Learning Universal Sentence Embeddings with Large-scale Parallel Translation Datasets

Anonymous ACL submission

Abstract

Although contrastive learning has greatly improved sentence representation, its performance is still limited by the size of monolingual sentence-pair datasets. Meanwhile, there exist large-scale parallel translation pairs (100x larger than monolingual pairs) that are highly correlated in semantic, but have not been utilized for learning universal sentence representation. Furthermore, given parallel translation pairs, previous contrastive learning frameworks can not well balance the monolingual embeddings’ alignment and uniformity which represent the quality of embeddings. In this paper, we build on the top of dual encoder and propose to freeze the source language encoder, utilizing its consistent embeddings to supervise the target language encoder via contrastive learning, where source-target translation pairs are regarded as positives. We provide the first exploration of utilizing parallel translation sentence pairs to learn universal sentence embeddings and show superior performance to balance the alignment and uniformity. We achieve a new state-of-the-art performance on the average score of standard semantic textual similarity (STS), outperforming both SimCSE and Sentence-T5, and the best performance in corresponding tracks on transfer tasks.

1 Introduction

It has been a fundamental problem in natural language processing to learn universal sentence embeddings that provide compact semantic representations (Reimers and Gurevych, 2019; Gao et al., 2021; Ni et al., 2021). Recently, contrastive learning (CL) which aims to learn effective representation by pulling semantically close neighbors together and separating non-neighbors, has widely attracted attention for building universal representations. Benefited from a powerful contrastive learning framework, scaling up the size of dataset greatly improves robustness and generalization of representations, as suggested by some previous works (Chen et al., 2020; Radford et al., 2021; Jia et al., 2021; Wang et al., 2021).

Gao et al. 2021 demonstrates that a contrastive objective can be extremely effective when coupled with pre-trained language models and sentence-pair datasets. However, the generality and capability of the language model are strictly limited by the size of existing sentence-pair datasets (Bowman et al., 2015; Williams et al., 2017). Meanwhile, there have accumulated large-scale parallel translation datasets (100x larger than existing monolingual sentence-pair datasets) in multilingual learning community (Yang et al., 2019a; Feng et al., 2020; Pan et al., 2021), which have not been utilized for learning universal sentence representations. Furthermore, given parallel translation pairs, previous contrastive learning frameworks (Radford et al., 2021; Gao et al., 2021) cannot well balance the alignment and uniformity (Wang and Isola, 2020) of monolingual sentence embeddings, where alignment calculates the expected distance between positive embeddings and uniformity measures how well the embeddings are uniformly distributed.

Suggested by Frozen (Tsimpoukelli et al., 2021)

\[1\] The alignment retains steady while uniformity improves.
in multimodal learning, freezing the language model and only updating the vision encoder enables strong generalization. In this paper, we build on the top of dual encoder (Radford et al., 2021; Yang et al., 2019b), and adopt a similar strategy as Frozen, where we freeze the source language encoder and only train the target language encoder for better monolingual sentence embeddings. The source language encoder constructs a large memory queue that stores negative embeddings, and provides consistent embeddings to supervise the target language encoder via contrastive learning, where source-target translation pairs are regarded as positives. Specifically, we utilize available large-scale Chinese-English translation datasets as source-target pairs to learn universal sentence embeddings in English scenarios. To obtain the source language (Chinese) encoder, instead of adopting a pre-trained model, we conduct the same protocol where a frozen pre-trained English encoder is utilized to supervise our source language (Chinese) encoder, and fine-tune it on Chinese NLI dataset for better performance. We initialize the target language (English) encoder with a pre-trained language model, such as BERT (Devlin et al., 2018) or RoBERTa (Liu et al., 2019). The illustration of training pipeline can be found in Figure 1.

We conduct a comprehensive evaluation protocol following SimCSE (Gao et al., 2021) on seven standard semantic textual similarity (STS) tasks (Agirre et al., 2012, 2013; Marelli et al., 2014; Agirre et al., 2014, 2015, 2016; Cer et al., 2017) and seven transfer tasks (Conneau and Kiela, 2018). We achieve a new state-of-the-art on STS tasks, outperforming SimCSE (Gao et al., 2021) and Sentence-T5 (Ni et al., 2021) by a large margin, and also achieve the best performance in corresponding tracks on transfer tasks evaluated by SentEval (Conneau and Kiela, 2018). On the average score of STS tasks, our pre-trained BERT$_{base}$ with or without fine-tuning surpasses SimCSE-BERT$_{base}$ by 4.39% and 3.25% respectively, and RoBERTa$_{large}$ achieves 85.58 on average. Surprisingly, BERT$_{base}$ with fine-tuning achieves better results than Sentence-T5 (11B) with only 1% parameters in comparison.

We summarize our contributions as below:

1. We provide the first exploration of utilizing existing large-scale parallel translation pairs for learning universal sentence representation.

2. We introduce a new cross-lingual contrastive learning framework to learn sentence embeddings that well balances alignment and uniformity.

3. Our approach achieves a new state-of-the-art on standard semantic textual similarity (STS), and the best performance in corresponding tracks on transfer tasks evaluated by SentEval.

2 Related Work

2.1 Universal Sentence Representation

Sentence representation is a well-studied area with many proposed methods (Mikolov et al., 2013; Pennington et al., 2014; Le and Mikolov, 2014). With the progress of pre-training, objectives like BERT (Devlin et al., 2018) and RoBERTa (Liu et al., 2019) are utilized to generate sentence embeddings. To derive semantically meaningful sentence embeddings that can be compared using cosine-similarity from BERT, Sentence-BERT (Reimers and Gurevych, 2019) uses siamese and triplet network structures. SimCSE (Gao et al., 2021) introduces a simple contrastive learning framework, which greatly improves state-of-the-art universal sentence embeddings on semantic textual similarity tasks both on unsupervised and supervised tracks. Sentence-T5 (Ni et al., 2021) investigates producing sentence embeddings from the pre-trained T5 (Raffel et al., 2019), then fine-tunes the model on natural language inference dataset and achieves the leading results in sentence embeddings benchmark datasets. These works are conducted on monolingual sentence-pair datasets, while not exploring existing large-scale parallel translation datasets. In this work, we provide an exploration of utilizing available parallel translation pairs for learning universal sentence embeddings.

2.2 Multilingual Learning

Multilingual learning has attracted increasing interests from the community. Parallel translation datasets have been widely leveraged for Neural Machine Translation (NMT) (Bahdanau et al., 2014; Wu et al., 2016), Semantic Retrieval (SR) (Wagner et al., 2001), Bitext Retrieval (Yang et al., 2019b,a) (BR) and Retrieval Question Answering (ReQA) (Kolomiyets and Moens, 2011), etc. Multilingual Universal Sentence Encoder (Yang et al., 2019b) conducts a multitask trained dual encoder to bridge 16 different languages, and achieves competitive results on SR, BR, ReQA tasks. LaBSE (Feng et al., 2021) and...
et al., 2020) adopts a dual encoder with additive margin softmax combined with masked language model (MLM) (Devlin et al., 2018) and translation language model (TLM) (Lample and Conneau, 2019) to improve multilingual sentence embeddings. mRASP2 (Pan et al., 2021) hypothesizes that inner multilingual representations leads to better multilingual translation performance. They regard a corresponding pair as a positive sample, and other in-batch samples including a variety of languages as negative samples, to establish a contrastive learning process. In this way, multiple languages representations are smoothly embedded into the same semantic space. Unlike previous works that focus on embedding text from multiple languages into the same semantic space, we propose utilizing corresponding parallel translation pairs as semantically close neighbors, pulling their embeddings together while pushing apart non-neighbors.

3 Proposed Approach

We start by briefly describing background and preliminaries in 3.1. Then, we introduce the design of our proposed contrastive framework for learning from parallel translation pairs in 3.2. Lastly, we provide analysis for our approach in 3.3.

3.1 Background

Scaling up the size of training dataset (Radford et al., 2021; Jia et al., 2021) has proved to be effective to improve robustness and generalization of representations in contrastive learning framework. However, previous works (Reimers and Gurevych, 2019; Gao et al., 2021) only utilize limited size of monolingual sentence pairs to learn universal sentence embeddings, such as MNLI datasets (Williams et al., 2017) and SNLI (Bowman et al., 2015). In contrast, there have existed large-scale well-annotated parallel translation pairs (100x larger than monolingual paired datasets) in the community of multilingual learning. Instead of training on limited monolingual sentence pairs, utilizing existing parallel translation datasets shows better flexibility and a potential to further improve the performance of sentence embeddings, where a parallel translation pair that is highly correlated in semantic can be treated as a positive sample.

Preliminaries. To utilize paired inputs, single multilingual encoder (Ma et al., 2020; Pan et al., 2021) and dual encoder (He et al., 2020; Radford et al., 2021; Ni et al., 2021) are the most commonly adopted strategies for learning multilingual representations. Multilingual encoder embeds sentences from different languages into a single semantic space using a unified encoder, based on the hypothesis that universal multilingual learning leads to better multilingual sentence representation. Its architecture is illustrated in A, Figure 2. Dual encoder, also known as two-tower, models the paired data with two independent encoders, and projects the embeddings of paired inputs into the same semantic space through joint training. Its architecture is illustrated in B, Figure 2.

Alignment and uniformity. Wang and Isola (2020) identifies two key properties related to contrastive learning that measure the quality of representations. The alignment calculates the expected distance between embeddings of the paired positive instances, while the uniformity measures how well the embeddings are uniformly distributed. Following Gao et al. (2021), we also use these metrics to demonstrate the inner workings of our approach.

3.2 Method

Although multilingual encoder and dual encoder can use parallel translation pairs straightforwardly,
they both suffer from the imbalance between alignment and uniformity, as source language encoder and target language encoder keep updating in the training process. In other words, while they pull the positive samples (source-target translation pairs) closer and the negative samples (source-non target translation pairs) farther away through an explicit contrastive learning objective, the alignment and uniformity of embeddings from monolingual sentence pairs cannot be guaranteed. Specifically, let \((s_i, t_i)\) denote the representation of a parallel translation pair generated by the source language encoder and target language encoder, respectively. We simplify the explicit contrastive objective as Eq 1.

\[
L_{\text{explicit}} = \alpha_1 * L_p - \alpha_2 * L_n
\]  

(1)

Where \(L_p\) and \(L_n\) represent the distance for positives and negatives of parallel translation pairs as defined in Eq 2 and Eq 3, \(\alpha\) denote the linear weights, \(D\) is a distance function, and \(i \neq j\). The explicit contrastive objective is to minimize the distance between positives and maximize the distance between negatives.

\[
L_p = D(s_i, t_i) + D(s_j, t_j)
\]  

(2)

\[
L_n = D(s_i, t_j) + D(s_j, t_i)
\]  

(3)

Given parallel translation pairs, we also define the implicit or actual objective that has not been considered into contrastive learning framework in Eq 4, which measures the alignment and uniformity of monolingual sentence embeddings. Although \(L_{\text{implicit}}\) is not considered in the explicit contrastive objective, we expect to retain good alignment and uniformity of monolingual sentence embeddings from the target encoder, as the actual objective is to learn monolingual universal sentence embeddings from parallel translation pairs.

\[
L_{\text{implicit}} = \beta_1 * L_p' - \beta_2 * L_n'
\]  

(4)

Where \(L_p'\) and \(L_n'\) represent the distance for positives and negatives of monolingual pairs as defined in Eq 5 and Eq 6. \(s_i^+\) and \(t_i^+\) represent the monolingual positive samples for \(s_i\) and \(t_i\), respectively. \(\beta\) denote linear weights.

\[
L_p' = D(s_i, s_i^+) + D(t_i, t_i^+)
\]  

(5)

\[
L_n' = D(s_i, s_j) + D(t_i, t_j)
\]  

(6)

In preliminaries, as shown in (A) and (B), Figure 2, the source language encoder keeps updating in training and can not provide consistent supervision for the target language encoder. The implicit objective for preliminaries is Eq 4, where the alignment and uniformity of source embeddings and target embeddings are both required to be implicitly optimized. However, given two independent implicit objectives, it becomes hard to find a local optimum through Eq 1 without any constraints.

To effectively improve the uniformity and retain the alignment simultaneously, and optimize the implicit objective (4) through an explicit objective (1), we propose to soften the implicit objective for better optimization with our modified architecture, built on the top of regular dual encoder. To be clear, we freeze the side of the source language encoder, so that the alignment and uniformity of source embeddings are frozen in the training. In this case, the implicit objective degrades to Eq 7.

\[
L_{\text{implicit}} = \beta_1 * D(t_i, t_i^+) - \beta_2 * D(t_i, t_j)
\]  

(7)

As the optimization space shrinks and the implicit objective relaxed, finding the local optimal solution becomes easier and more efficient. We show the differences between our approach (C) and preliminaries (A, B) in Figure 2.

3.3 Analysis

We first analyze the connection between our approach and SimCSE (Gao et al., 2021) and claim that the modified dual architecture with parallel translation pairs as input shares the same implicit...
contrastive objective as SimCSE with monolingual pairs as input. Then, we provide the visualization results of alignment and uniformity that show superior performance compared to preliminaries.

**Connection to SimCSE.** As shown in Figure 3, the explicit objective of SimCSE is defined in Eq 1. However, as SimCSE adopts a single monolingual encoder, the source and target language encoder refers to the same model. Given monolingual sentence pairs, $t_i = s_i^+$ is valid, and the implicit objective defined in Eq 4 is identical to its explicit objective. The alignment and uniformity of target language embeddings are optimized in the training. In our approach, as the source encoder is frozen, we soften the implicit objective to the alignment and uniformity of monolingual target embeddings as SimCSE. The only difference is that we optimize the target encoder implicitly with parallel translation pairs, while SimCSE optimizes explicitly with monolingual sentence pairs.

**Visualization of alignment and uniformity.** To validate the effectiveness of our approach, we take the checkpoint of our model and preliminaries every 100 steps during training and visualize their alignment and uniformity. The alignment and uniformity of target language embeddings can be found in 4.4.2 and the data used for visualization is in Appendix A. In A, Figure 4, we show the promising results of implicit objective (the alignment and uniformity of target encoder), given monolingual sentence pairs as input, where we greatly improve uniformity and retain a steady alignment, while others dramatically degrade alignment. In B, Figure 4, We also compare the convergence of explicit objective between three models. Starting from pre-trained checkpoints, all models greatly improve uniformity given parallel translation pairs as input. In contrast, we achieve a better training direction in alignment than other methods, which exhibits a more consistent convergence in cross-lingual training.

4 Experiments

We first describe the datasets in 4.1, and illustrate the training details in 4.2. Then in 4.3, we conduct comprehensive experiments to evaluate the effectiveness of our method. Lastly, we do ablation studies for further analyzing in 4.4.

4.1 Training Datasets

We adopt WMT and source-mixed datasets that have parallel translation pairs for cross-lingual contrastive learning, while the Chinese NLI dataset that has monolingual Chinese sentence pairs is only utilized for fine-tuning.

WMT Dataset\(^5\) is a common-used machine translation dataset composed of various sources. We perform an elaborate cleaning process following (Meng et al., 2020) to filter out low-quality pairs. We get 19,442,200 Chinese-English translation parallel pairs after cleaning.

Source-mixed Dataset collects from more open-sourced translation datasets built on the top of WMT dataset, including AIC (Wu et al., 2017), translation2019zh (Xu, 2019), UN Corpus (Ziemske et al., 2016), etc. Finally, we establish a

---

\(^5\)http://www.statmt.org/wmt20/
larger-scale dataset including 56,741,808 Chinese-English translation pairs. This dataset is used to show that further scaling up the size of the training set helps improve overall performance.

**Chinese NLI Dataset** is a Chinese Nature Language Inference dataset which is similar to NLI dataset (Bowman et al., 2015; Williams et al., 2017). We adopt the same method in SimCSE (Gao et al., 2021) to handle the Chinese NLI dataset: given one premise (sentence), we regard the absolutely true (entailment) sentence as the positive, and the definitely false (contradiction) sentence as the hard negative. We establish a dataset containing 315,298 triplets, and each triplet has 3 sentences: premise, positive, hard negative sentences.

### 4.2 Training Details

We elaborate the training details of our pipeline that is shown in Figure 1. We maintain a consistent memory queue (He et al., 2020) of negative embeddings, where the current mini-batch of the source language encoder’s embeddings are enqueued and the oldest are dequeued. The pooling method used in the training is [CLS] with an MLP layer following SimCSE. All experiments are conducted on 8 V100 GPUs. The batch size in experiments represents the batch size on each GPU.

#### 4.2.1 Training a Chinese Encoder

As shown in (A), Figure 1, the first step is to train a target language (Chinese) encoder. Specifically, we adopt the pre-trained SimCSE-RoBERTa\textsubscript{large} model as the source language (English) encoder, and initialize a Chinese RoBERTa\textsubscript{large} model\footnote{https://github.com/pluto-junzeng/CNSD} with pre-trained weights as the target language (Chinese) encoder. We adopt a series of hyperparameters from 4.2.2: learning rate is 5e-5, batch size is 200, queue size is 200,000, dropout is 0.1, and the input sentence length is 50. In addition, a cosine learning rate scheduler is applied for maintaining the consistency of training. We freeze the source language (English) encoder and only update the target language (Chinese) model. We evaluate every 250 training steps on the development set of Chinese STS-B and save the best checkpoint. The target language (Chinese) model is trained for 2 epochs on WMT or source-mixed dataset. To further boost the performance of the target language (Chinese) model, we fine-tune it on Chinese NLI dataset, with the same settings as described in section 4.2.3.

#### 4.2.2 Training an English Encoder

As shown in B, Figure 1, we train a target language (English) encoder that generates universal sentence embeddings. Specifically, we reuse the pre-trained Chinese encoder from 4.2.1 as the source language (Chinese) encoder and freeze its parameters. We evaluate every 250 training steps on the development set of STS-B and save the best checkpoint.

**Effect of Temperature.** Temperature is a crucial factor which impacts training convergence and the overall performance in contrastive learning. We evaluate several temperatures recommended by previous works (Gao et al., 2021; Ni et al., 2021; Radford et al., 2021), including 0.05, 0.01, parameter 1 (a learnable parameter in training). As shown in Table 1, a parameter 1 works best.

<table>
<thead>
<tr>
<th>Temperature</th>
<th>0.01</th>
<th>0.05</th>
<th>Parameter 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>BERT\textsubscript{base}</td>
<td>81.59</td>
<td>86.93</td>
<td>87.73</td>
</tr>
</tbody>
</table>

Table 1: Effect of the temperature.

For BERT\textsubscript{base} (or RoBERT\textsubscript{a\textsubscript{base}}), the learning rate is we-4, batch size is 400, queue size is 10000, temperature is parameter 1 and the dropout is defaulted set as 0.1. We leverage the cosine learning rate scheduler to adjust the learning rate dynamically. In the term of RoBERT\textsubscript{a\textsubscript{large}} (or BERT\textsubscript{large}), we set the learning rate to 5e-5, batch size to 200, queue size to 200,000, all other hyperparameters keep the same as BERT\textsubscript{a\textsubscript{base}}. Refer to appendix B for grid search of hyperparameters.

#### 4.2.3 Fine-tune on NLI Dataset

We investigate the effect of scaling up training dataset by fine-tuning on NLI dataset. The NLI dataset contains 275,602 samples, and each sample consists of a query sentence, a positive sentence, and a hard negative sentence. Following the similar training setting as SimCSE, we set the learning rate to 1e-5, batch size to 128, dropout to 0.1, temperature to 0.05, and input length to 50 for small models (BERT\textsubscript{base} and RoBERT\textsubscript{a\textsubscript{base}}). While for large models (BERT\textsubscript{a\textsubscript{large}} and RoBERT\textsubscript{a\textsubscript{large}}), we set batch size to 96.

### 4.3 Evaluation Results

Following Gao et al., we evaluate our models on seven transfer and seven STS tasks by SentEval

---

\footnote{https://huggingface.co/hfl/chinese-RoBERTa-wwm-ext-large}
tools. As the main goal of learning sentence embeddings is to cluster semantically similar sentences, we also take STS result as the main metric.

Semantic textual similarity tasks. We evaluate our approach under zero-shot and fine-tuned settings, respectively. To fairly compare with previous works (Gao et al., 2021; Ni et al., 2021), we adopt seven STS tasks including STS 2012–2016 (Agirre et al., 2012, 2013, 2014, 2015, 2016), STS Benchmark (Cer et al., 2017) and SICK-Relatedness (Marelli et al., 2014). STS tasks are widely used in measuring the discriminative power of sentence embeddings. In STS, sentence embeddings are evaluated by how well their cosine similarities correlate with human-annotated similarity scores. Suggested by Reimers et al., 2016; Gao et al., 2021, we also report Spearman’s correlation coefficients to evaluate the performance.

We start from pre-trained checkpoints of BERT or RoBERTa as the backbone. We divide the comparison into 3 tracks for a comprehensive comparison: BERT track, RoBERTa track, and state-of-the-art track. Specifically, BERT track includes Sentence-BERT (Reimers and Gurevych, 2019), CT-BERT (Carlsson et al., 2020), and Sim-BERT. RoBERTa track includes SimRoBERTa and Sentence-RoBERTa. In the term of the state-of-the-art track, we compare with Sentence-T5 (Ni et al., 2021) 11B model, which contains 11 billion parameters. Table 2 reports the evaluation results on seven STS tasks. Our approach can substantially improve results on all the datasets with or without extra NLI supervision, greatly outperforming the previous state-of-the-art models. Specifically, our approach outperforms the averaged Spearman’s correlation of SimCSE by 1.27-2.65 under a zero-shot setting.

Table 2: Comparison with previous state-of-the-art works in STS tasks. All results are from Gao et al., 2021; Ni et al., 2021; Reimers and Gurevych, 2019; WMT and SMD represent the model is trained on WMT dataset and source-mixed dataset, respectively. The pooling methods used for comparison can be found in Appendix C, and the Ours-RoBERTa_large(WMT)’s pooling method is [CLS] with MLP.
shot setting in all tracks. When using NLI datasets, Ours-BERT\textsubscript{base} further pushes the state-of-the-art results from 84.94 to 85.15. The gains are more pronounced on RoBERTa encoders, and our method achieves 85.58 with RoBERTa\textsubscript{large}.

**Transfer Tasks.** We evaluate on the following transfer tasks: MR (Pang and Lee, 2005), CR (Hu and Liu, 2004), SUBJ (Pang and Lee, 2004), MPQA (Wiebe et al., 2005), SST-2 (Socher et al., 2013), TREC (Voorhees and Tice, 2000) and MRPC (Dolan and Brockett, 2005). We employ the default configurations from SentEval. Results on transfer tasks are shown in Appendix Table 7.

Benefited from the large scale of parallel translation datasets that boosts the power of contrastive learning, our method learns more generalized sentence representations than previous approaches, and improves performance on transfer tasks.

### 4.4 Ablation Studies

We investigate the impact of source language encoder and contrastive objectives. We use BERT\textsubscript{base} (WMT) without fine-tuning as our benchmark.

#### 4.4.1 The effect of source language encoder

To analyze the role of source language encoder, we train a SimCSE-RoBERTa\textsubscript{large} model on the Chinese NLI dataset directly and use it as the source language (Chinese) encoder. For comparison, we train two RoBERTa\textsubscript{large} models on the WMT dataset following the steps in 4.2.1 with and without fine-tuning. Then, we train three target language (English) encoders as 4.2.2 given different source language models and evaluate them on the SST-B development set. We report the results in Table 3. We also directly evaluate the source language (Chinese) encoder on the Chinese STS-B test dataset. The results are in Table 4. All results reveal the superior performance of our approach.

### 4.4.2 The effect of contrastive objectives

In 3.1, we describe preliminaries in contrastive learning for handling paired data. Figure 2 shows the differences. To show the effectiveness of our cross-lingual contrastive learning scheme, we train models with multilingual encoder, dual encoder and our modified dual architecture, respectively, and evaluate their performance on STS-B development set. For dual encoder, we adopt the pre-trained source language (Chinese) encoder from 4.2.1 and a pre-trained RoBERTa\textsubscript{base}, then train it via contrastive learning. For multilingual encoder, we adopt a RoBERTa\textsubscript{base}-xlm (Lample and Conneau, 2019) model that accepts multilingual input. For our modified dual architecture, we use the same source and target encoder as dual encoder, while keeping the source encoder frozen. All models are trained on WMT dataset.

<table>
<thead>
<tr>
<th>Model</th>
<th>SimCSE\textsubscript{CN}</th>
<th>Ours</th>
<th>Ours+F</th>
</tr>
</thead>
<tbody>
<tr>
<td>STS-B\textsubscript{CN}</td>
<td>81.13</td>
<td>81.13</td>
<td><strong>83.37</strong></td>
</tr>
</tbody>
</table>

Table 4: Performance of source language encoders on Chinese STS-B test dataset. SimCSE\textsubscript{CN} represents the Chinese SimCSE-RoBERTa\textsubscript{large}. Ours+F and Ours are RoBERTa\textsubscript{large} that trained by our strategy with and without fine-tuning, respectively.

#### Models

<table>
<thead>
<tr>
<th>Models</th>
<th>Multilingual</th>
<th>Dual</th>
<th>Ours</th>
</tr>
</thead>
<tbody>
<tr>
<td>STS-B</td>
<td>71.02</td>
<td>73.13</td>
<td><strong>86.82</strong></td>
</tr>
</tbody>
</table>

Table 5: The effect of contrastive objectives. Dual, Multilingual and Ours represent dual encoder, multilingual encoder and our modified dual encoder.

For a fair comparison, we unify the hyperparameters of different objectives: batch size is 128, learning rate is 2e-4, queue size\(^8\) is 0, temperature is parameter 1. The only difference between dual encoder and ours is whether the source language encoder is frozen in the training. Table 5 shows the effectiveness of our approach.

### 5 Conclusion

In this work, we provide the first exploration of utilizing existing large-scale parallel translation pairs for learning universal sentence representation, propose a modified dual architecture that well balances the alignment and uniformity of embeddings. We demonstrated that our method achieves a new state-of-the-art on standard semantic textual similarity (STS), and the best performance on corresponding tracks on transfer tasks, outperforming both SimCSE and Sentence-T5.

\(^8\)We gather the samples from other GPUs, so the comparative samples in contrastive learning are 128×8=1024.
References


Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In International conference on machine learning, pages 1188–1196. PMLR.


## A Validation Set for Visualization

For monolingual sentence-pair dataset, we adopt the STS-B development set and the same settings as the SimCSE (Gao et al., 2021). For parallel translation dataset, UN Corpus development set is used for our visualization. We take out the first 1000 data of the UN Corpus development set. Then, we use the first 250 as positive samples, and replace the Chinese sentence in the last 750 pairs with other Chinese sentences (randomly selected in remaining data in the UN Corpus development set) as negative samples to build a visual validation set of parallel translation data.

## B Hyperparameters

We also provide comprehensive analysis of hyperparameters on cross-lingual contrastive learning, including the size of memory queue, learning rate and batch size. We perform grid-search of batch size $\in \{128, 256, 400, 512\}$, learning rate $\in \{5e^{-5}, 1e^{-4}, 2e^{-4}, 5e^{-4}\}$ and queue size $\in \{1024, 4096, 10000, 50000\}$ for BERT base, and batch size $\in \{64, 128, 200\}$, learning rate $\in \{1e^{-5}, 2e^{-5}, 5e^{-5}, 1e^{-4}\}$ and queue size $\in \{10000, 50000, 200000, 300000\}$ for RoBERTa large. We evaluate on STS-B development set. The results are shown in Table 6.

<table>
<thead>
<tr>
<th>BERT base</th>
<th>BERT large</th>
<th>RoBERTa base</th>
<th>RoBERTa large</th>
</tr>
</thead>
<tbody>
<tr>
<td>Batch size</td>
<td>400</td>
<td>200</td>
<td>400</td>
</tr>
<tr>
<td>Learning rate</td>
<td>2e-4</td>
<td>5e-5</td>
<td>2e-4</td>
</tr>
<tr>
<td>Queue size</td>
<td>10 T</td>
<td>200 T</td>
<td>10 T</td>
</tr>
</tbody>
</table>

Table 6: Our setting of batch sizes, queue size and learning rates for different models. T represents a thousand.

## C The Effect of Pooling

Suggested by Gao et al. (2021), pooling strategies make differences in the performance. Li et al. (2020) shows that taking the average embeddings of the pre-trained model leads to better performance than [CLS]. Here, we consider three different pooling settings: (1) Average Pooling, (2) [CLS] with MLP, (3) [CLS] without MLP. Table 8 shows the comparison between different pooling methods. We evaluate on STS-B development set. As shown, we find that CLS without MLP method was the best.
<table>
<thead>
<tr>
<th>Model</th>
<th>MR</th>
<th>CR</th>
<th>SUBJ</th>
<th>MPQA</th>
<th>SST</th>
<th>TREC</th>
<th>MRPC</th>
<th>Avg</th>
</tr>
</thead>
<tbody>
<tr>
<td>InferSent-GloVe</td>
<td>81.57</td>
<td>86.54</td>
<td>90.38</td>
<td>84.18</td>
<td>88.20</td>
<td>75.77</td>
<td>85.59</td>
<td></td>
</tr>
<tr>
<td>Universal Sentence Encoder</td>
<td>80.09</td>
<td>85.19</td>
<td>93.98</td>
<td>86.70</td>
<td>86.38</td>
<td>93.20</td>
<td>70.14</td>
<td>85.10</td>
</tr>
<tr>
<td>SBERT_base</td>
<td>83.64</td>
<td>89.43</td>
<td>94.39</td>
<td>89.86</td>
<td>89.60</td>
<td>76.00</td>
<td>87.41</td>
<td></td>
</tr>
<tr>
<td>SimCSE-BERT_base</td>
<td>82.69</td>
<td>89.25</td>
<td>94.81</td>
<td>89.59</td>
<td>88.40</td>
<td>73.51</td>
<td>86.51</td>
<td></td>
</tr>
<tr>
<td>Ours-BERT_base(SMD)</td>
<td>85.78</td>
<td>91.26</td>
<td>94.90</td>
<td>91.41</td>
<td>90.77</td>
<td>91.40</td>
<td>77.74</td>
<td>89.04</td>
</tr>
<tr>
<td>SRoBERTa_base</td>
<td>84.91</td>
<td>90.83</td>
<td>92.56</td>
<td>88.75</td>
<td>90.50</td>
<td>88.60</td>
<td>78.14</td>
<td>87.76</td>
</tr>
<tr>
<td>SimCSE-RoBERTa_base</td>
<td>84.92</td>
<td>92.00</td>
<td>94.11</td>
<td>89.82</td>
<td>91.27</td>
<td>88.75</td>
<td>75.65</td>
<td>88.08</td>
</tr>
<tr>
<td>SimCSE-RoBERTa_large</td>
<td><strong>88.12</strong></td>
<td>92.37</td>
<td>95.11</td>
<td>90.49</td>
<td><strong>92.75</strong></td>
<td>91.80</td>
<td>76.64</td>
<td>89.61</td>
</tr>
<tr>
<td>Ours-RoBERTa_base(SMD)</td>
<td>88.02</td>
<td><strong>92.45</strong></td>
<td><strong>95.45</strong></td>
<td>91.23</td>
<td>92.70</td>
<td><strong>94.80</strong></td>
<td>76.17</td>
<td><strong>90.12</strong></td>
</tr>
</tbody>
</table>

Table 7: Performance on transfer tasks. Results are from Gao et al.; Ni et al.; Reimers and Gurevych. SMD represents the model is pre-trained on source-mixed dataset. The models in comparison are both fine-tuned.

<table>
<thead>
<tr>
<th>Models</th>
<th>[CLS] w/M</th>
<th>AVG</th>
<th>[CLS] wo/M</th>
</tr>
</thead>
<tbody>
<tr>
<td>BERT_base</td>
<td>85.19</td>
<td>87.28</td>
<td><strong>88.08</strong></td>
</tr>
</tbody>
</table>

Table 8: The effect of different pooling methods. [CLS] w/M and [CLS] wo/M represent [CLS] with or without an MLP layer, respectively.

works the best for our models. In addition, we adopt the [CLS] with MLP as the fine-tuned models pooling method, as suggested by SimCSE (because we fine-tune our models by SimCSE method).