*re*CSE: Portable Reshaping Features for Sentence Embedding in Self-supervised Contrastive Learning

Anonymous ACL submission

Abstract

We propose *re*CSE, a self-supervised contrastive learning sentence representation framework based on feature reshaping. This framework is different from the current advanced models that use discrete data augmentation methods, but instead reshapes the input features of the original sentence, aggregates the global information of each token in the sentence, and alleviates the common problems of representation polarity and GPU memory consumption linear increase in current advanced models. In addition, our *re*CSE has achieved competitive performance in semantic similarity tasks. And the experiment proves that our proposed feature reshaping method has strong universality, which can be transplanted to other self supervised contrastive learning frameworks and enhance their representation ability, even achieving state-of-the-art performance. [1]

1 Introduction

Self-supervised sentence representation tasks (Le-Khac et al., 2020), which involve obtaining vector embeddings with rich semantic information from raw text in a self-supervised manner and can adapt to various downstream tasks without fine-tuning, have gained renewed attention due to the rise of contrastive learning (Chopra et al., 2005; Hadsell et al., 2006; Oord et al., 2018). Previous studies have directly employed pre-trained language models (PLM), such as BERT (Devlin et al., 2018) and RoBERTa (Liu et al., 2019), to derive high-quality sentence representations which still perform poorly in specific downstream tasks (e.g. semantic similarity task (Reimers and Gurevych, 2019)) without fine-tuning. Consequently, contrastive learning facilitates the emergence of more advanced methods (Gao et al., 2021; Yan et al., 2021) which subsequently assume a dominant position in the domain of sentence representation tasks.

The concept of contrastive learning suggests that the crux of self-supervised sentence representations learning hinges on the acquisition of suitable positive and negative samples from unlabeled data. Yan et al. (2021) employs various surface-level data augmentation techniques to derive positive samples from the original sentences. In contrast, Gao et al. (2021) adopts a more sophisticated approach that implicitly treats dropout (Hinton et al., 2012) as the baseline method for data augmentation. Specifically, Gao et al. (2021) feeds each sentence from a batch into a pre-trained BERT or RoBERTa model twice, applying independently sampled dropout masks for each pass. Consequently, Gao et al. (2021) considers the two distinct embeddings derived from the same original sentence as "positive sample pairs". Meanwhile, sentences from the same mini-batch that are not part of these pairs are categorized as "negative samples".

Further researchers (Wang and Dou, 2023; Wu et al., 2021; Shi et al., 2023; Chuang et al., 2022; Xu et al., 2024; Zhao et al., 2024; Zhuo et al., 2023) have identified several drawbacks associated with the generation of positive samples based on the same original sentence and dropout. These insights have spurred the development of numerous advanced works that address these issues, resulting in significant performance improvements. Although these works target different problems, they share a *discrete augmentation* method: the introduction of supplementary samples tailored to the specific challenges they aim to overcome. Although the approach of incorporating supplementary samples has yielded impressive and intuitive performance in downstream tasks, it concurrently introduces novel challenges: (1) The incorporation of supplementary samples is anticipated to augment the **polarity** of the sentence representation model. (2) Introducing supplementary samples is anticipated to escalate the **GPU memory requirements** for model training. And we conduct experiments to substantiate

[1] Our code is available at https://github.com/Anonymous/Anonymous.git

the aforementioned argument in Section 3.

To address the aforementioned challenges, we consider the following issues:

- Is there a strategy that can enhance the understanding of the overall semantics of sentences by contrastive learning frameworks, thereby enhancing their capacity for sentence representation without the incorporation of supplementary samples?

- How to design the contrastive learning framework that enhances sentence representation without resulting in a linear increase in GPU memory.

To sum up, we introduce *re*CSE, a novel contrastive learning framework for sentence embedding that eschews the introduction of supplementary samples in favor of portable feature **re**shaping. Specifically, a single word is insufficient to encapsulate the full semantic content of a sentence. Merely tokenizing and encoding a sentence does not effectively capture its global information. Consequently, our proposed *re*CSE approach reconstructs the tokenized features and integrates the comprehensive information of the sentence into the features. We integrate this process with a contrastive learning loss to fortify the original contrastive learning framework, thereby enhancing its capacity for sentence representation. In addition, to mitigate GPU memory consumption, we design the feature reshaping process as an independent module, or "pendant", of the original contrastive learning framework. This design operates independently, without necessitating additional inputs to the models embedded within the original framework. Through this "time for space" methodology, we effectively compress the GPU memory consumption throughout the training process.

In short, we make the following contributions:

- We propose *re*CSE, a contrastive learning framework that enhances sentence representation capabilities through feature reshaping alone, eliminating the need for supplementary samples.

- We innovatively decouple the feature reshaping process from the contrastive learning framework, enabling the feature reshaping and embedding models to operate independently, which significantly reduces GPU memory consumption.

- We are surprised to discover that feature reshaping exhibits portability, and our discovery corroborate through experimental validation on alternative sentence representation models.

2 Related Work and Background

2.1 Sentence Representation Learning

As a foundational task in the domain of natural language processing (NLP), sentence representation learning has garnered sustained interest over time. Wu et al. (2010) and Tsai (2012) employ a bag-of-word model to represent sentences, whereas Kiros et al. (2015) and Hill et al. (2016) categorize the task directly as a context prediction challenge. Recently, the proliferation of pre-trained language models (Devlin et al., 2018; Liu et al., 2019; Brown et al., 2020) has led many researchers to opt for utilizing models such as BERT (Devlin et al., 2018) and RoBERTa (Liu et al., 2019) to derive sentence representations. Although sentence representations generated by these pre-trained models are theoretically versatile for adaptation to any downstream task, achieving competitive performance without subsequent fine-tuning remains challenging. Furthermore, some studies (Ethayarajh, 2019; Yan et al., 2021) have identified that employing the [CLS] token directly as the sentence representation or utilizing average pooling in the last layer may result in **anisotropy**, a phenomenon where the learned embeddings converge into a limited region. Subsequently, methods such as BERT-Flow (Li et al., 2020) and BERT-Whitening (Su et al., 2021) have been proposed and have effectively mitigated this issue. Most recently, contrastive learning (Gao et al., 2021) has emerged as the predominant approach for sentence representation tasks.

2.2 Contrastive Learning for NLP

Contrastive learning (He et al., 2020; Chen et al., 2020) has garnered significant success in natural language processing by minimizing the distance between positive samples and maximizing the separation between negative samples (Gao et al., 2021; Wu et al., 2021; Yan et al., 2021; Kim et al., 2021). Recently, the concepts of **alignment** and **uniformity** (Wang and Isola, 2020) have been introduced as metrics for assessing the quality of representations derived from contrastive learning. Alignment evaluates the proximity of positive sample pairs, whereas uniformity assesses the impact of anisotropy on the spatial distribution of embed-

dings.

Based on the aforementioned research, SimCSE (Gao et al., 2021) is introduced by researchers as a seminal contribution to the field. It leverages the inherent randomness of the dropout noise to enrich the latent space of semantically aligned sentences, thereby generating diverse sentence representations that constitute positive pairs. For an exhaustive treatment of this topic, Please refer to section 2.3. Subsequent researches have further enhanced the quality of sentence representations based on SimCSE. ESimCSE (Wu et al., 2021) mitigates model bias induced by sentence length by incorporating supplementary samples through word repetition. Concurrently, SNCSE (Wang and Dou, 2023) bolsters the capability of sentence representation by introducing the negation of original sentences as supplementary negative samples. OssCSE (Shi et al., 2023) directly introduces two different supplementary samples to counteract the bias stemming from the uniformity of surface structures, **etc**. These methods uniformly leverage the introduction of supplementary samples to bolster sentence representation capability. While the impact is considerable, this strategy may escalate computational demands and introduce polarity within sentence representations. On the other hand, PromptBERT (Jiang et al., 2022) enhances the quality of sentence embeddings produced by BERT (Devlin et al., 2018) in SimCSE framework, employing a prompt-based method (Zhou et al., 2022). DCLR (Zhou et al., 2022) concentrates on refining the capacity for negative sample selection. ArcCSE (Zhang et al., 2022b) directly targets the optimization of the objective function. Nonetheless, these methodologies are predominantly contingent upon the inherent quality of the training data, exhibit challenges in portability, and demonstrate only modest enhancements in the capability of sentence representation.

2.3 Background: SimCSE

This section provides a detailed introduction to the foundational framework employed in our study: SimCSE (Gao et al., 2021).

In the context of two semantically similar sentences, we define these as a sentence pair, denoted as $\{x_i, x_i^+\}$, and consider this pair to constitute a positive sample. The central tenet of SimCSE involves utilizing identical sentence to forge positive samples pair, i.e., $x_i = x_i^+$. Note that there is a dropout mask placed on the fully-connected layers and the attention probabilities in Transformer.

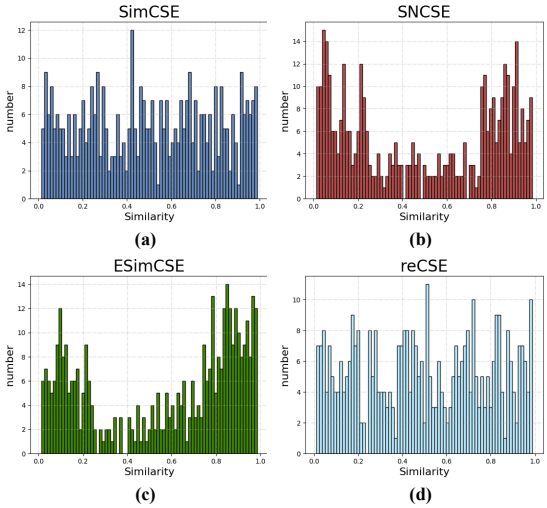

Figure 1: The distribution of representation polarity test results. The distribution of the framework (b, c) based on discrete data augmentation shows polarity (concavity), and the distribution of the basic SimCSE and our *re*CSE (a, d) is relatively uniform.

Consequently, the essence of constructing positive samples in SimCSE lies in encoding the same sentence x_i twice with distinct dropout masks z_i and z_i^+, thereby yielding two distinct embeddings that form a positive sample pair:

$$\mathbf{h}_i = f_\theta\left(x_i, z_i\right), \mathbf{h}_i^+ = f_\theta\left(x_i, z_i^+\right) \quad (1)$$

For each sentence within a mini-batch of size N, the contrastive learning objective with respect to x_i, given the embeddings h_i and h_i^+, is formulated as follows:

$$\ell_i = -\log \frac{e^{\mathrm{sim}\left(\mathbf{h}_i, \mathbf{h}_i^+\right)/\tau}}{\sum_{j=1}^{N} e^{\mathrm{sim}\left(\mathbf{h}_i, \mathbf{h}_j^+\right)/\tau}} \quad (2)$$

where τ is a temperature hyperparameter and $\mathrm{sim}\left(\mathbf{h}_i, \mathbf{h}_i^+\right)$ is the similarity metric, which is typically the cosine similarity function[2].

3 Probing for Motivations

In this section, we experimentally probe into the challenges associated with the current discrete enhancement-based sentence representation methods mentioned in Section 1.

3.1 Polarity by Incorporating Supplementary Samples

In essence, We believe that, from the perspective of semantic similarity, models augmented with supplementary samples are more likely to assign higher or

[2] $\mathrm{sim}\left(\mathbf{h}_i, \mathbf{h}_i^+\right) = \frac{\mathbf{h}_i^\top \mathbf{h}_i^+}{\|\mathbf{h}_i\| \cdot \|\mathbf{h}_i^+\|}$

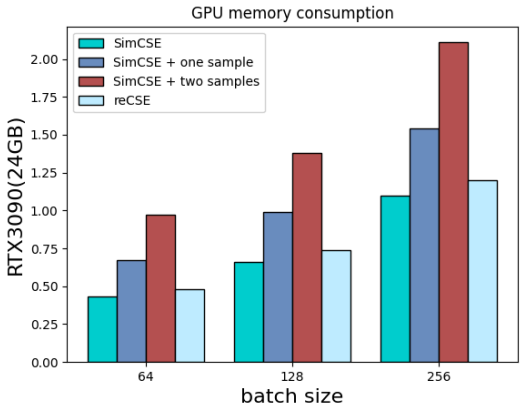

Figure 2: The impact of discrete data augmentation on GPU memory consumption and the test results of *re*CSE on GPU memory consumption. The y-axis scale is measured in RTX3090 (24GB) units. As more types of additional samples are introduced, the GPU memory consumption for training also increases linearly.

lower scores to sentences based on their similarity rather than "appropriate" ones. To verify this issue, we define a set of scores for $[0, 1, 2, 3]$, where 0 indicates that two sentences are completely dissimilar, while 3 signifies that they are highly similar. Then we employed GPT-4 (Achiam et al., 2023) to generate 100 sentence pairs for each score, resulting in a total of 400 sentence pairs. Subsequently, we utilized SimCSE (Gao et al., 2021), SNCSE (Wang and Dou, 2023) and ESimCSE (Wu et al., 2021), where SNCSE and ESimCSE incorporates additional samples based on SimCSE, to evaluate these pairs. The final statistical results are depicted in Figure 1 (a,b,c), which shows that after introducing supplementary samples, the predicted results have obvious polarity.

3.2 Linear Increase in GPU Consumption

Note that introducing supplementary samples is anticipated to escalate the GPU memory requirements for model training. Specifically, this augmentation increases the number of sentences processed in each training batch, consequently amplifying the GPU memory overhead. Notably, the diversity of supplementary samples is positively correlated with the extend of the increased memory overhead. We further conduct a preliminary experiment to substantiate our hypothesis. Employing SimCSE as the foundational model, we simulate the incorporation of supplementary samples through the application of additional dropout layers to the input sentences. We monitor the GPU memory consumption throughout the model training process. The results of this experiment are delineated in Figure 2.

4 Method

The *re*CSE framework we proposed is shown in Figure 3, which can be divided into three components: (1) the feature reshaping, (2) the prompt-based original data augmentation, (3) and the contrastive learning mechanism that integrates these reshaped features. Next we will provide a detailed introduction to each component.

4.1 Feature Reshaping

The feature reshaping part is the main innovation of our proposed *re*CSE. This part aims to reshaping the input features without generating supplementary samples, thereby enhance the focus on global information contained within the sentences, as shown in Figure 4. The ultimate objective is to integrate these reshaped features into contrastive learning framework.

Next, we offer an exhaustive exposition on our feature reshaping:

Given a sentence of n tokens $s = \{s^1, s^2, \ldots, s^n\}$, we initially employ a tokenizer to derive the original feature $x = \{x^1, x^2, \ldots, x^n\}$:

$$\{x^i\}_{i=1}^n = \text{Tokenizer}(\{s^i\}_{i=1}^n\}) \tag{3}$$

Note that the original feature x can be conceptualized as an $1 \times n$ matrix, where each element represents information pertaining to the respective token. Based on this conceptualization, we enhance the feature by augmenting its dimensionality and densifying it, thereby transforming x into an $n \times n$ matrix X:

$$X = \sqrt{x^{\mathrm{T}} \cdot x} \tag{4}$$

The matrix X derived from Eq. 4 can be partitioned into two components:

$$X = \text{diag}(x^1, x^2, \cdots, x^n) + \hat{X} \tag{5}$$

where the first part is a diagonal matrix, with its diagonal elements corresponding to those of the original features of corresponding tokens $\{x^i\}_{i=1}^n$, and the second part is a symmetric matrix $\hat{X}$ with diagonal elements of 0, with its off-diagonal elements representing the correlations features between the original features of distinct tokens $(x^i, x^j)_{i \neq j}^n$. In essence, the i-th column of matrix X comprises

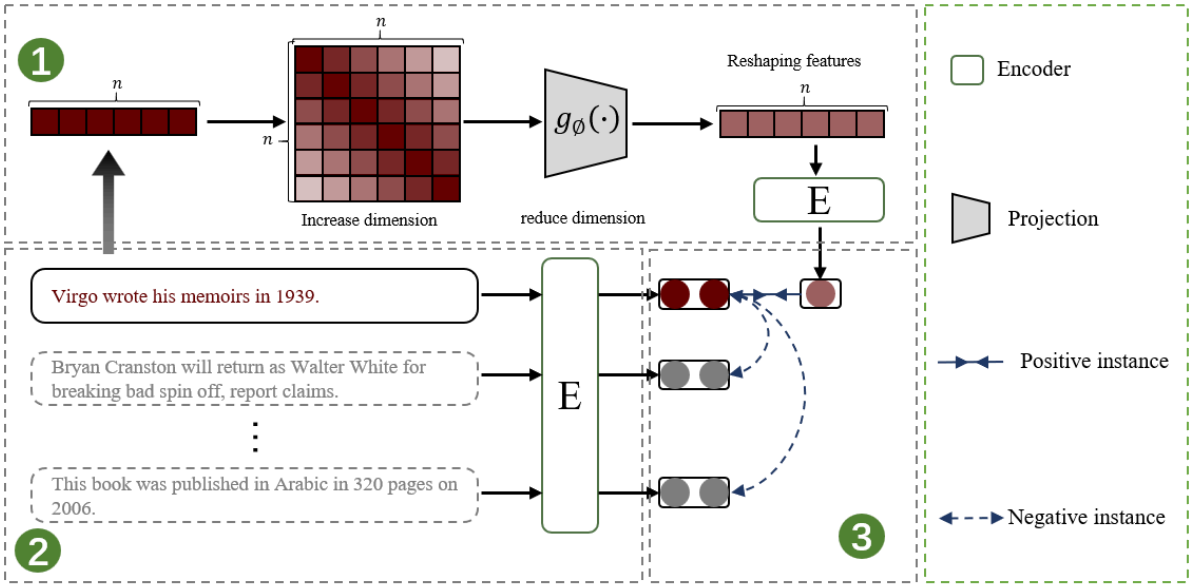

Figure 3: The main framework of *re*CSE. We adopt a modular design to reduce GPU memory consumption

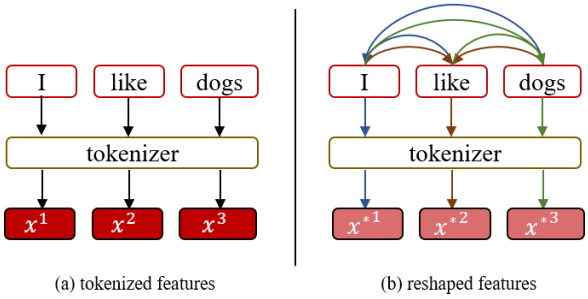

Figure 4: The original input features are based solely on a single token (a), while the reshaped features contain the global information of each token in the sentence (b).

the features of token x^i as well as the correlation features with respect to other tokens.

Ultimately, we need to compress matrix X back to its original dimensions, thereby extracting the reshaped features, which are input into the encoder:

$$x^* = g_\phi(X) \tag{6}$$

where g represents a linear projection operation applied per column of the matrix, designed to aggregate the $n \times n$ matrix X along its columns via a linear transformation, thereby reduce it to original $1 \times n$ dimension.

4.2 Prompt-based Data Augmentation

In the processing of the original sentence, We first use different prompts (Brown et al., 2020) to enhance sentence representation:

$$\begin{aligned} &\textit{The sentence : " } s \textit{ " mean } [MASK]. \\ &\textit{The sentence of " } s^+ \textit{ " means } [MASK]. \end{aligned} \tag{7}$$

We adhere to the SimCSE methodology (Gao et al., 2021) and employ dropout as the minimum data augmentation unit. For a set of sentence features $\{x_i\}_{i=1}^N$, we construct positive sample pair by encoding each input sentence feature x_i twice with different dropout masks:

$$\boldsymbol{h}^z = f_\theta(x_i, z), \boldsymbol{h}^{z'} = f_\theta(x_i, z') \tag{8}$$

where z and z' denote different dropout masks, $f_\theta(\cdot)$ is a pre-trained language encoder such as BERT and RoBERTa. To sum up, distinct embeddings $\boldsymbol{h}^z$ and $\boldsymbol{h}^{z'}$ based on identical input, constitute the positive sample pair we need. Because of the adoption of prompts, we ultimately embed the hidden state of the special [MASK] token $h_{[MASK]}$ as the input sample. We added an additional MLP layer with tanh activation function on $h_{[MASK]}$ to obtain h:

$$\begin{aligned} \boldsymbol{h}^z &= Tanh(MLP(\boldsymbol{h}^z_{[MASK]})) \\ \boldsymbol{h}^{z'} &= Tanh(MLP(\boldsymbol{h}^{z'}_{[MASK]})) \end{aligned} \tag{9}$$

Additionally, other inputs in the same mini-batch are categorized as negative samples.

4.3 Contrastive Learning with Reshaped Features

We employ the infoNCE loss (He et al., 2020) as the training objective for our proposed *re*CSE, which embodies the concept of contrastive learning through a straightforward cross-entropy loss. Furthermore, for a collection of input sentences $\{s_i\}_{i=1}^{N}$, the procedures delineated in the preceding sections facilitate the acquisition of three distinct embedding representations: the original sentence embedding h_i^z, the positive sample embedding $h_i^{z'}$, and the reshaped embedding h_i^*.

For h_i^z and $h_i^{z'}$, our training objectives are as follows:

$$\ell_{\mathrm{CL}} = -\log \frac{e^{\cos_\mathrm{sim}\left(h_i^z, \boldsymbol{h}_i^{z'}\right)/\tau}}{\sum_{j=1}^{N} e^{\cos_\mathrm{sim}\left(\boldsymbol{h}_i^z, h_j^{z'}\right)/\tau}} \tag{10}$$

where N is the size of the mini-batch, τ is a temperature parameter.

For the reshaped embedding representation h_i^*, we consider it as an additional positive sample and endeavor to approximate it to h_i^z and $h_i^{z'}$ as closely as possible. The specific equation is as follows:

$$\ell_{re} = -\sum_{Z \in \{z, z'\}} \log \frac{e^{\cos_\mathrm{sim}\left(h_i^Z, h_i^*\right)/\tau'}}{\sum_{j=1}^{N} e^{\cos_\mathrm{sim}\left(h_i^Z, h_j^*\right)/\tau'}} \tag{11}$$

where τ' is a temperature parameter. Let λ denote the trade-off hyperparameter that balances the two objectives, we formulate the final loss as:

$$\ell = \lambda \ell_{\mathrm{CL}} + (1-\lambda)\max(\ell_{\mathrm{CL}}, \ell_{re}) \tag{12}$$

We don't mandate ℓ_{re} as an obligatory training objective, instead, we prioritize its optimization during the initial stages of training. Specifically, when the ℓ_{re} value is minimal, our optimization strategy focus solely on ℓ_{CL}.

5 Experiments

5.1 Setup

Following Gao et al. (2021), we randomly select 1,000,000 sentences from English Wikipedia to form our input sentence corpus. We conduct experiments utilizing BERT (Devlin et al., 2018) and RoBERTa (Liu et al., 2019) as encoders, respectively. During evaluation phase, we employ semantic similarity tasks (STS) to evaluate the sentence representation capability of the proposed *re*CSE framework. The STS task is designed to measure the semantic similarity between sentences and represents one of the most widely utilized benchmark datasets for assessing self-supervised sentence embeddings. The task encompasses seven sub-tasks, specifically STS-12 to STS-16, STS-B, and SICK-R (Agirre et al., 2012, 2013, 2014, 2015, 2016; Cer et al., 2017; Marelli et al., 2014).

In detail, our *re*CSE is implemented through Torch 1.7.1 and huggingface transformers (Wolf et al., 2020). We conduct experiments using batch sizes of 128 and smaller on computing nodes equipped with Nvidia GTX 3090 GPUs, while other experiments are performed on a single A100 GPU. Additionally, for all experiments, we set the dropout rate to 0.1. We train the model for a total of 3 epochs, evaluate every 250 steps, and select the model parameters that demonstrated the highest performance.

5.2 Main Results

Our main experiment results are presented in Table 1. It is observable that, with the exception of OssCSE (Shi et al., 2023), which is not open source, our proposed *re*CSE exhibits a marginally lower average performance compared to state-of-the-art models, such as SNCSE (Wang and Dou, 2023), yet it retains strong competitiveness. However, it is important to note that the models currently exhibiting the highest average performance, such as SNCSE and OssCSE, employ discrete data augmentation techniques. These techniques introduce additional samples to bolster the sentence representation capabilities. And a distinct polarity in the ability to represent sentences is evident. Upon excluding these models that utilize discrete augmentation (marked with $*$ in Table 1) our proposed *re*CSE demonstrates the most superior performance and there is almost no introduction of polarity in representational ability as well (We will prove this in the next section). Furthermore, our method, which does not introduce any additional samples, is theoretically universal and can be integrated with any form of discrete enhancement, which is the "portability" mentioned in this work. We have also substantiated this claim through experiments detailed in next section.

Specifically, our proposed *re*CSE has demonstrated significant improvements over the baseline model SimCSE in all subtasks. When employing BERT as the encoder, our method realizes the maximum enhancement on the SICK-R benchmark, outperforming SimCSE by 3.68. Even though the

Semantic Textual Similarity (STS) Benchmark								
Model-BERT$_{base}$	**STS12**	**STS13**	**STS14**	**STS15**	**STS16**	**STS-B**	**SICK-R**	**Avg.**
SimCSE (Gao et al., 2021)	68.69	82.05	72.91	81.15	79.39	77.93	70.93	76.15
MoCoSE (Cao et al., 2022)	71.58	81.40	74.47	83.45	78.99	78.68	72.44	77.27
InforMin-CL (Chen et al., 2022)	70.22	83.48	75.51	81.72	79.88	79.27	71.03	77.30
MixCSE (Zhang et al., 2022a)	71.71	83.14	75.49	83.64	79.00	78.48	72.19	77.66
DCLR* (Zhou et al., 2022)	70.81	83.73	75.11	82.56	78.44	78.31	71.59	77.22
ArcCSE* (Zhang et al., 2022b)	72.08	84.27	76.25	82.32	79.54	79.92	72.39	78.11
PCL* (Wu et al., 2022)	72.74	83.36	76.05	83.07	79.26	79.72	72.75	78.14
ESimCSE* (Wu et al., 2021)	**73.40**	83.27	77.25	82.66	78.81	80.17	72.30	78.27
DiffCSE* (Chuang et al., 2022)	72.28	84.43	76.47	83.90	80.54	80.59	71.29	78.49
miCSE (Klein and Nabi, 2022)	71.71	83.09	75.46	83.13	80.22	79.70	73.62	78.13
SNCSE* (Wang and Dou, 2023)	70.67	**84.68**	76.99	83.69	80.51	81.35	74.77	78.97
OssCSE†,* (Shi et al., 2023)	71.78	84.40	**77.71**	**83.95**	79.92	80.57	**75.25**	**79.08**
(ours) *re*CSE	72.03	84.61	75.46	83.72	80.45	80.71	74.61	78.68
Model-RoBERTa$_{base}$	**STS12**	**STS13**	**STS14**	**STS15**	**STS16**	**STS-B**	**SICK-R**	**Avg.**
SimCSE (Gao et al., 2021)	70.16	81.77	73.24	81.36	80.65	80.22	68.56	76.57
DCLR* (Zhou et al., 2022)	70.01	83.08	75.09	83.66	81.06	81.86	70.33	77.87
PCL* (Wu et al., 2022)	71.13	82.38	75.40	83.07	81.98	81.63	69.72	77.90
ESimCSE* (Wu et al., 2021)	69.90	82.50	74.68	83.19	80.30	80.99	70.54	77.44
DiffCSE* (Chuang et al., 2022)	70.05	83.43	75.49	82.81	82.12	82.38	71.19	78.21
SNCSE* (Wang and Dou, 2023)	70.62	84.42	77.24	84.85	81.49	83.07	72.92	79.23
OssCSE†,* (Shi et al., 2023)	72.28	**85.27**	**79.51**	84.77	**82.32**	**83.55**	**75.54**	**80.46**
(ours) *re*CSE	73.72	84.11	76.47	83.97	81.42	83.14	72.52	79.19

Table 1: The results of semantic similarity task test. $*$ denotes the use of *discrete augmentation*, and † means this framework isn't open-source.

STS-16 benchmark shows the least improvement, it still reflects a 1.06 gain. Utilizing RoBERTa as the encoder, our method garners a 3.96 improvement on the SICK-R benchmark and a 0.77 improvement on the STS-16 benchmark. These results substantiate the effectiveness of our approach.

6 Further Analysis

6.1 Sentence Representation Polarity Analysis

In the preceding introduction, we identify that numerous sentence representation models exhibit polarity. Specifically, from the perspective of semantic similarity, these models consistently assign either higher or lower scores to a sentence pair. We posit that the underlying cause of this polarity is attributable to the presence of discrete data augmentation methods. We have also demonstrated this in the preceding introduction. Furthermore, since our *re*CSE framework does not employ discrete data augmentation, it is theoretically devoid of the introduction of polarity.

To substantiate our claim, we conduct additional test using our *re*CSE on 400 sentences generated by GPT-4 (mentioned in the Section 3). The result is depicted in Figure 1 (d), which clearly demonstrate that our proposed *re*CSE framework does not introduce polarity, thereby highlighting the superiority of our approach.

6.2 Reducing GPU Memory Consumption Analysis

In our analysis of SimCSE, we identify that the encoder based on pre-trained language models, such as BERT or RoBERTa, is the primary contributor to GPU memory consumption during training. Specifically, when the batch size is set to N, each encoder input comprises $2 \times N$ sentences. Advanced works, such as SNCSE, OssCSE, etc., often employ discrete data augmentation to introduce supplementary samples, which increase the encoder's input during training to $3 \times N$ even $4 \times N$ sentences or higher. This approach increases the consumption of GPU memory and loses generality, which is not conducive to the continued research of subsequent researchers.

Based on the above considerations, we decompose the *re*CSE framework into three sequential steps, as depicted in Figure 3. To curtail GPU memory consumption during training, we took inspiration from the CoCo dataset's processing methodology (Lin et al., 2014), opting to isolate the initial step entirely. Initially, we reshaped and stored the input sentence features. Subsequently, we augmented the original sentences with prompt and

Semantic Textual Similarity (STS) Benchmark								
Model-BERT$_{base}$	**STS12**	**STS13**	**STS14**	**STS15**	**STS16**	**STS-B**	**SICK-R**	**Avg.**
ESimCSE (Wu et al., 2021)	73.40	83.27	77.25	82.66	78.81	80.17	72.30	78.27
ESimCSE + *Feature reshaping*	**73.67**	**84.09**	**78.28**	**83.76**	**78.89**	**81.25**	**73.34**	**79.03**
SNCSE (Wang and Dou, 2023)	**70.67**	84.68	76.99	83.69	**80.51**	81.35	**74.77**	78.97
SNCSE + *Feature reshaping*	70.54	**84.99**	**77.84**	**84.71**	80.49	**81.87**	73.64	**79.26**
Model-RoBERTa$_{base}$	**STS12**	**STS13**	**STS14**	**STS15**	**STS16**	**STS-B**	**SICK-R**	**Avg.**
ESimCSE (Wu et al., 2021)	69.90	82.50	74.68	83.19	80.30	**80.99**	70.54	77.44
ESimCSE + *Feature reshaping*	**70.11**	**82.92**	**75.77**	**84.07**	**80.71**	80.98	**71.34**	**78.37**
SNCSE (Wang and Dou, 2023)	70.62	84.42	77.24	84.85	**81.49**	83.07	72.92	79.23
SNCSE + *Feature reshaping*	**70.63**	**85.04**	**78.25**	**84.86**	81.25	**83.98**	**73.07**	**79.88**

Table 2: Portability test results based on semantic similarity task.

dropout. Ultimately, the third step involved extracting and training the stored reshaped representations. Since the two encoders within the *re*CSE framework do not operate concurrently, an increase in the required training time is inevitable. Nevertheless, the GPU memory consumption will not surpass the maximum cost in first two steps. Theoretically this design has led to significant decrease in GPU memory consumption compared to prior studies and provide higher reference for subsequent researches.

To substantiate our argument, we execute a comparative GPU memory consumption test on the proposed *re*CSE with batch size $[64, 128, 256]$ as in the Introduction section, and the final result is shown in Figure 2. It can be seen that our *re*CSE has almost the same memory consumption as SimCSE (Gao et al., 2021), which demonstrates the effectiveness of our method in reducing GPU memory consumption.

6.3 Portability analysis

Through further analysis, we determine that our proposed feature reshaping method is universally applicable and is compatible with commonly used discrete data augmentation. To validate this hypothesis, we transplant our feature reshaping method into several state-of-the-art contrastive learning models that are currently available as open-source, including ESimCSE (Wu et al., 2021) and SNCSE (Wang and Dou, 2023). Subsequently,and conduct semantic similarity testing. The test results are shown in Table 2. It is evident that both ESimCSE and SNCSE exhibit significant performance enhancements following the introduction of the proposed feature reshaping. Notably, when employing BERT as the encoder, SNCSE's performance even surpasses that of the state-of-the-art model (OssCSE (Shi et al., 2023), 79.08) despite not being open source.

7 Conclusion

In this paper, we investigate the challenges associated with current advanced self-supervised contrastive learning frameworks for sentence representation. Specifically, we introduce two key issues: the polarity in representational capacity due to discrete augmentation and the linear escalation of GPU memory consumption during training as a result of incorporating additional samples. To address these challenges, we propose a novel feature reshaping and introduce *re*CSE, a self-supervised contrastive learning framework for sentence representation based on feature reshaping. Experimental results demonstrate that our proposed *re*CSE framework achieves competitive performance in semantic similarity tasks without a corresponding increase in GPU memory consumption. Additionally, we have showcased the portability of our feature reshaping method across other self-supervised contrastive learning frameworks. In general, we anticipate that our research will facilitate future researchers' recognition of the limitations inherent in discrete data augmentation methods, thereby inspiring the development of more universally applicable and efficacious approaches.

8 Limitations

Our work employs a "modular design" to mitigate GPU memory consumption. However, this approach may results in an extended training duration. Furthermore, the projection algorithm in the feature reshaping is currently rudimentary, indicat-

ing scope for further refinement and optimization.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

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 M Cer, Mona T Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Rada Mihalcea, German Rigau, and Janyce Wiebe. 2014. Semeval-2014 task 10: Multilingual semantic textual similarity. In *SemEval@ COLING*, 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.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

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 ACL*.

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*.

Shaobin Chen, Jie Zhou, Yuling Sun, and He Liang. 2022. An information minimization contrastive learning model for unsupervised sentence embeddings learning. In *COLING*.

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.

Sumit Chopra, Raia Hadsell, and Yann LeCun. 2005. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pages 539–546. IEEE.

Yung-Sung Chuang, Rumen Dangovski, Hongyin Luo, Yang Zhang, Shiyu Chang, Marin Soljačić, Shang-Wen Li, Wen-tau Yih, Yoon Kim, and James Glass. 2022. Diffcse: Difference-based contrastive learning for sentence embeddings. *arXiv preprint arXiv:2204.10298*.

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*.

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? comparing the geometry of bert, elmo, and gpt-2 embeddings. *arXiv preprint arXiv:1909.00512*.

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.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738.

Felix Hill, Kyunghyun Cho, and Anna Korhonen. 2016. Learning distributed representations of sentences from unlabelled data. *arXiv preprint arXiv:1602.03483*.

Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*.

Ting Jiang, Jian Jiao, Shaohan Huang, Zihan Zhang, Deqing Wang, Fuzhen Zhuang, Furu Wei, Haizhen Huang, Denvy Deng, and Qi Zhang. 2022. Prompt-bert: Improving bert sentence embeddings with prompts. *arXiv preprint arXiv:2201.04337*.

Taeuk Kim, Kang Min Yoo, and Sang-goo Lee. 2021. Self-guided contrastive learning for bert sentence representations. *arXiv preprint arXiv:2106.07345*.

Ryan Kiros, Yukun Zhu, Russ R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Skip-thought vectors. *Advances in neural information processing systems*, 28.

Tassilo Klein and Moin Nabi. 2022. micse: Mutual information contrastive learning for low-shot sentence embeddings. *arXiv preprint arXiv:2211.04928*.

Phuc H Le-Khac, Graham Healy, and Alan F Smeaton. 2020. Contrastive representation learning: A framework and review. *Ieee Access*, 8:193907–193934.

Bohan Li, Hao Zhou, Junxian He, Mingxuan Wang, Yiming Yang, and Lei Li. 2020. On the sentence embeddings from pre-trained language models. *arXiv preprint arXiv:2011.05864*.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer.

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*.

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.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

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

Zhan Shi, Guoyin Wang, Ke Bai, Jiwei Li, Xiang Li, Qingjun Cui, Belinda Zeng, Trishul Chilimbi, and Xiaodan Zhu. 2023. Osscse: Overcoming surface structure bias in contrastive learning for unsupervised sentence embedding. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7242–7254.

Jianlin Su, Jiarun Cao, Weijie Liu, and Yangyiwen Ou. 2021. Whitening sentence representations for better semantics and faster retrieval. *arXiv preprint arXiv:2103.15316*.

Chih-Fong Tsai. 2012. Bag-of-words representation in image annotation: a review. *International Scholarly Research Notices*, 2012(1):376804.

Hao Wang and Yong Dou. 2023. Sncse: Contrastive learning for unsupervised sentence embedding with soft negative samples. In *International Conference on Intelligent Computing*, pages 419–431. Springer.

Tongzhou Wang and Phillip Isola. 2020. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International conference on machine learning*, pages 9929–9939. PMLR.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Lei Wu, Steven CH Hoi, and Nenghai Yu. 2010. Semantics-preserving bag-of-words models and applications. *IEEE Transactions on Image Processing*, 19(7):1908–1920.

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

Xing Wu, Chaochen Gao, Liangjun Zang, Jizhong Han, Zhongyuan Wang, and Songlin Hu. 2021. Esimcse: Enhanced sample building method for contrastive learning of unsupervised sentence embedding. *arXiv preprint arXiv:2109.04380*.

Jiahao Xu, Charlie Soh Zhanyi, Liwen Xu, and Lihui Chen. 2024. Blendcse: Blend contrastive learnings for sentence embeddings with rich semantics and transferability. *Expert Systems with Applications*, 238:121909.

Yuanmeng Yan, Rumei Li, Sirui Wang, Fuzheng Zhang, Wei Wu, and Weiran Xu. 2021. Consert: A contrastive framework for self-supervised sentence representation transfer. *arXiv preprint arXiv:2105.11741*.

Yanzhao Zhang, Richong Zhang, Samuel Mensah, Xudong Liu, and Yongyi Mao. 2022a. Unsupervised sentence representation via contrastive learning with mixing negatives. *AAAI*.

Yuhao Zhang, Hongji Zhu, Yongliang Wang, Nan Xu, Xiaobo Li, and Binqiang Zhao. 2022b. 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.

Fufangchen Zhao, Guoqiang Jin, Rui Zhao, Jiangheng Huang, and Fei Tan. 2024. Simct: A simple consistency test protocol in llms development lifecycle. *arXiv preprint arXiv:2407.17150*.

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

Wenjie Zhuo, Yifan Sun, Xiaohan Wang, Linchao Zhu, and Yi Yang. 2023. Whitenedcse: Whitening-based contrastive learning of sentence embeddings. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12135–12148.