (Accuracy)
BERT-large (Devlin et al., 2019)	1024	334M	72.0%
BERT-large (ours)	1024	334M	73.9%
BERT-xlarge (ours)	2048	1270M	54.3%

Table 1: Increasing hidden size of BERT-large leads to worse performance on RACE.

Authors ask:

Is having better NLP models as easy as having larger models?

ALBERT: A Lite BERT

- 2가지의 parameter reduction techniques 이용
  - Factorized Embedding Parameterization
  - Cross-layer Parameter Sharing

- 이런 기법들을 이용하여 BERT-large 대비 18배 적은 parameter 개수의 모델로 트레이닝 속도를 1.7배 빠르게 함

ALBERT: A Lite BERT

- 또한 모델의 성능을 높이기 위해 BERT 모델의 NSP에 대비되는 SOP(Sentence-Order Prediction) 과 함께 SOP를 위한 self-supervised loss를 제안

Sentence A = The computer is not working.
Sentence B = It's unable to start.
Label = IsNextSentence

Sentence A = The computer is not working.
Sentence B = Coffee is very tasty.
Label = NotNextSentence

NSP(Next Sentence Prediction) of BERT

Factorized Embedding Parameterization

Factorized Embedding Parameterization

- In BERT: WordPiece embedding size E ≡ hidden layer size H
  - 모델링 관점에서:
    - WordPiece embedding은 context-independent representation 학습
    - Hidden layer embedding은 context-dependent representation을 학습
  - 실제적 관점에서:
    - E ≡ H 인 경우, H가 증가하면 임베딩 행렬의 크기인 V x E 크기가 매우 커짐
    - 이는 매개 변수 개수 증가에 핵심적 영향을 끼치며, 심지어 학습 중에 이 값들이 업데이트 되는 경우는 아주 가끔

Factorized Embedding Parameterization

- 따라서 ALBERT는 V x H를 분해!
  - H size hidden space에 WordPiece 별 one-hot vector를 투영하는 대신 E size 낮은 차원 embedding space에 투영한 다음 hidden space 투영
  - 따라서 embedding parameter 개수 획기적으로 감소
    - O(V x H) -> O(V x E + E x H)
  - 이는 H ≫ E 로 설정함으로서 가능

Cross-layer Parameter Sharing

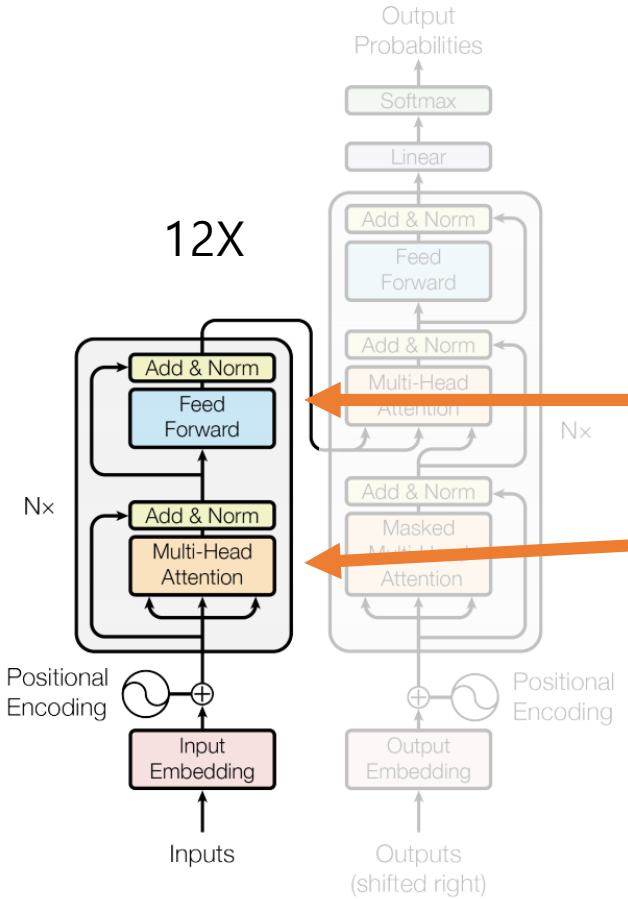


Figure 1: The Transformer - model architecture.

- Parameter의 효율 향상을 위해!
- 기존 방식:
  - 계층 간 FFN parameter sharing
  - Attention parameter 등 일부 sharing
- ALBERT는 계층 간 모든 parameter 공유하는 것을 기본으로 채택

Cross-layer Parameter Sharing

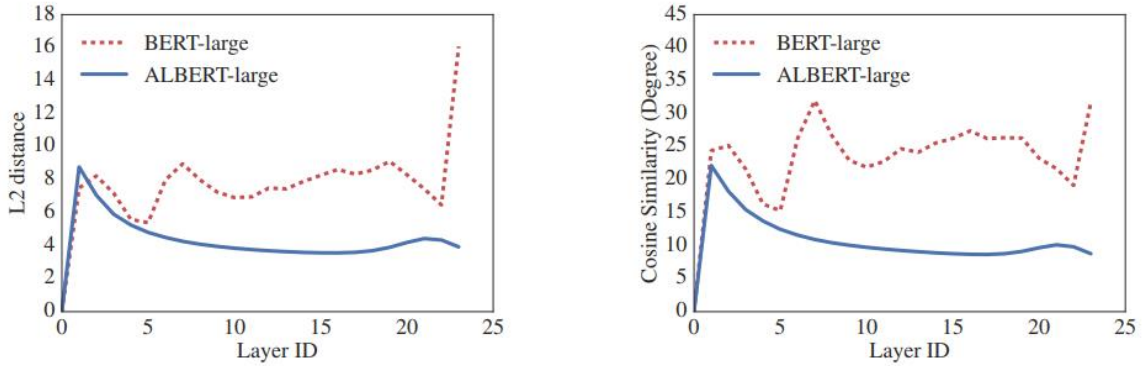


Figure 2: The L2 distances and cosine similarity (in terms of degree) of the input and output embedding of each layer for BERT-large and ALBERT-large.

- 각 Layer의 입력/출력 벡터 embedding의 L2 distance 및 Cosine similarity를 구함

Cross-layer Parameter Sharing

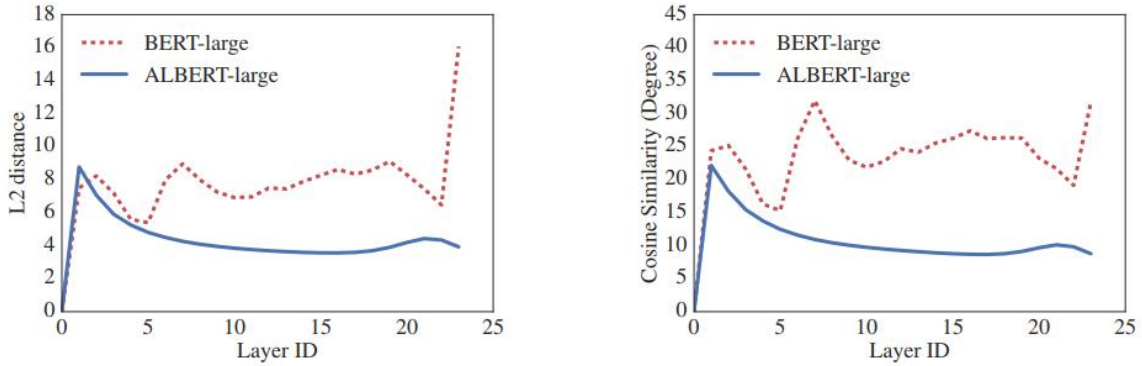


Figure 2: The L2 distances and cosine similarity (in terms of degree) of the input and output embedding of each layer for BERT-large and ALBERT-large.

- Layer 간 전이가 BERT보다 ALBERT에서 훨씬 smoother
- 이는 Cross-layer PS가 전체 네트워크를 안정화시키는 영향을 준다는 것을 증명

Inter-sentence Coherence Loss

- BERT는 Masked Language Modeling (MLM) 과 함께 Next-sentence Prediction (NSP) 이라는 두 가지 loss를 사용
- NSP는 두 segment가 연속적으로 나타나는 지 여부를 예측
  - Positive example: 코퍼스 상에서 연속된 문장 취함
  - Negative example: 다른 문서에서 문장을 pairing
  - Positive/Negative example은 동일한 빈도로 샘플링

Inter-sentence Coherence Loss

- NSP는 문장 pair 간 관계에 대한 Inference가 필요한 task에서 성능 향상을 위해 고안되었음
- 그러나 후속 연구에서 NSP가 task 수행 성능에 미치는 영향을 신뢰할 수 없다는 의견이 제시됨 (MLM의 영향이 훨씬 큼)

Inter-sentence Coherence Loss

- 이에 따라 저자들은 ALBERT에 일관성 (coherence)을 기초로 한 loss를 제안함
- 같은 문서에서 샘플링 된 두 문장이 학습 데이터로 주어졌을 때:
  - Positive example: 두 문장의 순서가 올바른 경우
  - Negative example: 두 문장의 순서가 뒤집어진 경우

Model Setup

Model		Parameters	Layers	Hidden	Embedding	Parameter-sharing
BERT	base	108M	12	768	768	False
	large	334M	24	1024	1024	False
	xlarge	1270M	24	2048	2048	False
ALBERT	base	12M	12	768	128	True
	large	18M	24	1024	128	True
	xlarge	60M	24	2048	128	True
	xxlarge	235M	12	4096	128	True

Table 2: The configurations of the main BERT and ALBERT models analyzed in this paper.

확연히 Parameter의 수가 BERT 대비 감소

Experiments

- Input의 형태는 “[CLS] x1 [SEP] x2 [SEP]” 의 형태
- Vocab size = 30,000
- Batch size = 4096
- Cloud TPU V3 64-1024 장 사용 (모델 크기에 따라)

Experiments

Model		Parameters	SQuAD1.1	SQuAD2.0	MNLI	SST-2	RACE	Avg	Speedup
BERT	base	108M	90.5/83.3	80.3/77.3	84.1	91.7	68.3	82.1	17.7x
	large	334M	92.4/85.8	83.9/80.8	85.8	92.2	73.8	85.1	3.8x
	xlarge	1270M	86.3/77.9	73.8/70.5	80.5	87.8	54.3	76.7	1.0
ALBERT	base	12M	89.3/82.1	79.1/76.1	81.9	89.4	63.5	80.1	21.1x
	large	18M	90.9/84.1	82.1/79.0	83.8	90.6	68.4	82.4	6.5x
	xlarge	60M	93.0/86.5	85.9/83.1	85.4	91.9	73.9	85.5	2.4x
	xxlarge	235M	**94.1/88.3**	**88.1/85.1**	**88.0**	**95.2**	**82.3**	**88.7**	1.2x

Table 3: Dev set results for models pretrained over BOOKCORPUS and Wikipedia for 125k steps. Here and everywhere else, the Avg column is computed by averaging the scores of the downstream tasks to its left (the two numbers of F1 and EM for each SQuAD are first averaged).

Model	E	Parameters	SQuAD1.1	SQuAD2.0	MNLI	SST-2	RACE	Avg
ALBERT base not-shared	64	87M	89.9/82.9	80.1/77.8	82.9	91.5	66.7	81.3
	128	89M	89.9/82.8	80.3/77.3	83.7	91.5	67.9	81.7
	256	93M	90.2/83.2	80.3/77.4	84.1	91.9	67.3	81.8
	768	108M	90.4/83.2	80.4/77.6	84.5	92.8	68.2	82.3
ALBERT base all-shared	64	10M	88.7/81.4	77.5/74.8	80.8	89.4	63.5	79.0
	128	12M	89.3/82.3	80.0/77.1	81.6	90.3	64.0	80.1
	256	16M	88.8/81.5	79.1/76.3	81.5	90.3	63.4	79.6
	768	31M	88.6/81.5	79.2/76.6	82.0	90.6	63.3	79.8

Table 4: The effect of vocabulary embedding size on the performance of ALBERT-base.

Experiments

Model		Parameters	SQuAD1.1	SQuAD2.0	MNLI	SST-2	RACE	Avg
ALBERT base E=768	all-shared	31M	88.6/81.5	79.2/76.6	82.0	90.6	63.3	79.8
	shared-attention	83M	89.9/82.7	80.0/77.2	84.0	91.4	67.7	81.6
	shared-FFN	57M	89.2/82.1	78.2/75.4	81.5	90.8	62.6	79.5
	not-shared	108M	90.4/83.2	80.4/77.6	84.5	92.8	68.2	82.3
ALBERT base E=128	all-shared	12M	89.3/82.3	80.0/77.1	82.0	90.3	64.0	80.1
	shared-attention	64M	89.9/82.8	80.7/77.9	83.4	91.9	67.6	81.7
	shared-FFN	38M	88.9/81.6	78.6/75.6	82.3	91.7	64.4	80.2
	not-shared	89M	89.9/82.8	80.3/77.3	83.2	91.5	67.9	81.6

Table 5: The effect of cross-layer parameter-sharing strategies, ALBERT-base configuration.

Experiments

Number of layers	Parameters	SQuAD1.1	SQuAD2.0	MNLI	SST-2	RACE	Avg
1	18M	31.1/22.9	50.1/50.1	66.4	80.8	40.1	52.9
3	18M	79.8/69.7	64.4/61.7	77.7	86.7	54.0	71.2
6	18M	86.4/78.4	73.8/71.1	81.2	88.9	60.9	77.2
12	18M	89.8/83.3	80.7/77.9	83.3	91.7	66.7	81.5
24	18M	**90.3/83.3**	**81.8/79.0**	83.3	91.5	**68.7**	**82.1**
48	18M	90.0/83.1	**81.8/78.9**	**83.4**	**91.9**	66.9	81.8

Table 7: The effect of increasing the number of layers for an ALBERT-large configuration.

Hidden size	Parameters	SQuAD1.1	SQuAD2.0	MNLI	SST-2	RACE	Avg
1024	18M	79.8/69.7	64.4/61.7	77.7	86.7	54.0	71.2
2048	60M	83.3/74.1	69.1/66.6	79.7	88.6	58.2	74.6
4096	225M	**85.0/76.4**	**71.0/68.1**	**80.3**	**90.4**	**60.4**	**76.3**
6144	499M	84.7/75.8	67.8/65.4	78.1	89.1	56.0	74.0

Table 8: The effect of increasing the hidden-layer size for an ALBERT-large 3-layer configuration.

Number of layers	SQuAD1.1	SQuAD2.0	MNLI	SST-2	RACE	Avg
12	94.0/88.1	88.3/85.3	87.8	95.4	82.5	88.7
24	94.1/88.3	88.1/85.1	88.0	95.2	82.3	88.7

Table 10: The effect of a deeper network using an ALBERT-xxlarge configuration.

Experiments

Models	SQuAD1.1 dev	SQuAD2.0 dev	SQuAD2.0 test	RACE test (Middle/High)
Single model (from leaderboard as of Sept. 23, 2019)				
BERT-large	90.9/84.1	81.8/79.0	89.1/86.3	72.0 (76.6/70.1)
XLNet	94.5/89.0	88.8/86.1	89.1/86.3	81.8 (85.5/80.2)
RoBERTa	94.6/88.9	89.4/86.5	89.8/86.8	83.2 (86.5/81.3)
UPM	-	-	89.9/87.2	-
XLNet + SG-Net Verifier++	-	-	90.1/87.2	-
ALBERT (1M)	94.8/89.2	89.9/87.2	-	86.0 (88.2/85.1)
ALBERT (1.5M)	**94.8/89.3**	**90.2/87.4**	**90.9/88.1**	**86.5 (89.0/85.5)**
Ensembles (from leaderboard as of Sept. 23, 2019)				
BERT-large	92.2/86.2	-	-	-
XLNet + SG-Net Verifier	-	-	90.7/88.2	-
UPM	-	-	90.7/88.2	
XLNet + DAAF + Verifier	-	-	90.9/88.6	-
DCMN+	-	-	-	84.1 (88.5/82.3)
ALBERT	**95.5/90.1**	**91.4/88.9**	**92.2/89.7**	**89.4 (91.2/88.6)**

Table 14: State-of-the-art results on the SQuAD and RACE benchmarks.

Conclusion & Implications

- ALBERT는 무작정 모델 크기만 키워 성능을 내는 현재 NLP 연구 트렌드에 의미있는 연구 결과를 남김
- Parameter의 개수보다 각 Layer와 Parameter의 의미를 파악, 그것을 의미있고 효율적으로 사용하는 것이 성능 향상에 영향

- 아무리 그래도 1060으로 학습은 힘들다~ 이말이야