Contrastive Learning for Generalizable Sentence Embeddings

Stanford CS224N Default Project

Shenghan Chen
Graduate School of Education
Stanford University
csh3@stanford.edu

Abstract

Recent works have proven contrastive learning an effective framework for producing generalizable sentence embeddings. In this project, supervised and unsupervised contrastive training is explored in conjunction with task-specific supervision to produce high-quality sentence embeddings for multiple tasks: sentiment classification, paraphrase detection, and semantic textual similarity prediction. In unsupervised settings, a fusion training approach is adopted, where the same batch of STS data is trained first against the supervised classification objective and then against the unsupervised contrastive objective. It leads to the best overall score of 0.520 on the dev sets, among all unsupervised methods explored in this project. In supervised settings, the paraphrases in the Quora dataset are used as positive pairs. During task-specific fine-tuning, it is helpful to add MLP networks on top of BERT embeddings as predition heads, while sharing the first layer between paraphrase and STS tasks. Multi-task average loss turns out to work the best, as compared to alternative approaches like gradient surgery or round robin training. Finally, A model ensemble combines the best-performing models on each single task and give the best results on the test sets: a sentiment classification accuracy of 0.521, a paraphrase detection accuracy of 0.830, a semantic textual similarity of 0.753, and an overall accuracy of 0.701. In summary, contrastive learning allows flexible construction of instance pairs from various kinds of datasets. It is demonstrated to be an effective pre-training method for sentence embeddings across different tasks.

1 Key Information to include

- Mentor: Hans Hanley
- External Collaborators (if you have any): None
- Sharing project: None

2 Introduction

This project aims to produce generalizable sentence representation for 3 different tasks: sentiment classification, paraphrase detection, and semantic textual similarity prediction. Traditional machine learning methods are known to solve many tasks rather effectively. A deep model usually has one single objective during its training stage. Hence, in order to tackle multiple tasks, it would require several disjoint models and separate training for each task, which in turn produces different representations of the same data. On the other hand, unified representation is often preferable in NLP, because its application to a broad range of tasks indicates true semantic understanding.

Contrastive learning is a promising framework for this purpose, not only for its wide application across various domains, but also for its proved effectiveness for learning semantic similarity. Furthermore, an unsupervised contrastive approach holds the potential of scaling to training truly massive language models.

The main goal of this project is to answer the question: how contrastive learning can be used to improve the generalizability of sentence embeddings, as measured by performance on the 3 predefined tasks. The core idea of contrastive learning is to pull together embeddings of semantically similar sentences, while pushing apart those of dissimilar sentences.

Both supervised and unsupervised contrastive training approaches are explored in conjunction with task-specific supervision. In unsupervised settings, different ways of combining contrastive training and task supervision are experimented with. The best-performing model adopts a fusion training approach, where the same batch of STS data is trained first against the supervised classification objective and then against the unsupervised contrastive objective. It leads to an overall score of 0.520 on the dev sets, best among all unsupervised methods explored in this project. In supervised settings, the paraphrases in the Quora dataset are used as positive pairs, with non-paraphrases as negative pairs. During task-specific fine-tuning, a small MLP network is added on top of BERT embeddings as the predition head for each task, and the first layer is shared between paraphrase and STS tasks. Multi-task average loss turns out to work the best among all task-specific objectives, including gradient surgery and round robin training. Finally, A model ensemble combines the best-performing models on each single task and give the best results on the test sets: a sentiment classification accuracy of 0.521, a paraphrase detection accuracy of 0.830, a semantic textual similarity of 0.753, and an overall accuracy of 0.701.

3 Related Work

The concept of contrastive learning may trace back to the domain of computer vision. Contrastive Predictive Coding (Oord et al., 2018) was among the first such methods developed for self-supervised vision learning. It was followed by MoCo (He et al., 2020) and SimCLR (Chen et al., 2020) with significantly improved performance on image classification tasks. In particular, SimCLR considers different data augmentation of the same image as positive instances, and other examples in a batch as negative instances.

As suggested by the name, SimCSE (Gao et al., 2021) can be considered the NLP counterpart of SimCLR. In this paper, both unsupervised and supervised approaches were proposed for learning semantic similarity. In absence of labeled data, the creative use of standard transformer dropout as data augmentation achieves surprisingly decent results. Whereas strong supervision from the NLI datasets further boosts model performance, especially when pairs labeled "contradiction" are taken as hard negatives. Unlike SimCLR, SimCSE does not require a huge batch size, making it suitable for this project.

Besides contrastive learning, many other methods have also been extensively researched in pursuit of better sentence embeddings. Multiple Negatives Ranking Loss (Henderson et al., 2017) is an alternative objective for semantic similarity learning; when cosine similarity is used as the scoring function, it is equivalent to the contrastive objective in Gao et al. (2021). For text classification tasks, Sun et al. (2019) proposed additional pre-training with target-domain data; for semantic similarity prediction, cosine-similarity fine-tuning (Reimers and Gurevych, 2019) was proposed on STS datasets. To explicitly deal with multi-task learning, methods like Gradient Surgery (Yu et al., 2020) are proposed to resolve conflicting gradient updates for different tasks.

4 Approach

4.1 Contrastive Pre-Training

In a nutshell, my approach fine-tunes BERT embeddings in combination with task-specific supervised training, mainly by optimizing against this simple contrastive learning objective:

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

for positive pairs $\left(x_i, x_i^+\right)$ in a mini-batch of N pairs, where $\mathbf{h}_i$ and $\mathbf{h}_i^+$ denote the representations of x_i and x_i^+, τ is a temperature hyperparameter, and $sim\left(\mathbf{h}_1, \mathbf{h}_2\right)$ is the cosine similarity $\frac{\mathbf{h}_1^T \mathbf{h}_2}{\|\mathbf{h}_1\| \cdot \|\mathbf{h}_2\|}$.

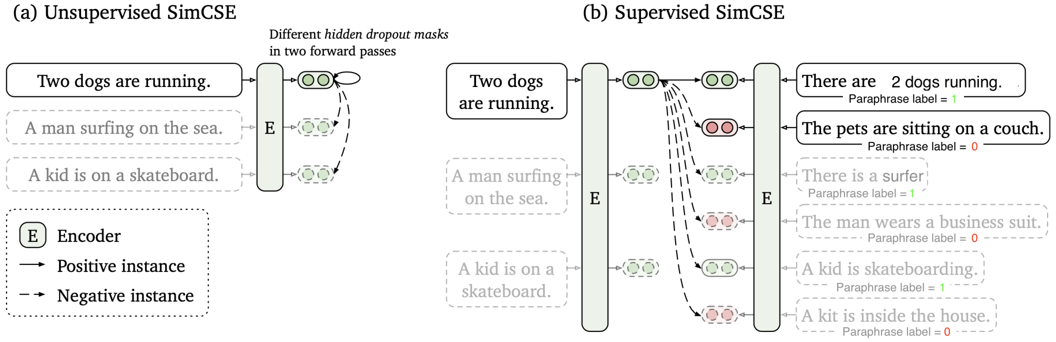

Figure 1: Pair construction in the Quora dataset. Adapted from Gao et al. (2021)

An important decision lies in how the positive pairs are chosen. In the unsupervised approach, standard dropouts are applied in the transformer layers of BERT, and the positive pairs are merely the final BERT embeddings (i.e. hidden state of the [CLS] token) of the same sentence, but from two different passes.

In the supervised approach, sentence pairs come from the Quora dataset where pairs that are paraphrases (with label 1) are taken as positive pairs. Sentence pairs that are not paraphrases (with label 0) are taken as negative pairs, whose contrastive loss is maximized instead. See Figure 1 for an intuitive illustration. In each epoch, this is followed by unsupervised training on the other 2 datasets sequentially, since there is no such structure in them (pairs with label 0 or 5 are relatively few in the STS dataset). For comparison, a NLI dataset (SNLI and MNLI combined) are also included for supervised contrastive pre-training as in the original SimCSE paper, either before, after, or in place of the contrastive pre-training on the 3 default datasets.

Inspired by SimCLR (Chen et al., 2020), a small (2-layer) MLP network is used on top of the CLS embeddings for contrastive training and thrown away in later predictions.

4.2 Task-Specific Supervised Fine-Tuning

Instead of using the CLS embeddings directly, a small (2-layer) MLP network with dropout is added in the prediction head for each of the 3 tasks. Semantic textual similarity is computed as the cosine similarity between the 2 transformed embeddings; paraphrase detection is done with a bilinear function between the two; the first layer of the MLP networks is shared between STS and paraphrase detection heads. Other 2 settings are also experimented with: (1) prediction heads without additional layers, and (2) completely shared heads between STS and paraphrase predictions.

Another important decision concerns how the contrastive training is combined with task-specific supervised training. In the early stage, only sentiment classification on the STS dataset is used as the supervision. The following 3 settings are experimented with: (1) first task-specific supervised training and then contrastive training; (2) first contrastive training and then task-specific supervised training; (3) fusion training, where the same batch of STS data is trained first against the classification objective (Cross Entropy Loss), and then against the contrastive objective.

In the final model, a multi-task training procedure is performed as follows. In each iteration, a batch is sampled from each of the 3 datasets, and the 3 tasks are trained together by optimizing the average loss of the three. For balance among the 3 tasks, the SST and the Quora datasets are proportionally down-sampled to the same length as the STS dataset (which is the smallest in size). For comparison, the following settings are also explored: (1) batch round robin, where in each iteration the model is optimized against the single task objective on a batch from each dataset sequentially; (2) epoch round robin, where in each epoch the model is trained on each entire dataset sequentially; (3) gradient surgery (Yu et al., 2020), in order to combine the 3 single-task losses; (4) single-task training with a contrastively pre-trained model. Finally, a model ensemble is formed from the models giving the best single-task performance.

4.3 Baselines

As the baseline for unsupervised methods, the minBERT model trained on the SST dataset from the first part is used directly for the multi-task evaluation. Both paraphrase detection and semantic textual similarity are predicted with a cosine similarity function between the final BERT embeddings of the sentence pairs (i.e. hidden state of the [CLS] tokens), without introducing additional layers or parameters.

The baseline for supervised methods includes additional MLP layers (described above) for the prediction heads and is directly fine-tuned against the task-specific objectives, without contrastive pre-training.

4.4 Attributions

The above learning objective is the same as in the original SimCSE paper Gao et al. (2021), whereas sentence pair construction from the Quora dataset is my original. Code for gradient surgery is taken off-the-shelf from https://github.com/WeiChengTseng/Pytorch-PCGrad but does not contribute to the final results (see Section 6); code for contrastive pre-training on the NLI dataset is adapted from https://github.com/princeton-nlp/SimCSE; all other code besides the starter code, including added layers of the model, task-specific fine-tuning, and the contrastive pre-training procedure is implemented by myself.

5 Experiments

5.1 Data

The 3 given datasets are used for unsupervised contrastive training and task-specific supervised training. Quora dataset and an additional NLI dataset (SNLI and MNLI combined) are used for supervised contrastive training, since the best results from the original SimCSE paper were trained on these NLI datasets. The combined NLI dataset has 276k samples, where each sample consists of a positive sentence pair plus a hard negative. The positive pairs are sentence pairs with the label "entailment" in the original NLI datasets, whereas the negative pairs are labeled as "contradiction". They are already formatted for contrastive training and made available by the original authors at https://huggingface.co/datasets/princeton-nlp/datasets-for-simcse/resolve/main/nli_for_simcse.csv.

5.2 Evaluation method

This project is evaluated on the 3 pre-defined default tasks.

5.3 Experimental details

All reported models are trained with a default learning rate of 1e-05, a default temperature parameter of 0.05, and a default batch size of 16. Different batch sizes are also explored at 32 and 64.

In unsupervised settings, the model is trained on first SST data, then Quora data, and finally SemEval STS data for 10 epochs. In each epoch, 6% of the Quora training data is randomly sampled to achieve a balance between the 3 tasks while also improving training efficiency. The model performance is evaluated on the dev split after each epoch, and the parameters giving the best dev accuracy are saved as the final models.

In supervised settings, contrastive pre-training is performed on the 3 datasets for 5 epochs and on the combined NLI dataset for 3 epochs. Fewer epochs are also experimented with. Different pre-training learning rates are also explored at 2e-5 and 3e-5. Task-specific training is conducted for 10 epochs. The MLP heads have a hidden dimension of 768 and dropout probability of 0.3 at each linear layer.

5.4 Results

5.4.1 Unsupervised Methods

The results for unsupervised methods on the dev sets are summarized in Table 1. Fusion training, where the same batch of STS data is trained first against the classification objective and then against the contrastive objective, leads to the best overall score.

Table 1: Results for Unsupervised Methods

Method	Overall	Sentiment Cls. Acc.	Paraphrase Acc.	STS Corr.
Vanilla BERT (Baseline)	0.391	**0.516**	0.388	0.270
Contrastive First	0.449	0.479	0.401	0.466
Supervised First	0.502	0.422	0.402	0.682
Fusion	**0.520**	0.465	**0.410**	**0.685**

5.4.2 Supervised Methods

The main results for supervised settings are summarized in Table 2 (with best results before ensemble in bold). See Section 6 for full results given by models with different design choices. Note that the *single task* results come from 3 different models, all contrastively pre-trained with the same procedure (i.e. on the 3 given datasets), but each fine-tuned against the single-task objective. Notably, it takes all major models only a few fine-tuning epochs to reach the best dev accuracy (see Figure 2 for the comparison). The final model produces the following results on the test sets: sentiment classification accuracy of 0.521, paraphrase detection accuracy of 0.830, semantic textual similarity of 0.753, and overall accuracy of 0.701.

Table 2: Results for Supervised Methods

Pre-Training Data	Overall	Sentiment Cls. Acc.	Paraphrase Acc.	STS Corr.
None (Baseline)	0.540	0.490	0.726	0.404
NLI	0.661	0.500	0.746	**0.735**
3 Given	**0.664**	**0.516**	0.815	0.661
3 Given (*Single Task*)	/	0.502	**0.836**	0.642
Ensemble (dev)	0.706	0.521	0.836	0.761
Ensemble (test)	0.701	0.521	0.830	0.753

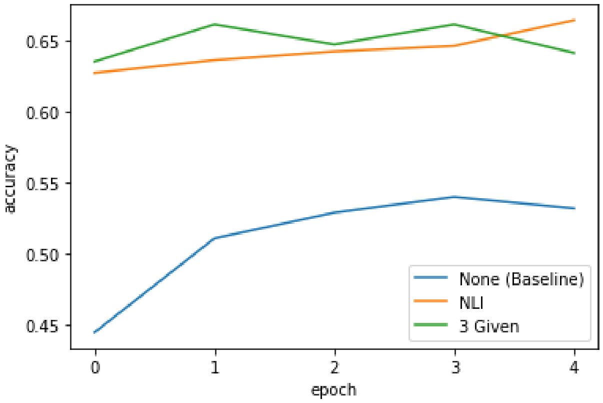

Figure 2: Overall dev Accuracy of Supervised Models

6 Analysis

Thorough ablation studies are conducted to scrutinize each of the following design choices. Note that all comparisons in every single table referenced below differ only by the highlighted characteristics; all other omitted aspects are kept constant.

6.1 MLP Networks in Prediction Heads

Adding MLP layers on top of prediction heads boosts model performance as compared to the vanilla BERT model (Table 3). Sharing the first linear layer leads between paraphrase detection and STS heads to slightly better results than having one distinct MLP network for each task, which works better than sharing the whole prediction head among the two. This is expected, since the 3 semantically different tasks benefit from different transformations of the same pre-trained sentence embeddings, while paraphrase detection and STS tasks are more similar in nature.

Table 3: Effect of MLP Networks in Prediction Heads

Settings	Overall	Sentiment Cls. Acc.	Paraphrase Acc.	STS Corr.
All Different	0.530	0.496	0.732	0.361
Shared Heads	0.494	0.497	0.625	0.361
Shared 1st Linear Layer	0.539	0.514	0.734	0.370

6.2 Unsupervised vs. Supervised Contrastive Training

Overall, supervised contrastive training produces better results than the unsupervised counterpart. This is consistent with the original SimCSE paper and expected, as both the Quora and the NLI datasets provide high-quality supervision for learning semantic similarity and distinguishing dissimilar sentence pairs. Whereas in unsupervised settings, the model is learning to represent a single sentence at a time, only with other random sentences as negative examples. On the other hand, contrastive pre-training on all 3 datasets works better than supervised training on the Quora dataset alone (Table 4), even though the other two are only trained on in an unsupervised way. This is likely because exposure to in-domain samples benefits the downstream task-specific training with the same data.

Table 4: Effect of Negative Pairs and More Data

Settings	Overall	Sentiment Cls. Acc.	Paraphrase Acc.	STS Corr.
Quora Only (No Negative Pairs)	0.620	0.387	0.730	0.743
Quora Only	0.623	0.521	0.781	0.565
All 3 Given	0.651	0.488	0.784	0.683

6.3 Contrastive Training Datasets and Negative Pairs

For supervised contrastive pre-training, there is no substantial difference in model performance between Quora and NLI datasets (Table 5). This disagrees with the results in the original SimCSE paper where NLI datasets lead to better results. This may be explained by the relevance of the Quora dataset and hence effectively in-domain training, as it is directly connected with one of the tasks. Furthermore, pre-training with both NLI and the 3 given datasets sequentially turns out to hurt performance, especially when the NLI dataset are trained after. This might be caused by the model overfitting to the pre-training data.

Table 5: Effect of Different Datasets for Contrastive Pre-Training

Pre-Training Data	Overall	Sentiment Cls. Acc.	Paraphrase Acc.	STS Corr.
NLI before 3 Given	0.628	0.476	0.794	0.614
3 Given before NLI	0.604	0.431	0.790	0.592
NLI	0.661	0.500	0.746	0.735
3 Given	0.664	0.516	0.815	0.661

Adding sentence pairs that are not paraphrase in the Quora dataset slightly as negative pairs improves model performance (Table 4), presumably because they constitute better negative pairs than random sentences in the batch, but not explicitly contradictory to one another like NLI negative pairs.

6.4 Multi-Task Objective vs. Single-Task Training

Multi-task training, with an average loss of the 3 single-task losses, proves to work better than both round robin training (Table 6) and gradient surgery (Table 7). This partially makes sense, as the primary objective against which the model is optimized is the average of the 3 single-task objectives.

Table 6: Effect of the Multi-Task Objective

Method	Overall	Sentiment Cls. Acc.	Paraphrase Acc.	STS Corr.
Epoch Round Robin	0.493	0.507	0.519	0.453
Batch Round Robin	0.491	0.510	0.520	0.442
Multi-Task Average Loss	0.540	0.490	0.726	0.404

Table 7: Effect of Gradient Surgery

Objective	Overall	Sentiment Cls. Acc.	Paraphrase Acc.	STS Corr.
Gradient Surgery	0.625	0.480	0.800	0.596
Multi-Task Average Loss	0.628	0.476	0.794	0.614

Surprisingly, fine-tuning the contrastively pre-trained model on a single dataset does not always lead to better accuracy in the corresponding task (Table 2). In fact, only the paraphrase detection task benefits from this single-task training. This is likely due to the relatively large size of the Quora dataset, so the other 2 tasks benefit from the richer semantic representations trained jointly with all 3 tasks. This is especially the case for the STS task, since it is semantically similar to the paraphrase detection task, yet the SemEval STS dataset is comparably small.

6.5 Hyperparameters

Table 8: Effect of Longer Contrastive Pre-Training Epochs on NLI Dataset

# of Epochs	Overall	Sentiment Cls. Acc.	Paraphrase Acc.	STS Corr.
1	0.630	0.500	0.730	0.658
3	0.661	0.500	0.746	0.735

The model ensemble combines the best-performing models on each single task and give the best results in all 3 tasks as expected (Table 2). Longer epoches of contrastive pre-training does lead to slightly better performance, for both NLI (Table 8) and the 3 given datasets (Table 9). The model performance seems robust to the choice of batch size. Different learning rates do result in slightly different performance (Table 10), and moderately larger learning rates tend to make the model training converge faster.

7 Conclusion

In summary, contrastive pre-training indeed benefits all 3 downstream tasks. Supervised pre-training works better than unsupervised pre-training as expected. Either NLI or Quora dataset provides effective supervision, but combining both sequentially would hurt performance. Pre-training on all 3 given datasets works better than supervised pre-training on Quora dataset alone. Including negative pairs only slightly boosts model performance. This proves contrastive learning a powerful framework across domains, as it allows flexible construction of instance pairs from various kinds of datasets.

For task-specific fine-tuning, small MLP networks with dropout on top of the CLS embeddings works well as prediction heads. Sharing early MLP layers between similar tasks also helps. Multi-task training, with an average loss of the 3 single-task losses, turns out better than alternative approachs such as round robin training, gradient surgery, and even single-task training in most cases. Finally, the

Table 9: Effect of Longer Contrastive Pre-Training Epochs on 3 Given Datasets

# of Epochs	Overall	Sentiment Cls. Acc.	Paraphrase Acc.	STS Corr.
2	0.591	0.406	0.697	0.671
3	0.610	0.360	0.724	0.748

Table 10: Effect of Different Learning Rates for Contrastive Pre-Training

Learning Rate	Overall	Sentiment Cls. Acc.	Paraphrase Acc.	STS Corr.
1e-5	0.637	0.497	0.806	0.609
2e-5	0.664	0.516	0.815	0.661
3e-5	0.658	0.496	0.799	0.680

model ensemble is able to combine the best single-task performance for all 3 tasks, with an overall accuracy of 0.701 on the test sets (sentiment classification accuracy: 0.521, paraphrase detection accuracy: 0.830, semantic textual similarity: 0.753).

There is still plenty of room for future work. (1) Different regularization methods can be explored to avoid the models from overfitting too soon. (2) Given enough computing budget, a more thorough hyperparameter search can be performed to further boost model performance. (3) In the fine-tuning phase, explicit learning rate schedule can also be utilized. (4) Time permitting, alignment and uniformity metrics (Wang and Isola, 2020) can be computed to systematically examine the quality of sentence embeddings produced by contrastive pre-training. (5) Lastly, more effort should be spent on understanding and improving on the STS task, since it has a larger gap with the best results on the leaderboards.

References

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR.

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

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

Matthew Henderson, Rami Al-Rfou, Brian Strope, Yun-Hsuan Sung, László Lukács, Ruiqi Guo, Sanjiv Kumar, Balint Miklos, and Ray Kurzweil. 2017. Efficient natural language response suggestion for smart reply. *arXiv preprint arXiv:1705.00652*.

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

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

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to fine-tune bert for text classification? In *Chinese Computational Linguistics: 18th China National Conference, CCL 2019, Kunming, China, October 18–20, 2019, Proceedings 18*, pages 194–206. Springer.

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

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. 2020. Gradient surgery for multi-task learning. *Advances in Neural Information Processing Systems*, 33:5824–5836.