Locality Preserving Sentence Encoding

Changrong Min[1], Yonghe Chu[2], Liang Yang[1], Bo Xu[1], and Hongfei Lin[1]
[1]Dalian University of Technology, Dalian, China
[2]Henan University of Technology, Zhengzhou, China
11909060@mail.dlut.edu.cn, yonghechu@163.com
{liang, xubo, hflin}@dlut.edu.cn

Abstract

Although researches on word embeddings have made great progress in recent years, many tasks in natural language processing are on the sentence level. Thus, it is essential to learn sentence embeddings. Recently, Sentence BERT (SBERT) is proposed to learn embeddings on the sentence level, and it uses the inner product (or, cosine similarity) to compute semantic similarity between sentences. However, this measurement cannot well describe the semantic structures among sentences. The reason is that sentences may lie on a manifold in the ambient space rather than distribute in a Euclidean space. Thus, cosine similarity cannot approximate distances on the manifold. To tackle the severe problem, we propose a novel sentence embedding method called **S**entence **BERT** with **L**ocality **P**reserving (SBERT-LP), which discovers the sentence submanifold from a high-dimensional space and yields a compact sentence representation subspace by locally preserving geometric structures of sentences. We compare the SBERT-LP with several existing sentence embedding approaches from three perspectives: sentence similarity, sentence classification, and sentence clustering. Experimental results and case studies demonstrate that our method encodes sentences better in the sense of semantic structures.

1 Introduction

Word embeddings aim to learn semantically meaningful word representations based on distribution hypothesis (Mikolov et al., 2013). Both context-free (Pennington et al., 2014) and contextual (Peters et al., 2018) word embeddings have made great progress in various downstream tasks: Text Classification (Aggarwal and Zhai, 2012), Dialogue System (Chen et al., 2017) and Text Clustering (Allahyari et al., 2017). However, in the real world, most Natural Language Processing tasks are on the sentence level. Hence, recent studies (Lin et al., 2017; Wang and Kuo, 2020) encode sentences into a dense vector space, which is described as the sentence space. These sentence embedding approaches generally fall into two categories: one is based on supervised learning, including: InferSent (Conneau et al., 2017), Universal Sentence Encoder (Cer et al., 2018) and SBERT (Reimers and Gurevych, 2019). While the other one is based on unsupervised learning, such as SkipThought vectors (Kiros et al., 2015), FastSent (Hill et al., 2016) and Transformer-based Sequential Denoising Auto-Encoder (TSDAE) (Wang et al., 2021). The unsupervised way overcomes the limitation of labeled data in different domains and data annotations, to some extent. Both of them represent a sentence as a point in the sentence space, where similar sentences are close.

There are two important problems in text processing: how to represent texts and how to evaluate their semantic similarity (He et al., 2004). Recently, various strategies have been taken to represent a sentence. For example, the SBERT (Reimers and Gurevych, 2019)learns semantic sentence representations with a Siamese Network on top of BERT. Additionally, some variants have been proposed such as SBERT-WK (Wang and Kuo, 2020) and BERT-flow (Li et al., 2020). The sentence space of the SBERT is associated with a Euclidean structure and the cosine similarity is employed to measure the semantic similarity. However, previous studies have demonstrated that human-generated text data are probably sampled from a submanifold of the ambient Euclidean space (Cai et al., 2005). As a result, sentence representations yielded from the SBERT may lie on a manifold, which is either linear or non-linear. The semantic similarity between sentences is the shortest distance, which may be curves, on the manifold. Hence, making use of the cosine similarity to approximate the length of a curve is inaccurate.

For obtaining correct semantic structures of the

Findings of the Association for Computational Linguistics: EMNLP 2021, pages 3050–3060
November 7–11, 2021. ©2021 Association for Computational Linguistics

sentence space, one straight way is to calculate the geodesic distance (Varini et al., 2006), which is the length of the shortest path between two points on the possibly curvy manifold (Ghojogh et al., 2020). However, because of requiring traversing from one point to another on the manifold, the geodesic distance is hard to approximate. Therefore, we aim to find an optimal Euclidean subspace of the sentence manifold. In the subspace, cosine similarity is effective to measure sentence semantic relations.

For implementing it, we borrow the idea of Locally Linear Embedding (Roweis and Saul, 2000), which is an effective way to develop low dimensional representations when data arises from sampling a probability distribution on a manifold (Cai et al., 2005). Then, we propose the Sentence BERT with Locality Preserving (SBERT-LP), which marries up the locality property and BERT. Our method highlights the local geometric structures of sentences. To be specific, the SBERT-LP firstly discovers the intrinsic manifold structure from the original sentence space. A novel Euclidean sentence subspace is then learned from the sub-manifold by preserving local geometric information of sentences. The local geometric structures are defined by each sentence and its neighbor sentences. Preserving locality avoids losing too much useful information of sentences during the projection. Finally, cosine similarity between sentences is consistent with their semantic similarity. Our contributions are summarized in three-folds:

(1) We theoretically analyze from the perspective of the manifold hypothesis that why the BERT-induced sentence embeddings show poor performance when retrieving semantically similar sentences.

(2) We propose the SBERT-LP, which obtains better representations in the sense of semantic structure by using locality preserving. Sentences related to the same semantics are still close to each other in the new Euclidean subspace. Our model is unsupervised and without any fine-tuning.

(3) We conduct experiments on three tasks. Experimental results and case studies demonstrate that the SBERT-LP is superior to other existing sentence embedding methods on various tasks.

2 Related work

Existing sentence embedding approaches are divided into two categories: non-parametric sentence embeddings and parametric sentence embeddings (Wang and Kuo, 2020).

The non-parameterized way is to derive sentence embeddings from pre-trained word embeddings (Mikolov et al., 2013; Pennington et al., 2014) via linear aggregations. For example, SIF (Arora et al., 2017) uses smooth inverse frequency to weigh each word in a sentence and remove some special directions with PCA. Besides, uSIF (Ethayarajh, 2018) builds upon the random walk model by setting the probability of word generation inversely related to the angular distance between the word and sentence embeddings. Although the non-parametric methods have been proved to be efficient, neglecting word orders and sentence structures degrades their performance.

In order to incorporate richer sentence information, parametric sentence embeddings are proposed. For example, SkipThought (Kiros et al., 2015) borrows the idea of skip-gram in word2vec. It encodes sentences intending to predict adjacent sentences. With the success of BERT (Devlin et al., 2019) on various NLP tasks (Sun et al., 2019; Clinchant et al., 2019), some BERT-based sentence embedding methods have been proposed recently. In addition to the SBERT, the SBERT-WK encodes sentences with QR factorization, re-weighting each word in a sentence. Furthermore, BERT-flow (Li et al., 2020) leverages Normalized Flows to transform the BERT sentence space into a standard Gaussian latent space that is isotropic. It concludes that the inner product may not accurately represent semantic similarity in the sentence space because of the non-smoothing semantic structure. In contrast, the SBERT-LP analyzes and solve the cosine metric problem of the SBERT sentence space on a manifold. Our work is inspired by the investigation of local geometry in the word space (Hasan and Curry, 2017; Yonghe et al., 2019). These methods solve semantic problems in word space. Since both word and sentence embeddings share the same high-dimensional space, problems with word embeddings may exist in sentence embeddings (Li et al., 2020). To the best of our knowledge, this paper is the first to solve the semantic metric problem in the sentence space with the incorporation of locality preserving ability.

3 Methodology

In this section, we first give a brief introduction to SBERT. Then, we will show how to effectively preserve the locality of sentences to solve the problem

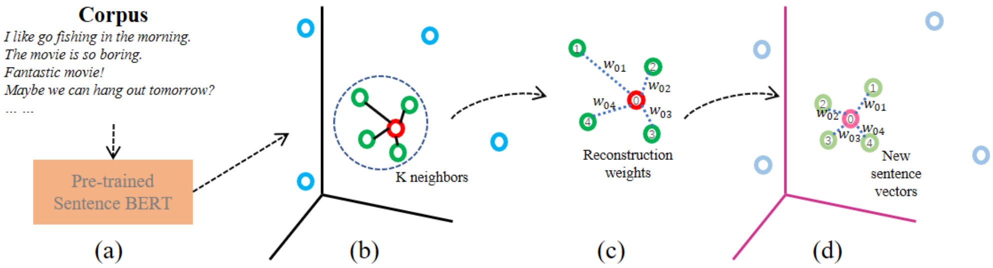

Figure 1: The architecture of SBERT-LP: (a) obtaining the high-dimensional sentence space from pre-trained SBERT; (b) constructing the kNN graph on the sentence submanifold; (c) calculating the optimal reconstruction weights of each sentences on the submanifold; (d) encoding sentences to a new Euclidean subspace, which has better semantic structures.

of semantic similarity metrics.

3.1 Sentence BERT

The sentence BERT (SBERT) is an efficient way to produce semantically meaningful sentence embeddings. It integrates the Siamese network with a pre-trained BERT language model. The SBERT pre-trained sentence embedding is trained on the SNLI and Multi-Genre NLI, and it uses cosine similarity to obtain semantic similarity between sentences. More details are provided in (Reimers and Gurevych, 2019).

3.2 Sentence BERT with Locality Preserving

To solve the semantic metric problem in the sentence space, we develop the SBERT-LP to encode sentences. Specifically, our method first constructs an adjacency graph, which captures the local geometrical structure of the original sentence space. Then, a new Euclidean subspace for sentence representation is learned by leveraging Locally Linear Embedding. The new subspace allows cosine similarity to metric semantic similarity correctly.

3.2.1 Problem Definition

Given a set of sentences S $=\{s_1,\ s_2,\ \dots s_m\}$, we first use SBERT to obtain high-dimensional representations of S. The representations denote as D $=\{d_1, d_2, \dots d_m\}$, where $d_i \in \mathbb{R}^n$. The problem is how to find a lower-dimensional embedding y_i of d_i so that $\left|{y_i}^\top y_j\right|$ can represent the correct semantic relationship between d_i and d_j.

3.2.2 Locality Preserving Embedding

Learning sentence embeddings via preserving the locality of each sample is divided into the following four steps:

Step 1: Obtaining the original sentence space from pre-trained sentence embeddings

Given a set of sentences S $=\{s_1, s_2, \dots, s_m\}$, where m is the total number of sentences. We make use of the SBERT to project sentences into a high-dimensional sentence space:

$$d_i = SBERT\left(s_i\right) \tag{1}$$

where $d_i \in \mathbb{R}^n$, and n is the dimensionality of the sentence space. In this paper, we use *BERT-base* and *BERT-large* pre-trained model, respectively. Therefore, the corresponding values of n are 768 and 1024 respectively.

Step 2: Constructing a k-Nearest Neighbors graph of sentences

We denote sentence representations obtained by SBERT as D=$\{d_1, d_2, \dots, d_m\}$. For all sentences on the sub-manifold, we construct a k-Nearest Neighbors graph. Specifically, we first calculate pairwise Euclidean distance between sentences. Then, we select the top-k nearest sentences as the neighbors of each sentence. Let $d_{ij} \in \mathbb{R}^n$ denote the j-th neighbor of the i-th sentence vector d_i and let the matrix $\mathbb{R}^{n\times k} \ni \mathbf{D}_i := [\mathbf{d}_{i1}, \dots, \mathbf{d}_{ik}]$ represent the k neighbors of d_i.

Step 3: Reconstructing sentences via local geometric structures on the manifold

The third step is to find the optimal reconstruction weights of every sentence based on the kNN graph. To optimize the linear reconstruction in the sentence space, we formulate it as:

$$\text{mini}_W\ \varepsilon(W) := \sum_{i=1}^{m} \left\| \mathbf{d}_i - \sum_{j=1}^{k} w_{ij} \mathbf{d}_{ij} \right\|_2^2 \tag{2}$$

where weights of each sentence subject to $\sum_{i=1}^{m} w_{ij} = 1, \forall i \in \{1, \dots, m\}$. $\mathbb{R}^{n\times k} \ni$

$W := [w_1, \ldots, w_m]^\top$ represents the reconstruction weight matrix. $\mathbb{R}^k \ni w_i := [w_{i1}, \ldots, w_{ik}]^\top$ denotes the reconstruction weights of the *i*-th sentence.

Then, the objective function can be restated as:

$$\varepsilon(W) = \sum_{i=1}^{m} || \mathbf{d}_i - \mathbf{D}_i w_i ||_2^2 \tag{3}$$

Then, we can imply that $\mathbf{d}_i = \mathbf{d}_i \mathbf{1}^\top w_i$ from the weight constraint. The objective can be further simplified as:

$$\begin{aligned} &\|\mathbf{d}_{\boldsymbol{i}} - \mathbf{D} w_i\|_2^2 = \left\| \mathbf{d}_i \mathbf{1}^\top w_i - \mathbf{D}_i w_i \right\|_2^2 \\ &= \left\| \left(\mathbf{d}_i \mathbf{1}^\top - \mathbf{D}_i \right) w_i \right\|_2^2 \\ &= w_i^\top \left(\mathbf{d}_i \mathbf{1}^\top - \mathbf{D}_i \right)^\top \left(\mathbf{d}_i \mathbf{1}^\top - \mathbf{D}_i \right) w_i \\ &= w_i^\top \mathbf{G}_i w_i \end{aligned} \tag{4}$$

where $\mathbf{G}_i$ is a gram matrix: $\mathbb{R}^{k \times k} \ni \mathbf{G}_i := \left(\mathbf{d}_i \mathbf{1}^\top - \mathbf{D}_i \right)^\top \left(\mathbf{d}_i \mathbf{1}^\top - \mathbf{D}_i \right)$. Eventually, we rewrite the objective function (2) as:

$$\begin{aligned} &\min_{w_i{}_{i=1}^m} \sum_{i=1}^{n} w_i^\top \mathbf{G}_i w_i \\ &\textit{subject to}: \ \mathbf{1}^\top w_i = 1, \forall i \in \{1, \ldots, m\}. \end{aligned} \tag{5}$$

For finding the optimal W, we first define the Lagrangian for equation (5) as $\mathcal{L}$:

$$\mathcal{L} = \sum_{i=1}^{m} w_i^\top \mathbf{G}_i w_i - \sum_{i=1}^{m} \lambda_i \left(\mathbf{1}^\top w_i - 1 \right) \tag{6}$$

Then, we set the derivative of Lagrangian to zero:

$$\begin{aligned} &\tfrac{\partial \mathcal{L}}{\partial w_i} = 2\mathbf{G}_i w_i - \lambda_i \mathbf{1} = 0 \\ &\Longrightarrow w_i = \tfrac{1}{2} \mathbf{G}_i^{-1} \lambda_i \mathbf{1} = \tfrac{\lambda_i}{2} \mathbf{G}_i^{-1} \mathbf{1}. \\ &\tfrac{\partial \mathcal{L}}{\partial \lambda} = \mathbf{1}^\top w_i - 1 {=} 0 \Longrightarrow \mathbf{1}^\top w_i = 1 \end{aligned} \tag{7}$$

We combine the two derivative results in Eq.(7):

$$\frac{\lambda_i}{2} \mathbf{1}^\top \mathbf{G}_i^{-1} \mathbf{1} = 1 \Longrightarrow \lambda_i = \frac{2}{\mathbf{1}^\top \mathbf{G}_i^{-1} \mathbf{1}} \tag{8}$$

Making use of Eqs. (7) and (8), we then have:

$$w_i = \frac{\lambda_i}{2} \mathbf{G}_i^{-1} \mathbf{1} = \frac{\frac{2}{\mathbf{1}^\top \mathbf{G}_i^{-1} \mathbf{1}}}{2} \mathbf{G}_i^{-1} \mathbf{1} {=} \frac{\mathbf{G}_i^{-1} \mathbf{1}}{\mathbf{1}^\top \mathbf{G}_i^{-1} \mathbf{1}} \tag{9}$$

Finally, we obtain the optimal reconstruction weights W. Actually, each sentence and its neighbors reflect local geometric structures of the sentence manifold. The optimal weights indicate in what proportion the information should be passed from the neighbors.

Step 4: Finding the optimal Euclidean sentence subspace

The SBERT-LP aims to make the locality (the optimal weights) on the sentence manifold be maintained within the Euclidean sentence sub-space. Thus, in this step, we encode sentences into the Euclidean sub-space with the locality on the sentence manifold. Then, we formulate the optimization problem of this embedding as:

$$\underset{\mathrm{Y}}{\operatorname{minimize}} \sum_{i=1}^{m} \left\| \mathbf{y}_i - \sum_{j=1}^{m} \overline{w}_{ij} \mathbf{y}_j \right\|_2^2 \tag{10}$$

subjects to $\frac{1}{m} \sum_{i=1}^{m} \mathbf{y}_i \mathbf{y}_i^\top = \mathbf{I}$, and $\sum_{i=1}^{m} \mathbf{y}_i = \mathbf{0}$. $\mathbf{I}$ is the identity matrix, while $\mathbf{y}_i \in \mathbb{R}^p$ is the *i*-th embedded sentence, and p is the dimensionality of the Euclidean sentence embeddings. Then, we denote the set of embedded sentences as a matrix: $\mathrm{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_m]^\top$, and $\mathrm{Y} \in \mathbb{R}^{m \times p}$. $\overline{w}_{ij}$ is weight between two sentences. If the *j*-th sentence is the neighbor of the *i*-th sentence, the $\overline{w}_{ij}$ is set to w_{ij}, which we have obtained in the third step. Otherwise, it equals to zero. Then the weight $\overline{w}_{ij}$ can be formulated as:

$$\overline{w}_{ij} := \begin{cases} w_{ij} & \text{if } d_j \in D_i \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

We then define the weight for the *i*-th sentence as: $\overline{w}_i = [w_{i1}, w_{i2}, \ldots, w_{im}]^\top$. Besides, we set a one-hot vector: $\mathbf{1}_i = [0, \ldots, 1, \ldots, 0]^\top$ where *i*-th element is one while the others are zero. Then, the objective function can be rewritten as:

$$\begin{aligned} &\sum_{i=1}^{m} || \mathbf{y}_i - \sum_{j=1}^{m} \overline{w}_{ij} \mathbf{y}_j ||_2^2 \\ &= \sum_{i=1}^{m} || \mathbf{Y}^\top \mathbf{1}_i - \mathbf{Y}^\top \overline{\boldsymbol{w_i}} ||_2^2 \end{aligned} \tag{12}$$

The formula is then simplified into matrix form:

$$\sum_{i=1}^{m} \left\| \mathbf{Y}^\top \mathbf{1}_i - \mathbf{Y}^\top \overline{\boldsymbol{w}}_i \right\|_2^2 |= \left\| \mathbf{Y}^\top (\mathbf{I} - \overline{\boldsymbol{W}})^\top \right\|_F^2 \tag{13}$$

where $\overline{\boldsymbol{W}}=[\overline{w}_1, \overline{w}_2, \ldots, \overline{w}_m]^\top$, $\overline{\boldsymbol{W}} \in \mathbb{R}^{m \times m}$, while $._F$ denotes the Frobenius norm of matrix. We further simplified the Eq. (13) as:

$$\begin{aligned} \left\| \mathbf{Y}^\top (\mathbf{I} - \overline{\boldsymbol{W}})^\top \right\|_F^2 &= \operatorname{tr}\left((\mathbf{I} - \overline{\boldsymbol{W}}) \mathbf{Y} \mathbf{Y}^\top (\mathbf{I} - \overline{\boldsymbol{W}})^\top \right) \\ &= \operatorname{tr}\left(\mathbf{Y}^\top (\mathbf{I} - \overline{\boldsymbol{W}})^\top (\mathbf{I} - \overline{\boldsymbol{W}}) \mathbf{Y} \right) \\ &= \operatorname{tr}\left(\mathbf{Y}^\top \mathbf{M} \mathbf{Y} \right) \end{aligned} \tag{14}$$

where $\operatorname{tr}(\,\cdot\,)$ is the trace of matrix and $\mathbf{M}=(\mathbf{I} - \overline{\boldsymbol{W}})^\top (\mathbf{I} - \overline{\boldsymbol{W}})$, $\mathbf{M} \in \mathbb{R}^{m \times m}$. Then, the objective function in Eq. (10) is formulated as:

$$\min_{\mathbf{Y}} \operatorname{tr}\left(\mathbf{Y}^\top \mathbf{M} \mathbf{Y} \right) \tag{15}$$

Therefore, if we ignore the second constraint, the Lagrangian $\mathcal{L}'$ for Eq. (15) is:

$$\mathcal{L}' = \operatorname{tr}\left(\mathbf{Y}^\top \mathbf{M} Y \right) - \operatorname{tr}\left(\mathbf{\Lambda}^\top \left(\frac{1}{n} \mathbf{Y}^\top \mathbf{Y} - \mathbf{I} \right) \right) \tag{16}$$

where $\mathbf{\Lambda} \in \mathbb{R}^{\mathbf{m} \times \mathbf{m}}$ is a diagonal matrix including the Lagrange multipliers. Then, we set the derivative of $\mathcal{L}'$ to zero:

$$\frac{\partial \mathcal{L}'}{\partial \mathbf{Y}} = 2\mathbf{M}\mathbf{Y} - \frac{2}{n} \mathbf{Y} \mathbf{\Lambda} = 0 \Longrightarrow \mathbf{M}\mathbf{Y} = \mathbf{Y} \left(\frac{1}{m} \mathbf{\Lambda} \right) \tag{17}$$

Thus, the columns of $\mathbf{Y}$ are the eigenvectors of $\mathbf{M}$, and the $\mathbf{Y}$ represents the target sentence embeddings.

4 Experiments

In this section, we perform experiments on three tasks to demonstrate the effectiveness of the SBERT-LP. We firstly introduce experimental settings for the datasets and hyper-parameters. Then we compare the SBERT-LP with several state-of-the-art sentence encoding methods. Finally, we analyze the effect of different parameters on the SBERT-LP, and we make use of some cases from STS datasets to illustrate the effectiveness of our model on semantic metric recovery. Sentence embeddings aim to cluster semantically similar sentences. Therefore, we mainly focus on the performance of different models on the STS task and take the results of the other two tasks as references.

4.1 Experimental Settings and Datasets

To verify that SBERT-LP is able to learn better sentence representations in the sense of semantics, we set three downstream tasks: Semantic Textual Similarity, Text Classification, and Text Clustering, respectively. We obtain high-dimensional sentence embeddings from two pre-trained models without fine-tuning: SBERT-base and SBERT-large. Fifteen datasets are leveraged for three tasks:

(1) For the Semantic Textual Similarity task, we use seven standard semantic textual similarity datasets: the STS tasks 2012-2016 (Agirre et al., 2012, 2013, 2014, 2015, 2016), the STS benchmark (Cer et al., 2017), and the SICK-Relatedness datasets (Marelli et al., 2014). The datasets are labeled between 0 and 5 on the semantic similarity of sentence pairs.

(2) For the Text Classification task, we use seven standard datasets in the SentEval (Conneau and Kiela, 2018) to evaluate sentence embedding approaches: MR (Pang and Lee, 2005), CR (Hu and Liu, 2004), SUBJ (Pang and Lee, 2004), MPQA (Wiebe et al., 2005), SST (Socher et al., 2013), TREC (Li and Roth, 2002) and MRPC (Dolan et al., 2004).

(3) For the Text Clustering task, we make use of the 20 Newsgroup dataset for evaluation.

4.2 Baselines

We compare the BSERT-LP with several groups of state-of-the-art methods for sentence representation learning:

(1) non-parameterized sentence encoders: Avg. GloVe embeddings; Avg. BERT embeddings; Avg. Fasttext embeddings (Joulin et al., 2017); BERT CLS-TOKEN.

(2) parameterized sentence encoders: InferSent-GloVe; Universal Sentence Encoder; SBERT; SBERT-WK; BERT-flow.

4.3 Evaluation on Semantic Textual Similarity

4.3.1 Task Description

We evaluate the model for STS without leveraging any STS specific training data. We directly evaluate sentence embedding methods on the test data and compute the cosine similarity between sentences as the similarity score. The metric is Spearman's correlation, which is the same as (Reimers and Gurevych, 2019).

4.3.2 Results and Analysis

In table 1, we report the performances of the different sentence embedding methods in terms of Spearman's correlation on the STS datasets. From

Models	STS12	STS13	STS14	STS15	STS16	STS-b	SICK-R
Avg. GloVe embeddings	55.14	70.66	59.73	68.25	63.66	58.02	53.76
Avg. BERT embeddings	38.78	57.98	57.98	63.15	61.06	46.35	58.40
BERT-CLS-TOKEN	20.16	30.01	20.09	36.88	38.08	16.50	42.63
InferSent-GloVe	52.86	66.75	62.15	72.77	66.87	68.03	65.65
Universal Sentence Encoder	64.49	67.80	64.61	76.83	73.18	74.92	76.69
SBERT$_{base}$	70.97	76.53	73.19	79.09	74.30	77.03	72.91
SBERT$_{large}$	72.27	78.46	74.90	80.99	76.25	79.23	73.75
BERT$_{base}$-flow	68.95	78.48	77.62	81.95	78.94	81.03	74.97
BERT$_{large}$-flow	70.19	80.27	78.85	**82.97**	**80.57**	81.18	74.52
SBERT-WK	68.20	68.80	74.30	77.50	77.00	-	-
SBERT$_{base}$-LP	**73.11**	80.90	74.71	76.04	72.56	78.86	75.94
SBERT$_{large}$-LP	72.12	**84.59**	**78.88**	80.42	76.50	**82.31**	**76.70**

Table 1: Spearman coefficient results for different models on the STS task. The best results are bolded and the second-best results are underlined.

this table, the proposed model SBERT-LP markedly outperforms the other competing methods in terms of the metric. Specifically, we can see that the SBERT-LP can improve the performance significantly compared with SBERT. This confirms that the SBERT-LP does a better job than SBERT in capturing semantic similarity between sentences by preserving local geometric structures of each sentence lying on the submanifold embedded in the ambient space. Besides, the SBERT-LP yields better results than the SBERT-flow, which is a strong baseline for sentence embedding, on five datasets. It is reasonable to say that the manifold distribution hypothesis of sentences is more efficient for sentence representations in the sense of semantic structures, compared with the Gaussian latent space.

4.4 Evaluation on Text Classification

4.4.1 Task Description

SBERT leverages Logistic Regression as the classifier on the text classification task. However, parameters in LR classifier may influence the experimental results. Hence, we make use of the non-parametric k-nearest neighbor (kNN) algorithm as the classifier. The distance metric of kNN is the Euclidean distance, while the k is set to 3 empirically. Accuracy is leveraged to evaluate the classification performance of models.

4.4.2 Results and Analysis

The Accuracy comparison results of the seven SentEval datasets are depicted in table 2. Even though transfer learning is not the purpose of SBERT-LP, it outperforms other state-of-the-art sentence embeddings methods on three datasets. We can observe from these results that SBERT-LP performs better than SBERT. Therefore, we can attribute the improvement achieved by SBERT-LP over SBERT and its variants to locality preserving character, which is brought LLE. However, the result of the SBERT-LP on the TREC dataset is not satisfactory. The reason is that the USE is trained on question-answer tasks, which are the same type with the TREC dataset (Reimers and Gurevych, 2019).

4.5 Evaluation on Text Clustering

4.5.1 Task Description

We make use of K-means (MacQueen et al., 1967), which is based on a distance metric, for clustering. Four indicators are employed to evaluate the performance: Mutual Information (MI), Normalized Mutual Information (NMI), Adjusted Rand Index (ARI), and Purity.

4.5.2 Results and Analysis

As shown in table 3, it is worth mentioning that the SBERT-LP significantly outperforms SBERT. This provides empirical evidence that accounting for the better semantic relationships among sentences obtained from the SBERT-LP encodes the clustering structure even better. Similar sentences are closer in the sentence space given by SBERT-LP, while dissimilar sentences are further apart. However, we can also find that the Universal Sentence Encoder (USE) achieves the best in terms of all metrics. The reason is that the USE has more intra-class consistency compared to other sentence embedding methods.

Models	MR	CR	SUBJ	MPQA	SST	TREC	MRPC
Avg. GloVe embeddings	82.94	85.25	92.51	90.02	83.69	78.40	80.00
Avg. BERT embeddings	85.15	89.43	**96.06**	91.19	86.55	89.00	80.99
Avg. fast-text embeddings	83.36	86.54	93.12	91.14	83.96	89.20	79.36
BERT CLS-TOKEN	82.87	84.19	93.88	88.50	82.10	81.40	78.78
InferSent-GloVe	85.53	89.03	93.86	92.60	88.08	85.20	75.88
Universal Sentence Encoder	80.09	85.19	93.98	86.70	86.38	**93.20**	70.14
SBERT$_{base}$	86.96	93.38	93.07	93.71	90.88	81.40	83.07
SBERT$_{large}$	**89.15**	94.38	93.33	93.80	**92.92**	79.80	83.94
SBERT$_{base}$-LP	87.19	93.64	93.23	93.88	91.32	83.06	83.94
SBERT$_{large}$-LP	88.89	**94.65**	93.58	**94.18**	92.75	83.60	**84.35**

Table 2: The accuracy of different models on the text classification task. The best results are bolded and the second-best results are underlined.

4.6 Parameters Analysis

Having shown the superiority of the SBERT-LP, in this section, we compare the performance in different neighborhood numbers and the performance in different dimensionalities.

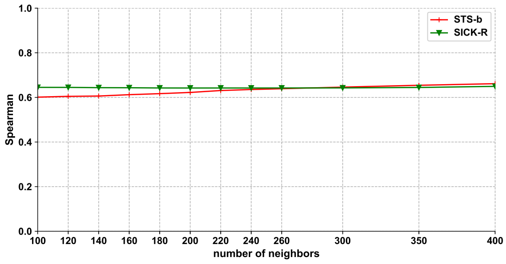

Figure 2: Spearman's coefficient of SBERT-LP on STS-b and SICK-R datasets with different number of neighbors.

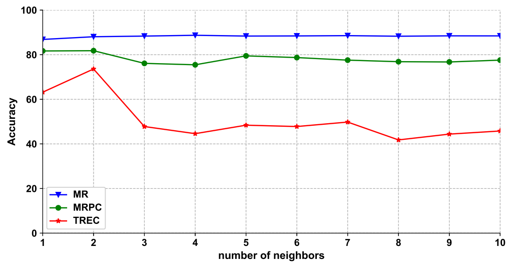

Figure 3: Accuracy of SBERT-LP on three SentEval datasets with different number of neighbors.

4.6.1 Selection of the number of neighbors

Our method is based on LLE, thus the selection of the number of neighbors is very important for constructing the local geometric structure on the sentence manifold. And several different algorithms, such as Residual Variance and Procrustes Statistics, have been proposed to find the optimal number of neighbors (Ghojogh et al., 2020). However, we experimentally find that the number of neighbors obtained by these methods is not optimal. Therefore, grid search is employed to get the optimal number of neighbors. Figures 2 and 3 demonstrate the relationship between the number of neighbors and the performance of different downstream tasks.

4.6.2 Dimensionality of the Euclidean embeddings

The dimensionality of the original sentence space is usually 768 or 1024. Although high-dimensionality sentence representations contain a wealth of semantic information, only part of the information can benefit downstream tasks. Besides, overwhelmingly complex sentence feature sets will slow the classification or regression models down and make finding global optima difficult. SBERT-LP improves this problem to a large extent. Specifically, it maps sentences into a lower-dimensional space, which reduces the number of learnable parameters for downstream tasks.

We experimentally observe that there are no specific laws for the selection of dimensions. To be specific, the dimensionality of the target space often varies greatly from task to task. For example, for Sentiment Analysis, the classification result is optimal when the dimensions are in the range of 16-64. While the optimal range is 128-300 for the STS task. This may be due to the fact that universal sentence embeddings obtained by SBERT-LP contain much less information related to sentiment than semantic information. More details are reported in figure 2.

Models	MI	NMI	ARI	Purity
Avg. GloVe embeddings	1.0444	0.3558	0.1797	0.3327
Avg. BERT embeddings	0.7320	0.2520	0.1030	0.2380
Avg. fast-text embeddings	0.5491	0.1886	0.0708	0.2057
BERT CLS-TOKEN	0.1056	0.0361	0.0104	0.1020
Universal Sentence Encoder	**1.6585**	**0.5628**	**0.3732**	**0.5740**
SBERT$_{base}$	0.9659	0.3255	0.1745	0.3540
SBERT$_{large}$	0.9412	0.3166	0.1590	0.3330
SBERT$_{base}$-LP	1.2400	0.4467	0.1917	0.4570
SBERT$_{large}$-LP	1.3171	0.4594	0.2656	0.4906

Table 3: Performance of different models on the text clustering. The best results are bolded and the second-best results are underlined.

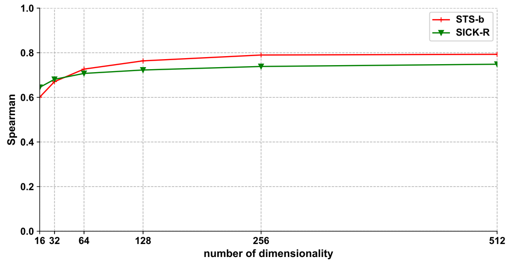

Figure 4: Spearman's coefficient of the SBERT-LP on STS-b and SICK-R datasets with different number of dimensionalities.

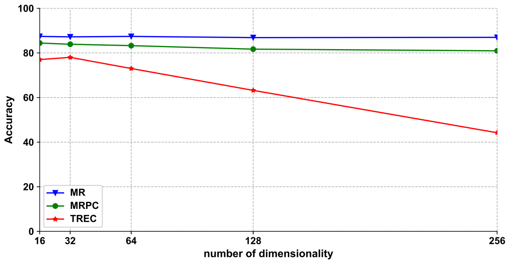

Figure 5: Accuracy of SBERT-LP on three SentEval datasets with different number of dimensionalities.

4.7 Qualitative Analysis

To verify that the SBERT-LP can make cosine similarity a valid measure of semantic similarity, we select some cases for illustration. The cases are shown in table 4.

The two pairs of sentences and labels are selected from the STS13 dataset. The labels demonstrate that the semantic distance between the two sentences of sentence pair 0 should be smaller than that of sentence pair 1. However, following the result of Sim_1, we can observe the relationship between the two pairs of sentence is reversed by the SBERT. The phenomenon shows that making use of the cosine similarity to capture semantic structures of the SBERT is invalid. Then, the result of Sim_2 shows that the SBERT-LP well solves the semantic similarity problem existing in the SBERT. To be specific, the SBERT-LP takes advantage of the locality preservation property to transform the sentence manifold in the ambient space into Euclidean sentence embeddings while keeping the semantic relationships between sentences unchanged.

Order	Sentence_0	Sentence_1	Sim_0	Sim_1	Sim_2
0	*the words in this frame describe a period of time, as opposed to a point in time.*	*the period during which something is functional (as between birth and death);*	2.2	0.6408	0.2335
1	*torres moving on after Olympic bid fails*	*torres finishes 4th, misses out on sixth Olympics*	3.0	0.6236	0.4234

Table 4: sentence pairs and their similarity scores given by cosine similarity. Sim_0 is the manual label; Sim_1 is given by SBERT; Sim_2 is given by the SBERT-LP.

5 Conclusion

In this paper, we propose the SBERT-LP that is simple yet effective. To solve the metric problem in the sentence space, this method well exploits the idea of locality preserving to recovery the cosine similarity. It not only captures the sentence submanifold but also rebuilds a Euclidean sentence subspace. Experimental results on three tasks demonstrate that the SBERT-LP learns better sentence representations in the sense of semantic structures.

References

Charu C Aggarwal and ChengXiang Zhai. 2012. A survey of text classification algorithms. In *Mining text data*, pages 163–222. Springer.

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

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

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

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

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

Mehdi Allahyari, Seyedamin Pouriyeh, Mehdi Assefi, Saied Safaei, Elizabeth D Trippe, Juan B Gutierrez, and Krys Kochut. 2017. A brief survey of text mining: Classification, clustering and extraction techniques. *arXiv preprint arXiv:1707.02919*.

Sanjeev Arora, Yingyu Liang, and Tengyu Ma. 2017. A simple but tough-to-beat baseline for sentence embeddings. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Deng Cai, Xiaofei He, and Jiawei Han. 2005. Document clustering using locality preserving indexing. *IEEE Transactions on Knowledge and Data Engineering*, 17(12):1624–1637.

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

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

Hongshen Chen, Xiaorui Liu, Dawei Yin, and Jiliang Tang. 2017. A survey on dialogue systems: Recent advances and new frontiers. *Acm Sigkdd Explorations Newsletter*, 19(2):25–35.

Stephane Clinchant, Kweon Woo Jung, and Vassilina Nikoulina. 2019. On the use of BERT for neural machine translation. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 108–117, Hong Kong. Association for Computational Linguistics.

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

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680, Copenhagen, Denmark. Association for Computational Linguistics.

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

Bill Dolan, Chris Quirk, and Chris Brockett. 2004. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 350–356, Geneva, Switzerland. COLING.

Kawin Ethayarajh. 2018. Unsupervised random walk sentence embeddings: A strong but simple baseline. In *Proceedings of The Third Workshop on Representation Learning for NLP*, pages 91–100, Melbourne, Australia. Association for Computational Linguistics.

Benyamin Ghojogh, Ali Ghodsi, Fakhri Karray, and Mark Crowley. 2020. Locally linear embedding and its variants: Tutorial and survey. *arXiv preprint arXiv:2011.10925*.

Souleiman Hasan and Edward Curry. 2017. Word re-embedding via manifold dimensionality retention. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 321–326, Copenhagen, Denmark. Association for Computational Linguistics.

Xiaofei He, Deng Cai, Haifeng Liu, and Wei-Ying Ma. 2004. Locality preserving indexing for document representation. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 96–103.

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

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431, Valencia, Spain. Association for Computational Linguistics.

Ryan Kiros, Yukun Zhu, Ruslan Salakhutdinov, Richard S. Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Skip-thought vectors. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 3294–3302.

Bohan Li, Hao Zhou, Junxian He, Mingxuan Wang, Yiming Yang, and Lei Li. 2020. On the sentence embeddings from bert for semantic textual similarity. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9119–9130.

Xin Li and Dan Roth. 2002. Learning question classifiers. In *COLING 2002: The 19th International Conference on Computational Linguistics*.

Zhouhan Lin, Minwei Feng, Cícero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. 2017. A structured self-attentive sentence embedding. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

James MacQueen et al. 1967. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA.

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

Tomás Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 3111–3119.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 271–278, Barcelona, Spain.

Bo Pang and Lillian Lee. 2005. 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'05)*, pages 115–124, Ann Arbor, Michigan. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

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

Sam T Roweis and Lawrence K Saul. 2000. Nonlinear dimensionality reduction by locally linear embedding. *science*, 290(5500):2323–2326.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to fine-tune bert for text classification? In *China National Conference on Chinese Computational Linguistics*, pages 194–206. Springer.

Claudio Varini, Andreas Degenhard, and Tim W Nattkemper. 2006. Isolle: Lle with geodesic distance. *Neurocomputing*, 69(13-15):1768–1771.

Bin Wang and C-C Jay Kuo. 2020. Sbert-wk: A sentence embedding method by dissecting bert-based word models. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2146–2157.

Kexin Wang, Nils Reimers, and Iryna Gurevych. 2021. Tsdae: Using transformer-based sequential denoising auto-encoder for unsupervised sentence embedding learning. *arXiv preprint arXiv:2104.06979*.

Janyce Wiebe, Theresa Wilson, and Claire Cardie. 2005. Annotating expressions of opinions and emotions in language. *Language resources and evaluation*, 39(2):165–210.

Chu Yonghe, Hongfei Lin, Liang Yang, Yufeng Diao, Shaowu Zhang, and Fan Xiaochao. 2019. Refining word representations by manifold learning. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 5394–5400. International Joint Conferences on Artificial Intelligence Organization.