

# SentiCSE: A Sentiment-aware Contrastive Sentence Embedding Framework with Sentiment-guided Textual Similarity

Jaemin Kim<sup>1,2\*</sup>, Yohan Na<sup>1\*</sup>, Kangmin Kim<sup>1,3</sup>, Sang Rak Lee<sup>1</sup>, Dong-Kyu Chae<sup>1†</sup>

<sup>1</sup>Hanyang University, <sup>2</sup>LG Electronics, <sup>3</sup>BHSN  
{jaemink, nayohan, kevin7133, sangrak, dongkyu}@hanyang.ac.kr

## Abstract

Recently, sentiment-aware pre-trained language models (PLMs) demonstrate impressive results in downstream sentiment analysis tasks. However, they neglect to evaluate the quality of their constructed sentiment representations; they just focus on improving the fine-tuning performance, which overshadows the representation quality. We argue that without guaranteeing the representation quality, their downstream performance can be highly dependent on the supervision of the fine-tuning data rather than representation quality. This problem would make them difficult to foray into other sentiment-related domains, especially where labeled data is scarce. We first propose *Sentiment-guided Textual Similarity* (SgTS), a novel metric for evaluating the quality of sentiment representations, which is designed based on the degree of equivalence in sentiment polarity between two sentences. We then propose SentiCSE, a novel Sentiment-aware Contrastive Sentence Embedding framework for constructing sentiment representations via combined word-level and sentence-level objectives, whose quality is guaranteed by SgTS. Qualitative and quantitative comparison with the previous sentiment-aware PLMs shows the superiority of our work. Our code is available at: <https://github.com/nayohan/SentiCSE>

**Keywords:** Sentiment analysis, sentiment-aware PLMs, representation learning

## 1. Introduction

*Sentiment analysis* has been attracting great attention from researchers in not only the field of natural language processing (NLP) but also various other domains such as recommender systems, management information systems, and social media analysis. Sentiment analysis can play a critical role in the decision-making process by obtaining public opinion about specific products, services, or social issues (Zhang et al., 2018; Qi et al., 2015). In particular, more and more people are actively expressing their opinions online these days, which makes sentiment analysis even more important (Yadav and Vishwakarma, 2020).

Recently, pre-trained language models (PLMs) such as BERT (Devlin et al., 2019) and GPT (Radford et al., 2018) have accomplished remarkable performance in various NLP tasks by capturing semantic information and constructing contextualized representations. Under this guiding trend, a number of researches on sentiment analysis employed the PLMs as a backbone and made the sentiment-related applications as a downstream task. However, directly employing the PLMs has a limitation in sentiment analysis because there is a significant difference between *textual similarity*

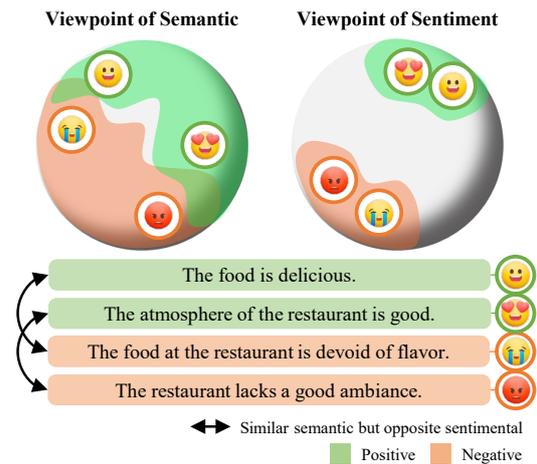


Figure 1: The difference between sentence embedding from a semantic perspective and a sentiment perspective, which shows the necessity of focusing on embedding methods from a sentiment perspective for sentiment analysis, instead of the traditional semantic perspective.

*in terms of semantics and textual similarity from the viewpoint of sentiment.* For example, as illustrated in Figure 1, the two sentences “The food is delicious.” and “The atmosphere of the restaurant is good.” are very different in terms of semantics, but are similar with regards to sentiment.

For further improvement, recent studies investigated the approaches toward building *sentiment-aware* PLMs: They introduced sentiment information in words, phrases, and sentences and designed sentiment-related pre-training objectives into PLMs. Notable examples include phrase-level

\*Co-first authors.

<sup>2,3</sup>This work was done while the authors were at Hanyang University. Currently, Jaemin Kim and Kangmin Kim are working as NLP research engineers at LG Electronics and BHSN respectively.

†Corresponding author.

polarity prediction adopted by SentiBERT (Yin et al., 2020), combined word-level and sentence-level polarity prediction proposed by SentiLARE (Ke et al., 2020), word-level and sentence-level polarity and emoticon prediction performed by SentiX (Zhou et al., 2020), and prediction of the replaced sentiment words and sentence-level contrastive learning in SentiWSP (Fan et al., 2022).

Even though they have achieved great performance in their downstream sentiment analysis tasks, quantitative evaluation of the quality of their constructed sentiment representations has been largely neglected. We argue that the quality of sentiment representations should be guaranteed because, without it, the low-quality representations of a backbone model may lead to the risk of poor *few-shot* performance when applied to other sentiment-related domains with limited labeled data (Tian et al., 2020b; Rizve et al., 2021). In light of the data scarcity in many real-world scenarios (Gunel et al., 2021), it is crucial to consider few-shot learning rather than just focusing on fine-tuning performance, which would be mainly due to the supervision of sufficient fine-tuning labels.

This paper presents a novel sentiment-aware pre-training framework that ensures high-quality sentiment representations. First, we propose **Sentiment-guided Textual Similarity (SgTS)**, a novel metric that quantitatively evaluates the quality of sentiment representations by measuring the extent of equivalence in sentiment polarity between two provided sentences. We then propose **SentiCSE**: Sentiment-aware Contrastive Sentence Embedding, which is highly focused on constructing sentiment representations by utilizing sentiment-related linguistic knowledge, towards achieving better performance on downstream sentiment-related tasks in label-scarce scenarios. SentiCSE has the following two pre-training objectives. First, the word-level objective aims to reconstruct sentences where each predicted word has the same sentiment polarity with the masked token, towards a better understanding of sentiment in text rather than semantics. Second, the sentence-level objective utilizes a quadruple of sentences (two positives and two negatives), and puts it to the supervised contrastive learning objective, where the sentences with the same polarity attract each other, and those with different polarity, which can be considered as negatives, repel each other. Through our extensive experiments under both the full label (linear probing) and the few-shot settings, we confirm that SentiCSE not only achieves state-of-the-art performance but also exhibits the best quality of sentiment representations using much less data.

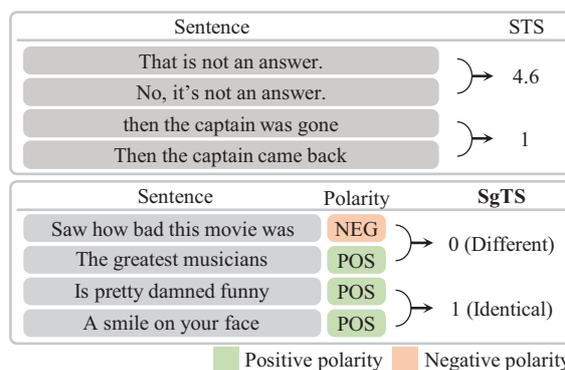


Figure 2: Comparison of STS and our SgTS. STS measures similarity of two sentences based on contextual semantics while SgTS judges similarity based on their sentiment polarities.

## 2. SgTS

To the best of our knowledge, there has been no method to quantitatively evaluate the quality of sentiment representations. Traditionally, *Semantic Textual Similarity (STS)* (Agirre et al., 2012, 2013, 2014, 2015, 2016; Cer et al., 2017; Marelli et al., 2014) has been a popular task that measures the degree of semantic equivalence between two sentences and has been employed by the original PLMs to measure the quality of their contextualized representations by using Spearman’s correlation (Reimers and Gurevych, 2019). However, directly applying STS to evaluate the quality of sentiment representations is not feasible because the textual similarity between two sentences in terms of semantics is quite different from the similarity in terms of sentiment.

To fill this research gap, we propose an SgTS (**Sentiment-guided Textual Similarity**) metric that can evaluate the quality of sentiment representations. We first design an SgTS task, inspired by the STS task, which aims to measure the sentiment similarity of two given sentences. Figure 2 illustrates our SgTS task. We assigned a ground-truth similarity score of 0 (different) or 1 (identical) to each pair of two sentences based on their sentiment polarities. In this way, we can automatically construct an SgTS benchmark (i.e., a corpus of sentence pairs with the similarity labels of 0 or 1) based on any dataset with the sentiment labels on sentences. Finally, the quality of sentiment representation is measured by the Spearman correlation between the ground-truth similarity scores (0 or 1) and the predicted scores computed by cosine similarity between two given sentence embeddings.

## 3. SentiCSE

This section presents SentiCSE, an effective framework for learning high-quality sentiment represen-

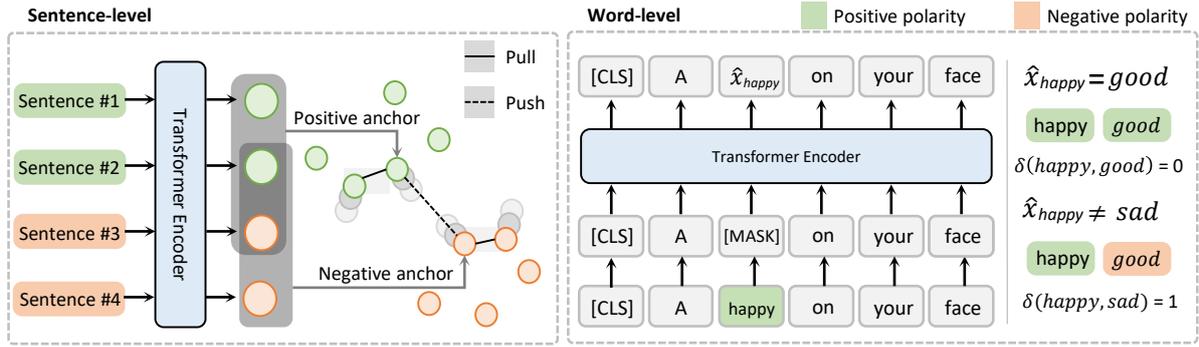


Figure 3: The overview of SentiCSE. In the sentence-level objective, we take two anchors, one for ‘positive’ polarity and the other for ‘negative’ polarity. We then encourage the sentiment representations to be more close to the corresponding sentences belonging to the same polarity, and to be far from the corresponding sentences associated with different polarities. In the word-level objective, our model tries to predict the masked words as in conventional MLM. But the error signal is dropped if the model matches the polarity of the predicted word with the masked word, even though the word prediction was not correct.

tations. It borrows the encoder of backbones such as BERT or RoBERTa. Its overall framework is illustrated in Figure 3. SentiCSE consists of two sentiment-aware pre-training objectives: the Word-level objective that recovers the sentiment polarity of masked tokens and the Sentence-level objective based on supervised contrastive learning. The main idea of our design is to allow SentiCSE to improve the quality of its sentiment representation by leveraging the characteristics of sentiment polarity.

### 3.1. Sentiment-aware Word-level Objective

Formally, let  $X = \{x_1, x_2, \dots, x_n\}$  be a text sequence of length  $n$ , where each  $x_i$  is a word in the Byte-Pair Encoding vocabulary (Radford et al., 2019) with the size of 50,265. In the original masked language model (MLM), a certain percentage of words in a given sentence is randomly masked and then predicted by the model (Devlin et al., 2019; Clark et al., 2020). In our work, we selectively mask the sentiment words included in the SentiWordNet lexicon (Baccianella et al., 2010). More specifically, within the word listed in SentiWordNet, we refer to its sentiment scores (i.e., *positivity* and *negativity*). According to the two scores, we acquire the word’s polarity based on the following polarity function:

$$S(x) = \begin{cases} +, & \text{if } S_{pos}(x) > 0 \text{ and } S_{neg}(x) == 0, \\ -, & \text{if } S_{pos}(x) == 0 \text{ and } S_{neg}(x) > 0, \\ multi, & \text{if } S_{pos}(x) > 0 \text{ and } S_{neg}(x) > 0, \end{cases} \quad (1)$$

where  $S_{pos}(x)$  and  $S_{neg}(x)$  indicate the positivity and negativity scores of the given word  $x$ , respectively. Among the words in a sentence, only the words having either ‘+’ (positive) or ‘-’ (negative)

polarity are candidate tokens for masking. We do not mask the words with ‘*multi*’ (multi-polarity) because such words are positive in some sentences and negative in some other sentences, which would make training unstable. This setting helps the training to focus more on sentiment-related information. We mask 10% of the candidate tokens; we empirically found that the masking ratio of 10% balances the trade-off between providing sufficient sentiment information to learn good representations and promoting training efficiency.

After generating a corrupted sentence, our model is trained to predict the masked words with the same sentiment as the original words while also predicting the masked words as original words. Formally, let  $\tilde{X} = \{\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n\}$  be the corrupted version of  $X$ . The encoder encodes  $\tilde{X}$  to contextualized representations  $H = \{h_1, h_2, \dots, h_n\}$ . Let  $k$  denote the index of the masked words; then each  $h_k$  is fed to a SoftMax layer:  $P_\theta(m_k|M) = \text{Softmax}(W \cdot h_k + b)$  where  $\theta$  denotes the model parameters.

After then, we replace the masked word  $\tilde{x}_k$  with the predicted word  $\hat{x}_k$  that has the highest softmax probability. We then introduce an indicator function  $\delta(x_k, \hat{x}_k)$  that outputs 0 if the polarities of the two given words  $x_k$  and  $\hat{x}_k$  are identical, and 1 if they are different, based on SentiWordNet:

$$\delta(x_k, \hat{x}_k) = \begin{cases} 0, & \text{if } S(x_k) == S(\hat{x}_k) \\ 1, & \text{otherwise.} \end{cases} \quad (2)$$

This indicator function is applied to the cross-entropy loss for a single sentence  $X$  is:

$$\mathcal{L}_w^X = \sum_{k \in \mathcal{I}} -\log P_\theta(x_k|\tilde{X}) \cdot \delta(x_k, \hat{x}_k), \quad (3)$$

where  $\mathcal{I}$  denotes the set of indices of the masked

words. Then the training objective for a batch is:

$$\mathcal{L}_w = \sum_{i=1}^N \mathcal{L}_w^{X_i}, \quad (4)$$

where  $N$  is the batch size.

By doing so, the loss caused by word prediction that matches the polarity of the original word is removed, even if the words themselves are not matched. This objective makes the model focus more on understanding linguistic knowledge in terms of sentiment.

### 3.2. Sentiment-aware Sentence-level Objective

Here, we adopt the contrastive learning framework demonstrated in SimCSE (Gao et al., 2021a). Especially, we choose *supervised* contrastive learning because it can explicitly define which examples should be attracted or should be repelled, leveraging negatives. We believe that the sentiment analysis task benefits from negatives, which can be easily chosen based on the sentiment polarities of sentences.

Formally, we compose a contrastive example with a quadruple of sentences  $q_i : (p_i, p_i^+, n_i, n_i^+)$ , where  $p_i$  and  $p_i^+$  are two different positive sentences and  $n_i$  and  $n_i^+$  are two different negative sentences. We then define two objectives,  $\mathcal{L}_{pos}^{q_i}$  and  $\mathcal{L}_{neg}^{q_i}$ , each pulling the given two positive sentences and pushing the two negative sentences close to each other, respectively:

$$-\log \frac{e^{sim(p_i, p_i^+)/\tau}}{e^{sim(p_i, p_i^+)/\tau} + \sum_{j=1}^N (\alpha \cdot e^{sim(p_i, n_j)/\tau})}, \quad (5)$$

$$-\log \frac{e^{sim(n_i, n_i^+)/\tau}}{e^{sim(n_i, n_i^+)/\tau} + \sum_{j=1}^N (\alpha \cdot e^{sim(n_i, p_j^+)/\tau})}, \quad (6)$$

where  $sim(\cdot)$  computes the cosine similarity between the embeddings of two given sentences,  $\alpha$  indicates the weight on negative samples, and  $\tau$  stands for temperature. The hyperparameters  $\alpha$  and  $\tau$  control the effectiveness of negative samples. Our sentence-level objective for a batch  $\mathcal{L}_s$  simply combines the two above losses by:

$$\mathcal{L}_s = \frac{1}{N} \sum_{i=1}^N (\mathcal{L}_{pos}^{q_i} + \mathcal{L}_{neg}^{q_i}). \quad (7)$$

With this objective, SentiCSE strives to learn high-quality representations by pulling neighbors sharing similar sentiment together and pushing apart non-neighbors with different sentiment.

The final objective function of our SentiCSE can be denoted as:

$$\mathcal{L}_{SentiCSE} = \lambda_w \cdot \mathcal{L}_w + \mathcal{L}_s, \quad (8)$$

| Dataset         | IMDB   | SST-2  | Yelp-2  | Amazon  | MR    |
|-----------------|--------|--------|---------|---------|-------|
| Avg. length     | 302    | 14     | 176     | 42      | 27    |
| % of SentiWords | 6.3%   | 8.3%   | 5.1%    | 6.0%    | 9.9%  |
| # train         | 25,000 | 67,349 | 560,000 | 160,000 | 8,530 |
| # valid         | -      | 872    | -       | 4,000   | 1,066 |
| # test          | 25,000 | 1,821  | 38,000  | 4,000   | 1,066 |

Table 1: Data statistics.

| Model     | Backbone | Pre-trained dataset |        |      |     |    | # Sentences |
|-----------|----------|---------------------|--------|------|-----|----|-------------|
|           |          | Wiki                | Amazon | Yelp | SST | MR |             |
| SentiBERT | BERT     |                     |        |      | ✓   |    | 0.067M      |
| SentiX    | BERT     |                     | ✓      |      | ✓   |    | 240M        |
| SentiLARE | RoBERTa  |                     |        | ✓    |     |    | 6.7M        |
| SentiWSP  | ELECTRA  | ✓                   |        |      |     |    | 0.5M        |
| SentiCSE  | RoBERTa  |                     |        |      |     | ✓  | 0.008M      |

Table 2: The default pre-training corpora with their number of sentences and backbones adopted by each method.

where  $\lambda_w$  controls the importance of the word-level loss. We empirically choose the values of  $\lambda_w$  as 0.15. Guided by the combined word-level and sentence-level objectives, SentiCSE learns representations that are more reflective of the sentiment polarity of the sentences.

## 4. Experimental Settings

We conducted extensive experiments to evaluate our SentiCSE. This section summarizes the implementation details of our experiments.

### 4.1. Dataset

We employed the following four datasets that include sufficient sentiment words: **IMDB** (Maas et al., 2011), **SST-2** (Socher et al., 2013), **Yelp-2** (Zhang et al., 2015), **Amazon** (Ni et al., 2019), and **MR** (Pang and Lee, 2005). All the datasets include review sentences collected from various domains (movie, restaurant, dental clinic, etc.), whose sentiment labels are given by human experts or automatically judged based on the corresponding review scores. Table 1 shows their statistics.

### 4.2. Baselines

We employed the four sentiment-aware PLMs including **SentiBERT** (Yin et al., 2020), **SentiX** (Zhou et al., 2020), **SentiLARE** (Ke et al., 2020), and **SentiWSP**<sup>1</sup> (Fan et al., 2022). We also employed supervised **SimCSE** (Gao et al., 2021a), **BERT** (Devlin et al., 2019), and **RoBERTa** (Liu et al., 2019) as

<sup>1</sup>Actually, SentiWSP adopts quite different configurations from the other baselines, such as the use of ELECTRA, Wiki dataset, and the unsupervised contrastive learning. And its performance is not on par with the other baselines (SgTS: Avg. 0.02, linear probing: Avg. 85.76, 1-shot: Avg. 56.23, 5-shot: Avg. 63.67). Hence, it is not compared in most of our comparative results.

baselines. The pre-training corpora and the backbones adopted in each baseline can be found in Table 2. BERT and RoBERTa were pre-trained on the Wiki corpus (Devlin et al., 2019) same with SentiWSP; supervised SimCSE was trained on the NLI dataset (Bowman et al., 2015; Williams et al., 2018) transformed for contrastive learning.

For a fair comparison, there can be two perspectives on what should be selected as the pretraining data and backbone model for each compared method. The first idea is to follow each baseline’s own configuration; more specifically, we use the pretraining dataset and backbone model chosen by the authors of the respective baseline. This protocol has been adopted by most of the baseline papers (SentX, SentiLARE, and SentiWSP), in the belief that it would be inappropriate to use a different backbone model or pretraining corpus which were not selected by the authors of each baseline. In contrast, the second perspective is that a fair evaluation requires the use of the same pretraining data and backbone models. We try our best to conduct the comparative experiments by following both two perspectives, though the second perspective is harder to implement because the baselines generally adopt different pretraining datasets and backbones with each other.

### 4.3. Pre-training Environment

We briefly explain how we pre-trained SentiCSE. We chose MR as the default pretraining dataset, but other dataset can be used. For each positive sentence contained in a training set, we composed a quadruple of sentences by randomly sampling another positive sentence and two negative sentences from the training set (in the case where MR is adopted as a pretraining dataset, 4,265 quadruples will be generated). For the hyperparameters, we fixed the maximum sentence length as 128, the number of layers (Transformer blocks) as 12, and the embedding dimension as 768. The learning rate and the batch size were set to 1e-5 and 64, respectively. Under this setting, if we use MR, the pre-training takes only 3 hours, which proceeds 20,000 steps with four NVIDIA A30 GPUs. For the validation step, we constructed the SgTS benchmark (if MR is used, the SgTS-MR benchmark is composed of 429 sentence pairs with 0 or 1 labels, based on the 872 sentences included in MR’s validation set). For every 500 step, we evaluated the SgTS performance to get the best checkpoint (In the case where MR is used, we obtained the best checkpoint at 14,000 step).

### 4.4. Details in Downstream Tasks

To evaluate the downstream performance of SentiCSE, we employed the linear probing and few-

| Model                       | SgTS        |             |             |             |             |             |
|-----------------------------|-------------|-------------|-------------|-------------|-------------|-------------|
|                             | IMDB        | SST-2       | Yelp-2      | Amazon      | MR          | Avg.        |
| BERT                        | 0.01        | 0.08        | 0.09        | 0.15        | 0.07        | 0.06        |
| SimCSE <sub>BERT</sub>      | 0.16        | 0.13        | 0.24        | 0.19        | 0.13        | 0.18        |
| SentiBERT                   | 0.13        | 0.17*       | 0.12        | 0.09        | 0.18        | 0.14        |
| SentiX                      | 0.62        | 0.48        | 0.77*       | 0.52*       | <b>0.39</b> | 0.56        |
| SentiCSE <sub>BERT</sub>    | <b>0.64</b> | <b>0.72</b> | <b>0.76</b> | <b>0.37</b> | 0.63*       | <b>0.62</b> |
| Roberta                     | 0.06        | 0.05        | 0.06        | 0.02        | 0.04        | 0.06        |
| SimCSE <sub>RoBERTa</sub>   | 0.21        | 0.11        | 0.26        | 0.20        | 0.19        | 0.19        |
| SentiLARE                   | 0.48        | 0.38        | 0.65*       | 0.36        | <b>0.57</b> | 0.46        |
| SentiCSE <sub>RoBERTa</sub> | <b>0.77</b> | <b>0.72</b> | <b>0.82</b> | <b>0.56</b> | 0.69*       | <b>0.71</b> |

Table 3: Representation quality measured by our SgTS task. We employ Spearman’s correlation with the “all” setting. We divided the compared methods into two groups: the BERT-based models and RoBERTa-based ones. The highest scores for each data are highlighted in bold face. For each data column, the scores with \* indicate the results when the same dataset is used for pretraining and a downstream task.

shot settings. Linear probing only trains the linear classification layer on top of the fixed input representations. We followed the default configurations for linear probing, which can be found in the SentiEval toolkit (Conneau and Kiela, 2018). Since the configurations for IMDB and Yelp-2 are not available, we borrowed the settings for SST-2. Since IMDB and Yelp-2 do not provide validation data, we randomly sampled 10% of their training sets to construct the validation sets.

In contrast to linear probing supervised by the entire training sentences, the few-shot setting assumes a data-scarce situation. This setting limits the number of accessible samples for each class (positive or negative) to  $K$ . For example, in the 5-shot setting, we randomly sampled 10 sentences (5 for each class) without replacement. Among the rest training data, we randomly sampled 500 examples for the validation, which was conducted every 20 steps with the early stop patience value as 5. Since the number of training examples is extremely small, the random seed may have a lot of influence on performance. To mitigate this issue, we repeated 10 experiments with 10 random seeds and then reported the average of the results (Günel et al., 2021). We tested SentiCSE on the 1-shot and 5-shot settings.

## 5. Results and Analyses

This section reports the results of our extensive experiments.

### 5.1. Quality of Representations (SgTS)

First, we qualitatively and quantitatively evaluate the quality of sentiment representations of each model. Figure 4 compares each model’s visualization of sentiment representation on each downstream dataset: for each data, we let each

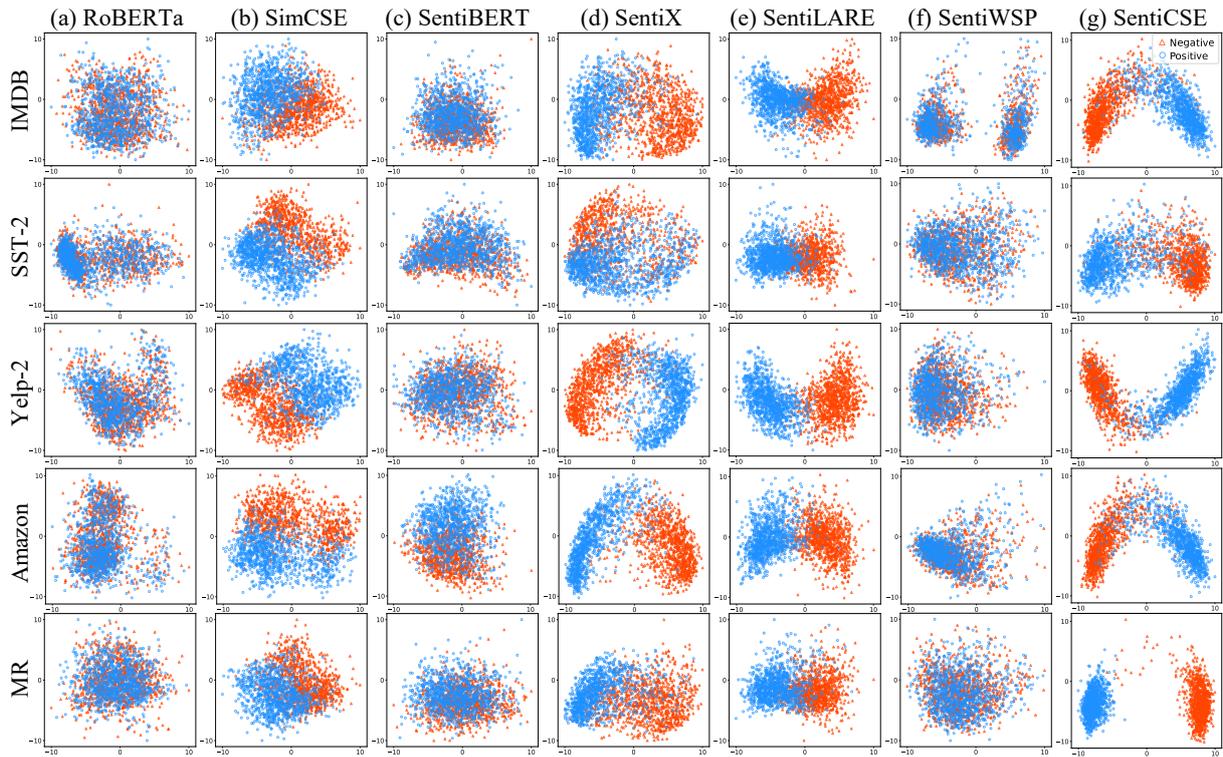


Figure 4: Visualized embeddings of the sentences included in each dataset. Each model is pre-trained on its own pretraining data, specified in Table 2. PCA is used for visualization.

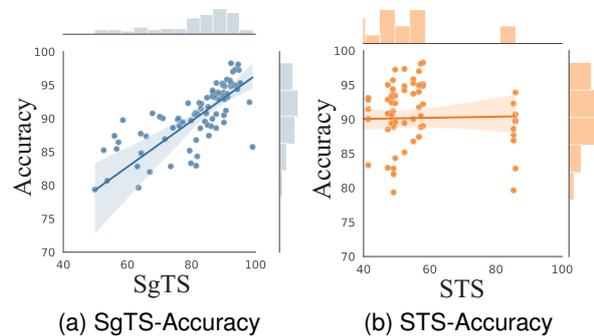


Figure 5: (a): Correlation between our SgTS and few-shot accuracy. (b): Correlation between STS and few-shot accuracy. We can see there is a strong correlation between our SgTS and the accuracy of sentiment analysis ( $\rho = 0.7$  with  $p$ -value  $< 0.01$ ) while there does not seem to be a correlation between STS and accuracy.

model encode sentences in the dataset to 768-dimensional embedding vectors. We then projected them to 2D space via PCA (principal component analysis). We observed that SentiX, SentiLARE, and SentiCSE exhibit better quality of representations, which indicates their representations contain much more sentiment information than the other models (SentiWSP, RoBERTa and SimCSE) pre-trained on a dataset that is not related to sentiment analysis. We observed that our SentiCSE exhibits better representations than the baselines because

it makes the distance between the two clusters (positive sentence embeddings and negative sentence embeddings) far enough.

In addition, Table 3 reports the representation quality in terms of the proposed SgTS metric, measured by Spearman correlation between the Cosine similarity between two sentence embeddings and the corresponding ground-truth labels (0: 'different' or 1: 'identical'). We observed that our SentiCSE exhibits the highest value, which indicates that two sentences with the same sentiment polarity are close to each other in the embedding space, and far from each other if their polarities are different. We also observed that SentiX and SentiLARE exhibit relatively high scores on Yelp-2. This also shows the effectiveness of our SgTS as an evaluation method for the quality of sentiment representation, because Yelp-2 is the pretraining dataset of SentiX and SentiLARE. In addition, the sentiment-aware PLMs except our SentiCSE show relatively low score on MR and SST-2, where the average length of the sentences is shorter than that of the other dataset. This might imply that the previous sentiment-aware PLMs rely on sufficient semantic information for constructing sentiment representations.

Finally, Figure 5 clearly depicts the strong correlation between the few-shot accuracy (this will be reported in the next subsection) and our SgTS metric. We computed the Pearson Correlation Coeffi-

| Model                       | 1-shot accuracy |              |              |              |              | 5-shot accuracy |              |              |              |              |
|-----------------------------|-----------------|--------------|--------------|--------------|--------------|-----------------|--------------|--------------|--------------|--------------|
|                             | IMDB            | SST2         | Yelp-2       | Amazon       | MR           | IMDB            | SST2         | Yelp-2       | Amazon       | MR           |
| BERT                        | 52.08           | 50.26        | 56.76        | 52.98        | 52.24        | 54.02           | 54.26        | 62.64        | 58.10        | 54.38        |
| SimCSE <sub>BERT</sub>      | 54.08           | 61.74        | 66.20        | 60.92        | 61.64        | 71.26           | 66.82        | 81.58        | 73.58        | 67.16        |
| SentiBERT                   | 51.40           | 55.60*       | 59.64        | 54.90        | 54.88        | 57.76           | 64.84*       | 70.20        | 67.02        | 64.90        |
| SentiX                      | 74.64           | 64.96        | 87.66*       | 86.14*       | <b>65.06</b> | <b>83.68</b>    | 72.32        | 93.40*       | 92.32*       | <b>76.68</b> |
| SentiCSE <sub>BERT</sub>    | <b>76.08</b>    | <b>87.88</b> | <b>81.62</b> | <b>82.24</b> | 85.82*       | 81.84           | <b>93.26</b> | <b>87.64</b> | <b>84.82</b> | 86.14*       |
| RoBERTa                     | 52.00           | 54.54        | 56.56        | 52.84        | 53.82        | 60.30           | 49.80        | 72.42        | 64.58        | 56.78        |
| SimCSE <sub>RoBERTa</sub>   | 59.04           | 61.06        | 68.44        | 58.40        | 61.72        | 74.72           | 68.08        | 86.62        | 75.14        | 71.56        |
| SentiLARE                   | 70.20           | 74.26        | 87.00*       | 84.58        | <b>68.68</b> | 87.18           | 80.10        | 93.28*       | <b>91.06</b> | <b>82.34</b> |
| SentiCSE <sub>RoBERTa</sub> | <b>82.64</b>    | <b>92.92</b> | <b>89.72</b> | <b>89.04</b> | 87.38*       | <b>88.12</b>    | <b>94.50</b> | <b>92.08</b> | 90.40        | 88.00*       |

Table 4: Comparative results on the 1-shot and 5-shot settings on each downstream dataset. In this table, we divided the compared methods into two groups: the BERT-based models and RoBERTa-based ones. Then we separately compared them within each group. For each data column, the accuracy with \* indicate the results when the same dataset is used for pretraining and a downstream task. We do not highlight such cases in bold face. Instead, we highlight the best accuracy achieved by the models pre-trained on a different dataset. SentiCSE achieves an impressive few-shot performance in most cases, owing to its high-quality sentiment representations.

| Model                       | IMDB         | SST-2        | Yelp-2       | Amazon       | MR           |
|-----------------------------|--------------|--------------|--------------|--------------|--------------|
| BERT                        | 85.25        | 85.44        | 89.75        | 86.44        | 80.68        |
| SimCSE <sub>BERT</sub>      | 86.91        | 87.73        | 92.29        | 88.60        | 79.64        |
| SentiBERT                   | 87.40        | 90.25*       | 90.76        | 87.33        | 84.80        |
| SentiX                      | <b>94.20</b> | 89.45        | 97.33*       | 94.82*       | <b>85.18</b> |
| SentiCSE <sub>BERT</sub>    | 90.63        | <b>95.30</b> | <b>93.12</b> | <b>89.93</b> | 85.74*       |
| RoBERTa                     | 82.82        | 79.36        | 88.87        | 81.98        | 50.38        |
| SimCSE <sub>RoBERTa</sub>   | 90.73        | 89.68        | 93.89        | 89.82        | 82.83        |
| SentiLARE                   | <b>94.84</b> | 92.20        | 98.26*       | <b>95.10</b> | <b>89.02</b> |
| SentiCSE <sub>RoBERTa</sub> | 94.03        | <b>95.18</b> | <b>95.86</b> | 93.69        | 89.49*       |

Table 5: Comparison of linear probing performance (i.e., a full resource setting).

cient among the SgTS scores and the corresponding few-shot accuracy scores, and got  $\rho = 0.7$  with  $p$ -value  $< 0.01$ , which indicates that high-quality representation guaranteed by SgTS results in the high few-shot performance. Such strong relationship implies that our SgTS serves as a meaningful quantitative metric for measuring the quality of sentiment representation. In contrast, the relationship between the traditional STS and the accuracy of sentiment analysis does not appear to be significant. This result sheds a new light on how to pre-train a language model for better sentiment analysis.

## 5.2. Performance of Downstream Tasks

First, Table 4 reports the comparative results on the few-shot setting. We observed that SentiCSE achieved significant performance improvement. A notable point is that SentiCSE even outperforms SentiX and SentiLARE on the Amazon and Yelp-2 dataset, which are their pretraining datasets; it also beats SentiBERT by a wide margin on SST2, on which SentiBERT was pre-trained.

Next, Table 5 shows results on the linear probing setting that provides full resource. Again, SentiCSE exhibits impressive performance: SentiCSE outper-

forms SentiBERT on SST2 where SentiBERT was pre-trained. SentiCSE slightly underperforms SentiLARE on IMDB. SentiCSE also underperforms SentiLARE and SentiX on Yelp-2 and Amazon, however, considering that Yelp-2 and Amazon are their pretraining datasets, we believe that the results still support the effectiveness of our work.

Another noteworthy point is that the baselines were pre-trained on very large corpus such as Yelp, which involved significant computational resources and time. In contrast, SentiCSE was pre-trained on MR, which contains only 1.52% of the training data of Yelp-2. As a result, SentiCSE requires much less resources in terms of data, computation, and cost than the baselines for both pre-training and downstream tasks, but is able to learn high-quality sentiment representations and provides comparable downstream performance between them.

Finally, we try to compare SentiCSE with the baselines using the same pretraining dataset and backbone model. For this experiment, we implemented various versions of SentiCSE using different backbones (BERT and RoBERTa) and different pretraining dataset (SST2, Yelp2, and Amazon). Table 6 reports the results. Again, we confirm that SentiCSE generally outperforms the baselines, even though SentiCSE gives up its best configuration (MR and RoBERTa) and is adapted to each baseline’s best configuration.

## 5.3. Further Analysis

This subsection provides additional results from our ablation studies, hyperparameter analysis, and qualitative analysis.

First, we examined the impact of each component in SentiCSE. Table 7 shows the performance of each ablation model on linear probing. In the sentence-level objective, we observed that using

| Dataset | Model                  | 1-shot accuracy |              |              |              |              | 5-shot accuracy |              |               |              |              |
|---------|------------------------|-----------------|--------------|--------------|--------------|--------------|-----------------|--------------|---------------|--------------|--------------|
|         |                        | IMDB            | SST2         | Yelp-2       | Amazon       | MR           | IMDB            | SST2         | Yelp-2        | Amazon       | MR           |
| SST2    | SentiBERT $\diamond$   | 51.40           | 55.60        | 59.64        | 54.90        | 54.88        | 57.76           | 64.84        | 70.20         | 67.02        | 64.90        |
|         | SentiCSE $\diamond$    | <b>74.68</b>    | <b>91.82</b> | <b>82.00</b> | <b>81.24</b> | <b>86.94</b> | <b>81.86</b>    | <b>92.80</b> | <b>88.02</b>  | <b>86.38</b> | <b>90.24</b> |
| Yelp2   | SentiX $\diamond$      | 74.64           | 64.96        | 87.66        | 86.14*       | 65.06        | 83.68           | 72.32        | 93.40         | 92.32*       | 76.68        |
|         | SentiCSE $\diamond$    | 69.22           | 68.10        | 91.14        | 85.48        | 63.76        | 84.24           | 86.48        | <b>95.24</b>  | 89.74        | 80.86        |
|         | SentiLARE $\spadesuit$ | 70.20           | 74.26        | 87.00        | 84.58        | 68.68        | 87.18           | 80.10        | 93.28         | 91.06        | 82.34        |
|         | SentiCSE $\spadesuit$  | <b>76.64</b>    | <b>84.78</b> | <b>94.26</b> | <b>89.28</b> | <b>73.20</b> | <b>87.98</b>    | <b>87.66</b> | 95.12         | <b>92.68</b> | <b>86.36</b> |
| Amazon  | SentiX $\diamond$      | 74.64           | 64.96        | 87.66*       | 86.14        | 65.06        | 83.68           | 72.32        | <b>93.40*</b> | 92.32        | 76.68        |
|         | SentiCSE $\diamond$    | <b>75.16</b>    | <b>78.56</b> | <b>91.16</b> | <b>93.16</b> | <b>78.02</b> | <b>86.64</b>    | <b>85.18</b> | 92.98         | <b>93.86</b> | <b>85.44</b> |

Table 6: Comparative results on the 1-shot and 5-shot settings on each downstream dataset. In this table, we divided the compared methods into three groups in terms of pretraining data. Then we separately compared them within each group. Again, SentiCSE achieves meaningful few-shot performance in most cases.  $\diamond$ : BERT-base models.  $\spadesuit$ : RoBERTa-based models.

| Loss                                    | IMDB         | SST-2        | Yelp-2       | Amazon       | MR           | Avg.         |
|-----------------------------------------|--------------|--------------|--------------|--------------|--------------|--------------|
| $\mathcal{L}_{pos}$                     | 93.40        | 94.95        | 94.60        | 92.13        | 86.59        | 92.33        |
| $\mathcal{L}_{neg}$                     | 93.18        | 94.84        | 95.02        | 92.38        | 87.99        | 92.68        |
| $\mathcal{L}_{pos} + \mathcal{L}_{neg}$ | 93.61        | 95.30        | 95.16        | 92.28        | 88.18        | 92.91        |
| $\mathcal{L}_w + \mathcal{L}_{pos}$     | 93.03        | 95.18        | 95.37        | 92.60        | 86.87        | 92.61        |
| $\mathcal{L}_w + \mathcal{L}_{neg}$     | 93.10        | 94.61        | <b>95.38</b> | 92.42        | 88.65        | 92.83        |
| $\mathcal{L}_{SentiCSE}$                | <b>93.89</b> | <b>95.30</b> | 95.36        | <b>93.23</b> | <b>88.93</b> | <b>93.34</b> |

Table 7: Ablation of each component.

| $\lambda_w$ | IMDB         | SST-2        | Yelp-2       | Amazon       | MR           | Avg          |
|-------------|--------------|--------------|--------------|--------------|--------------|--------------|
| 0.15        | 93.89        | <b>95.30</b> | 95.36        | 93.42        | <b>88.93</b> | <b>93.88</b> |
| 0.2         | 93.72        | 95.07        | <b>95.99</b> | 93.01        | 87.90        | 93.14        |
| 0.25        | <b>93.91</b> | 95.18        | 95.86        | <b>93.27</b> | 88.37        | 93.32        |

Table 8: Impact of  $\lambda_w$  on the performance.

| Ratio | IMDB         | SST-2        | Yelp-2       | Amazon       | MR           | Avg          |
|-------|--------------|--------------|--------------|--------------|--------------|--------------|
| 0.1   | 94.03        | 95.18        | <b>95.86</b> | <b>93.69</b> | <b>89.49</b> | <b>93.65</b> |
| 0.5   | 93.89        | <b>95.30</b> | 95.36        | 93.23        | 88.93        | 93.31        |
| 0.9   | <b>94.06</b> | 94.84        | 95.46        | 93.19        | 88.37        | 93.18        |

Table 9: Sensitivity test on the masking ratio.

only  $\mathcal{L}_{neg}$  provides a better result than using only  $\mathcal{L}_{pos}$ . This might be because of the phenomenon that diverse and rich vocabulary is used in negative reviews, while relatively consistent vocabulary is used in positive reviews (Fang and Zhan, 2015). Using  $\mathcal{L}_{pos}$  and  $\mathcal{L}_{neg}$  together provides higher performance than using only one of them, which confirms the effectiveness of taking both polarity labels (positive and negative) as anchors. Adding  $\mathcal{L}_w$  to each sentence-level objective consistently improves performance, which confirms that understanding sentence sentiment at word-level benefits learning sentiment representations. Furthermore, Figure 6 shows the visualization results obtained from each ablation. Notably, we can observe that contrastive learning using two sentences of different sentiment polarities as each anchor leads to higher quality representations compared to using only one sentence of a single sentiment polarity. We also confirm that the word-level loss plays an important role.

Next, we report the performance of SentiCSE ac-

ording to varying  $\lambda_w$  and the masking ratio used for the word-level objective. We tested different masking ratios with  $\{0.1, 0.5, 0.9\}$  and  $\lambda_w$  with  $\{0.15, 0.2, 0.25\}$ . Tables 7, 8 and 9 report the linear probing performance in each case. We observed that their impact was not very significant.

Finally, we qualitatively evaluate the sentiment representation quality of SentiCSE through a case study. We fed one positive and one negative statement each about the taste of the food, the mood in the restaurant, and the friendliness of the staff for a particular restaurant into the models to obtain embeddings. We then searched for two sentences with the highest embedding similarity to the query, "The atmosphere of the restaurant is good", which is another positive statement about the atmosphere of the restaurant. The results, compared across the baselines, are shown in Table 10. For SentiBERT and SentiWSP, the most similar sentence is positive, but the next similar sentence is a negative review about the service of the staff, which is contextually similar to the atmosphere of the restaurant. In the case of SentiLARE, the first sentence was a positive review, but the second was a negative review about the mood in the restaurant. SentiX recognized the negative review as the most similar sentence. On the other hand, our SentiCSE correctly identified positive reviews about the restaurant, highlighting its promising quality of sentence representation compared to the baselines in terms of sentiment analysis.

## 6. Related Work

This section focuses on PLMs specially designed for sentiment analysis because this line of research is the most relevant to our work.

SentiBERT (Yin et al., 2020) embeds a two-level attention mechanism on top of the BERT representation, and introduces the binary constituency parse tree in order to capture phrase-level compositional sentence semantics. SentiLARE (Sentiment-

| Query Sentence                                                                                  |
|-------------------------------------------------------------------------------------------------|
| The atmosphere of the restaurant is good.                                                       |
| SentiBERT                                                                                       |
| 1) The food is delicious.<br>2) The service at the restaurant is discourteous.                  |
| SentiX                                                                                          |
| 1) The service at the restaurant is discourteous.<br>2) The food is delicious.                  |
| SentiLARE                                                                                       |
| 1) The restaurant has attentive and friendly staff.<br>2) The restaurant lacks a good ambiance. |
| SentiWSP                                                                                        |
| 1) The food is delicious.<br>2) The service at the restaurant is discourteous.                  |
| SentiCSE                                                                                        |
| 1) The food is delicious.<br>2) The restaurant has attentive and friendly staff.                |

Table 10: From several positive and negative reviews about a restaurant, the two sentences with the highest embedding similarity to the query were identified by each model. Cosine similarity was used as the similarity metric.

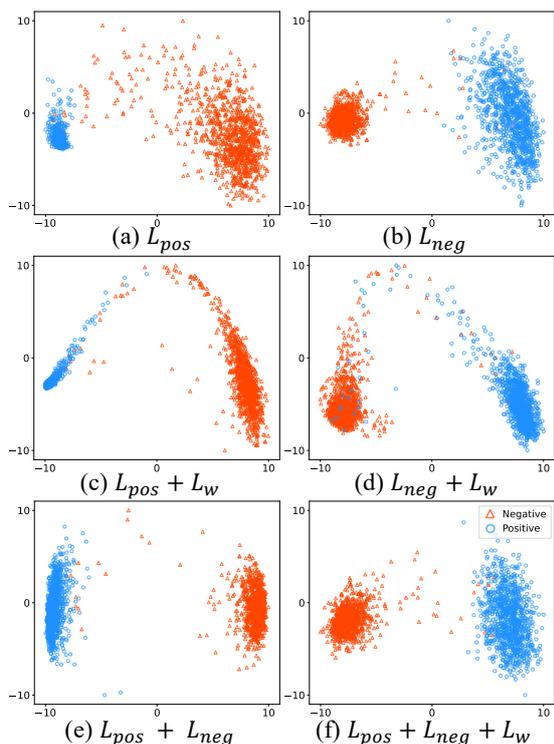


Figure 6: Embedding visualizations of each ablation. PCA is used for visualization.

Aware Language Representation Learning) (Ke et al., 2020) introduces sentiment-related linguistic knowledge obtained from SentiWordNet into PLMs, aiming at constructing knowledge-aware language representations. SKEP (Sentiment Knowledge Enhanced Pre-training) (Tian et al., 2020a) designs three objective functions that predict sentiment-

related knowledge to construct a unified sentiment representation for multiple downstream sentiment analysis tasks. SentiX (Zhou et al., 2020) explores domain-invariant sentiment knowledge from large-scale review dataset and designs the cross-domain sentiment classification tasks.

Very recently, SimCSE (Gao et al., 2021b) successfully introduces the contrastive learning paradigm to sentence embedding. Motivated by this work, sentence-level contrastive learning has been adopted to sentiment analysis research. SCAPT (Supervised ContrAstive Pre-Training) (Li et al., 2021) aligns the representation of implicit sentiment expressions with the same polarity in order to capture both implicit and explicit sentiment orientation. SentiWSP (Sentiment-Aware Word and Sentence Level Pre-training) (Fan et al., 2022) combines the word-level task that adopts MLM in sentiment word prediction and the sentence-level task which employs contrastive learning.

We summarize the differences between the previous works and our method as follows: (1) In the word-level task, we focus on matching the polarity of words, rather than matching the words themselves. (2) In the sentence-level task, we employ supervised contrastive learning that enables us to specify negatives to benefit from constructing sentiment representations. (3) We design the SgTS task to evaluate the quality of representations.

## 7. Conclusion

This paper presents SgTS (Sentiment-guided Textual Similarity), a metric for evaluating sentiment representations. We then propose SentiCSE which focuses on the understanding sentiment of sentences via the word-level and sentence-level objectives, whose representation quality is guaranteed by SgTS. Experimental results confirm that SentiCSE shows higher than or comparable performance to the state-of-the-art baselines while requiring much less amount of data. We believe that our work sheds light on the importance of the quality of sentiment representations in the downstream sentiment analysis tasks with limited labeled data.

## 8. Acknowledgement

This work was partly supported by (1) the National Research Foundation of Korea (NRF) grant funded by the Korea government (\*MSIT) (No.2018R1A5A7059549) and (2) the Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2020-0-01373, Artificial Intelligence Graduate School Program (Hanyang University)). \*Ministry of Science and ICT

## 9. References

- Eneko Agirre, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Inigo Lopez-Gazpio, Montse Maritxalar, Rada Mihalcea, et al. 2015. Semeval-2015 task 2: Semantic textual similarity, english, spanish and pilot on interpretability. In Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015), pages 252–263.
- Eneko Agirre, Carmen Banea, Claire Cardie, Daniel 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 Claramunt, and Janyce Wiebe. 2016. Semeval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation. In SemEval-2016. 10th International Workshop on Semantic Evaluation; 2016 Jun 16-17; San Diego, CA. Stroudsburg (PA): ACL; 2016. p. 497-511. ACL (Association for Computational Linguistics).
- Eneko Agirre, Daniel Cer, Mona Diab, 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.
- Eneko Agirre, Mona Diab, Daniel Cer, and Aitor Gonzalez-Agirre. 2012. Semeval-2012 task 6: a pilot on semantic textual similarity. In Proceedings of 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, pages 385–393.
- Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2010. [SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining](#). In Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10), Valtella, Malta. European Language Resources Association (ELRA).
- Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. [A large annotated corpus for learning natural language inference](#). In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.
- Daniel Cer, Mona Diab, Eneko Agirre, Inigo 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.
- Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. [ELECTRA: Pre-training text encoders as discriminators rather than generators](#). In ICLR.
- 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).
- 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.
- Shuai Fan, Chen Lin, Haonan Li, Zhenghao Lin, Jinsong Su, Hang Zhang, Yeyun Gong, Jian Guo, and Nan Duan. 2022. [Sentiment-aware word and sentence level pre-training for sentiment analysis](#). In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 4984–4994, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Xing Fang and Justin Zhan. 2015. Sentiment analysis using product review data. Journal of Big Data, 2(1):1–14.
- Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021a. [SimCSE: Simple contrastive learning of sentence embeddings](#). In Proceedings

- of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021b. [Simcse: Simple contrastive learning of sentence embeddings](#). In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 6894–6910.
- Beliz Gunel, Jingfei Du, Alexis Conneau, and Veselin Stoyanov. 2021. [Supervised contrastive learning for pre-trained language model fine-tuning](#). In International Conference on Learning Representations.
- Pei Ke, Haozhe Ji, Siyang Liu, Xiaoyan Zhu, and Minlie Huang. 2020. [SentiLARE: Sentiment-aware language representation learning with linguistic knowledge](#). In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 6975–6988, Online. Association for Computational Linguistics.
- Zhengyan Li, Yicheng Zou, Chong Zhang, Qi Zhang, and Zhongyu Wei. 2021. [Learning implicit sentiment in aspect-based sentiment analysis with supervised contrastive pre-training](#). In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 246–256, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. [Roberta: A robustly optimized bert pretraining approach](#). arXiv preprint arXiv:1907.11692.
- Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. [Learning word vectors for sentiment analysis](#). In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.
- 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).
- Jianmo Ni, Jiacheng Li, and Julian McAuley. 2019. [Justifying recommendations using distantly-labeled reviews and fine-grained aspects](#). 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 188–197, Hong Kong, China. Association for Computational Linguistics.
- 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.
- Zirun Qi, Veda Storey, and Wael Jabr. 2015. [Sentiment analysis meets semantic analysis: Constructing insight knowledge bases](#).
- Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. 2018. [Improving language understanding by generative pre-training](#). In OpenAI Technical Report. OpenAI.
- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. [Language models are unsupervised multitask learners](#). In OpenAI Technical Report.
- 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.
- Mamshad Nayeem Rizve, Salman Khan, Fahad Shahbaz Khan, and Mubarak Shah. 2021. [Exploring complementary strengths of invariant and equivariant representations for few-shot learning](#). In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 10836–10846.
- 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.

conference on computational linguistics, pages 568–579.

Hao Tian, Can Gao, Xinyan Xiao, Hao Liu, Bolei He, Hua Wu, Haifeng Wang, and Feng Wu. 2020a. [SKEP: Sentiment knowledge enhanced pre-training for sentiment analysis](#). In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 4067–4076, Online. Association for Computational Linguistics.

Yonglong Tian, Yue Wang, Dilip Krishnan, Joshua B Tenenbaum, and Phillip Isola. 2020b. Rethinking few-shot image classification: a good embedding is all you need? In Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16, pages 266–282. Springer.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. [A broad-coverage challenge corpus for sentence understanding through inference](#). 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 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Ashima Yadav and Dinesh Kumar Vishwakarma. 2020. Sentiment analysis using deep learning architectures: a review. Artificial Intelligence Review, 53(6):4335–4385.

Da Yin, Tao Meng, and Kai-Wei Chang. 2020. [SentiBERT: A transferable transformer-based architecture for compositional sentiment semantics](#). In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 3695–3706, Online. Association for Computational Linguistics.

Lei Zhang, Shuai Wang, and Bing Liu. 2018. Deep learning for sentiment analysis: A survey. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 8(4):e1253.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. [Character-level convolutional networks for text classification](#). In Advances in Neural Information Processing Systems, volume 28. Curran Associates, Inc.

Jie Zhou, Junfeng Tian, Rui Wang, Yuanbin Wu, Wenming Xiao, and Liang He. 2020. Sentix: A sentiment-aware pre-trained model for cross-domain sentiment analysis. In Proceedings of the 28th international