Composition-contrastive Learning for Sentence Embeddings

Sachin Chanchani and **Ruihong Huang**
Texas A&M University
{chanchanis, huangrh}@tamu.edu

Abstract

Vector representations of natural language are ubiquitous in search applications. Recently, various methods based on contrastive learning have been proposed to learn textual representations from unlabelled data; by maximizing alignment between minimally-perturbed embeddings of the same text, and encouraging a uniform distribution of embeddings across a broader corpus. Differently, we propose maximizing alignment between texts and a composition of their phrasal constituents. We consider several realizations of this objective and elaborate the impact on representations in each case. Experimental results on semantic textual similarity tasks show improvements over baselines that are comparable with state-of-the-art approaches. Moreover, this work is the first to do so without incurring costs in auxiliary training objectives or additional network parameters.[1]

1 Introduction

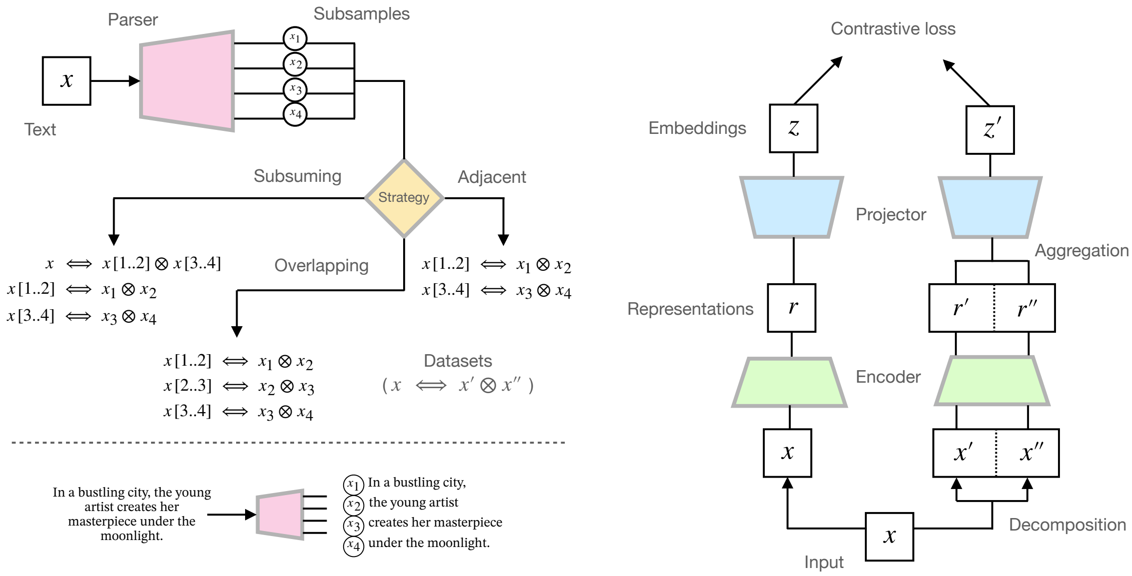
Significant progress has been made on the task of learning universal sentence representations that can be used for a variety of natural language processing tasks without task-specific fine-tuning (Conneau et al., 2017, Cer et al., 2018, Kiros et al., 2015, Logeswaran and Lee, 2018, Giorgi et al., 2021a, Yan et al., 2021, Gao et al., 2021, Chuang et al., 2022a). Recent works have shown the potential to learn good sentence embeddings without labeled data by fine-tuning pre-trained language models (PLMs) using the unsupervised framework introduced in SimCLR (Chen et al., 2020), adapted to the natural language processing (NLP) domain. In computer vision (CV), SimCLR exploits a series of transformations (blurs, crops, color distortions, etc.) to construct positive pairs from otherwise unique data points. A cross entropy objective (InfoNCE; Oord et al., 2018) is then applied to minimize distance between representations originating from the same datum, while maximizing the distance to all other points in a mini-batch. The success of the framework in computer vision is due largely to the diversity of augmentations used for creating positive pairs, which leave the identity of the original example intact while reducing pairwise mutual information in the input space (Tian et al., 2020; Wu et al., 2020; Purushwalkam and Gupta, 2020).

Constructing positive pairs via discrete augmentations have not been effective when applying the same objective to sentence embeddings. In fact, Gao et al. (2021) perform an ablation study of textual augmentations (e.g., cropping, synonym replacement) and find that training on these pairs hurts downstream performance on semantic textual similarity (STS) tasks. Instead, they observe that minimal (10%) dropout noise can be used to create positive pairs on-the-fly, and empirically results in stronger representations. This framework relying on nearly identical pairs is known as SimCSE. Since the dropout noise exists as a regularization component of the BERT architecture (Devlin et al., 2019a), explicit augmentations are unnecessary, making it a simple yet effective framework for unsupervised learning of sentence embeddings.

Here, we make a case for composition as augmentation, by exploiting its presence in language as a signal for learning sentence encoders. We conduct a series of experiments to illustrate the impact of training on positive examples derived by averaging representations of textual constituents in the latent space. Following previous works, we benchmark the proposed strategy on 7 STS tasks. Our results show that it is feasible to significantly improve upon SimCSE without making expensive architectural modifications or changing the overall training objective. We hope our findings can inspire new avenues of inquiry in text representation learning that draw on long-standing notions in semantics and linguistics.

[1] Code, pre-trained models, and datasets will be available at github.com/perceptiveshawty/CompCSE.

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 15836–15848
July 9-14, 2023 ©2023 Association for Computational Linguistics



(a) subsampling strategies

(b) training objective

Figure 1: An overview of composition-based contrastive learning. Subsampling strategies used to expand the training dataset are illustrated on the left-hand side, where the bidirectional arrows indicate positive pairs, brackets indicate spans of text, and all other pairs are the standard in-batch negatives. The $\otimes$ operation is a shorthand for the augmentation strategy integrated with the framework and depicted on the right: along with dropout noise, examples are decomposed in the input space and constituents are independently passed through the encoder. Resultant `[CLS]` tokens are then aggegrated and passed through a linear projector before computing the contrastive loss.

2 Background and Related Work

2.1 Unsupervised Contrastive Learning

Contrastive learning (Hadsell et al., 2006) aims to learn vector-valued representations of data without relying on annotations. Meaning is derived from these representations based on their proximity to other points in the same space, e.g. two images of dogs will be closer in space than a dog and a chair. Several works have theoretically verified the utility of representations derived from contrastive learning (Arora et al., 2019; Lee et al., 2020; Tosh et al., 2020) under various assumptions; Chen et al. (2020) showed that SimCLR can even outperform supervised counterparts on CV transfer learning benchmarks. In SimCLR (and SimCSE), the learning objective for an example is:

$$l_i = -log \frac{e^{sim(z_i, z_i^+)/\tau}}{\sum_{j=1}^{N} e^{sim(z_i, z_j^+)/\tau}}, \tag{1}$$

where $z_i = f(x_i), z_i^+ = f(x_i^+)$ are vector representations of an input and its corresponding augmented positive, τ is a temperature hyperparameter, $sim(.,.)$ is cosine similarity, and N is batch size.

Drawbacks of InfoNCE. In examination of eq. 1, it is evident that InfoNCE uniformly repels examples in the mini-batch besides the minimally augmented positive. Consequentially, the resulting embeddings show poor group-wise discrimination, especially in language, since it is likely that different examples in the batch can have different relative similarities to a given anchor. Another consequence of the unsupervised InfoNCE objective is dimensional collapse, wherein embedding vectors are mostly differentiated by a small proportion of the feature axes; thus under-utilizing the full expressive capacity of the encoder. This was theoretically posited in Jing et al. (2022). They prove that minimal augmentation, coupled with an over-parameterized network, results in low rank solutions to the unsupervised contrastive objective. We hypothesize that this is closely tied to short-cut learning (Robinson et al., 2021a) —- in the context of sentence embeddings, Wu et al. (2022c) observed that spurious features related to the lengths of sentences are relied on to solve the contrastive objective. Such solutions can yield non-generalizable features that poorly represent data from new domains.

Qualifying the representation space. Wang and Isola (2020) proposed two metrics to measure the quality of embeddings derived through contrastive learning. First, *alignment* measures on average the proximity of pairs of examples that *should* be close in space, i.e. for a set of positive pairs p_{pos} and their normalized representations $f(x), f(x^+)$:

$$\ell_{\text{align}} \triangleq \underset{(x,x^+)\sim p_{\text{pos}}}{\mathbb{E}} \|f(x) - f(x^+)\|^2. \quad (2)$$

Conversely, *uniformity* measures how scattered the embeddings are upon the unit hypersphere:

$$\ell_{\text{uniform}} \triangleq \log \underset{x,y \overset{i.i.d.}{\sim} p_{\text{data}}}{\mathbb{E}} e^{-2\|f(x)-f(y)\|^2}, \quad (3)$$

where p_{data} denotes the full data distribution. We use these metrics to explore the advantages and drawbacks of various augmentations in contrastive pre-training, similarly to Gao et al. (2021).

2.2 Learning Sentence Embeddings

Early works. First approaches to learning sentence embeddings span unsupervised (Kiros et al., 2015; Hill et al., 2016; Logeswaran and Lee, 2018), and supervised (Conneau et al., 2017; Cer et al., 2018; Reimers and Gurevych, 2019) methods which have been studied extensively in the literature. More recent work has focused on unsupervised contrastive learning with the advent of SimCSE (Gao et al., 2021), which passes the same sentence to a language model twice; the independent dropout masks sampled in the two forward passes encode the sentence at slightly different positions in vector space. A cross-entropy objective is then used to maximize the probability of top-1 proximity between positives while uniformly repelling other examples.

Successors to SimCSE. Works that follow SimCSE attempt to improve the framework with auxiliary training objectives (Chuang et al., 2022a; Nishikawa et al., 2022; Zhou et al., 2023; Zhang et al., 2022; Wu et al., 2022b; Wang et al., 2022), verbalized or continuous prompts (Wang et al., 2022; Yuxin Jiang and Wang, 2022), instance generation or weighting strategies (Zhou et al., 2022), momentum encoders with negative sample queues (He et al., 2020), or entirely new parameters with secondary networks (Wu et al., 2022a). Many works combine several of these components, making it difficult to discern their impact in isolation. As the design choices have become more intricate and less parameter-efficient, performance on STS benchmarks has too become saturated.

3 Composition-based Contrastive Learning

Our augmentation strategy retains the simplicity and efficiency of SimCSE, as illustrated in Figure 1. Specifically, it requires just one additional forward pass that is ultimately compensated by a non-trivial reduction in convergence time (§6). Beginning with a corpus of unlabelled sentences $\{x_i\}_{i=1}^m$, we consider x_i^+ only in the latent space, as a composition of the representations of $(x_i^{'+}, x_i^{''+})$. A simple (and effective) way to curate $(x_i^{'+}, x_i^{''+})$ is to split the tokens of x_i in half, and encode the left and right phrases in independent forward passes through the encoder and linear projector. After obtaining their respective `[CLS]` token representations $(z_i, z_i^{'+}, z_i^{''+})$, $(z_i^{'+}, z_i^{''+})$ is aggregrated and taken to be the corresponding positive example for z_i. The training objective for a single pair is then the same as in eq. 1, where $z^+ = aggregate(z_i^{'+}, z_i^{''+})$. We experiment with aggregation methods in §5, and find that the best approach varies according to the size and type of underlying PLM. In our final model based on $\text{BERT}_{\texttt{base}}$, we find that this manner of augmentation is especially suitable for the scheme proposed in DirectCLR (Jing et al., 2022), which aims to directly mitigate dimensional collapse by computing the loss from eq. 1 on a subset of the embedding vector axes before backpropagating to the entire representation.

Decomposition as data augmentation. To explain the motivation for decomposing examples in the input space, we can consider an example from the development subset of STS-B labelled as having high semantic similarity:

```
A man is lifting weights in a garage.
A man is lifting weights.
```

There are two semantic atoms at play in the first text: 1) a man is lifting weights, and 2) a man is in a garage. The similarity between the two texts can only be considered high based on the first atom; lifting weights. It cannot be said that there is a general relation between *being in a garage* and *lifting weights* - a garage is equally, if not more likely to be related to cars, parking, or storage, yet this does not preclude a connection between them. It is only through the composition of both atoms that we can relate the two. Thus, there is a need for sentence encoders to learn more generalized phrase representations; to at least implicitly abide by principles of semantic compositionality. The challenge in enforcing this kind of constraint through a contrastive objective is in the choice of data — it would require a corpus where lexical collocations are encountered across a diverse set of contexts.

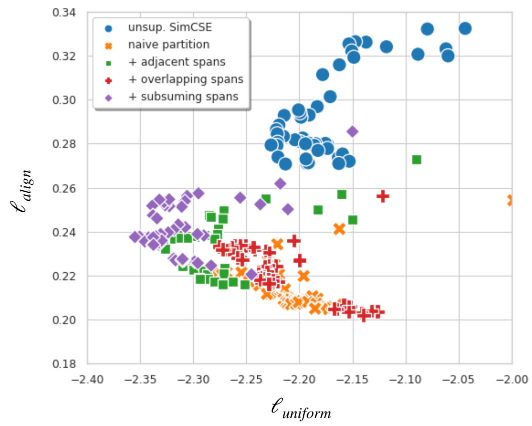


Figure 2: ℓ_{align}-ℓ_{uniform} tradeoff for subsampling strategies explored in this work. Measurements are taken every 10 training steps on the development subset of STS-B, for 500 steps with BERT$_{\texttt{base}}$. The ideal trajectory is the negative direction for both axes/metrics.

Subsampling from decomposed inputs. To further examine the effect of decomposition in the input space, we leverage a pre-trained discourse parser[2] to extract atomic semantic units from each unique example in the training set; typically simple phrases or clauses. We experiment with 3 kinds of strategies (Figure 1a) to expand the training set, besides considering our augmentation in isolation: let $C = \{x_{i,k}\}_{k=1}^{c}$ represent the c non-overlapping phrases extracted from an input x_i :

- **adjacent spans** are sampled by taking each unique pair in C such that there is no overlap between inputs;
- **overlapping and adjacent spans** are sampled by taking (potentially) overlapping pairs in C;
- **overlapping, adjacent, and subsuming spans** are sampled by recursively partitioning the elements of C in half, i.e. maximizing the lexical overlap of extracted input samples.

Impact on the representation space. A consequence of expanding the training set with subsamples is the presence of harder in-batch negatives. Prior work has demonstrated that this is generally beneficial to contrastive learning (Robinson et al., 2021b; Kalantidis et al., 2020; Zhang and Stratos, 2021). Following Gao et al. (2021), we measure the uniformity and alignment of representations obtained for the development set of STS-B to understand the effect of training with additional subsamples. STS-B is comprised of pairs of sentences accompanied by a score between 1-5 indicating degree of semantic similarity. We take all pairs as p_{data}, and pairs with a score greater than 4 as p_{pos}. Both metrics are measured every 10 steps for 500 training steps, to understand the direction in which each of our strategies drives the encoder.

As shown in Figure 2, any of the subsampling strategies can bring non-trivial improvements over unsupervised SimCSE in both alignment and uniformity. Specifically, expanding the training set with subsamples (+ *adjacent,* + *overlapping,* + *subsuming*) encourages a more uniform embedding distribution. On the other hand, forgoing subsampling for just the compositional augmentation (*naive partition*) achieves the better alignment while retaining the uniformity of SimCSE. This is because we leave the self-prediction objective intact, while increasing its difficulty: although subsamples are potentially highly related, positive pairs are only curated from the exact same text. As a consequence, the underlying PLM is forced to effectively distinguish examples with high lexical overlap — which is precisely the intuition underlying DiffCSE Chuang et al. (2022b), and other discriminative pre-training objectives.

4 Experiment

Setup. In our experiments, we modify the public PyTorch implementation[3] of SimCSE to support our proposed augmentation and subsampling methods. All of our language models are initialized from pre-trained BERT/RoBERTa checkpoints (Devlin et al., 2019b; Liu et al., 2019), except the randomly-initialized MLP over the `[CLS]` representation. For all models, we employ the scheme illustrated in Figure 1 and report the best results after training with or without the 3 subsampling strategies. We keep the best checkpoints after evaluating on the development set of STS-B every 125 steps during training. Batch size is fixed at 64 for all models; for base and large sized models, learning rates are fixed to 3e-5 and 1e-5 respectively. Besides those covered in 5, extensive hyperparameter searches were not conducted in this work.

Data. We use the same 1 million randomly sampled sentences[4] as SimCSE for training, be-

[2] https://github.com/seq-to-mind/DMRST_Parser

[3] https://github.com/princeton-nlp/SimCSE

[4] https://huggingface.co/datasets/princeton-nlp/datasets-for-simcse

PLM	Method	STS12	STS13	STS14	STS15	STS16	STS-B	SICK-R	Avg.
BERT$_{\texttt{base}}$	SimCSE♣	68.40	82.41	74.38	80.91	78.56	76.85	72.23	76.25
	L2P-CSR♡	70.21	83.25	75.42	82.34	78.75	77.8	72.65	77.20
	DCLR♠	70.81	83.73	75.11	82.56	78.44	78.31	71.59	77.22
	MoCoSE♢	71.58	81.40	74.47	83.45	78.99	78.68	72.44	77.27
	ArcCSE†	72.08	84.27	76.25	82.32	79.54	79.92	72.39	78.11
	PCL‡	72.74	83.36	76.05	83.07	79.26	79.72	72.75	78.14
	∗SimCSE (*w/ comp.*)	72.14	84.06	75.38	**83.82**	80.43	80.29	71.12	78.18
	ESimCSE◯	**73.40**	83.27	**77.25**	82.66	78.81	80.17	72.30	78.27
	SNCSE$	70.67	**84.79**	76.99	83.69	**80.51**	**81.35**	**74.77**	**78.97**
BERT$_{\texttt{large}}$	SimCSE ♣	70.88	84.16	76.43	84.50	79.76	79.26	73.88	78.41
	DCLR ♠	71.87	84.83	77.37	84.70	79.81	79.55	74.19	78.90
	L2P-CSR ♡	71.44	85.09	76.88	84.71	80.00	79.75	74.55	78.92
	MoCoSE ♢	74.50	84.54	77.32	84.11	79.67	80.53	73.26	79.13
	ESimCSE◯	73.21	85.37	77.73	84.30	78.92	80.73	74.89	79.31
	ArcCSE†	73.17	86.19	77.90	84.97	79.43	80.45	73.50	79.37
	∗SimCSE (+ *subsum.*)	**75.10**	86.57	77.70	84.72	80.25	80.17	73.21	79.67
	PCL‡	74.89	85.88	78.33	85.30	80.13	81.39	73.66	79.94
	SNCSE$	71.94	**86.66**	**78.84**	**85.74**	**80.72**	**82.29**	**75.11**	**80.19**
RoBERTa$_{\texttt{base}}$	SimCSE♣	70.16	81.77	73.24	81.36	80.65	80.22	68.56	76.57
	ESimCSE◯	69.90	82.50	74.68	83.19	80.30	80.99	70.54	77.44
	L2P-CSR♡	71.69	82.43	74.55	82.15	**81.81**	81.36	70.22	77.74
	DCLR♠	70.01	83.08	75.09	83.66	81.06	81.86	70.33	77.87
	∗SimCSE (*w/ comp.*)	**72.56**	83.33	73.67	83.36	81.14	80.71	70.39	77.88
	PCL‡	71.54	82.70	75.38	83.31	81.64	81.61	69.19	77.91
	SNCSE$	70.62	**84.42**	**77.24**	**84.85**	81.49	**83.07**	**72.92**	**79.23**
RoBERTa$_{\texttt{large}}$	∗SimCSE (*w/ comp.*)	72.32	84.19	75.00	84.83	81.27	82.10	70.99	78.67
	SimCSE♣	72.86	83.99	75.62	84.77	81.80	81.98	71.26	78.90
	DCLR♠	73.09	84.57	76.13	85.15	81.99	82.35	71.80	79.30
	PCL‡	**73.76**	84.59	76.81	85.37	81.66	82.89	70.33	79.34
	ESimCSE◯	73.20	84.93	76.88	84.86	81.21	82.79	72.27	79.45
	L2P-CSR♡	73.29	84.08	76.65	85.47	82.70	82.15	72.36	79.53
	SNCSE$	73.71	**86.73**	**80.35**	**86.80**	**83.06**	**84.31**	**77.43**	**81.77**

Table 1: The performance on STS tasks (Spearman's correlation) for different sentence embedding models. Results are imported as follows — ♣: Gao et al. (2021), ♡: Zhou et al. (2023), ♠: Zhou et al. (2022), ♢: Cao et al. (2022), †: Zhang et al. (2022), ‡: Wu et al. (2022a), ◯: Wu et al. (2022c), $: Wang et al. (2022), ∗: our results.

sides incorporating the subsampling strategies from §3. We evaluate on 7 semantic textual similarity tasks: STS 2012-2016, STS-Benchmark, SICK-Relatedness (Agirre et al., 2012, 2013, 2014, 2015, 2016; Cer et al., 2017; Marelli et al., 2014) and report averaged Spearman's correlation across all available test subsets. We employ the modified SentEval[5] (Conneau and Kiela, 2018) package accompanying the source code of SimCSE for fair comparison with other works.

Baselines. We compare our results with many contemporaries: ESimCSE (Wu et al., 2022c), SNCSE (Wang et al., 2022), PCL (Wu et al., 2022a), DCLR (Zhou et al., 2022), ArcCSE (Zhang et al., 2022), MoCoSE (Cao et al., 2022), and L2P-CSR (Zhou et al., 2023). We consider SimCSE (Gao et al., 2021) as our baseline, since we leave its training objective and network architecture intact.

Results. We can observe in Table 1 that our methods bring non-trivial improvements to SimCSE with both BERT encoders, as well as RoBERTa$_{\texttt{base}}$. In fact, we achieve an average F1 score within 0.8 points of SNCSE-BERT$_{\texttt{base}}$ (Wang et al., 2022). SNCSE exploits biases in test sets by engineering hard negatives via explicitly negated sentences — the impact of this strategy is more apparent in the results utilizing RoBERTa, where there is parity in all works besides SNCSE. In the case of BERT$_{\texttt{large}}$, the gap in performance between our approach and SNCSE is narrower at 0.52 points. A clear failure of the composition-augmented objective presents itself in the results with RoBERTa$_{\texttt{large}}$. This could be attributed to poor hyperparameter settings, or a fundamental incompatibility between our approach and the model size/RoBERTa pre-training objective, since other works achieve better results with this PLM.

[5]https://github.com/facebookresearch/SentEval

5 Ablation

We ablate several aspects of the approach to understand their impact in isolation. We first consider the subsampling strategy, or lack thereof, in which each model achieves the best STS-B development set performance. These are then tied to each model in subsequent ablations.

Including subsamples. In the process of designing DeCLUTR, Giorgi et al. (2021b) report gains from subsampling more than one anchor per input document. In our experiments, we find that the aligment-uniformity trade-off differs between BERT$_{\texttt{large}}$ and BERT$_{\texttt{base}}$, ie. different strategies can be better suited to different PLMs. In Table 2, we show that including subsamples is beneficial to the BERT$_{\texttt{large}}$ PLM, but harmful to BERT$_{\texttt{base}}$. This is likely a result of the difference in no. of parameters — the smaller PLM may not possess the expressive capacity to distinguish highly related texts without suffering a degeneration in alignment. With RoBERTa$_{\texttt{base}}$, we observe that subsampling non-overlapping spans gives the best results, whereas none of our strategies appeared compatible with RoBERTa$_{\texttt{large}}$.

PLM	Method	STS-B
BERT$_{\texttt{base}}$	SimCSE	81.47
	w/ composition	**83.97**
	Additional subsampling:	
	+ *adjacent*	83.39
	+ *overlapping*	83.18
	+ *subsuming*	82.97
BERT$_{\texttt{large}}$	SimCSE	84.41
	w/ composition	84.79
	Additional subsampling:	
	+ *adjacent*	84.84
	+ *overlapping*	85.01
	+ *subsuming*	**85.06**
RoBERTa$_{\texttt{base}}$	SimCSE	83.91
	w/ composition	**84.14**
	Additional subsampling:	
	+ *adjacent*	84.00
	+ *overlapping*	84.10
	+ *subsuming*	82.92
RoBERTa$_{\texttt{large}}$	SimCSE	**85.07**
	w/ composition	<u>84.80</u>
	Additional subsampling:	
	+ *adjacent*	83.91
	+ *overlapping*	82.74
	+ *subsuming*	83.33

Table 2: Development set results of STS-B after varying the subsampling strategy on different-sized PLMs.

Aggregration method. In SBERT (Reimers and Gurevych, 2019), and in historically effective works such as InferSent (Conneau et al., 2017), PLMs are fine-tuned with a cross entropy loss to predict whether two sentences u and v entail or contradict eachother. Their pooler is a concatenation of the two sentence embeddings, along with second-order features such as the element-wise difference, $|u - v|$. We experiment with these aggregration methods, as well as simpler choices such as element-wise sums/averages. We can see in Table 3 that simply interpolating the embeddings is preferable to other methods for BERT-based encoders. We postulate that this interpolation functions as a form of self-distillation, and amplifies the salience of desirable sparse features correlated with sentential context (Wen and Li, 2021). For RoBERTa, we find that concatenating the first and last halves of the representations is better. Since RoBERTa does not use the next-sentence prediction (NSP) objective, its embeddings will not encode sentential knowledge. Averaging RoBERTa embeddings may not correlate well with real tokens in its vocabulary, whereas concatenating the first and last halves of constituent embeddings retains localized token-level information, making it a better choice in this case.

Aggregration	STS-B
BERT$_{\texttt{base}}$	
sum	83.92
avg.	**83.97**
concat first & last half	83.01
concat + project	69.24
concat w/ abs. difference + project	68.79
RoBERTa$_{\texttt{base}}$	
sum	84.00
avg.	84.08
concat first & last half	**84.14**
concat + project	65.02
concat w/ abs. difference + project	65.44

Table 3: Results of different aggregration methods for composing z^+ in the latent space. Results are based on BERT$_{\texttt{base}}$ on the development set of STS-B.

Composing z vs. z^+. In our training objective, there are two sets of sentence representations, one derived from pure dropout noise, and the second by averaging the coordinates of constituent representations. However, for each sentence we can: 1) compose the anchor z in latent space, which means other in-batch examples are repelled from a synthetic example's coordinate, 2) compose the positive z^+, which means synthetic coordinates are repelled from representations of real examples, or 3) compose both z and z^+ in the latent space. In Table 4, we can see that with BERT$_{\texttt{base}}$, we found the best results by directly embedding the anchor sentence, and composing z^+ from constituents.

BERT$_{\texttt{base}}$	**Compose**	z	z^+	*Both*
	STS-B	83.61	**83.97**	83.81

Table 4: Differences between having compositional anchors and positives. In the *Both* case, the model framework is symmetric in that both anchors and positives are composed of constituent representations. Results are based on BERT$_{\texttt{base}}$ on the development set of STS-B.

Number of partitions. Within our framework, we can aggregrate the embeddings of two or more phrases. Increasing the number of phrases increases the number of forward passes, and magnifies the impact of dropout noise. We find that partitioning into more than two bins is detrimental to the objective (Table 5), though perhaps this is the case because the evaluation data consists mostly of short-length sentences.

BERT$_{\texttt{base}}$	**Partitions**	2	3	4
	STS-B	**83.97**	83.48	83.52

Table 5: Impact of splitting examples into more than 2 bins. Results are based on BERT$_{\texttt{base}}$ with the development set of STS-B.

Hyperparameter d_0. In our experiments with BERT$_{\texttt{base}}$, computing the contrastive loss on a subvector of (z_i, z_i^+) is complementary to composing z_i^+ in the latent space. When $d_0 \to d$, our training objective is the exact same as in all *CSE works, ie. computing the loss on all coordinates of (z_i, z_i^+). For BERT$_{\texttt{base}}$, we search $d_0 \in \{192, 256, 384\}$ with the compositional augmentation in isolation (*w/ composition*); for BERT$_{\texttt{large}}$, $d_0 \in \{320, 384, 512\}$ with the expanded training set of subsamples (+ *subsuming*). Our results in Table 6 indicate that taking a subvector to compute the loss is beneficial for BERT$_{\texttt{base}}$, but the entire vector is necessary for BERT$_{\texttt{large}}$. With RoBERTa encoders, we aggregrate embeddings by concatenating the first and last halves of the phrase embeddings, so d_0 is inapplicable.

BERT$_{\texttt{base}}$	d_0	*192*	*256*	*384*	*768*
	STS-B	83.88	**84.11**	83.17	83.97
BERT$_{\texttt{large}}$	d_0	*320*	384	*512*	*1024*
	STS-B	84.61	84.94	84.98	**85.06**

Table 6: Varying the size of the subvector used to compute InfoNCE, as proposed in Jing et al. (2022). Results are based on the development set of STS-B.

6 Analysis

Stability and efficiency of training. Successors to SimCSE have incrementally improved STS performance while disproportionately driving up resource requirements. This limits accessibility to practitioners who wish to learn embeddings from their own corpora, perhaps in other languages. Differently, our approach relies on a single additional forward pass while converging much faster than SimCSE. In Figure 3, we compare our BERT$_{\texttt{base}}$ model's evaluation curve to SimCSE's for 1000 training steps in the same setting. We observe that composition as augmentation greatly speeds up convergence, with evaluation metrics plateauing much faster, and more stably than SimCSE. In fact, on a single NVIDIA A100 GPU (40GB), our model can finish training in under 15 minutes.

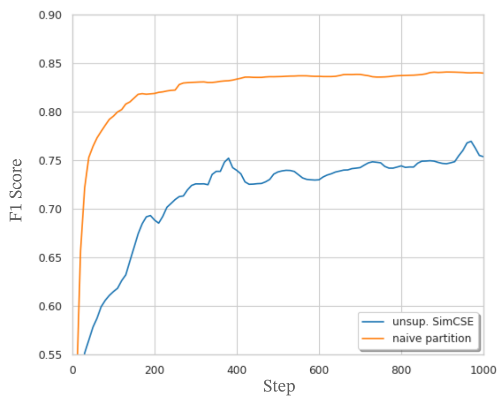


Figure 3: Evaluation curve for BERT$_{\texttt{base}}$, using dropout noise as augmentation (unsup. SimCSE) and latent space composition (naive partition). The y-axis reflects performance on the development set of STS-B.

Text length as a feature. To investigate the structure of the learned space, In Figure 5, we visualize embeddings of sentences from the development set of STS-B after down-projecting to 2D Euclidean space. We employ UMAP (McInnes et al., 2018) with cosine distance as the metric to preserve local and global topological neighborhoods. The same

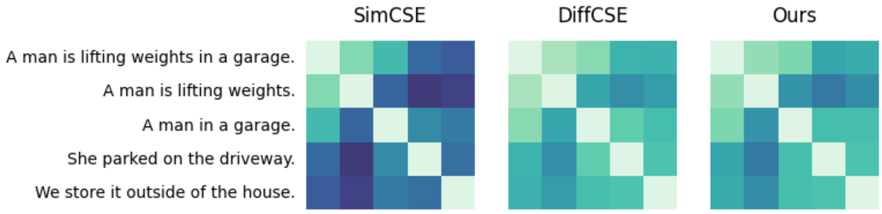
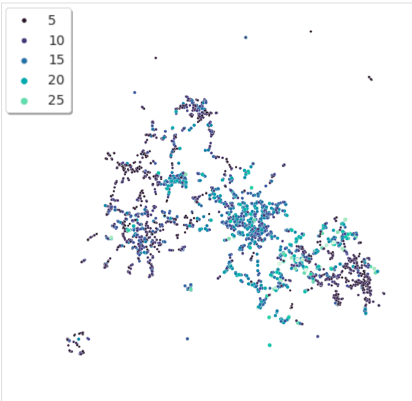
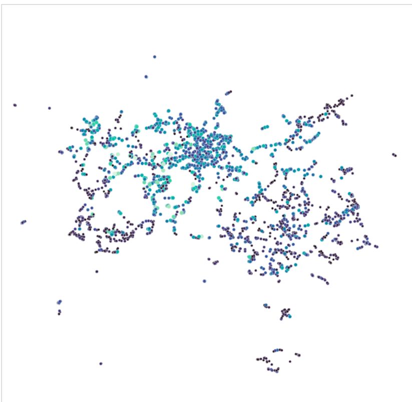


Figure 4: A heatmap displaying the pairwise similarities computed by SimCSE, DiffCSE, and our model, with the same underlying PLM BERT$_{\texttt{base}}$ and color scale. Lighter colors indicate higher similarity.



(a) unsup. SimCSE



(b) w/ compositional augmentations

Figure 5: 2D UMAP projection of the representations of all sentences from the validation subset of STS-B. Color indicates word count.

parameters are used to compute the embeddings in Figure 5a and 5b, which are derived from dropout noise, and composition-based augmentations (*w/ composition*) respectively. In Figure 5a, we can observe several clusters of dark points corresponding to shorter sentences. This corroborates our intuition that minimal augmentation to create positive pairs can lead to shortcut learning, wherein text length is relied upon to solve the training objective. In contrast, we see a more scattered distribution of points in Figure 5b, particularly with shorter sentences. Coupled with the improved performance on STS tasks, we can conclude that our framework is less prone to learning from spurious correlations.

Learned similarity metric. Returning to the example initially posed in §3, we show in Figure 4 similarity scores for pairs of examples computed by our BERT$_{\texttt{base}}$ model, as well as the corresponding DiffCSE and SimCSE variants. Notice that all three assign higher similarities between anchor: "A man is lifting weights in a garage", and phrases: "A man is lifting weights", "A man in a garage". However, despite their equal constitution in the anchor text, SimCSE incorrectly assesses a higher similarity between the anchor and the first phrase, whereas DiffCSE and our model better capture the equivalence in similarity. The same occurs with anchor: "We store it outside of the house", and texts: "A man is in a garage", "She parked on the driveway"; despite both being unrelated to the anchor, SimCSE spuriously assigns a higher affinity to the former. Overall, we observed parity in the similarity assessments given by our model and DifCSE, which validates the ability of our approach to remedy the suboptimal alignment of SimCSE without explicit incentive.

7 Conclusion

In summary, we proposed a new way to construct positive pairs for unsupervised contrastive learning frameworks relying on pre-trained language models. Our experiments on STS tasks verified the effectiveness of the approach, which achieved competitive results with more complex learning methods, with the benefit of stabilizing and reducing the overall cost of training. We provided empirical studies and qualitative examinations into our approach, verifying its ability to train sentence encoders with better alignment. We believe this work can foster new avenues of inquiry in contrastive learning, especially those that draw upon a *human* cognition of language.

8 Limitations

There are several limitations in this work. First, we have not explored how to make use of composition-based augmentations in the supervised setting. A second limitation is a lack of theoretical grounding in the impact of our latent space composition. Finally, we have not explored interoperability with other training objectives.

9 Note on Ethics

We do not believe there are significant ethical considerations stemming from our work, except those that accompany the use of language models and unlabelled corpora in general. Pre-trained language models, including BERT and RoBERTa, are known to learn and reiterate harmful prejudices. Although our pre-training corpus is sourced from Wikipedia and cited in several related works, it cannot be feasibly vetted for explicit or inappropriate content.

Acknowledgements

We thank the anonymous reviewers for their valuable feedback and input. We also thank Haotian Xu, for his insight and suggestions that shaped the course of this work. We gratefully acknowledge support from National Science Foundation (NSF) via the awards IIS-1942918 and IIS-2127746. Portions of this research were conducted with the advanced computing resources provided by Texas A&M High Performance Research Computing.

References

Eneko Agirre, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Iñigo Lopez-Gazpio, Montse Maritxalar, Rada Mihalcea, German Rigau, Larraitz Uria, and Janyce Wiebe. 2015. SemEval-2015 task 2: Semantic textual similarity, English, Spanish and pilot on interpretability. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 252–263, Denver, Colorado. Association for Computational Linguistics.

Eneko Agirre, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Rada Mihalcea, German Rigau, and Janyce Wiebe. 2014. SemEval-2014 task 10: Multilingual semantic textual similarity. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 81–91, Dublin, Ireland. Association for Computational Linguistics.

Eneko Agirre, Carmen Banea, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Rada Mihalcea, German Rigau, and Janyce Wiebe. 2016. SemEval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 497–511, San Diego, California. Association for Computational Linguistics.

Eneko Agirre, Daniel Cer, Mona Diab, and Aitor Gonzalez-Agirre. 2012. SemEval-2012 task 6: A pilot on semantic textual similarity. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 385–393, Montréal, Canada. Association for Computational Linguistics.

Eneko Agirre, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, and Weiwei Guo. 2013. *SEM 2013 shared task: Semantic textual similarity. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity*, pages 32–43, Atlanta, Georgia, USA. Association for Computational Linguistics.

Sanjeev Arora, Hrishikesh Khandeparkar, Mikhail Khodak, Orestis Plevrakis, and Nikunj Saunshi. 2019. A theoretical analysis of contrastive unsupervised representation learning. In *36th International Conference on Machine Learning, ICML 2019*, 36th International Conference on Machine Learning, ICML 2019, pages 9904–9923. International Machine Learning Society (IMLS). Publisher Copyright: © 2019 International Machine Learning Society (IMLS).; 36th International Conference on Machine Learning, ICML 2019 ; Conference date: 09-06-2019 Through 15-06-2019.

Rui Cao, Yihao Wang, Yuxin Liang, Ling Gao, Jie Zheng, Jie Ren, and Zheng Wang. 2022. Exploring the impact of negative samples of contrastive learning: A case study of sentence embedding. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3138–3152, Dublin, Ireland. Association for Computational Linguistics.

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14, Vancouver, Canada. Association for Computational Linguistics.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder for English. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 169–174, Brussels, Belgium. Association for Computational Linguistics.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning (ICML)*, pages 1597–1607.

Yung-Sung Chuang, Rumen Dangovski, Hongyin Luo, Yang Zhang, Shiyu Chang, Marin Soljacic, Shang-Wen Li, Scott Yih, Yoon Kim, and James Glass. 2022a. DiffCSE: Difference-based contrastive learning for sentence embeddings. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4207–4218, Seattle, United States. Association for Computational Linguistics.

Yung-Sung Chuang, Rumen Dangovski, Hongyin Luo, Yang Zhang, Shiyu Chang, Marin Soljačić, Shang-Wen Li, Wen-tau Yih, Yoon Kim, and James Glass. 2022b. DiffCSE: Difference-based contrastive learning for sentence embeddings.

Alexis Conneau and Douwe Kiela. 2018. SentEval: An evaluation toolkit for universal sentence representations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. European Language Resources Association (ELRA).

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 670–680.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019a. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019b. BERT: Pre-training of deep bidirectional transformers for language understanding.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

John Giorgi, Osvald Nitski, Bo Wang, and Gary Bader. 2021a. DeCLUTR: Deep contrastive learning for unsupervised textual representations. In *Association for Computational Linguistics and International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pages 879–895.

John Giorgi, Osvald Nitski, Bo Wang, and Gary Bader. 2021b. DeCLUTR: Deep contrastive learning for unsupervised textual representations. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 879–895, Online. Association for Computational Linguistics.

Raia Hadsell, Sumit Chopra, and Yann LeCun. 2006. Dimensionality reduction by learning an invariant mapping. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 1735–1742. IEEE.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning.

Felix Hill, Kyunghyun Cho, and Anna Korhonen. 2016. Learning distributed representations of sentences from unlabelled data. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1367–1377, San Diego, California. Association for Computational Linguistics.

Li Jing, Pascal Vincent, Yann LeCun, and Yuandong Tian. 2022. Understanding dimensional collapse in contrastive self-supervised learning. In *International Conference on Learning Representations*.

Yannis Kalantidis, Mert Bulent Sariyildiz, Noe Pion, Philippe Weinzaepfel, and Diane Larlus. 2020. Hard negative mixing for contrastive learning. In *Neural Information Processing Systems (NeurIPS)*.

Ryan Kiros, Yukun Zhu, Ruslan Salakhutdinov, Richard S Zemel, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. 2015. Skip-thought vectors. In *Advances in Neural Information Processing Systems (NIPS)*, pages 3294–3302.

J. Lee, Qi Lei, Nikunj Saunshi, and Jiacheng Zhuo. 2020. Predicting what you already know helps: Provable self-supervised learning. In *Neural Information Processing Systems*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Lajanugen Logeswaran and Honglak Lee. 2018. An efficient framework for learning sentence representations. In *International Conference on Learning Representations (ICLR)*.

Marco Marelli, Stefano Menini, Marco Baroni, Luisa Bentivogli, Raffaella Bernardi, and Roberto Zamparelli. 2014. A SICK cure for the evaluation of compositional distributional semantic models. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 216–223, Reykjavik, Iceland. European Language Resources Association (ELRA).

Leland McInnes, John Healy, and James Melville. 2018. Umap: Uniform manifold approximation and projection for dimension reduction.

Sosuke Nishikawa, Ryokan Ri, Ikuya Yamada, Yoshimasa Tsuruoka, and Isao Echizen. 2022. EASE: Entity-aware contrastive learning of sentence embedding. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3870–3885, Seattle, United States. Association for Computational Linguistics.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding.

Senthil Purushwalkam and Abhinav Gupta. 2020. Demystifying contrastive self-supervised learning: Invariances, augmentations and dataset biases.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Joshua Robinson, Li Sun, Ke Yu, Kayhan Batmanghelich, Stefanie Jegelka, and Suvrit Sra. 2021a. Can contrastive learning avoid shortcut solutions? In *Advances in Neural Information Processing Systems*, volume 34, pages 4974–4986. Curran Associates, Inc.

Joshua David Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. 2021b. Contrastive learning with hard negative samples. In *International Conference on Learning Representations*.

Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. 2020. What Makes for Good Views for Contrastive Learning? In *Advances in Neural Information Processing Systems*, volume 33, pages 6827–6839. Curran Associates, Inc.

Christopher Tosh, Akshay Krishnamurthy, and Daniel J. Hsu. 2020. Contrastive learning, multi-view redundancy, and linear models. *ArXiv*, abs/2008.10150.

Hao Wang, Yangguang Li, Zhen Huang, Yong Dou, Lingpeng Kong, and Jing Shao. 2022. SNCSE: Contrastive learning for unsupervised sentence embedding with soft negative samples.

Tongzhou Wang and Phillip Isola. 2020. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning (ICML)*, pages 9929–9939.

Zixin Wen and Yuanzhi Li. 2021. Toward Understanding the Feature Learning Process of Self-supervised Contrastive Learning.

Mike Wu, Chengxu Zhuang, Milan Mosse, Daniel Yamins, and Noah Goodman. 2020. On Mutual Information in Contrastive Learning for Visual Representations.

Qiyu Wu, Chongyang Tao, Tao Shen, Can Xu, Xiubo Geng, and Daxin Jiang. 2022a. PCL: Peer-contrastive learning with diverse augmentations for unsupervised sentence embeddings.

Xing Wu, Chaochen Gao, Zijia Lin, Jizhong Han, Zhongyuan Wang, and Songlin Hu. 2022b. InfoCSE: Information-aggregated contrastive learning of sentence embeddings.

Xing Wu, Chaochen Gao, Liangjun Zang, Jizhong Han, Zhongyuan Wang, and Songlin Hu. 2022c. ESimCSE: Enhanced sample building method for contrastive learning of unsupervised sentence embedding.

Yuanmeng Yan, Rumei Li, Sirui Wang, Fuzheng Zhang, Wei Wu, and Weiran Xu. 2021. ConSERT: A contrastive framework for self-supervised sentence representation transfer. In *Association for Computational Linguistics and International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pages 5065–5075.

Linhan Zhang Yuxin Jiang and Wei Wang. 2022. Improved universal sentence embeddings with prompt-based contrastive learning and energy-based learning.

Wenzheng Zhang and Karl Stratos. 2021. Understanding hard negatives in noise contrastive estimation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1090–1101, Online. Association for Computational Linguistics.

Yuhao Zhang, Hongji Zhu, Yongliang Wang, Nan Xu, Xiaobo Li, and Binqiang Zhao. 2022. A contrastive framework for learning sentence representations from pairwise and triple-wise perspective in angular space. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4892–4903, Dublin, Ireland. Association for Computational Linguistics.

Kun Zhou, Beichen Zhang, Xin Zhao, and Ji-Rong Wen. 2022. Debiased contrastive learning of unsupervised sentence representations. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6120–6130, Dublin, Ireland. Association for Computational Linguistics.

Kun Zhou, Yuanhang Zhou, Xin Zhao, and Ji-Rong Wen. 2023. Learning to perturb for contrastive learning of unsupervised sentence representations.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
We provided a negative result in Section 4. We also dedicated a section to this end: Section 8.

☑ A2. Did you discuss any potential risks of your work?
Provided in the appendix, although it is mostly a general statement on the use of pre-trained language models and unlabelled corpora.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Sections 1-2: "Vector representations of natural language are ubiquitous in search, retrieval and reranking applications. Various methods based on contrastive learning have been proposed to learn textual representations from unlabelled data. These approaches maximize agreement between minimally perturbed texts, while uniformly repelling examples within a broader corpus."
Section 3: "Different from previous works, we propose to maximize agreement between sentences and a composition of their semantic constituents. We consider several interpretations of this objective and elaborate the impact on resultant embeddings in each case."
Section 4: "Experimental results on semantic textual similarity tasks demonstrate improvements over baselines that are on par with contemporary approaches."
Sections 1-6: "Moreover, this work is the first to do so without incurring costs in auxiliary training objectives or additional network parameters."

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Sections 4-6.

☑ B1. Did you cite the creators of artifacts you used?
Sections 4-6.

☐ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Not applicable. Left blank.

☐ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Not applicable. Left blank.

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Not applicable. Left blank.

☐ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Not applicable. Left blank.

☒ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
We include a footnote which links to a GitHub page where these statistics are available in detail; see Section 5. Related works do not report these statistics in the body or appendix of their paper.

C ☑ Did you run computational experiments?

Sections 3-6.

☒ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
We utliize BERT and RoBERTa, and cite their papers where these parameters are listed. Additionally, we report on efficiency as a benefit of our approach in Section 6.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Section 4

☒ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
We did not conduct extensive hyperparameter searches and did not repeat experiments due to time and resource constraints.

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
We provide footnotes with links to all source codes utilized in this work; Sections 3-5

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.