Can Contrastive Learning Refine Embeddings

Lihui Liu[1], Jinha Kim[2(✉)], and Vidit Bansal[2]

[1] University of Illinois at Urbana-Champaign, Champaign, IL 61820, USA
lihuil2@illinois.edu
[2] Amazon, Seattle, WA 98109, USA
{jinhak,bansalv}@amazon.com

Abstract. Recent advancements in contrastive learning have revolutionized self-supervised representation learning and achieved state-of-the-art performance on benchmark tasks. While most existing methods focus on applying contrastive learning on input data modalities like images, natural language sentences, or networks, they overlook the potential of utilizing output from previously trained encoders. In this paper, we introduce SIMSKIP, a novel contrastive learning framework that specifically refines the input embeddings for downstream tasks. Unlike traditional unsupervised learning approaches, SIMSKIP takes advantage of the output embedding of encoder models as its input. Through theoretical analysis, we provide evidence that applying SIMSKIP does not lead to larger upper bounds on downstream task errors than that of the original embedding which is SIMSKIP's input. Experiment results on various open datasets demonstrate that the embedding by SIMSKIP improves the performance on downstream tasks.

1 Introduction

Embedding symbolic data such as text, graphs, and multi-relational data has become a key approach in machine learning and AI [24]. The learned embeddings can be utilized in various applications. For instance, in NLP, word embeddings generated by WORD2VEC [23] or BERT [4] have been employed in tasks like question answering and machine translation. In the field of graph learning, embeddings of graphs like NODE2VEC [7] and DEEPWALK [28] have been used for node classification and link prediction in social networks. Similarly, in computer vision, image embeddings such as ResNet [9] can be used for image classification.

Despite the progress in representation learning, learning effective embeddings remains a challenging problem. Deep learning models often require a large amount of labeled training data, which can be costly and limit their applicability. Additionally, the learned embeddings often perform well on one task but not on others.

Contrastive learning has the advantage of being able to learn representations without label information, thus saving a significant amount of human effort and resources that would have been used for data labeling. The fundamental idea of contrastive learning is to bring together an anchor and a "positive" sample in the embedding space while pushing apart the anchor from many "negative" samples [3]. As there are no labels available,

L. Liu—Work conducted while the author was an intern at Amazon.

A. Meroño Peñuela et al. (Eds.): ESWC 2024, LNCS 14664, pp. 236–250, 2024.
https://doi.org/10.1007/978-3-031-60626-7_13

a positive pair often consists of data augmentations of the sample, and negative pairs are formed by the anchor and randomly chosen samples from the minibatch [3]. Although the concept is simple, recent research has shown that contrastive learning methods can achieve comparable results to supervised methods [13,29].

Given the success of contrastive learning, a logical question to ask is whether using the output of another embedding model as input to contrastive learning can further refine the embedding space and make it perform better for downstream tasks. This is the question we aim to answer in this paper. We propose a new approach, called SIMSKIP, that takes the output embedding of another model as input and applies contrastive learning on it. Our proposed method aims to fine-tune the input embedding space, making it more robust for downstream tasks. We theoretically prove that after applying SIMSKIP on the input embedding space, for a downstream task, the error upper bound of the new learned fine-tuned embedding will not be larger than that of the original embedding space. We conduct extensive experiments on various datasets and downstream tasks to evaluate the performance of our proposed approach and compare it with other state-of-the-art methods. The results show that the proposed SIMSKIP can refine the input embedding space and achieve better performance on downstream tasks.

In summary, the main contributions of this paper are:

- **Problem Definition.** To the best of our knowledge, we are the first to propose and investigate the use of contrastive learning to improve the robustness of embedding spaces.
- **Algorithm** We propose a skip-connection-based contrastive learning model, SIMSKIP, and theoretically prove that it can reduce the error upper bound of downstream tasks.
- **Empirical Evaluations**. We conduct extensive experiments on several real-world datasets and various downstream tasks. The results of our experiments demonstrate the effectiveness of SIMSKIP.

2 Preliminaries and Problem Definition

2.1 Contrastive Learning

Contrastive learning aims to learn effective representations by pulling semantically similar samples together and pushing dissimilar samples apart [6]. In self-supervised setting as such contrastive learning, constructing positive and negative pairs from unlabeld dataset through data augmentation is critical. For example, in visual representations, an effective approach is to generate two augmented images from one input image and use them as the positive pair, while other images in the same mini-batch are treated as negative pairs of the input image. There are several different data augmentation methods such as cropping, flipping, distortion, and rotation [3]. In node representations in graphs, one idea is to use the neighborhood of the given node as positive pairs, while nodes that are farther away are treated as negative pairs. For graph-level representations, operations such as node deletion and edge deletion can be used to generate positive augmentations of the input graph.

After building positive and negative pairs, neural network-based encoders are used to learn representation vectors from augmented data examples. Various network architectures such as ResNet [3] for images and BERT [6] for text can be used. The output representation vectors of the encoders are used as the final embedding of the input data. To learn an effective embedding space discriminating positive and negative pairs, a simpler neural network called projector is stacked on top of a encoder and the contrastive loss is applied against the projector output. A commonly used projector is an MLP with one or two layers, which is simple to implement.

In training, first, a random sample of N examples is taken for a mini-batch. Then, N pairs are constructed from N samples through data augmentation, which lead $2N$ examples total in the mini-batch. N augmented pair of an input data point are treated as the positive pair in the mini-batch. For each augmented positive pair (i, j), the remaining $2N-2$ example are used to construct negative examples (i, k). The commonly used contrastive learning loss is

$$l_{i,j} = -\log \frac{\exp(\text{sim}(z_i, z_j)/\tau)}{\sum_{k=1}^{2N} \mathbf{I}_{k \neq i,j} \exp(\text{sim}(z_i, z_k)/\tau)} \tag{1}$$

where τ is the temperature, sim is a similarity function such as the cosine similarity, $z_i (= p(f(x_i))$ is the output of the projector p which takes the output of the encoder f [3].

2.2 Problem Statement

In this paper, we focus on investigating whether contrastive learning can refine the embeddings for downstream tasks. Given an input dataset $D = \{d_i\}_1^N$, an arbitrary embedding function $h()$, and its output embedding $\mathcal{X}$, where $x_i \in \mathcal{X}$ is the embedding of data point d_i ($x_i = h(d_i)$), our goal is to design a new embedding function f such that $f(h(d_i))$ performs no worse than $h(d_i)$ given an arbitrary downstream task T.

3 Method

In the previous sections, we outlined the concept of unsupervised contrastive learning. In this section, we will delve into the specifics of using SIMSKIP that refines pre-existing embeddings.

3.1 Contrastive Learning Limitation

The architecture of contrastive learning ensures that augmentations of the same data point are close to each other in the embedding space. However, this alone does not guarantee that the learned embeddings are suitable for downstream tasks. As shown in Fig. 1, assuming we have eight input embedding points that belong to two different classes, red and blue. When adding Gaussian noise to the original embedding points to create their augmentations, the augmented positive points are represented by the circles around the points on the left side of Fig. 1. When there are two different contrastive

learning encoders, f_1 and f_2, they will map all augmentations of the same data point to the embedding points close to that of the original data point in the contrastive embedding space. If the two augmentations are denoted as x_{i1} and x_{i2}, $\text{sim}(f_1(x_{i1}), f_1(x_{i2}))$ will be close to 1 (the same applies to f_2). Since all the other augmentation examples are treated as negative examples, it is clear that the contrastive loss of $f_1(x)$ will be very similar as that of $f_2(x)$ when they map the original data points as shown in the right side of Fig. 1.

Even though the contrastive loss of f_1 and f_2 are very similar, the performance of the downstream classification task may differ between the two embedding spaces. For example, f_1 separates the red and blue points into distinct clusters, which makes it easy for the downstream classification task to accurately classify them. However, in the embedding space created by f_2, the red and blue points are mixed together, which results in poor performance for the downstream classification task.

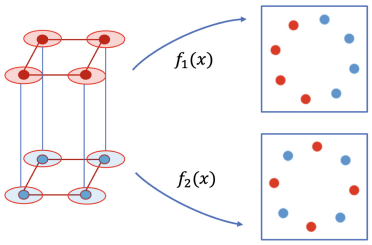

Fig. 1. The problem of existing unsupervised contrastive learning

3.2 SimSkip Details

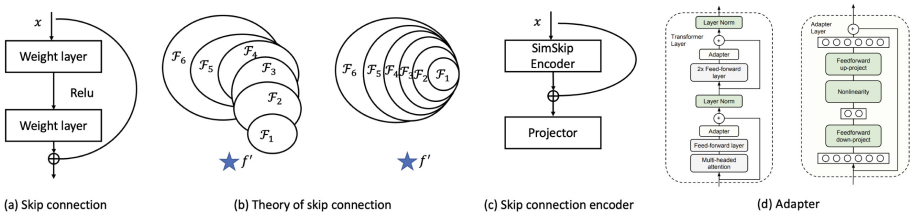

Fig. 2. The Skip Connection. The picture of Adapter is from [10].

To address this problem, we introduce skip connection contrastive learning. This method is similar in principle to ResNet [9] as illustrated in Fig. 2. The idea behind skip connections is that it retains the expressiveness of the original network. A specific network architecture defines a class of functions F that it can represent. Suppose $f^{'}$ is the optimal function we aim to find. If it is within $\mathcal{F}$, we are in good shape. However,

it is often the case that it is not. Therefore, our goal is to find the best approximation of $f^{'}$ within $\mathcal{F}$.

A naive way to achieve this is to increase the depth and width of the neural network. By adding more layers and neurons, the network can represent a new class of functions $\mathcal{F}'$, which is more expressive than $\mathcal{F}$. In general, we expect that $f_{\mathcal{F}'}$ would be better than $f_{\mathcal{F}}$, as a more expressive function class should be able to capture more complex patterns in the data. However, this may not be the case. In fact, increasing the depth and width of the network can lead to a worse $f_{\mathcal{F}'}$, as illustrated by Fig. 2(b). In this example, even though $\mathcal{F}_6$ is larger than $\mathcal{F}_3$, its optimal approximation is farther from the optimal function f'.

To solve this problem, Kaiming et al. in [9] proposes to use skip-connection to avoid the aforementioned issue from the non-nested function classes. The idea of skip-connection is that it can create nested function classes where $\mathcal{F}_1 \subseteq ... \subseteq \mathcal{F}_6$ as shown on the right of Fig. 2(b). Because the larger function classes contain the smaller ones, it can guarantee that increasing them strictly increases the expressive power of the network. For deep neural networks, if we can train the newly-added layer into an identity function $f(x) = x$, the new model will be as effective as the original model. As the new model may get a better solution to fit the training dataset, the added layer might make it easier to reduce training errors.

Building on the idea of incorporating skip connections, we propose a model named SIMSKIP that utilizes contrastive learning to refine embedding based on the original input embedding. The architecture of SIMSKIP is illustrated in Fig. 2(c). SIMSKIP consists of two components: a skip connection based encoder and a projector. The detail of the encoder can be found in Fig. 3.

The projector is a Multi-layer Perceptron (MLP) with one hidden layer, represented as $W_2\sigma(W_1x)$, where σ is a ReLU non-linearity. By incorporating skip connections, the expressive power of the network (contrastive learning encoder) is increased. Therefore, the new learned embedding should perform at least as well as the original embedding in downstream tasks.

3.3 Data Augmentation

Data augmentation is commonly used in contrastive learning to generate positive samples for a given data point. However, when the input to the model is the output embedding of another model, traditional data augmentation methods are not applicable. Image-based techniques like cropping, resizing, cut-out, and color distortion, as well as Sobel filtering, can only be applied to images [3]. Other methods such as node deletion and edge deletion for graphs are also not suitable for this purpose. Designing an effective data augmentation strategy is critical for contrastive learning methods.

Inspired by [30], in this paper, we use two types of data augmentation to embedding output of an encoder network – masking and Gaussian noise.

A - Random Masking. Random masking is applied to the input embedding. Specifically, given an input embedding $\mathbf{e}_i \in R^d$, a random vector $M \in \{0,1\}^d$ is created where 0 indicates that the element will be masked and 1 indicates no change. The number of 0 s in M is drawn from a Bernoulli distribution $B(1-\alpha)$, where α is a hyper-

parameter. The output after applying random masking is $\mathbf{e}_i \circ M$, where $\circ$ represents element-wise multiplication.

B - Gaussian Noise. When adding Gaussian noise to the input embedding $\mathbf{e}_i \in R^d$, a random vector $\epsilon \sim N(0, \mathbf{I})$ is first sampled from a multi-variable Gaussian distribution, where $\epsilon \in R^d$ and each element in ϵ is drawn from a Gaussian distribution with zero mean and unit variance. The output after adding the Gaussian noise is $\mathbf{e}_i + \delta \circ \epsilon$, where δ is a hyper-parameter.

3.4 Theoretical Proof

In this section, we theoretically demonstrate why SIMSKIP may refine it's input embedding. Here, 'refine' means that the embedding which SIMSKIP produces has no worse downstream performance than that of the original embedding which is SIMSKIP's input. We initially establish the upper bound for the loss in any downstream task within the context of contrastive unsupervised learning, as demonstrated in [31]. Then, we prove that using a skip connection-based network as the contrastive learning encoder can achieve a smaller or equal loss upper bound for downstream classification tasks compared to using original input embedding directly.

A - Preliminary. Let $\mathcal{X}$ denote the set of all possible data points. Let $f_1(x)$ represent an arbitrary neural network that takes x as its input. Then, a neural network with skip connection f_2 can be denoted as

$$f_2(x) = f_1(x) + x = f_1(x) + f_I \tag{2}$$

where $f_I = x$.

B - Downstream Task Loss for Contrastive Unsupervised Learning. In this section, we present an upper bound for the loss of a supervised downstream task which uses representation learned by any contrastive learning, as originally shown in [31].

In unsupervised learning, given a contrastive encoder f, the primary objective is to make ensure that the embeddings of the positive pair (x^+, x), generated by the function f, are close to each other, while the embeddings of the negative pair (x^-, x), generated by the same function, are far away from each other. Contrastive learning assumes access to similar data in the form of pairs (x, x^+) that come from a distribution D_{sim} as well as k i.i.d. negative samples $x_1^-, x_2^-, ..., x_k^-$ from a distribution D_{neg} that are presumably unrelated to x. Learning is done over $\mathcal{F}$, a class of representation functions $f : \mathcal{X} \longrightarrow R^d$ where f is the embedding function. The quality of the representation function f (contrastive encoder) is evaluated by its performance on a multi-class classification task $\mathrm{T} \in \mathcal{T}$ using linear classification. A multi-class classifier for $\mathrm{T} \in \mathcal{T}$ is a function g whose output coordinates are indexed by the classes c in task $\mathrm{T} \in \mathcal{T}$. For example, in SimCLR [3], the encoder is denoted as f and a linear classifier is used as the projector. So the whole framework of SimCLR can be expressed as $g(x) = wf(x)$. The loss considered in [31] is the logistic loss $l(v) = \log_2(1 + \sum_i \exp(-v_i))$ for $v \in R^d$. Then the supervised loss of the downstream task classifier g is

$$L_{sup}(\mathrm{T}, g) = E[l(\{g(x)_c - g(x)_{c'}\}_{c \neq c'})] \tag{3}$$

where c and c' are different classes. For simplicity, we use $L_{sup}(f)$ to denote the downstream loss of the model with function f which satisfies $L_{sup}(\mathrm{T}, g)$ and $g(x) = wf(x)$.

We outline the objective of contrastive learning: k denotes number of negative samples used for training. The unsupervised loss can be defined as

$$L_{un}(f) = E[l(\{f(x)^T(f(x^+) - f(x^-)\}_{i=1}^k)] \tag{4}$$

After training, suppose $\hat{f}$ is the function which can minimizes the empirical unsupervised loss and we denote its corresponding loss for supervised downstream task as $L_{sup}(\hat{f})$. According to the theorem 4.1 in [31], we have

$$L_{sup}(\hat{f}) <= \alpha L_{un}(f) + \eta Gen_M + \epsilon \tag{5}$$

where Gen is the generalization error which is defined as

$$Gen_M = O(R\sqrt{k}\frac{R_s(\mathcal{F})}{M} + (R^2 + logk)\sqrt{\frac{log\frac{1}{\epsilon}}{M}}) \tag{6}$$

and M is the sample size, $R_s(\mathcal{F})$ is the Rademacher average of $\mathcal{F}$ [31], $\mathcal{F}$ is the function space defined by f, and R is a constant which satisfies $||f(x)|| <= R$ for any x. This shows that the supervised task loss $L_{sup}(\hat{f})$ is bounded by the unsupervised loss, $L_{un}(f)$.

C - Skip-Connection Based Model. Suppose we use neural network with skip connection (f_2) to learn the contrastive embedding, according to Eq. (2), we have

$$L_{un}(f_2) = E[l(\{f_2(x)^T(f_2(x^+) - f_2(x^-)\}_{i=1}^k)] \tag{7}$$

$$= E[l(\{(f_I(x) + f_1(x))^T(f_I(x^+) + f_1(x^+) - f_I(x^-) - f_1(x^-)\}_{i=1}^k)] \tag{8}$$

where l is the logistic loss. Suppose the learned $f_1(x)$ is a trivial identity matrix I. As x is closer to x^+ than to x^-, $f_I(x)^T(f_I(x^+) - f_I(x^-)) = f_I(x)^T f_I(x^+) - f_I(x)^T f_I(x^-) >= 0$ holds. Accordingly,

$$\begin{aligned} L_{un}(f_2) = L_{un}(f_I + f_I) &= E[l(4\{f_I(x)^T(f_I(x^+) - f_I(x^-)\}_{i=1}^k)] \\ &\leq E[l(\{f_I(x)^T(f_I(x^+) - f_I(x^-)\}_{i=1}^k)] = L_{un}(f_I) \end{aligned}$$

holds because l is monotonically decreasing. This means the upper bound of skip connection contrastive learning loss $L_{un}(f_2)$ is smaller than $L_{un}(f_I)$ which is the contrastive learning error of the original embedding. If SIMSKIP learns f_1 which is not an identity matrix through contrastive learning process, $L_{un}(f_2)$ is trivially less than $L_{un}(f_I + f_I)$, which induces that $L_{un}(f_2) \leq L_{un}(f_I)$ always holds. Therefore, the upper bound for using skip connections for contrastive learning should be lower.

We have observed that the proposed SIMSKIP exhibits several similar properties to Adapter [10] in that both employ skip connection as their fundamental components as shown in Fig. 2(d). However, Adapter is embedded within each layer of Transformer [33], while SIMSKIP is positioned outside the original model $h()$. Although

Adapter has been widely used in many Transformer [33] based models [4,21], no theoretical proof has been given thus far. In this work, we present the first theoretical proof demonstrating why skip connection-based refinement does not degrade downstream tasks.

4 Experiments

Throughout the experiments, we want to show the effectiveness of SIMSKIP through downstream task metric improvement and its wide applicability to various pre-trained embeddings over different modalities including shallow knowledge graph embedding, deep graph neural network embedding, image embedding, and text embeddings. The datasets and benchmark methods used in the study are initially described, followed by the presentation of experimental results.

4.1 Experimental Setting

The study utilizes five datasets, as outlined below:

- The **movieQA** is a movie knowledge graph derived from the WikiMovies Dataset. It includes over 40,000 triples that provide information about movies.
- The **STL10** is an image dataset for image classification task. It has 10 classes and has 500 96×96 training images along with 800 test image per class.
- The **CIFAR10** is an image dataset for image classification tasks. It has 10 classes and has 6,000 32×32 color images per class. The dataset is split into 50000 training images and 10000 test images.
- The **Cora** is a graph dataset for node classification. It consists of 2,708 scientific publications as nodes with seven classes and 5,429 citations as edges.
- The **Pubmed** is a graph dataset for node classification. It consists of 19,717 scientific publication in Pubmed as nodes with three classes and 44,338 citations as edges.

The following methods are employed to learn the input embedding for contrastive learning:

- FedE [2] is a Federated Knowledge Graph embedding framework that focuses on learning knowledge graph embeddings by aggregating locally-computed updates. For the local Knowledge Graph embedding, we employed TransE [1]. This framework includes a client for each knowledge graph and a server for coordinating embedding aggregation.
- SimCLR [3] is a simple framework for contrastive learning of image representations. It first learns generic representations of images on an unlabeled dataset and then can be fine-tuned with a small number of labeled images to achieve good performance for a given classification task.
- GraphSAGE [8] is a general, inductive graph neural network (GNN) that leverages node feature information (e.g., text attributes). It samples and aggregates a nodes neighborhood's features to generate node embeddings.

– SimCSE [6] is a self-supervised text embedding that refines any pre-training transformer-based language models. Its main idea is to apply contrastive learning by treating two text embeddings obtained from the same input text with different dropout as positive pairs.

Throughout the experiment, we adhere to the original baseline experiment settings when running baselines. The embedding dimensions are 128 for FedE, 128 for SIMCLR, 128 for GraphSage, and 768 for SimCSE. As for our proposed SIMSKIP, we explore various hyper-parameters over learning rate of 0.001, 0.0003, 000003, and 0.00001, and report its optimal performance. The encoder architecture of SIMSKIP is shown in Fig. 3. Layer 1 and layer 2 have the same structure, which contains a linear layer, a batch norm layer, a ReLU and a dropout layer. The projector is a two-layer feed forward network. When the dimension of the original embedding is d, the number of parameters for the encoder is $d \times d/2$ for layer 1, $d/2 \times d$ for layer 2, and $d \times d$ for the linear layer. The number of parameters for the project is $d \times d$ for both layer 1 and layer 2. For data augmentation, the masking augmentation randomly masked 20% of the vector, and Gaussian noise augmentation added noise sampled from a Gaussian distribution with mean 0 and variance 0.13.

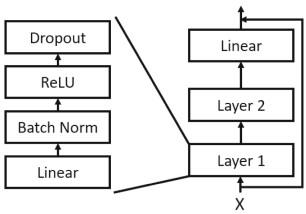

Fig. 3. The SimSkip Encoder. Layer 1 and Layer 2 have the same architecture.

4.2 SIMSKIP for Federated Knowledge Graph Embedding

This section evaluates SIMSKIP's performance of refining the embedding learned by federated learning. To have the federated knowledge graph learning setting, two knowledge graphs are randomly sampled from the movieQA knowledge graph. FedE [2] is used to learn the entity embeddings in a federated manner.

Table 1. Accuracy on different downstream tasks for movieQA knowledge graph

	KNN	Genre Classification	Movie Recommendation
FedE	58.5	84.8	7.47
SIMSKIP+ mask	62.8	**86.1**	**7.67**
SIMSKIP+ gaussian	**63.3**	86.0	7.01

In this experiment, we use three different downstream tasks - k-nearest neighbor same genre prediction (KNN), genre classification and movie recommendation. KNN is that, given a query movie, take 10-nearest movies to the query movie in the embedding space and count how many movies in those 10 movies have the same genre as the query movie. Genre classification downstream task predicts the genre of a movie according to its embedding and a 3 layered MLP was employed as a downstream classifier. Movie recommendation task recommends new movies to users according to the user's watching history. The user's watching history data is from Netflix dataset[1]. Given a user's watching history $V_1, ..., V_N$ in a chronological order, we treat $V_1, .., V_{N-10}$ as the training data, and predict 10 movies the user is most likely to watch. Then, we calculate how many movies in the 10 predicted movies belong to $V_{N-9}, ..., V_N$.

The results of different methods are shown in Table 1. SIMSKIP improved accuracy of all three downstream tasks. For KNN task, the improvement is about 4%. For genre classification and movie recommendation, the average improvement is 2% and 0.2%, respectively.

4.3 SIMSKIP for Image Embedding

In this experiment, we test SIMSKIP's performance for refining the self-supervised image embedding. We first use SimCLR [3] to learn the embedding, then we further refine the embedding with SIMSKIP. STL10 and CIFAR10 datasets were used for evaluation. The downstream task is the image classification task and a 3-layer MLP was employed as a downstream classifier. Table 2 presents the downstream image classification accuracy and shows that SIMSKIP refines the embedding space and improves the downstream task accuracy about 1% in average.

Table 2. Image classification accuracy on STL10 and CIFAR10

Image Classification	STL10	CIFAR10
SimCLR	76.09	66.88
SIMSKIP+ mask	75.84	65.93
SIMSKIP+ gaussian	**77.73**	**67.02**

4.4 SIMSKIP for Node embedding learned by Supervised Learning

In this section, we test SIMSKIP's performance of refining the embedding learned by supervised learning. We use GraphSAGE [8] as the supervised embedding learner. Core and Pubmed were used for evaluation which GraphSAGE used for its evaluation. In the experiment, we first train GraphSAGE in supervised setting and treat the output of the second to last layer as the node embedding, then we apply SIMSKIP. The downstream task is the node classification task and the same classification head of GraphSAGE was used as a downstream node classifier.

[1] https://www.kaggle.com/code/laowingkin/netflix-movie-recommendation/data.

Table 3 shows that node classification accuracy on Cora and Pubmed datasets. Because we originally thought that the embedding trained by supervised learning should fit the downstream task best, further refining it with SIMSKIP won't improve the downstream task performance. However, the experiment results show that SIMSKIP further improved the downstream task performance even with the embedding learned by a supervised task.

Table 3. Node classification accuracy on Cora and Pubmed

	Cora	Pubmed
GraphSAGE	82.60	81.6
SIMSKIP+ mask	82.60	81.8
SIMSKIP+ gaussian	**82.90**	**82.9**

4.5 SIMSKIP for Transformer-based Text Embedding

In this section, we test SIMSKIP's performance of refining the self-supervised transformer-based text embedding. Specifically, we further refine the pre-trained SimCSE [6] embedding with SIMSKIP and apply SIMSKIP text embedding to various NLP downstream tasks including CR [11], MPQA [36], MR [27], MRPC [5], SST-2 [32], SUBJ [26], and TREC [34]. These tasks are also used in [6]. We use accuracy as the metric, which means that a higher value indicates better performance. The results are presented in Table 4. Our findings suggest that stacking multiple embedding enhancing techniques (see SimCSE + SIMSKIP) keeps improving the downstream task performance.

Table 4. NLP task accuracy for self-supervised text embedding

Model	CR	MPQA	MR	MRPC	SST2	SUBJ	TREC
SimCSE	85.99	88.5	80.54	73.65	**86.47**	94.8	**82.19**
SimCSE + SIMSKIP	**86.36**	88.27	**80.82**	74.71	85.55	94.93	80.8
SimCSE + SIMSKIP (mask+noise)	85.44	**88.56**	78.25	**75.01**	82.96	**95.28**	80.1

4.6 Ablation Study

In section, we assess the effect of the skip connection in SIMSKIP. For comparison, we implemented SIMSKIP$^-$ which is obtained by removing the skip connection from SIMSKIP. For original embedding, we used embedding learned by SimCLR, STL10 and CIFAR10 as datasets, image classification as the downstream task. Table 5 shows that the accuracy of the downstream task with SIMSKIP$^-$ is lower than SIMSKIP and even lower than that with the original SimCLR embedding (see Table 2). This aligns with our

findings in Subsect. 3.1 which states that a wider and deeper network does not necessarily lead to a better approximation of the optimal function (see Fig. 2(b)). When the skip connection is omitted, the initial embedding obtained from the contrastive encoder becomes randomly dispersed across the entire embedding space. Consequently, subsequent updates have limited impact. However, with the inclusion of a skip connection, we ensure that the initial embedding from the contrastive encoder retains original useful information, facilitating the effectiveness of the subsequent updating process.

Table 5. Ablation study of SIMSKIP on image classification downstream task (unit: accuracy)

	STL10	CIFAR10
SIMSKIP + mask	75.84	65.93
SIMSKIP + gaussian	**77.73**	**67.02**
SIMSKIP$^-$ + mask	47.1	56.5
SIMSKIP$^-$ + gaussian	56.6	66.7

5 Related Work

5.1 Representation Learning

The goal of representation learning is to learn low dimensional vectors of the input data so that similar data points will be close to each other in the vector space, while dissimilar data points will be far from each other. It has been applied in many applications, such as dialogue system [17,19], fact checking [15,18–20], and question answering [16] and so on. Representation learning methods like TransE [1], RESCAL [25] and DistMult [37] embed entities in the knowledge graph as points in the low dimensional Euclidean space and model relations as linear or bilinear transformation in the space.

5.2 Contrastive Learning

Contrastive Learning focuses on minimizing the distance between the target embedding (anchor) vector and the matching (positive) embedding vector [14,22,35,38], while maximizing the distance between the anchor vector and the non-matching (negative) embedding vectors. Recent work on contrastive learning have shown that discriminative or contrastive approaches can (i) produce transferable embeddings for visual objects through the use of data augmentation [3], and (ii) learn joint visual and language embedding space that can be used to perform zero-shot detection [12]. Given the sparseness and long-tailed property of scene graph datasets, application (i) of contrastive approach can help the model learn better visual appearance embeddings of (subject, object) pairs under limited resource settings. Moreover, in application (ii), contrastive learning gives a clearer separation of the visual embeddings and language embeddings compared to the traditional black-box neural fusion approaches [9], which allows more control over both the symbolic triples input and the final output embedding spaces.

One thing to note is that the direct comparison of SimSkip to other contrastive techniques is not the primary focus. The main claim of SimSkip is its ability to further enhance the quality of embeddings learned by other contrastive methods through a skip-connection based encoder-projector architecture with contrastive learning in terms of downstream task performance. Accordingly, SimSkip serves as a facilitator of other techniques rather than a direct competitor. Additionally, since the embedding enhancing capability of SimSkip originates from the architecture rather than a specific contrastive learning training technique, SimSkip benefits from integration with other state-of-the-art contrastive learning techniques in various dimensions such as loss function and data augmentation.

6 Conclusion

In this paper, we propose a skip connection based contrastive learning framework (SimSkip) which refine the input embedding space. We theoretically prove that the downstream task error upper bounds with using SimSkip embedding as its input will not be larger than that with the original embedding. The experiment results show the effectiveness of the proposed method. For future work, we intend to explore diverse data augmentation methods in embedding space and continue reducing the error bound in theoretical analysis. Besides, we plan to analyze how SimSkip and related architectures can address the issues raised in Fig. 1.

References

1. Bordes, A., N, U.: Translating embeddings for modeling multi-relational data. In: Advances in Neural Information Processing Systems, vol. 26 (2013)
2. Chen, M., Zhang, W., Yuan, Z., Jia, Y., Chen, H.: FedE: embedding knowledge graphs in federated setting. In: The 10th International Joint Conference on Knowledge Graphs, IJCKG 2021, pp. 80–88. Association for Computing Machinery, New York (2021). https://doi.org/10.1145/3502223.3502233
3. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: III, H.D., Singh, A. (eds.) Proceedings of the 37th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 119, pp. 1597–1607. PMLR (2020). https://proceedings.mlr.press/v119/chen20j.html
4. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding (2019)
5. Dolan, W.B., Brockett, C.: Automatically constructing a corpus of sentential paraphrases. In: Proceedings of the Third International Workshop on Paraphrasing (IWP2005) (2005). https://aclanthology.org/I05-5002
6. Gao, T., Yao, X., Chen, D.: SimCSE: simple contrastive learning of sentence embeddings. In: Empirical Methods in Natural Language Processing (EMNLP) (2021)
7. Grover, A., Leskovec, J.: node2vec: scalable feature learning for networks (2016). https://doi.org/10.48550/ARXIV.1607.00653, https://arxiv.org/abs/1607.00653
8. Hamilton, W., Ying, Z., Leskovec, J.: Inductive representation learning on large graphs. In: Guyon, I., et al. (eds.) Advances in Neural Information Processing Systems, vol. 30. Curran Associates, Inc. (2017). https://proceedings.neurips.cc/paper/2017/file/5dd9db5e033da9c6fb5ba83c7a7ebea9-Paper.pdf

9. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778 (2016). https://doi.org/10.1109/CVPR.2016.90
10. Houlsby, N., et al.: Parameter-efficient transfer learning for NLP (2019)
11. Hu, M., Liu, B.: Mining and summarizing customer reviews. In: Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2004, pp. 168–177. Association for Computing Machinery, New York (2004). https://doi.org/10.1145/1014052.1014073
12. Jiang, H., Wang, R., Shan, S., Chen, X.: Transferable contrastive network for generalized zero-shot learning. CoRR abs/1908.05832 (2019). http://arxiv.org/abs/1908.05832
13. Khosla, P., et al.: Supervised contrastive learning. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., Lin, H. (eds.) Advances in Neural Information Processing Systems, vol. 33, pp. 18661–18673. Curran Associates, Inc. (2020). https://proceedings.neurips.cc/paper_files/paper/2020/file/d89a66c7c80a29b1bdbab0f2a1a94af8-Paper.pdf
14. Khosla, P., et al.: Supervised contrastive learning. arXiv preprint arXiv:2004.11362 (2020)
15. Liu, L., Du, B., Fung, Y.R., Ji, H., Xu, J., Tong, H.: KompaRe: a knowledge graph comparative reasoning system. In: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2021, pp. 3308–3318. Association for Computing Machinery, New York (2021)
16. Liu, L., Du, B., Ji, H., Zhai, C., Tong, H.: Neural-answering logical queries on knowledge graphs. In: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2021, pp. 1087–1097. Association for Computing Machinery, New York (2021)
17. Liu, L., Hill, B., Du, B., Wang, F., Tong, H.: Conversational question answering with reformulations over knowledge graph. arXiv preprint arXiv:2312.17269 (2023)
18. Liu, L., Ji, H., Xu, J., Tong, H.: Comparative reasoning for knowledge graph fact checking. In: 2022 IEEE International Conference on Big Data (Big Data) (2022)
19. Liu, L., Tong, H.: Knowledge graph reasoning and its applications. In: Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pp. 5813–5814 (2023)
20. Liu, L., et al.: Knowledge graph comparative reasoning for fact checking: problem definition and algorithms. Data Eng. **45**, 19–38 (2022)
21. Liu, Y., et al.: RoBERTa: a robustly optimized BERT pretraining approach (2019)
22. Luo, Z., Xu, W., Liu, W., Bian, J., Yin, J., Liu, T.Y.: KGE-CL: contrastive learning of tensor decomposition based knowledge graph embeddings. In: Proceedings of the 29th International Conference on Computational Linguistics, pp. 2598–2607. International Committee on Computational Linguistics, Gyeongju (2022). https://aclanthology.org/2022.coling-1.229
23. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Burges, C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K. (eds.) Advances in Neural Information Processing Systems, vol. 26. Curran Associates, Inc. (2013). https://proceedings.neurips.cc/paper/2013/file/9aa42b31882ec039965f3c4923ce901b-Paper.pdf
24. Nickel, M., Kiela, D.: Poincaré embeddings for learning hierarchical representations. In: Guyon, I., et al. (eds.) Advances in Neural Information Processing Systems, vol. 30, pp. 6341–6350. Curran Associates, Inc. (2017). http://papers.nips.cc/paper/7213-poincare-embeddings-for-learning-hierarchical-representations.pdf
25. Nickel, M., Tresp, V., Kriegel, H.P.: A three-way model for collective learning on multi-relational data. In: Proceedings of the 28th International Conference on International Conference on Machine Learning, ICML 2011, pp. 809–816. Omnipress, Madison (2011)

26. Pang, B., Lee, L.: A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts (2004). https://doi.org/10.48550/ARXIV.CS/0409058, https://arxiv.org/abs/cs/0409058
27. Pang, B., Lee, L.: Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales. In: Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL 2005), pp. 115–124. Association for Computational Linguistics, Ann Arbor (2005). https://doi.org/10.3115/1219840.1219855, https://aclanthology.org/P05-1015
28. Perozzi, B., Al-Rfou, R., Skiena, S.: DeepWalk: online learning of social representations. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2014, pp. 701–710. Association for Computing Machinery, New York (2014). https://doi.org/10.1145/2623330.2623732
29. Robinson, J., Chuang, C.Y., Sra, S., Jegelka, S.: Contrastive learning with hard negative samples (2021)
30. Saunshi, N., Ash, J., Goel, S.: Understanding contrastive learning requires incorporating inductive biases. arXiv (2022)
31. Saunshi, N., Plevrakis, O., Arora, S., Khodak, M., Khandeparkar, H.: A theoretical analysis of contrastive unsupervised representation learning. In: Chaudhuri, K., Salakhutdinov, R. (eds.) Proceedings of the 36th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 97, pp. 5628–5637. PMLR (2019). https://proceedings.mlr.press/v97/saunshi19a.html
32. Socher, R., et al.: Recursive deep models for semantic compositionality over a sentiment treebank. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pp. 1631–1642. Association for Computational Linguistics, Seattle (2013). https://aclanthology.org/D13-1170
33. Vaswani, A., et al.: Attention is all you need (2017)
34. Voorhees, E.M., Tice, D.M.: Building a question answering test collection. In: Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2000, pp. 200–207. Association for Computing Machinery, New York (2000)
35. Wang, L., Zhao, W., Wei, Z., Liu, J.: SimKGC: simple contrastive knowledge graph completion with pre-trained language models. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics, Dublin (2022)
36. Wiebe, J., Wilson, T., Cardie, C.: Annotating expressions of opinions and emotions in language. Lang. Resour. Eval. **39**, 165–210 (2005)
37. Yang, B., tau Yih, W., He, X., Gao, J., Deng, L.: Embedding entities and relations for learning and inference in knowledge bases (2015)
38. You, Y., Chen, T., Sui, Y., Chen, T., Wang, Z., Shen, Y.: Graph contrastive learning with augmentations. In: Advances in neural information processing systems, vol. 33, pp. 5812–5823 (2020)