7.0. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks

2019년 8월 출간

논문 : https://arxiv.org/pdf/1908.10084.pdf

논문이 생각하는 문제

- Semantic Textual Similarity(STS)와 같은 두 문장간의 유사도 TASK는 모든 문장끼리 pairwise해야되는 단점이 있음
- 10,000개의 문장이 있으면 10000*9999/2 = 49,995,000 문장이 생성되고 학습&추론에서 오래걸림
- 이런 방식이 가끔 GloVe embedding(통계를 이용한 방식)보다 낮은 성능을 보이기도 함
- Poly-Encoder의 Attention은 투머치다~

논문 주장

- Task에 따라 적절한 objective function을 선정해서 학습하자~
- inference할때는 cosine similarity로만 해도 충분하다~

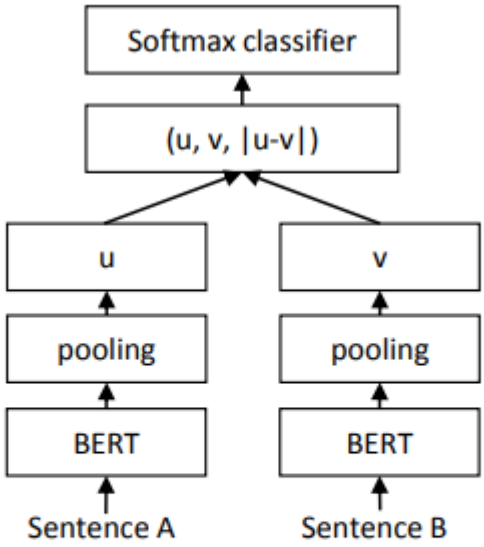

Figure 1: SBERT architecture with classification objective function, e.g., for fine-tuning on SNLI dataset. The two BERT networks have tied weights (siamese network structure).

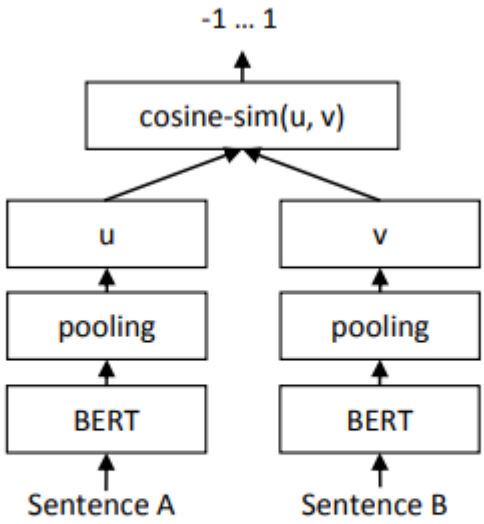

Figure 2: SBERT architecture at inference, for example, to compute similarity scores. This architecture is also used with the regression objective function.

모델 구조

- BERT Body share
- BERT output pooling
- bi-encoder

Pooling Strategy

- 문장 비교할때 사용할 토큰을 계산하는 방법
- 전체 토큰 평균하기, 더하기, <CLS>만 사용하기

Classification Objective Function

- fine-tuning에 기본적으로 사용되는 objective function
- Context, Candidate, element-wise difference를 concatenate해서 사용

$$o = \text{softmax}(W_t(u, v, |u - v|))$$

Regression Objective Function

- inference할때는 cosine similarity만 사용한다~
- 학습할때는 사용하지 않음

Triplet Objective Function

- FaceNet 논문을 참고해서 적용한 Objective Function
- vector space에서 Positive는 가깝게 Negative는 멀게 하는것이 목적
- 소스 확인해봤을땐 Task에 따라 fine-tuning에서의 사용여부가 달라짐
- s_a : Context
- s_p : Positive Candidate
- s_n : Negative Candidate
- ϵ : Hyper parameter (논문에서 1 사용)
- $||\cdot||$: Euclidean Distance

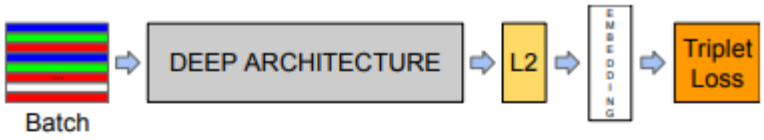

Figure 2. **Model structure.** Our network consists of a batch input layer and a deep CNN followed by L_2 normalization, which results in the face embedding. This is followed by the triplet loss during training.

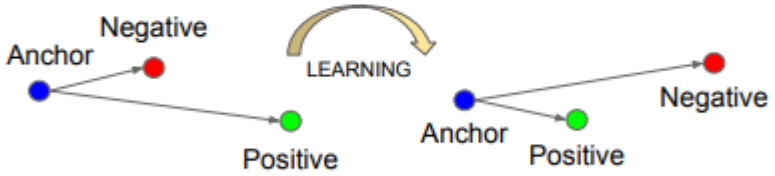

Figure 3. The **Triplet Loss** minimizes the distance between an *anchor* and a *positive*, both of which have the same identity, and maximizes the distance between the *anchor* and a *negative* of a different identity.

$$max(||s_a - s_p|| - ||s_a - s_n|| + \epsilon, 0)$$

실험결과

Model	STS12	STS13	STS14	STS15	STS16	STSb	SICK-R	Avg.
Avg. GloVe embeddings	55.14	70.66	59.73	68.25	63.66	58.02	53.76	61.32
Avg. BERT embeddings	38.78	57.98	57.98	63.15	61.06	46.35	58.40	54.81
BERT CLS-vector	20.16	30.01	20.09	36.88	38.08	16.50	42.63	29.19
InferSent - Glove	52.86	66.75	62.15	72.77	66.87	68.03	65.65	65.01
Universal Sentence Encoder	64.49	67.80	64.61	76.83	73.18	74.92	**76.69**	71.22
SBERT-NLI-base	70.97	76.53	73.19	79.09	74.30	77.03	72.91	74.89
SBERT-NLI-large	72.27	**78.46**	**74.90**	80.99	76.25	**79.23**	73.75	76.55
SRoBERTa-NLI-base	71.54	72.49	70.80	78.74	73.69	77.77	74.46	74.21
SRoBERTa-NLI-large	**74.53**	77.00	73.18	**81.85**	**76.82**	79.10	74.29	**76.68**

Table 1: Spearman rank correlation ρ between the cosine similarity of sentence representations and the gold labels for various Textual Similarity (STS) tasks. Performance is reported by convention as $\rho \times 100$. STS12-STS16: SemEval 2012-2016, STSb: STSbenchmark, SICK-R: SICK relatedness dataset.

Model	Spearman
Not trained for STS	
Avg. GloVe embeddings	58.02
Avg. BERT embeddings	46.35
InferSent - GloVe	68.03
Universal Sentence Encoder	74.92
SBERT-NLI-base	77.03
SBERT-NLI-large	79.23
Trained on STS benchmark dataset	
BERT-STSb-base	84.30 ± 0.76
SBERT-STSb-base	84.67 ± 0.19
SRoBERTa-STSb-base	$\mathbf{84.92} \pm 0.34$
BERT-STSb-large	$\mathbf{85.64} \pm 0.81$
SBERT-STSb-large	84.45 ± 0.43
SRoBERTa-STSb-large	85.02 ± 0.76
Trained on NLI data + STS benchmark data	
BERT-NLI-STSb-base	$\mathbf{88.33} \pm 0.19$
SBERT-NLI-STSb-base	85.35 ± 0.17
SRoBERTa-NLI-STSb-base	84.79 ± 0.38
BERT-NLI-STSb-large	$\mathbf{88.77} \pm 0.46$
SBERT-NLI-STSb-large	86.10 ± 0.13
SRoBERTa-NLI-STSb-large	86.15 ± 0.35

Table 2: Evaluation on the STS benchmark test set. BERT systems were trained with 10 random seeds and 4 epochs. SBERT was fine-tuned on the STSb dataset, SBERT-NLI was pretrained on the NLI datasets, then fine-tuned on the STSb dataset.

Model	r	ρ
Unsupervised methods		
tf-idf	46.77	42.95
Avg. GloVe embeddings	32.40	34.00
InferSent - GloVe	27.08	26.63
10-fold Cross-Validation		
SVR (Misra et al., 2016)	63.33	-
BERT-AFS-base	77.20	74.84
SBERT-AFS-base	76.57	74.13
BERT-AFS-large	78.68	76.38
SBERT-AFS-large	77.85	75.93
Cross-Topic Evaluation		
BERT-AFS-base	58.49	57.23
SBERT-AFS-base	52.34	50.65
BERT-AFS-large	62.02	60.34
SBERT-AFS-large	53.82	53.10

Table 3: Average Pearson correlation r and average Spearman's rank correlation ρ on the Argument Facet Similarity (AFS) corpus (Misra et al., 2016). Misra et al. proposes 10-fold cross-validation. We additionally evaluate in a cross-topic scenario: Methods are trained on two topics, and are evaluated on the third topic.

Model	**Accuracy**
mean-vectors	0.65
skip-thoughts-CS	0.62
Dor et al.	0.74
SBERT-WikiSec-base	0.8042
SBERT-WikiSec-large	**0.8078**
SRoBERTa-WikiSec-base	0.7945
SRoBERTa-WikiSec-large	0.7973

Table 4: Evaluation on the Wikipedia section triplets dataset (Dor et al., 2018). SBERT trained with triplet loss for one epoch.

Model	MR	CR	SUBJ	MPQA	SST	TREC	MRPC	Avg.
Avg. GloVe embeddings	77.25	78.30	91.17	87.85	80.18	83.0	72.87	81.52
Avg. fast-text embeddings	77.96	79.23	91.68	87.81	82.15	83.6	74.49	82.42
Avg. BERT embeddings	78.66	86.25	94.37	88.66	84.40	92.8	69.45	84.94
BERT CLS-vector	78.68	84.85	94.21	88.23	84.13	91.4	71.13	84.66
InferSent - GloVe	81.57	86.54	92.50	**90.38**	84.18	88.2	75.77	85.59
Universal Sentence Encoder	80.09	85.19	93.98	86.70	86.38	**93.2**	70.14	85.10
SBERT-NLI-base	83.64	89.43	94.39	89.86	88.96	89.6	**76.00**	87.41
SBERT-NLI-large	**84.88**	**90.07**	**94.52**	90.33	**90.66**	87.4	75.94	**87.69**

Table 5: Evaluation of SBERT sentence embeddings using the SentEval toolkit. SentEval evaluates sentence embeddings on different sentence classification tasks by training a logistic regression classifier using the sentence embeddings as features. Scores are based on a 10-fold cross-validation.

	NLI	STSb
Pooling Strategy		
MEAN	**80.78**	**87.44**
MAX	79.07	69.92
CLS	79.80	86.62
Concatenation		
(u, v)	66.04	-
$(\|u - v\|)$	69.78	-
$(u * v)$	70.54	-
$(\|u - v\|, u * v)$	78.37	-
$(u, v, u * v)$	77.44	-
$(u, v, \|u - v\|)$	**80.78**	-
$(u, v, \|u - v\|, u * v)$	80.44	-

Table 6: SBERT trained on NLI data with the classification objective function, on the STS benchmark (STSb) with the regression objective function. Configurations are evaluated on the development set of the STSb using cosine-similarity and Spearman's rank correlation. For the concatenation methods, we only report scores with MEAN pooling strategy.