Smoothed Contrastive Learning for Unsupervised Sentence Embedding

Xing Wu[1,2,3], **Chaochen Gao**[1,2]*, **Yipeng Su**[1], **Jizhong Han**[1], **Zhongyuan Wang**[3], **Songlin Hu**[1,2]†
[1]Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[2]School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
[3]Kuaishou Technology, Beijing, China
{wuxing,gaochaochen,suyipeng,hanjizhong,husonglin}@iie.ac.cn
wangzhongyuan@kuaishou.com

Abstract

Unsupervised contrastive sentence embedding models, e.g., unsupervised SimCSE, use the InfoNCE loss function in training. Theoretically, we expect to use larger batches to get more adequate comparisons among samples and avoid overfitting. However, increasing batch size leads to performance degradation when it exceeds a threshold, which is probably due to the introduction of false-negative pairs through statistical observation. To alleviate this problem, we introduce a simple smoothing strategy upon the InfoNCE loss function, termed Gaussian Smoothed InfoNCE (GS-InfoNCE). In other words, we add random Gaussian noise as an extension to the negative pairs without increasing the batch size. Through experiments on the semantic text similarity tasks, though simple, the proposed smoothing strategy brings improvements to unsupervised SimCSE. Our code are available at https://github.com/caskcsg/gsInfoNCE.

1 Introduction

Good sentence representation benefits many natural language processing tasks, and sentence representation learning has been widely studied (Logeswaran and Lee, 2018; Reimers and Gurevych, 2019). Contrastive learning has recently been proposed and extensively explored to learn high-quality sentence representations based on the pre-trained language models(Devlin et al., 2018; Liu et al., 2019). Contrastive learning aims to learn effective representation by pulling close semantically similar sentences while pushing apart dissimilar ones (Hadsell et al., 2006). Among those unsupervised sentence embedding learning methods with contrastive learning, the latest state-of-the-art method, as far as we know, is unsupervised SimCSE (unsup-SimCSE) (Gao et al., 2021). unsup-SimCSE implicitly hypothesizes "dropout" as minimal data augmentation and assumes a sentence is semantically more similar to its augmented counterpart than any other sentence. Though simple, unsup-SimCSE works surprisingly well, performing on par with previously supervised counterparts.

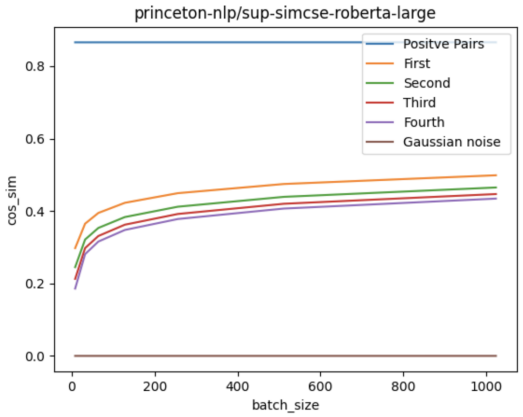

Figure 1: The changing trend of the cosine similarities of the negative pairs in the batch. As the batch_size increases, the mean values of the top 4 cosine similarities also increase, indicating negative pairs with lower confidence exists.

Theoretically, since contrastive learning is carried out among samples within a batch, increasing the batch size will probably bring more adequate comparisons and avoid overfitting. However, according to the original unsup-SimCSE paper (Gao et al., 2021), a larger batch size does not always lead to improvements. The performance even decreases when the batch size exceeds a threshold. We assume that as the batch size increases, more similar sentence samples are probably introduced and easily constitute false-negative pairs, which is detrimental to the learning of the model. We design a probing statistical experiment for different batch sizes to verify our assumption. We use the currently best semantic textual similarity model, i.e., the SimCSE-RoBERTa$_{large}$ (Gao et al., 2021) to measure the cosine similarity of sentence pairs. Randomly sampling a batch with N sentences, we

*The first two authors contribute equally.
†Corresponding author.

Proceedings of the 29th International Conference on Computational Linguistics, pages 4902–4906
October 12–17, 2022.

measure the similarity between all negative pairs within the batch. We calculate the batch's top 4 mean similarity values. We repeat the procedure 100 times and average to eliminate randomness. As shown in Figure1, the top 4 similarity values increase as the batch size increases. It means that, in a larger batch, there will be negative pairs comprised of more similar sentences. When the batch size does not exceed a threshold, the negative pairs of similar sentences are hard negatives and good for training. But when the batch size exceeds, false-negative pairs with higher similarity are introduced, which will mislead the model training. Therefore, achieving sufficient comparison for samples in a "confident" (not too large) batch is particularly important.

As shown in the figure 1, Gaussian noise is far away from all samples and can constitute a very confident negative pair with any sample within a batch. Therefore, we propose to add random Gaussian noise as an extension to the negative pairs without increasing the batch size [1]. In other words, we introduce a simple smoothing strategy upon the InfoNCE loss function by simply adding a Gaussian noise term to the denominator, termed Gaussian Smoothed InfoNCE (GS-InfoNCE). From two perspectives, the Gaussian noise term can be understood as a smoothing strategy. Firstly, the number of negative pairs in a given batch is limited and discrete, and these pairs are used to approximate the negative distribution. We can make the distribution smoother by adding random Gaussian noise to extend the negative pairs. Secondly, from the perspective of the loss function, the denominator of GS-InfoNCE's loss introduces an additional penalty term to avoid overfitting. Through experiments on the semantic text similarity (STS) tasks, GS-InfoNCE outperforms the state-of-the-art unsup-SimCSE by an average Spearman correlation of 1.38%, 0.72%, 1.17% and 0.28% on the base of BERT-base, BERT-large, RoBERTa-base and RoBERTa-large, respectively.

Our contributions can be summarized as follows: we propose GS-InfoNCE for unsup-SimCSE, by introducing a simple smoothing strategy upon the InfoNCE loss function to bring sufficient comparison for samples without increasing the batch size. Our approach can bring improvements to unsup-SimCSE with different model configurations through experiments.

[1] A contemporaneous work (Zhou et al., 2022) has also randomly initialized new negatives based on random Gaussian noises to simulate sampling within the whole semantic space, and devise a gradient-based algorithm to optimize the noise-based negatives.

2 Background: Contrastive Learning

Contrastive learning is a discriminative representation learning framework extensively used for unsupervised representation learning. The core idea is to compare a sentence with a semantically similar one (i.e., positive example) and many semantically dissimilar ones (i.e., negative examples). In this way, the semantically similar sentences are closer in the representation space, while the semantically dissimilar ones are farther apart.

InfoNCE (Chen et al., 2020) propose to take a cross-entropy objective with in-batch negatives, namely the InfoNCE objective function. It is a commonly used loss function for contrast learning by pulling similar sentences closer and pushing dissimilar ones apart in the representation space. Specifically, given a set of sentence pairs: $\mathcal{D} = \left\{\left(x_i, x_i^+\right)\right\}_{i=1}^m$, where x_i and x_i^+ are the ith pair of semantically related sentences. Let $\mathbf{h}_i$ and $\mathbf{h}_i^+$ denote the semantical representations of x_i and x_i^+, for a mini-batch with N pairs, the training loss for (x_i, x_i^+) is:

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

where τ is a temperature hyperparameter and $\text{sim}\left(\mathbf{h}_i, \mathbf{h}_i^+\right)$ is the similarity measurement function, which is typically the cosine similarity function.

Unsupervised SimCSE The idea of unsup-SimCSE is quite simple: each positive pair takes the same sentence as input and utilizes "dropout" as minimal data augmentation. In detail, it takes a collection of sentences $\{x_i\}_{i=1}^m$ and use $x_i^+ = x_i$. It feeds the same input to the encoder twice by applying different dropout masks on fully-connected layers and attention probabilities in the transformer. Through training, positive pair embeddings obtained are similar in the representation space.

3 Gaussian Smoothed InfoNCE

We introduce a Gaussian noise term to the InfoNCE loss function, termed Gaussian Smoothed InfoNCE (GS-InfoNCE). Given a Gaussian distribution as

follows: $G \sim N\left(\mu, \sigma^2\right)$, whose mean is μ, and the variance is σ^2, we randomly sample M Gaussian noise vectors from it with the same dimensions as the sentence vector. These vectors constitute high confident negative pairs with each sample in the batch to fill and smooth the representation space. Note that these Gaussian noise vectors will not participate in the positive pair constitution. In that way, the loss function of GS-InfoNCE is denoted as follows:

$$\ell_i = -\log \frac{e^{\text{sim}\left(\mathbf{h}_i, \mathbf{h}_i^+\right)/\tau}}{\sum_{j=1}^{N} e^{\text{sim}(\mathbf{h}_j, \mathbf{h}_i)/\tau} + \lambda \cdot \sum_{k=1}^{M} e^{\text{sim}(\mathbf{g}_k, \mathbf{h}_i)/\tau}} \tag{2}$$

where $\mathbf{g}_k$ is a random Gaussian noise vector, M is the number of Gaussian noise vectors involved in the calculation, and λ is a balance hyperparameter.

The python implementation of GS-InfoNCE is quite simple, with only three lines of codes based on the original InfoNCE implementation in unsup-SimCSE.

4 Experiments

We focus on unsup-SimCSE and replace the original InfoNCE objective loss function with GS-InfoNCE. Following (Gao et al., 2021), the main goal of sentence embeddings is to cluster semantically similar sentences. For a fair comparison, we conduct our experiments on seven semantic textual similarity (STS) tasks introduced below and take STS results to compare sentence embedding methods.

Semantic textual similarity tasks Semantic textual similarity measures the semantic similarity of any two sentences. STS 2012–2016 (Agirre et al., 2012, 2013, 2014, 2015, 2016)and STS Benchmark (Cer et al., 2017) are widely used semantic textual simiarlty benchmark datasets. Following unsup-SimCSE, we use Spearman correlation[2] to measure the correlation between the ranks of predicted scores and the ground-truth.

Training details The training details of unsup-SimCSE can be found in (Chen et al., 2020) and github[3]. Our experimental settings are consistent with the original method. For the Gaussian distribution, we empirically use the standard normal distribution, with $\mu = 0, \sigma^2 = 1$. Additionally, we set $\lambda = 1$ and $M = 3 \times batch\_size$ for all experiments. As illustrated in Figure 1, we have confirmed that increasing the batch size will introduce false-negative pairs with high similarity, so in our experiments, we set the batch size to a moderate size of 64. Following unsup-SimCSE, we conduct experiments on four commonly used models: BERT-base, BERT-large, RoBERTa-base and RoBERTa-large.

Model	SimCSE	+ GS-InfoNCE
BERT$_{base}$	64	64
BERT$_{large}$	64	64
RoBERTa$_{base}$	512	64
RoBERTa$_{large}$	512	64

Table 1: Comparison of $batch\_size$ with or without using GS-InfoNCE in unsup-SimCSE.

Main Results We list the experimental results in Table 2. On the BERT$_{base}$ model, in terms of Spearman correlation, our GS-InfoNCE brings an average increase of 1.38% over unsup-SimCSE on seven test sets, and the maximum gain on STS-B reach 2.85%. On the BERT$_{large}$ model, our GS-InfoNCE gives unsup-SimCSE an average improvement of 0.55% on the 7 test sets, although there is a slight decrease on the SICK15 and SICK-R data sets. On the RoBERTa$_{base}$ and RoBERTa$_{large}$ models, we have a similar situation, with an average improvement of 1.17% and 0.31% on the 7 test sets.

In general, the improvement brought by GS-InfoNCE to unsup-SimCSE is comprehensive. We can fully surpass the previous best model results with the same or smaller batch size in different model configurations, which well demonstrates that our smoothing strategy has played a key role. We believe that a finer search of the parameters can achieve better results and we leave it to our future work.

Analysis: Effect of hyperpraramter M Gaussian random noise constitutes high-confidence negative pairs with the sentences in a batch. M is the number of Gaussian noise vectors involved in the GS-InfoNCE calculation. We further explore the influence of M on the performance of GS-InfoNCE on BERT$_{base}$. We reuse the hyperparameters of the

[2] https://en.wikipedia.org/wiki/Spearman%27s_rank_correlation_coefficient

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

```
    # ... code from origianl unsup-SimCSE above...
    z1, z2 = pooler_output[:,0], pooler_output[:,1]
    cos_sim = cls.sim(z1.unsqueeze(1), z2.unsqueeze(0))
    reg_random = torch.normal(mean, std, size=(reg_size, hidden_size)).to(device)
    reg_cos_sim = cls.sim(z1.unsqueeze(1), reg_random.unsqueeze(0))
    cos_sim = torch.cat((cos_sim, reg_cos_sim),1).to(device)
    labels = torch.arange(cos_sim.size(0)).long().to(cls.device)
    loss_fct = nn.CrossEntropyLoss()
    # ... code from origianl unsup-SimCSE below...
```

Listing 1: Codes in red are regularization modifications to the original InfoNCE loss

Model	STS12	STS13	STS14	SICK15	STS16	STS-B	SICK-R	Avg.
SimCSE-BERT$_{base}$♣	68.40	82.41	74.38	80.91	78.56	76.85	72.23	76.25
+ GS-InfoNCE	**70.12**	**82.57**	**75.21**	**82.89**	**80.23**	**79.70**	**72.70**	**77.63**
SimCSE-BERT$_{large}$♣	70.88	84.16	76.43	**84.50**	79.76	79.26	**73.88**	78.41
+ GS-InfoNCE	**73.75**	**85.09**	**77.35**	84.44	**79.88**	**79.94**	73.48	**78.96**
SimCSE-RoBERTa$_{base}$♣	70.16	81.77	73.24	81.36	80.65	80.22	68.56	76.57
+ GS-InfoNCE	**71.12**	**83.24**	**75.00**	**82.61**	**81.36**	**81.26**	**69.62**	**77.74**
SimCSE-RoBERTa$_{large}$♣	**72.86**	83.99	75.62	**84.77**	**81.80**	81.98	71.26	78.90
+ GS-InfoNCE	71.76	**84.91**	**76.79**	84.35	81.74	**82.97**	**71.71**	**79.21**

Table 2: Sentence embedding performance on semantic textual similarity (STS) test sets in terms of Spearman's correlation. ♣ : results from the official published model by the unsup-SimCSE.

best-performing model and only vary the hyperparameter M. For each M, we train the model until convergence and then select the checkpoint that performs the best on the validation set to evaluate on the test set. The performance statistics are listed in Table 3. As M becomes larger, the performance of GS-InfoNCE on the test set slowly improves. When $M = 3$, the best performance is reached, after which the model performance begins to decline. In general, GS-InfoNCE is not sensitive to M (recommend < 8), making it feasible to apply easily in practical applications.

5 Related Work

Deep and wide models are prone to overfitting, and thus regularization strategies are important to improve their generalization ability. Among them, smoothing is a very commonly used method. (Szegedy et al., 2016; Müller et al., 2019) propose to use label smoothing as a regularization method that makes the clusters between categories more compact and avoids adversarial examples with over high confidence. Text smoothing(Wu et al., 2022; Zhu et al., 2019) also seems to be able to bring further improvements in tasks such as text classification and machine translation by smoothing the one-hot representation of the input text into the probability distribution representation of the dictionary. Our GS-InfoNCE can also be regarded as a smoothing strategy that makes the distribution of negative samples smoother by introducing multiple random Gaussian noise vectors as an extension of the negative examples. Compared with label smoothing and text smoothing, GS-InfoNCE directly uses the standard Gaussian distribution for sampling, largely saving computational costs.

bs=64	**0**×	**0.5**×	**1**×	**2**×
BERT$_{base}$	76.25	76.96	76.90	77.11
bs=64	**3**×	**4**×	**8**×	**16**×
BERT$_{base}$	**77.63**	76.81	76.94	75.57

Table 3: Effect of the hyperprarameter M on BERT$_{base}$. We set M as a multiple of batch size (bs=64). **0**× means the original SimCSE without using GS-InfoNCE.

6 Conclusion and Future Work

This paper proposes GS-InfoNCE for unsupervised SimCSE methods by introducing a simple smoothing strategy upon the InfoNCE loss function to bring sufficient comparison for samples without increasing the batch size. In the future, we will explore how to improve the generalization capability of GS-InfoNCE and verify its effectiveness on more contrastive learning methods.

References

Eneko Agirre, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Inigo Lopez-Gazpio, Montse Maritxalar, Rada Mihalcea, et al. 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.

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.

Eneko Agirre, Carmen Banea, Daniel Cer, Mona Diab, Aitor Gonzalez Agirre, Rada Mihalcea, German Rigau Claramunt, and Janyce Wiebe. 2016. Semeval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation. In SemEval-2016. 10th International Workshop on Semantic Evaluation; 2016 Jun 16-17; San Diego, CA. Stroudsburg (PA): ACL; 2016. p. 497-511. ACL (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.

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.

Daniel Cer, Mona Diab, Eneko Agirre, Inigo Lopez-Gazpio, and Lucia Specia. 2017. Semeval-2017 task 1: Semantic textual similarity-multilingual and cross-lingual focused evaluation. arXiv preprint arXiv:1708.00055.

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, pages 1597–1607. PMLR.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. arXiv preprint arXiv:2104.08821.

Raia Hadsell, Sumit Chopra, and Yann LeCun. 2006. Dimensionality reduction by learning an invariant mapping. In 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), volume 2, pages 1735–1742. IEEE.

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. arXiv preprint arXiv:1803.02893.

Rafael Müller, Simon Kornblith, and Geoffrey Hinton. 2019. When does label smoothing help? arXiv preprint arXiv:1906.02629.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. arXiv preprint arXiv:1908.10084.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2818–2826.

Xing Wu, Chaochen Gao, Meng Lin, Liangjun Zang, Zhongyuan Wang, and Songlin Hu. 2022. Text smoothing: Enhance various data augmentation methods on text classification tasks. arXiv preprint arXiv:2202.13840.

Kun Zhou, Beichen Zhang, Wayne Xin Zhao, and Ji-Rong Wen. 2022. Debiased contrastive learning of unsupervised sentence representations. arXiv preprint arXiv:2205.00656.

Jinhua Zhu, Fei Gao, Lijun Wu, Yingce Xia, Tao Qin, Wengang Zhou, Xueqi Cheng, and Tie-Yan Liu. 2019. Soft contextual data augmentation for neural machine translation. arXiv preprint arXiv:1905.10523.