SimCSE:

Simple Contrastive Learning of Sentence Embeddings

Tianyu Gao, Xingcheng Yao, Danqi Chen (*2021.04.*)

김선행, 김한길, 옥창원

2021. 08. 08.

SimCSE is…

A **simple contrastive sentence embedding framework**

→ Sentence embedding을 위해 contrastive learning을 아주 쉽게 이용할 수 있는 프레임워크

의의

- Text에 contrastive learning를 간단히 적용할 수 있는 프레임워크 제시

  Standard semantic textual similarity (STS) tasks에 대해 기존 best result를 훨씬 상회하는 결과를 보여줌

  - **Unsupervised: Dropout이라는 아주 일반적이고 간단한 방식 -> PLM 등에 확장 가능성**
  - Supervised: NLI Datasets 이용
- 다양한 실험을 통해 왜 성능이 좋게 나오는지를 설명하고자 노력

SimCSE is…

SimCSE: Simple Contrastive Learning of Sentence Embeddings

Tianyu Gao[†*] **Xingcheng Yao**[‡*] **Danqi Chen**[†]
[†]Department of Computer Science, Princeton University
[‡]Institute for Interdisciplinary Information Sciences, Tsinghua University
{tianyug,danqic}@cs.princeton.edu
yxc18@mails.tsinghua.edu.cn

Abstract

This paper presents SimCSE, a simple contrastive learning framework that greatly advances the state-of-the-art sentence embeddings. We first describe an unsupervised approach, which takes an input sentence and predicts *itself* in a contrastive objective, with only standard dropout used as noise. This simple method works surprisingly well, performing

	$\text{BERT}_{\text{base}}$
Unsupervised	
Avg. embeddings	56.7
IS-BERT (prev. SoTA)	66.6
SimCSE	**74.5** (+7.9%)
Supervised	
SBERT	74.9
SBERT-whitening (prev. SoTA)	77.0
SimCSE	**81.6** (+4.6%)

princeton-nlp / **SimCSE** Notifications Star 1.1k Fork 97

Contrastive Learning

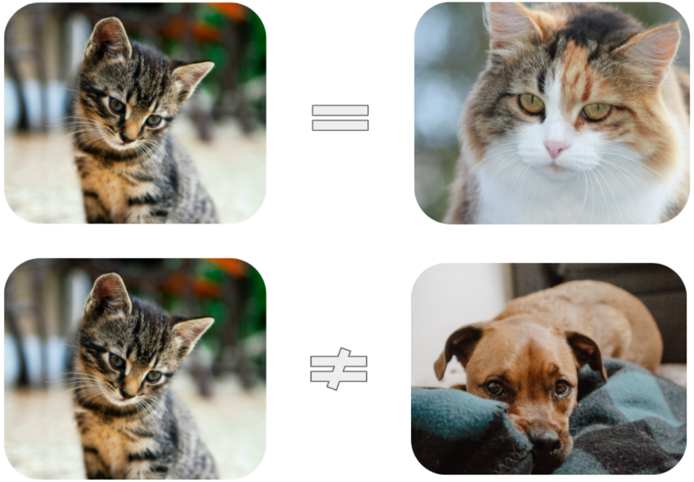
Contrastive Learning

- Self-supervised learning의 일종
- Positive pair(유사한 데이터)의 representation은 거리가 가까워 지도록 학습하고,
  Negative pair(다른 데이터)의 representation은 거리가 멀어지도록 학습을 시킨다.

Key question

1. How to construct good positive pair?

2. How to construct good negative pair?



Contrastive Learning(image): SimCLR

기 접근: Memory Bank

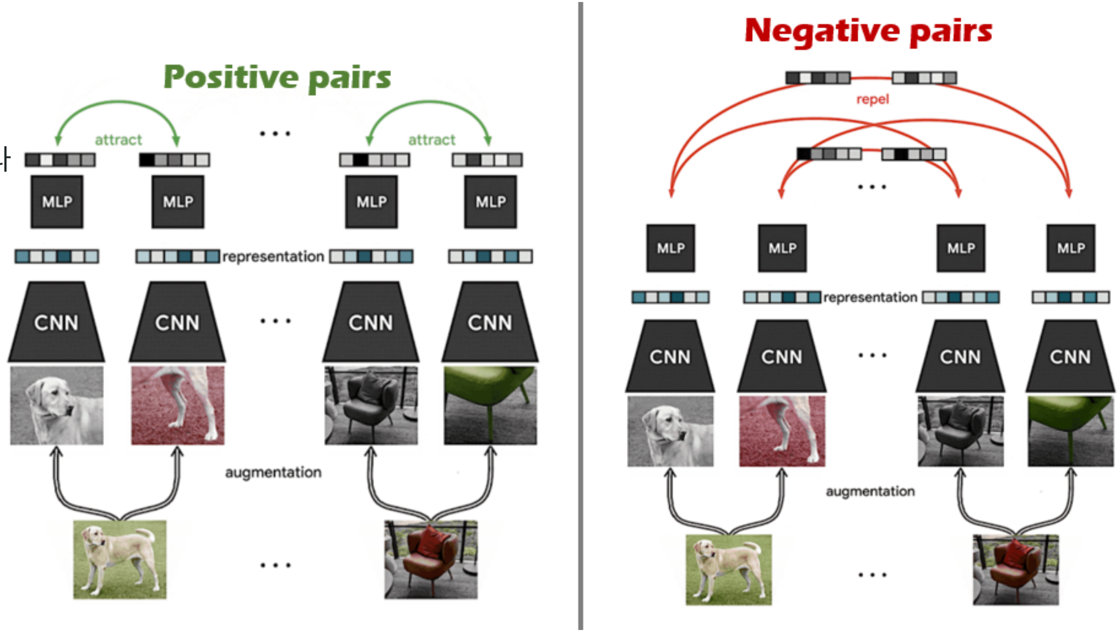
- 관측치 전체 embedding을 계속 저장하고 있어야 한다
- 랜덤 샘플링을 통해 negative example을 구성

→ 데이터 샘플별로 학습에 기여하는 정도가 다르다

⇨

SimCLR

하나의 배치 안에서 negative sample을 계산하자

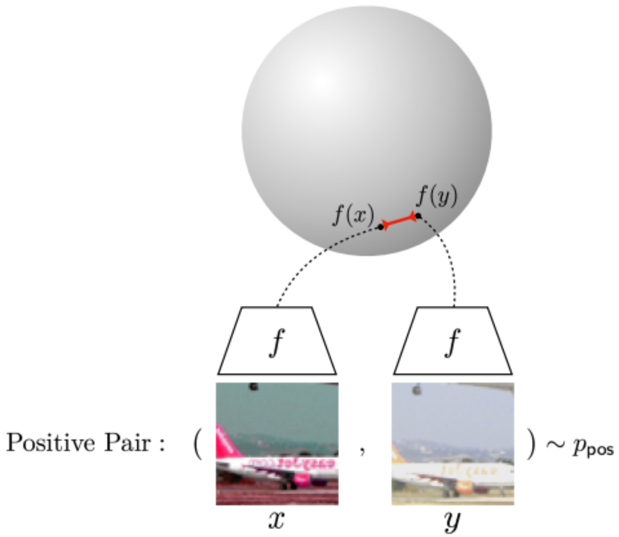


https://velog.io/@tobigs-gm1/Self-Supervised-Learning

Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020, November). A simple framework for contrastive learning of visual representations. In International conference on machine learning (pp. 1597-1607). PMLR

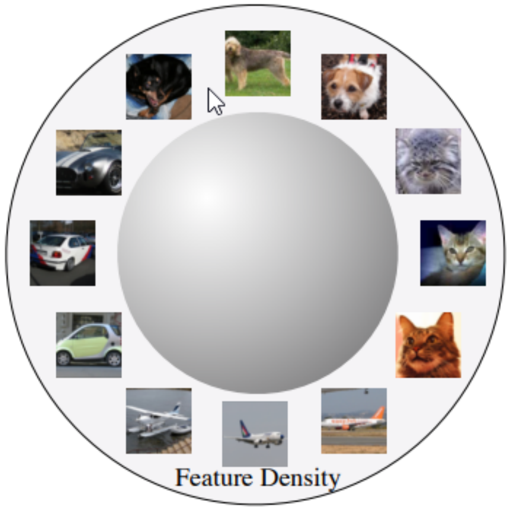
Contrastive representation Metric: Alignment and uniformity

둘 다 수치가 작을수록 좋음



Alignment: Similar samples have similar features.

“Expected distance
between embeddings of the paired instances”



Uniformity: Preserve maximal information.

“How well the embeddings are
uniformly distributed on each dimension”

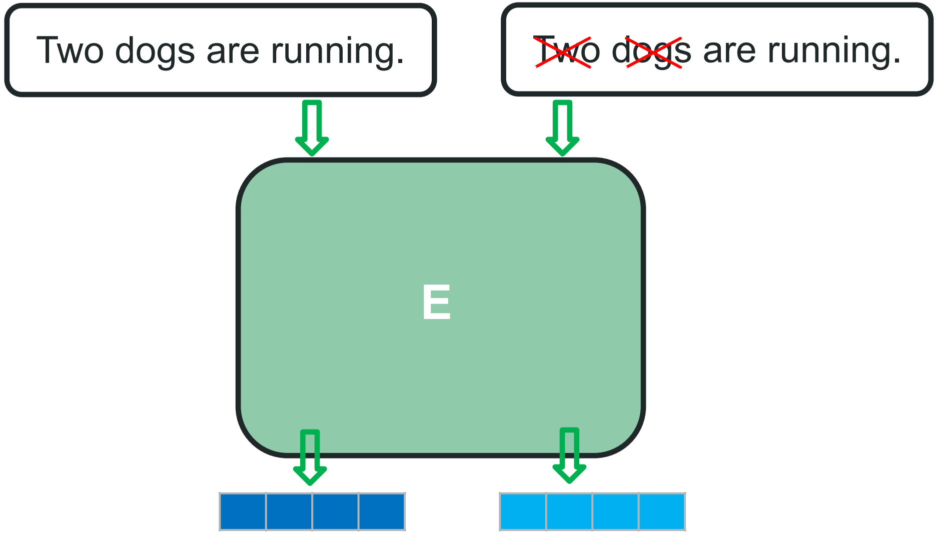
Wang, T., & Isola, P. (2020, November). Understanding contrastive representation learning through alignment and uniformity on the hypersphere

SimCSE

Unsupervised SimCSE

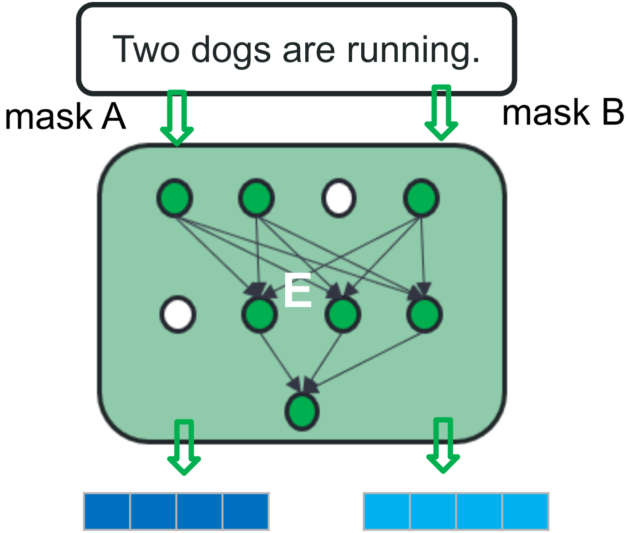
Clear(2020) / Coco-lm(2021)

Apply augmentation techniques
such as word deletion, reordering, and substitution

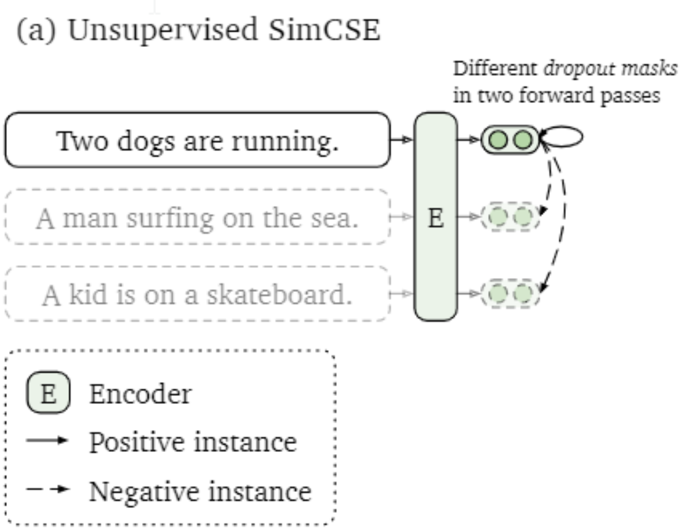


SimCSE

Apply **random mask for dropout**



Unsupervised SimCSE



- Paired(semantically related) example: $x_i^+ = x_i$.
- Representations: $\mathbf{h}_i^z = f_\theta(x_i, \boxed{z})$
  - Z: random mask for dropout
    (standard dropout mask in Transformers)
  - f: pre-trained language model such as BERT or RoBERTa

- Training objective for (x_i, x_i^+)

$$\ell_i = -\log \frac{e^{\mathrm{sim}(\mathbf{h}_i^{z_i}, \mathbf{h}_i^{z_i'})/\tau}}{\sum_{j=1}^{N} e^{\mathrm{sim}(\mathbf{h}_i^{z_i}, \mathbf{h}_j^{z_j'})/\tau}}, \qquad (4)$$

  - sim = cosine similarity $\frac{\mathbf{h}_1^\top \mathbf{h}_2}{\|\mathbf{h}_1\| \cdot \|\mathbf{h}_2\|}$
  - mini-ba…
  - Temperature τ= 0.05

Unsupervised SimCSE

1.Data augmentation 관점에서의 비교 실험 결과
•트레이닝은 English Wikipedia, 테스트는 STS-B development set 이용

Data augmentation			STS-B
None			**79.1**
Crop	*10%*	*20%*	*30%*
	75.4	70.1	63.7
Word deletion	*10%*	*20%*	*30%*
	74.7	71.2	70.2
Delete one word			74.8
w/o dropout			71.4
MLM 15%			66.8
Crop 10% + MLM 15%			70.8

Table 2: Comparison of different data augmentations on STS-B development set (Spearman's correlation). *Crop k%*: randomly crop and keep a continuous span with 100-*k*% of the length; *word deletion k%*: randomly delete *k*% words; *delete one word*: randomly delete one word; *MLM k%*: use $\text{BERT}_{\texttt{base}}$ to replace *k*% of words. All of them include the standard 10% dropout (except "w/o dropout").

각종 data augmentation 기법들을 적용하는 것이

적용하지 않고 10% dropout만 하는 경우(None)보다

두 문장의 유사도를 더 떨어뜨림

→ Text의 discrete nature에 따라 해당 data augmentation 기법들이

discrete한 augmentation이기 때문

Unsupervised SimCSE

- 정말 dropout 때문일까?

p	0.0	0.01	0.05	0.1
STS-B	64.9	69.5	78.0	**79.1**
p	0.15	0.2	0.5	*Fixed 0.1*
STS-B	78.6	78.2	67.4	45.2

Table 4: Effects of different dropout probabilities p on the STS-B development set (Spearman's correlation, $\text{BERT}_{\text{base}}$). *Fixed 0.1*: use the default 0.1 dropout rate but apply the same dropout mask on both x_i and x_i^+.

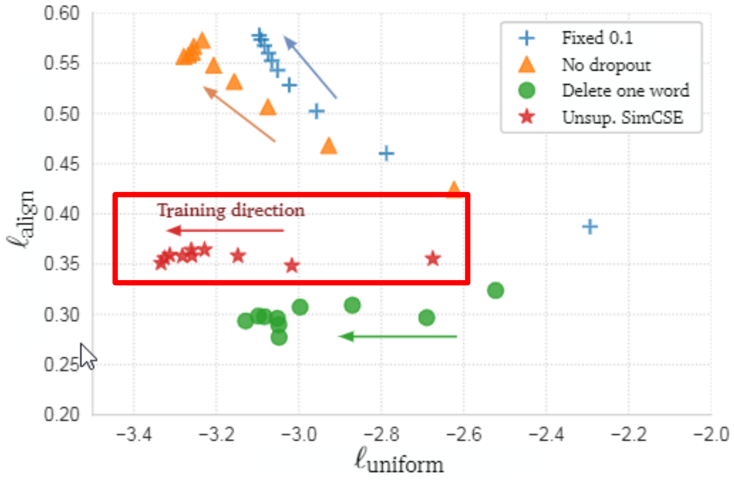


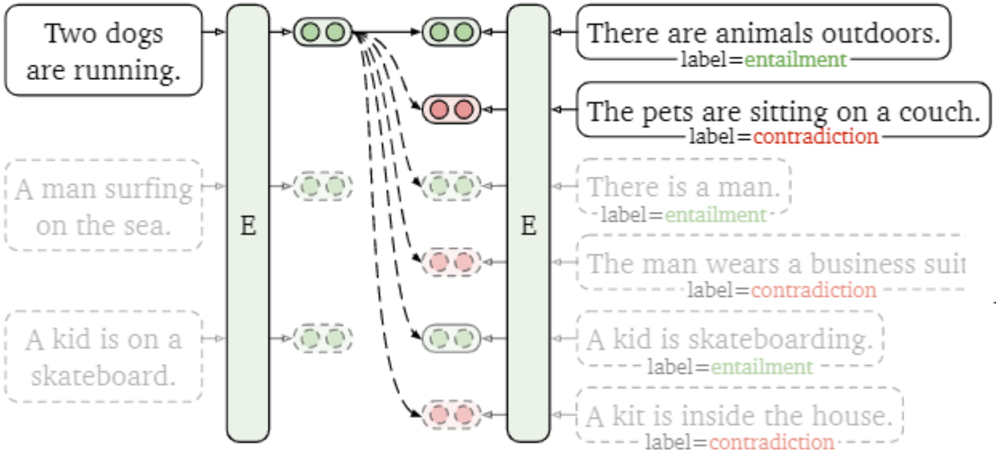
Figure 2: ℓ_{align}-ℓ_{uniform} plot for unsupervised SimCSE, "no dropout", "fixed 0.1" (same dropout mask for x_i and x_i^+ with $p = 0.1$), and "delete one word". We visualize checkpoints every 10 training steps and the arrows indicate the training direction. For both ℓ_{align} and ℓ_{uniform}, *lower numbers are better*.

No dropout 또는 fixed 0.1일 때 급격한 performance 감소
→ Dropout이 중요하게 역할함을 알 수 있음

Supervised SimCSE

STS dataset에서
기준 문장 & entailment 문장 → Positive pair로 이용

(b) Supervised SimCSE



- Unsupervised training objective for $x_i^+ = x_i$.

$$\ell_i = -\log \frac{e^{\mathrm{sim}(\mathbf{h}_i^{z_i}, \mathbf{h}_i^{z_i'})/\tau}}{\sum_{j=1}^{N} e^{\mathrm{sim}(\mathbf{h}_i^{z_i}, \mathbf{h}_j^{z_j'})/\tau}}, \quad (4)$$

- Supervised training objective for (x_i, x_i^+, x_i^-)

$$-\log \frac{e^{\mathrm{sim}(\mathbf{h}_i, \mathbf{h}_i^+)/\tau}}{\sum_{j=1}^{N} \left(e^{\mathrm{sim}(\mathbf{h}_i, \mathbf{h}_j^+)/\tau} + e^{\mathrm{sim}(\mathbf{h}_i, \mathbf{h}_j^-)/\tau} \right)}. \quad (5)$$

Supervised SimCSE

Dataset	sample	full
Unsup. SimCSE (1m)	-	79.1
QQP (134k)	81.8	81.8
Flickr30k (318k)	81.5	81.4
ParaNMT (5m)	79.7	78.7
SNLI+MNLI		
entailment (314k)	**84.1**	**84.9**
neutral (314k)[3]	82.6	82.9
contradiction (314k)	77.5	77.6
SNLI+MNLI		
entailment + hard neg.	-	**86.2**
+ ANLI (52k)	-	85.0

- Supervised가 대부분의 경우 unsupervised 보다 뛰어남.
- NLI (SNLI + MNLI) 데이터셋에 대해 학습한 모델이 가장 좋은 성능을 보임
  - 데이터 퀄리티 우수 & lexical overlap 낮음

* F1 measured between two bags of words)

- for the entailment pairs (SNLI + MNLI) is 39%,

- while they are 60% and 55% for QQP and ParaNMT.

- contradiction 문장을 활용하여 hard negatives를 이용하는 경우 성능 증가(84.9→86.2)
- 이전 연구에서 흔히 쓰는 dual-encoder는 오히려 성능을 저하시킴 (86.2→84.2)

Experiment

Exp1: STS(Semantic textual similarity tasks)

- Sentence embeddings의 주요 목적은 결국 semantically similar sentences를 잘 군집화 하는 것

  → STS results를 살펴보자!

- 모델 구조

  Pre-trained BERT/RoBERTa의 [CLS] 토큰 위에 다층 퍼셉트론(MLP) 레이어를 추가하여 학습

- 평가

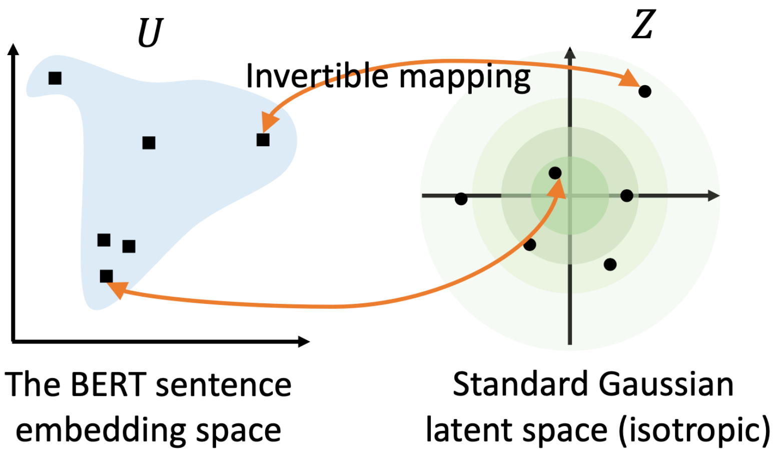
  Spearman's correlation (순위를 고려하는 것이 값 그 자체를 따지는 것 보다 본 실험에 더 적합)

Exp1: STS(Semantic textual similarity tasks)

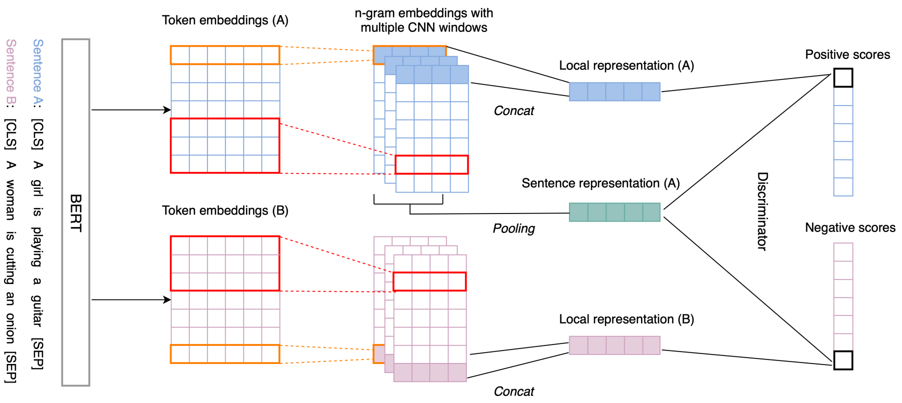
Model	STS12	STS13	STS14	STS15	STS16	STS-B	SICK-R	Avg.
Unsupervised models								
GloVe embeddings (avg.)♣	55.14	70.66	59.73	68.25	63.66	58.02	53.76	61.32
$\text{BERT}_{\text{base}}$ (first-last avg.)	39.70	59.38	49.67	66.03	66.19	53.87	62.06	56.70
$\text{BERT}_{\text{base}}$-flow	58.40	67.10	60.85	75.16	71.22	68.66	64.47	66.55
$\text{BERT}_{\text{base}}$-whitening	57.83	66.90	60.90	75.08	71.31	68.24	63.73	66.28
$\text{IS-BERT}_{\text{base}}$♡	56.77	69.24	61.21	75.23	70.16	69.21	64.25	66.58
∗ $\text{SimCSE-BERT}_{\text{base}}$	**66.68**	**81.43**	**71.38**	**78.43**	**78.47**	**75.49**	**69.92**	**74.54**
$\text{RoBERTa}_{\text{base}}$ (first-last avg.)	40.88	58.74	49.07	65.63	61.48	58.55	61.63	56.57
$\text{RoBERTa}_{\text{base}}$-whitening	46.99	63.24	57.23	71.36	68.99	61.36	62.91	61.73
∗ $\text{SimCSE-RoBERTa}_{\text{base}}$	**68.68**	**82.62**	**73.56**	**81.49**	**80.82**	**80.48**	**67.87**	**76.50**
∗ $\text{SimCSE-RoBERTa}_{\text{large}}$	**69.87**	**82.97**	**74.25**	**83.01**	**79.52**	**81.23**	**71.47**	**77.47**
Supervised models								
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
$\text{SBERT}_{\text{base}}$♣	70.97	76.53	73.19	79.09	74.30	77.03	72.91	74.89
$\text{SBERT}_{\text{base}}$-flow	69.78	77.27	74.35	82.01	77.46	79.12	76.21	76.60
$\text{SBERT}_{\text{base}}$-whitening	69.65	77.57	74.66	82.27	78.39	79.52	76.91	77.00
∗ $\text{SimCSE-BERT}_{\text{base}}$	**75.30**	**84.67**	**80.19**	**85.40**	**80.82**	**84.25**	**80.39**	**81.57**
$\text{SRoBERTa}_{\text{base}}$♣	71.54	72.49	70.80	78.74	73.69	77.77	74.46	74.21
$\text{SRoBERTa}_{\text{base}}$-whitening	70.46	77.07	74.46	81.64	76.43	79.49	76.65	76.60
∗ $\text{SimCSE-RoBERTa}_{\text{base}}$	**76.53**	**85.21**	**80.95**	**86.03**	**82.57**	**85.83**	**80.50**	**82.52**
∗ $\text{SimCSE-RoBERTa}_{\text{large}}$	**77.46**	**87.27**	**82.36**	**86.66**	**83.93**	**86.70**	**81.95**	**83.76**

Exp1: STS(Semantic textual similarity tasks)

BERT-flow / BERT-whitening



IS-BERT (Info-Sentence BERT)



[Li et al, 2020]

Exp1: STS(Semantic textual similarity tasks)

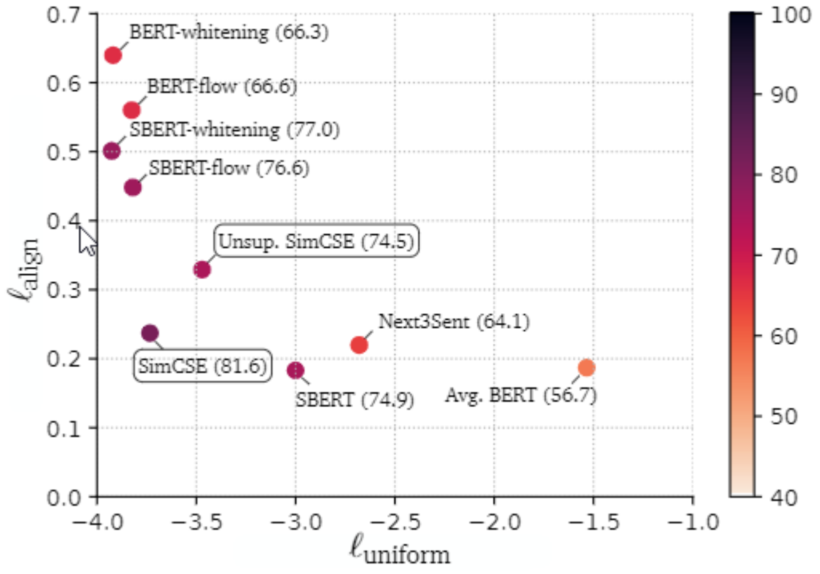


Figure 3: ℓ_{align}-ℓ_{uniform} plot of models based on $\text{BERT}_{\text{base}}$. Color of points and numbers in brackets represent average STS performance (Spearman's correlation). *Next3Sent*: "next 3 sentences" from Table 3.

- Pre-trained embedding는

  alignment는 좋으나, uniformity가 좋지 않음

- Post-processing method(BERT-flow, BERT-whitening)는

  uniformit를 크게 증가시키나, alignment를 안좋게 만듦

- 그에 반해 SimCSE는 alignment와 uniformity 둘다 증가

Exp2: Transfer tasks

Model	MR	CR	SUBJ	MPQA	SST	TREC	MRPC	Avg.
Unsupervised models								
GloVe embeddings (avg.)♣	77.25	78.30	91.17	87.85	80.18	83.00	72.87	81.52
Skip-thought♡	76.50	80.10	93.60	87.10	82.00	92.20	73.00	83.50
Avg. BERT embeddings♣	78.66	86.25	94.37	88.66	84.40	**92.80**	69.54	84.94
BERT-[CLS]embedding♣	78.68	84.85	94.21	88.23	84.13	91.40	71.13	84.66
IS-BERT$_{\text{base}}$♡	**81.09**	**87.18**	**94.96**	**88.75**	**85.96**	88.64	74.24	**85.83**
∗ SimCSE-BERT$_{\text{base}}$	80.41	85.30	94.46	88.43	85.39	87.60	71.13	84.67
w/ MLM	80.74	85.67	94.68	87.21	84.95	89.40	**74.38**	85.29
∗ SimCSE-RoBERTa$_{\text{base}}$	79.67	84.61	91.68	85.96	84.73	84.20	64.93	82.25
w/ MLM	**82.02**	**87.52**	**94.13**	**86.24**	**88.58**	**90.20**	**74.55**	**86.18**
∗ SimCSE-RoBERTa$_{\text{large}}$	80.83	85.30	91.68	86.10	85.06	89.20	**75.65**	84.83
w/ MLM	**83.30**	**87.50**	**95.27**	**86.82**	**87.86**	**94.00**	75.36	**87.16**
Supervised models								
InferSent-GloVe♣	81.57	86.54	92.50	90.38	84.18	88.20	75.77	85.59
Universal Sentence Encoder♣	80.09	85.19	93.98	86.70	86.38	93.20	70.14	85.10
SBERT$_{\text{base}}$♣	**83.64**	**89.43**	94.39	**89.86**	**88.96**	**89.60**	76.00	**87.41**
∗ SimCSE-BERT$_{\text{base}}$	82.69	89.25	**94.81**	89.59	87.31	88.40	73.51	86.51
w/ MLM	82.68	88.88	94.52	89.82	88.41	87.60	**76.12**	86.86
SRoBERTa$_{\text{base}}$	84.91	90.83	92.56	88.75	90.50	88.60	**78.14**	87.76
∗ SimCSE-RoBERTa$_{\text{base}}$	84.92	**92.00**	**94.11**	**89.82**	91.27	88.80	75.65	88.08
w/ MLM	**85.08**	91.76	94.02	89.72	**92.31**	**91.20**	76.52	**88.66**
∗ SimCSE-RoBERTa$_{\text{large}}$	88.12	92.37	95.11	90.49	92.75	91.80	76.64	89.61
w/ MLM	**88.45**	**92.53**	**95.19**	**90.58**	**93.30**	**93.80**	**77.74**	**90.23**

- Supervised model 의 경우 이전 접근 방식과 유사 또는 더 나은 성능을 보임
- 반면 unsupervised는 확실한 성능 우위를 보여주지 못함
- MLM 방식을 objective func에 추가하는 것이 SimCSE가 token-level knowledge를 잊지 않게 하는데 도움을 줌

$$\ell + \lambda \cdot \ell^{\text{mlm}}$$

Exp3: Ablation Study

Batch size	32	64	128	256	512	1024
STS-B	84.6	85.6	86.0	**86.2**	**86.2**	86.0

Table 8: Effect of different batch sizes (STS-B development set, Spearman's correlation, $\text{BERT}_{\texttt{base}}$).

Model	STS-B	Avg. transfer
[CLS]	**86.2**	85.8
First-last avg.	86.1	86.1
w/o MLM	**86.2**	85.8
w/ MLM		
$\lambda = 0.01$	85.7	86.1
$\lambda = 0.1$	85.7	**86.2**
$\lambda = 1$	85.1	85.8

Table 9: Ablation studies of different pooling methods and incorporating the MLM objective. The results are based on the development sets using $\text{BERT}_{\texttt{base}}$.

Thanks!

참고: Exp1: STS(Semantic textual similarity tasks)

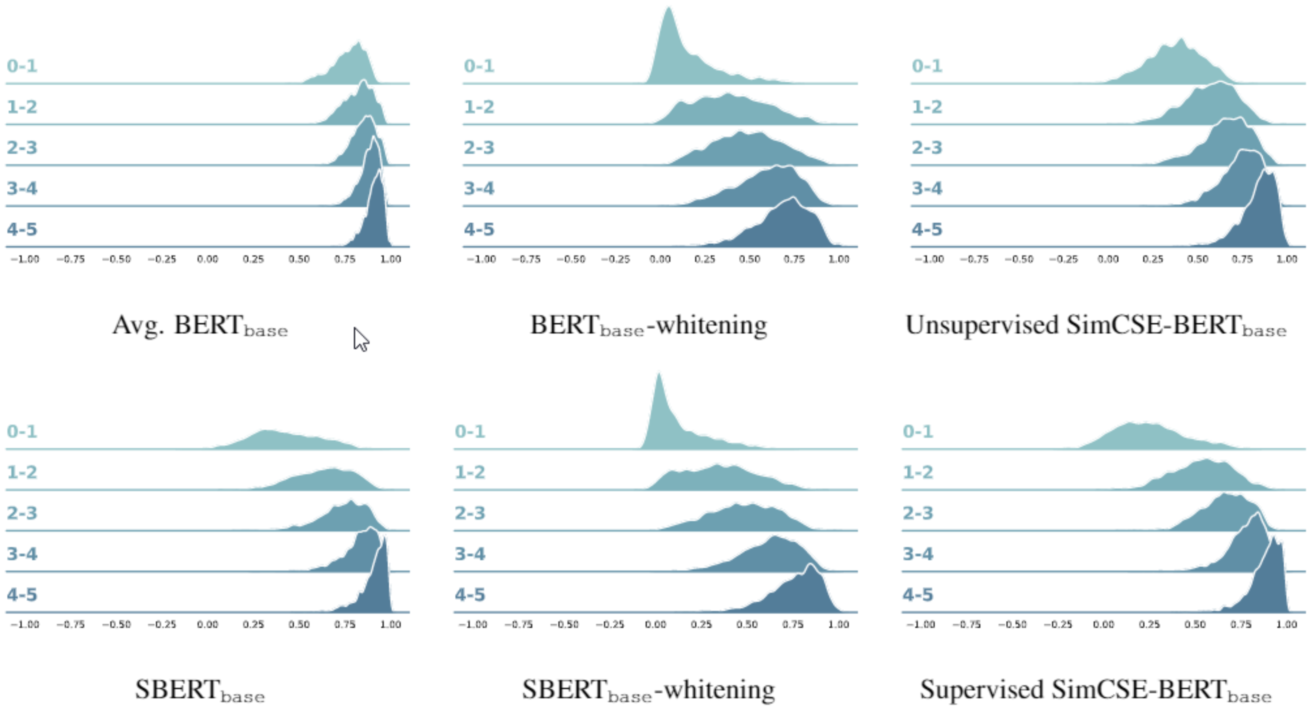


Figure 4: Density plots of cosine similarities between sentence pairs in full STS-B. Pairs are divided into 5 groups based on ground truth ratings (higher means more similar) along the y-axis, and x-axis is the cosine similarity.

참고: Exp1: STS(Semantic textual similarity tasks)

	$\mathbf{SBERT_{base}}$	$\mathbf{Supervised\ SimCSE\text{-}BERT_{base}}$
Query: A man riding a small boat in a harbor.		
#1	A group of men traveling over the ocean in a small boat.	A man on a moored blue and white boat.
#2	Two men sit on the bow of a colorful boat.	A man is riding in a boat on the water.
#3	A man wearing a life jacket is in a small boat on a lake.	A man in a blue boat on the water.
Query: A dog runs on the green grass near a wooden fence.		
#1	A dog runs on the green grass near a grove of trees.	The dog by the fence is running on the grass.
#2	A brown and white dog runs through the green grass.	Dog running through grass in fenced area.
#3	The dogs run in the green field.	A dog runs on the green grass near a grove of trees.

Table 10: Retrieved top-3 examples by SBERT and supervised SimCSE from Flickr30k (150k sentences).

참고: Exp2: Transfer tasks

- MR : 영화 리뷰로 긍부정으로 이루어진 데이터 세트
- SST : 감성 분석을 다루는 이진 분류 데이터셋
- CR : 크롤링으로 수집한 전자제품 리뷰 데이터셋
- TREC : train/dev/test 분할에 대한 1,229/65/68 질문과 53,417/1,117/1,442 질문-답변 쌍
- SUBJ : 전체 감정 극성(긍정 또는 부정) 또는 주관적 평가(예: "별 2개 반")와 관련하여 레이블이 지정된 영화 리뷰 문서 모음과 주관적 상태(주관 또는 객관적)
- MPQA : Multi-Perspective Question Answering 의견 및 기타 개인 상태(신념, 감정, 감정, 상상력 등)
- MRPC : Microsoft Research Paraphrase Corpus는 뉴스와이어 기사에서 수집된 5,801개의 문장 쌍으로 구성된 말뭉치입니다. 각 쌍은 의역인지 여부에 따라 사람 주석에 의해 레이블이 지정됩니다. 전체 세트는 훈련 부분 집합(4,076개 문장 쌍 중 2,753개가 의역)과 테스트 부분 집합(1,725개 쌍이 의역임)으로 나뉨.