applied sciences

MDPI

Article

Prefix Data Augmentation for Contrastive Learning of Unsupervised Sentence Embedding

Chunchun Wang and Shu Lv *

School of Mathematical Sciences, University of Electronic Science and Technology of China, Chengdu 611731, China; 202221110310@std.uestc.edu.cn
* Correspondence: lvshu@uestc.edu.cn

Abstract: This paper presents prefix data augmentation (Prd) as an innovative method for enhancing sentence embedding learning through unsupervised contrastive learning. The framework, dubbed PrdSimCSE, uses Prd to create both positive and negative sample pairs. By appending positive and negative prefixes to a sentence, the basis for contrastive learning is formed, outperforming the baseline unsupervised SimCSE. PrdSimCSE is positioned within a probabilistic framework that expands the semantic similarity event space and generates superior negative samples, contributing to more accurate semantic similarity estimations. The model's efficacy is validated on standard semantic similarity tasks, showing a notable improvement over that of existing unsupervised models, specifically a 1.08% enhancement in performance on BERTbase. Through detailed experiments, the effectiveness of positive and negative prefixes in data augmentation and their impact on the learning model are explored, and the broader implications of prefix data augmentation are discussed for unsupervised sentence embedding learning.

Keywords: contrastive learning; sentence embedding; prefix data augmentation

Citation: Wang, C.; Lv, S. Prefix Data Augmentation for Contrastive Learning of Unsupervised Sentence Embedding. *Appl. Sci.* **2024**, *14*, 2880. https://doi.org/10.3390/app14072880

Academic Editors: Francisco De Arriba-Pérez, Silvia García-Méndez and Enrique Costa-Montenegro

Received: 1 February 2024
Revised: 24 March 2024
Accepted: 25 March 2024
Published: 29 March 2024

1. Introduction

Sentence embedding learning has long been a core focus within the field of natural language processing, serving as a critical component in a wide array of downstream applications [1–6]. Researchers have been constructing sentence vector models through text-matching annotated datasets, where each sample is formatted as (Sentence A, Sentence B, Label). When comparing or ranking pairs of sentences for semantic similarity, we typically rely on the cosine value of the angle between sentence vectors for judgment, as shown in Equation (1). Here, A and B represent sentence vectors, and cos(A, B) denotes the cosine value of the angle between them. This equation is based on a standard coordinate basis, where the cosine value of the angle between different basis vectors is 0. Due to the high-dimensional nature of sentence vectors, using the cosine value of the angle between vectors to measure semantic similarity offers the following advantages: (1) it is unaffected by scale; (2) it effectively captures angular information; (3) it exhibits high computational efficiency; and (4) it remains effective in high-dimensional spaces (whereas metrics based on vector norms are susceptible to the curse of dimensionality in high-dimensional spaces). Gao et al. [7] discovered that the sentence vectors learned via Transformers exhibit anisotropy, a characteristic similarly identified in BERT and GPT-2 by Ethayarajh [8]. "Anisotropy" refers to the phenomenon where word embeddings occupy a narrow conical region in the vector space, implying that the coordinate system of the sentence vectors is not a standard coordinate basis, rendering the equality in Equation (1) invalid. Methods such as Bert-flow [9], Bert-whitening [10], IS-Bert [11], CT-Bert [12], and Simcse [13] aim to eliminate or mitigate the anisotropy of the learned sentence vectors, thereby enabling more accurate semantic similarity judgments between sentence pairs.

$$\cos(A, B) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} \quad (1)$$

In recent years, contrastive learning has seen successfully applied in sentence embedding learning [13–15]. The goal of contrastive learning is to bring positive sample pairs closer together in the encoding space, while pushing negative samples further apart. Wang et al. [16] introduced two key concepts for measuring the quality of contrastive learning: alignment and uniformity. For alignment, two samples forming a positive pair should be mapped to nearby features and thus be (mostly) invariant to unneeded noise factors. For uniformity, feature vectors should be roughly uniformly distributed on the unit hypersphere, preserving as much information from the data as possible. Therefore, constructing positive samples and selecting negative samples are crucial for enhancing the effectiveness of contrastive learning, among which SimCSE [13] stands out as a representative method. SimCSE's unsupervised model uses dropout [17] as a form of data augmentation to construct positive samples with minimal noise. Specifically, SimCSE processes N sentences as a batch, where each sentence undergoes two independent rounds of dropout masking before being input into a pretrained BERT model. The resulting embeddings from the same sentence serve as positive pairs, whereas embeddings from different sentences are treated as negative pairs.

SimCSE assumes that dropout serves as the "minimal" form of data augmentation, but a crucial weakness is overlooked. Because dropout aims to retain as much information from the original sentence as possible, including the positional information of each token, this approach inadvertently leads to a misunderstanding: sentences of the same length are perceived to have a higher probability of being semantically similar. Conversely, the unsupervised version of SimCSE does not effectively differentiate between various negative examples. The supervised version of SimCSE, which uses a supervised natural language inference (NLI) dataset, constructs hard negative samples to further enhance the effect of contrastive learning. Wang et al. [18] proposed SNCSE, which acquires word vectors through cue learning and syntactic parsing using spacy and constructs soft negative examples. Nishikawa et al. [19] proposed Ease, which generates positive sample pairs by associating entities sourced from Wikipedia. Negative entities are constrained to be the same type as the positive ones and are excluded if they appear on the same Wikipedia page. Randomly selected candidate entities that meet these criteria are used as hard negative data to construct triplet data. However, these methods for distinguishing negative samples are limited to annotated datasets and are not applicable in unsupervised learning scenarios.

To address these aforementioned issues, we propose PrdSimCSE, an unsupervised contrastive learning framework based on prefix data augmentation. As illustrated in Figure 1, PrdSimCSE uses prefix data augmentation (Prd) to construct positive samples and differentiate negative samples. Next, Section 2 introduces the background knowledge on text augmentation, including the basic concepts of sentence embedding and contrastive learning. Section 3 details the prefix data augmentation method, including how to determine positive and negative prefixes and their impact on the model. Section 4 outlines the unsupervised PrdSimCSE, covering algorithm design, experimental environment setup, parameter settings, datasets, and baseline comparisons. Section 5 describes the progress of prefix data augmentation determined through ablation studies, where we further discuss how to determine positive and negative prefixes and the semantic bias caused by prefix data augmentation. Section 6 discusses the relationship between semantic similarity events and prefix data augmentation, including the advantages and roles of positive and negative prefixes. Section 7 concludes the paper by providing the contributions of prefix data augmentation to unsupervised sentence embedding learning and directions for future research.

The contributions of this paper can be summarized as follows:

- This paper introduces a novel text data augmentation method, prefix data augmentation. Positive samples are constructed using modal particle prefixes combined with dropout, thus preserving the original semantics as much as possible while altering

the positional information of each token. This approach enhances and facilitates contrastive learning.

- Prefix data augmentation can also be used to modify the original semantics. By constructing prompts that reverse the semantics and using them as prefixes, the modified sentences can serve as negative samples. This method allows for the creation of a richer set of negative samples from unsupervised corpora, thereby increasing the discriminability between different negative samples.
- We developed a novel unsupervised sentence embedding learning approach, PrdSimCSE, which we used to construct both positive and negative samples through prefix data augmentation. Additionally, the approach treats other sentences within the same batch as negative samples.
- We evaluated PrdSimCSE across various datasets, and the experimental results demonstrated that our proposed PrdSimCSE achieves superior performance in sentence representation compared with prior approaches. Furthermore, through ablation studies, we further examined the effectiveness of PrdSimCSE and discussed the advantages and limitations of our method in detail.

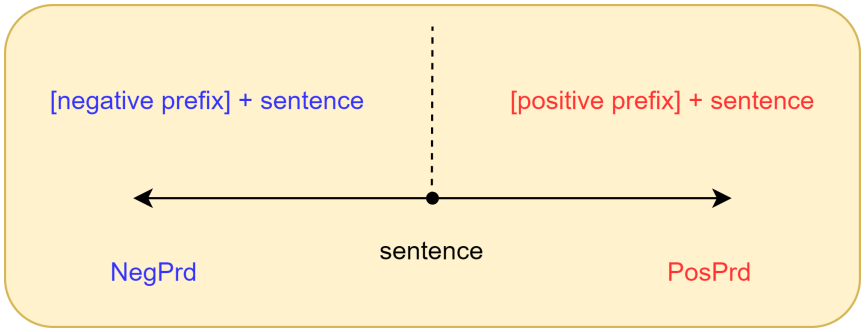

Figure 1. Prefix data augmentation. PosPrd: positive prefix. NegPrd: negative prefix.

2. Background

Text Augmentation Text augmentation can be categorized into two main types based on the generation method: back-translation and adding noise. Back-translation is a simple and efficient text augmentation technique that generates more high-quality samples on the basis of existing datasets by translating and then retranslating the text in scenarios with few samples [20,21]. However, back-translation carries an implicit prior, whereas the model is presented with input texts that, despite having different linguistic expressions, share the same semantics. Adding noise involves directly performing operations such as the addition, deletion, or replacement of sentences. The easy data augmentation (EDA) technique proposed by Wei et al. [22] is a compilation of such methods. EDA consists of four strategies: (1) synonym replacement (SR): randomly select nonstop words from a sentence and replace them with randomly chosen synonyms; (2) random insertion (RI): randomly identify a nonstop word in a sentence, find a synonym for it, and insert that synonym into a random position in the sentence; then, repeat this process n times; (3) in random swap (RS), two words in the sentence are randomly chosen, and their positions are swapped, repeating this process n times; and (4) random deletion (RD): each word in the sentence is randomly deleted with a probability p. Given that these methods involve random operations, a question arises: can the text's label remain unchanged after EDA operations? To address this concern, Xie et al. [23] proposed unsupervised data augmentation (UDA), where the core idea involves replacing a certain proportion of nonessential words in the text with unimportant words from a dictionary, thereby generating new texts. However, in the context of contrastive learning for sentence representation, using such text augmentation methods to construct positive samples for unsupervised learning typically results in lower performance compared with supervised models. SimCSE has achieved notable success in unsupervised learning, also demonstrating that dropout can serve as a "minimal" form of text data augmentation, offering an alternative to other text augmenta-

tion methods. However, dropout augmentation does not alter the positional information of words within a sentence, which can introduce new biases to the model. Inspired by SimCSE, we developed a text augmentation technique suitable for unsupervised learning: prefix data augmentation.

Sentence Embedding Sentence embedding learning aims to convert natural language text sequences into numerical sequences that machines can comprehend. Depending on whether the training corpus is labeled, sentence embedding learning processes can be categorized into supervised and unsupervised approaches. In this study, we primarily focused on unsupervised sentence embedding learning. Word2Vec [24] was one of the earlier models developed for the unsupervised learning of semantic knowledge from large text corpora. Word2Vec proposed the CBOW and skip-gram methods. In CBOW, surrounding words are predicted based on the center word; in skip-gram, the center word is predicted based on surrounding words. BERT [25] introduced the masked language model (MLM) and next sentence prediction (NSP) training methods, but the training data must be at the document level. Subsequent models such as CrossThought [26] and CMLM [27] face similar issues. SimCSE [13] is an unsupervised contrastive learning framework, enabling direct sentence embedding learning using widely available short texts. SimCSE also adapts well to downstream tasks that primarily involve short texts.

Contrastive Learning In the context of sentence embedding learning, the aim of contrastive learning is to train an encoder that produces similar encodings for sentences of the same class within the same dataset, while ensuring the encoding results for sentences of different classes are as dissimilar as possible. Suppose we have a set of samples to be learned, $x_0, x_+^1, \ldots, x_+^m, x_-^1, \ldots x_-^n$, where x_0 serves as the anchor sample, x_+ serves as the positive sample for x_0, and x_- serves as the negative sample for x_0. We drew inspiration from the infoNCE used in MoCo [28] and adopted the SimCSE approach, using other samples within the same batch as negative samples. In a single batch, the contrastive learning objective for the anchor sample x_0 is

$$loss = -\log \frac{\sum_{i=1}^{m} e^{\mathrm{sim}(x_0, x_+^i)/t}}{\sum_{i=1}^{m} e^{\mathrm{sim}(x_0, x_+^i)/t} + \sum_{j=1}^{n} e^{\mathrm{sim}(x_0, x_+^j)/t}} \tag{2}$$

where the batch size is denoted as $N = 1 + m + n$, where t is the temperature hyperparameter, and sim is the function used to compute the cosine similarity between two vectors. In our experiments, positive samples were generated only using PosPrd (positive prefix), represented as $m = 1$ in the formula. For the selection of negative samples, in addition to using other samples within the batch as negative samples, we employed NegPrd (negative prefix). Subsequently, we used pretrained models such as BERT and RoBERTa [29] and fine-tuned all parameters using the contrastive learning objective.

3. Prefix Data Augmentation

As the name suggests, prefix data augmentation involves changing text data by adding prefixes to augment the dataset. In theory, any text can serve as a prefix for other texts. However, from the perspective of constructing positive samples, we aimed for prefixes that did not alter the original sentence's semantics. When constructing negative samples, we preferred prefixes that reversed or disrupted the semantics of the original sentence as much as possible. Through multiple experiments and trials, we found that modal particle prefixes are excellent positive prefixes, whereas prompts that reverse the semantics are effective as negative prefixes.

In the field of computer vision, various effective data augmentation methods are employed to construct positive examples for contrastive learning, such as cropping, rotation, and color adjustments. These methods are employed because the information in images is continuous [30], and partial pixel blocks of an image can convey information. However, in natural language processing (NLP), the semantic information embedded in text data is discrete. As shown in Table 1, text augmentation methods such as Random deletion,

Word substitution, and Rearrangement substantially alter the semantics of the original sentence, resulting in decreased similarity between the modified and original sentences. However, the positive prefix preserves the original semantic information. SimCSE applies dropout as a method that effectively addresses the challenge posed by the discrete nature of semantics in text data. When using contrastive learning for sentence embeddings, dropout can serve as a "minimal" positive data augmentation method for constructing positive examples. However, the dropout method only deactivates certain positions in the sentence embeddings. Does constructing positive samples using this approach ensure the encoder is sensitive to the length and structure of sentences? To mitigate this issue, we introduced a simple meaningless prefix in PosPrd.

Table 1. An example of different data augmentation techniques for changing the semantic meaning of a sentence.

Data Augmentation	Sentence	Similarity
Origin	I love natural language processing.	1
Positive prefix	Um, I love natural language processing.	0.99
Random deletion	I natural language processing.	0.65
Word substitution	I love artificial language processing.	0.51
Rearrangement	natural language processing love I.	0.32

We found that specific semantic reversal prefixes can serve as negative data augmentation, used for constructing negative samples in contrastive learning. In the contrastive learning of sentence embeddings, a common approach involves using other sentences within the same batch as negative samples for the anchor sentence. This method is based on the assumption that in a rich training corpus, each sentence belongs to two semantically different categories. However, text data often contain a considerable amount of repetition. For instance, in restaurant reviews, many sentences may belong to the positive semantic category, whereas others belong to the negative semantic category. Consequently, during training, many negative samples may be "problematic", where two semantically similar sentences are treated as negative pairs, which is detrimental to contrastive learning. By introducing manually crafted negative prefixes, we ensure the original sentence is augmented into a category with the opposite semantic meaning, thereby constructing higher-quality negative samples. This approach contributes to increasing the effectiveness of contrastive learning, enabling the model to more accurately capture semantic distinctions.

4. Unsupervised PrdSimcse

4.1. Algorithm Design

PrdSimCSE uses a pretrained BERT (uncased) or RoBERTa as the starting point for training, with all data sharing the same encoder. As illustrated in Figure 2, the model receives three inputs: the original sample, a positive sample that has undergone prefix data augmentation, and a negative sample also enhanced with prefix data augmentation. The output generated from the original sample is used to calculate the contrastive learning loss according to Equation (2), after which gradients are back-propagated to update the parameters of the entire model.

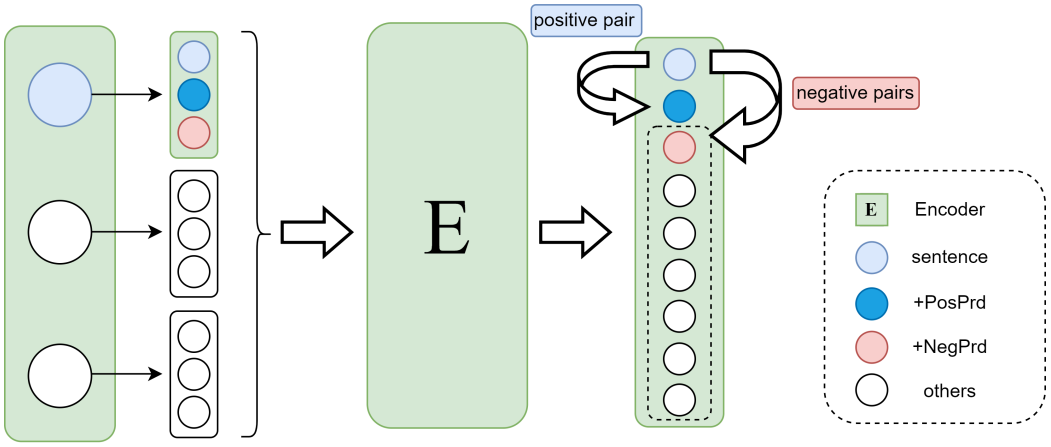

Figure 2. PrdSimCSE constructs positive and negative samples with different prefixes.

4.2. Experimental Environment Setup

Table 2 presents the computational environment of our model, and all experiments were conducted within this setup. We used Python 3.8 as our programming language. Additionally, we employed PyTorch (version 1.7.1) and the Transformers library (version 4.2.1) for the training, evaluation, and testing of PrdSimCSE.

Table 2. Experimental Environment Setup.

Attribute	Details
System Development Environment	Ubuntu22.04
Development Environment	Visual Studio Code (version 1.83)
CUDA version	11.8
GPU	NVIDIA 4080 (16 G)
CPU	Intel i9 13900KF
Memory	32 GB

4.3. Parameter Settings

For PrdSimCSE, we trained the model for three epochs, conducting evaluations every 250 steps. We performed a grid search over batch sizes of 32, 64, 128 and learning rates of 1×10^{-5}, 2×10^{-5}, 3×10^{-5}, 4×10^{-5}, 5×10^{-5} on the STS-B development set. The final hyperparameter settings are shown in Table 3. Additionally, we used dropout sampling with a dropout rate of 0.1 and employed AdamW as the optimizer, with a weight decay of 0.01. For evaluations, we selected the [cls] token as the sentence representation and retained the MLP layer. All experimental results were assessed using the Spearman correlation coefficient as the evaluation metric.

Table 3. Parameter Settings.

Model	$\text{Bert}_{\text{base}}$			$\text{RoBERTa}_{\text{base}}$		
Batch size	32	64	128	32	64	128
Learning rate	4×10^{-5}	4×10^{-5}	3×10^{-5}	3×10^{-5}	2×10^{-5}	1×10^{-5}

4.4. Datasets

The training corpus was derived from an unlabeled dataset collected by Gao et al. [13], consisting of 1 million English sentences randomly extracted from Wikipedia, with each data entry being a single English sentence. For the PrdSimCSE's evaluation, we employed seven semantic similarity datasets and nine transfer task datasets, which are briefly introduced here.

The semantic similarity tasks includes STS12 [31], STS13 [32], STS14 [33], STS15 [34], STS16 [35], SICK-R [36], and STS-B [37]. STS 12–16 are the datasets from the SemEval competitions from 2012 to 2016, respectively. STS-B and SICK-R are also datasets from the SemEval competition. In these datasets, each text pair was scored by humans on a scale of 0–5, indicating the level of similarity between the texts. For evaluation, we used the test sets from all the semantic similarity tasks.

The transfer tasks included MR [38], CR [39], SUBJ [40], MPQA [41], SST-2 [42], TREC [43], MRPC [44], BQ [45], and LCQMC [46]. Among these, MR, CR, SUBJ, and SST-2 are sentiment classification datasets; MPQA and BQ are question answering datasets; TREC and LCQMC are question classification datasets; MRPC is a dataset for sentence pair similarity. Additionally, BQ and LCQMC are Chinese datasets; we first translated them into English before inputting them into the model for testing. In essence, these tasks were all text classification tasks, where each sentence was associated with a label (or a matching sentence). During evaluation, the model received a sentence and then determined its category; finally, we compared this prediction with the sentence's label.

4.5. Baseline

Sentence embedding learning can be divided into supervised and unsupervised categories, with this study focusing on improvements in unsupervised contrastive learning. We used unsupervised models such as IS-BERT [11], CT-BERT [12], and SimCSE [13] as baselines. IS-BERT ensures maximal consistency between global and local features, CT-BERT aligns sentence embeddings of the same sentence from two different encoders, and SimCSE aligns different embeddings of the same sentence from the same encoder. Additionally, we compared our approach with postprocessing methods such as BERT-flow [9] and BERT-whitening [10], as well as with naïve baselines such as average GloVe embeddings [47] and average embeddings from the first and last layers of BERT. BERT-flow and BERT-whitening focus on eliminating anisotropy in sentence vectors at the BERT output without changing the encoder structure, whereas average GloVe embeddings and average first- and last-layer BERT embeddings retain the original information of the pretrained BERT. These were all reliable standards for evaluating the effectiveness of our model.

4.6. Main Results

In Table 4, we showcase the performance on the STS tasks. We replicated the results of SimCSE and, through comparison, we found that PrdSimCSE-BERT$_{\text{base}}$ substantially outperformed SimCSE-BERT$_{\text{base}}$ on STS13-STS16, STS-B, and SICK-R in terms of Spearman correlation coefficient, notably by 2.01% on STS-B. On STS12, PrdSimCSE performed consistently with SimCSE, producing an increase in the average Spearman correlation coefficient from 76.25% to 77.33%. Compared with the unsupervised models, PrdSimCSE outperformed CT-BERT by 5.28% and IS-BERT by 10.75% in terms of the average Spearman correlation coefficient. For the RoBERTa model, PrdSimCSE-RoBERTa$_{\text{base}}$ also improved upon SimCSE-RoBERTa$_{\text{base}}$'s 76.57% to 77.10%, particularly outperforming SimCSE by 4.81% on SICK-R.

In Table 5, we present the results for transfer tasks. Compared with SimCSE-BERT$_{\text{base}}$, PrdSimCSE-BERT$_{\text{base}}$ performed better on five of the datasets. However, on the TREC task, the accuracy of PrdSimCSE was 7.89% lower than that of SimCSE—a substantial difference that requires further consideration. TREC is a widely used dataset in the field of information retrieval, employed for evaluating the performance of information retrieval systems. The Text REtrieval Conference (TREC) Question Classification dataset contains 5500 labeled questions in the training set and another 500 in the test set. The dataset has 6 coarse class labels and 50 fine class labels. During testing, each sentence is assigned a label. We speculate that the reason for this discrepancy is because PrdSimCSE uses prefix data augmentation, resulting in a large number of identical sentence prefixes in the training corpus, which leads to considerable semantic bias when encoding sentences. This was further discussed in the ablation experiments.

Table 4. Sentence embedding performance on STS tasks (Spearman correlation, "all" setting). We highlight the highest numbers among models with the same pretrained encoder. ♣: results from [6]; ♡: results from [11]. ♢: results from ourselves; all other results are from [13].

Model	STS12	STS13	STS14	STS15	STS16	STS-B	SICK-R	Avg.
GloVe embedding (avg.) ♣	55.14	70.66	59.73	68.25	63.66	58.02	53.76	61.32
$BERT_{base}$ (first–last avg.)	39.7	59.38	49.67	66.03	66.19	53.87	62.06	56.7
$BERT_{base}$-flow	58.4	67.1	60.85	75.16	71.22	68.66	64.47	66.55
$BERT_{base}$-whitening	57.83	66.9	60.9	75.08	71.31	68.24	63.73	66.28
IS-$BERT_{base}$ ♡	56.77	69.24	61.21	75.23	70.16	69.21	64.25	66.58
CT-$BERT_{base}$	61.63	76.8	68.47	77.5	76.48	74.31	69.19	72.05
SimCSE-$BERT_{base}$	68.4	82.41	74.38	80.91	78.56	76.85	72.23	76.25
PrdSimCSE-$BERT_{base}$ ♢	68.21	83.56	75.44	81.71	79.79	78.86	73.75	77.33
$RoBERTa_{base}$(first-last avg.)	40.88	58.74	49.07	65.63	61.48	58.55	61.63	56.57
$RoBERTa_{base}$-whitening	46.99	63.24	57.23	71.36	68.99	61.36	62.91	61.73
SimCSE-$RoBERTa_{base}$	70.16	81.77	73.24	81.36	80.65	80.22	68.56	76.57
PrdSimCSE-$RoBERTa_{base}$ ♢	69.29	83.52	75.02	81.13	78.84	78.52	73.37	77.10

Table 5. Transfer task results of different sentence embedding models (measured as accuracy). ♣: results from [6]; ♡: results from [13]. ♢: results from ourselves. We highlight the highest numbers among models with the same pretrained encoder.

Model	MR	CR	SUBJ	MPQA	SST-2	TREC	MRPC	Avg.
GloVe embedding(avg.) ♣	77.25	78.3	91.17	87.85	80.18	83	72.87	81.52
BERT-[CLS]embedding ♣	78.68	84.85	94.21	88.23	84.13	91.40	71.13	84.66
SimCSE-$BERT_{base}$ ♡	81.18	86.46	94.45	88.88	85.5	89.8	74.43	85.81
PrdSimCSE-$BERT_{base}$ ♢	81.32	87.32	94.67	88.11	86.81	80.91	76.23	85.01

In Table 6, we present the evaluation results of PrdSimCSE on the two Chinese transfer task datasets. Because our model was trained on English corpora, direct evaluation using Chinese datasets resulted in very poor performance. We used Google Translate to translate all the data into English before conducting the evaluations. PrdSimCSE-$BERT_{base}$ outperformed SimCSE-$BERT_{base}$ by 1.63% on the BQ dataset and by 2.17% on LCQMC. PrdSimCSE-$RoBERTa_{base}$ exceeded SimCSE-$RoBERTa_{base}$ in performance by 0.85% on the BQ dataset and by 1.37% on LCQMC.

Table 6. Sentence embedding performance on two Chinese datasets.

Model	BQ	LCQMC
SimCSE-$Bert_{base}$	45.38	56.46
PrdSimCSE-$Bert_{base}$	47.01	58.63
SimCSE-$RoBERTa_{base}$	47.97	62.41
PrdSimCSE-$RoBERTa_{base}$	48.82	63.78

5. Ablation Studies

The success of PrdSimCSE can be attributed to two factors: positive and negative prefixes. Therefore, we conducted experiments from these two perspectives. Unless otherwise specified, all models described in this section were trained based on $Bert_{base}$ and were compared on the STS-B development set.

5.1. Progress of Prefix Data Augmentation

We aimed to understand the extent of improvement PrdSimCSE could achieve over SimCSE and, specifically, the contributions of positive and negative prefixes to PrdSimCSE. As shown in Table 7, both positive and negative prefixes were found to be effective data augmentation methods, and their combination led to further improvement in PrdSimCSE.

Table 7. Improvement on STS-B development sets produced by PosPrd or NegPrd over SimCSE. ♡: results from [13].

Model	STS-B
SimCSE ♡	82.5
+PosPrd	83.4
+NegPrd	82.9
PrdSimCSE	83.5

Sentence embeddings play a pivotal role across a spectrum of downstream tasks including text matching, sentiment analysis, and question answering systems. The advancements facilitated by Predix data augmentation in sentence embedding models hold the promise of enhancing the efficacy of these specific downstream applications. Drawing insights from the findings presented in Tables 4 and 5, when benchmarked against SimCSE-BERTbase, PrdSimCSE exhibited notable performance gains: a 2.01% enhancement in the STS-B text matching task, a 1.31% improvement in sentiment analysis as observed in the SST-2 task, and a 1.67% increase in accuracy for the question answering task BQ. Central to these tasks is the precise vectorization of textual data, a cornerstone in contemporary natural language processing (NLP) research. Retrieval methods relying on text embeddings have garnered significant traction within the NLP community. Text embedding libraries, by segmenting and organizing textual data followed by vectorization utilizing sentence embedding models, serve as pivotal components. Notably, the efficacy of these libraries hinges greatly upon the performance of the underlying sentence embedding models. With robust storage and retrieval capabilities, text embedding libraries stand poised to drive further innovation in NLP applications. The PrdSimCSE framework proposed herein is poised to catalyze advancements in this domain, bolstering the widespread adoption and utility of text embedding libraries.

5.2. Determining Positive Prefixes

The primary criterion for positive prefixes is that they should not distort the original semantics of the sentence. Before specific experiments, we deployed a semantic similarity calculation tool using the pipeline of the Transformers package. By adding prefixes to several example sentences and calculating their similarity to the original sentences, we selected sentence pairs with the highest semantic similarity to preliminarily determine several positive prefixes. For the convenience of later discussion, the abbreviations and meanings of these positive prefixes are explained here. one-[CLS], one-[MASK], one-um: a single [CLS], a single [MASK], and a single 'um' as prefixes, respectively. level-[CLS], level-[MASK], level-um: As sentence length increases, the number of prefix words also gradually increases, as demonstrated in the Table 8 outlining the specific rules. Subsequently, we used the corresponding positive prefixes for data augmentation in sentence embedding learning. The experimental results are shown in Table 9, where the positive prefix "level-um" achieved the best results. Additionally, using "level-[CLS]" as a positive prefix also yielded accurate results, whereas "one-[MASK]" performed poorly. Other prompting sentences led to a decrease in performance.

Table 8. Relationship between sentence length and number of words added as prefix

Sentence Length	Number of Words Added as Prefix
<8	0
[8, 16)	1
[16, 24)	2
[24, 32)	3
≥32	4

Table 9. Results of different types of PosPrd on STS-B. ♡: results from [13].

Type of PosPrd	STS-B
SimCSE ♡	82.5
one-[CLS]	81
one-[MASK]	81.8
one-um	81.5
level-[CLS]	82.8
level-[MASK]	80.6
level-um	83.4

5.3. Determining Negative Prefixes

Similar to the method of positive prefixes, we also experimented with various negative prefixes. Simply covering the original sentence by adding long prefixes loses the meaning of prefix data augmentation. However, short prefixes cannot reverse the semantics of the entire sentence. Based on this, we identified several appropriately sized semantic reversal cues. For the convenience of later discussion, the abbreviations and meanings of these negative prefixes are explained here. one-reversal: Using 'reversal' as a prefix. prefix1: "Sentence contradicts in time, place, people, number, emotion, type". prefix2: "Contradictory in time, location, individuals, quantity, emotion, and category within the sentence". prefix3: "The expression in terms of time, location, persons, number, emotion, and type in the following sentence is contradictory". Prefix3 did not override the original sentences, but reversed the meaning of the sentences as much as possible. Table 10 lists the performance of the negative prefixes in semantic similarity tasks.

Table 10. Results of different types of NegPrd on STS-B. ♡: results from [13].

Type of NegPrd	STS-B
SimCSE ♡	82.5
one-reversal	80.4
prefix1	81.3
prefix2	81.1
prefix3	82.9

5.4. Semantic Bias Caused by Prefix Data Augmentation

To further explore the semantic bias introduced by prefix data augmentation and verify the impact of sentence length on the encoding results, we conducted experiments on the SICK-R test set [36]. Based on whether the length of the shorter sentence in a sentence pair was less than eight, we divided the SICK-R test set into two groups and evaluated them separately, comparing the Spearman correlation coefficients. As shown in Table 11, both PrdSimCSE and SimCSE performed much worse on sentences with lengths > 8 than on those with lengths ≤ 8. However, upon further analysis of the original SICKR test set results, PrdSimCSE exhibited larger fluctuations, especially in sentences with lengths ≤ 8, where the fluctuation in PrdSimCSE was notably wider than that of SimCSE. This indirectly confirmed our speculation that models trained with Prd tend to produce semantic deviations when encoding sentences.

In English transfer tasks, we observed that PrdSimCSE significantly underperformed SimCSE on the TREC dataset [43], a discrepancy related to this specific error. PrdSimCSE produced a larger error for sentences shorter than eight words. The TREC test set contains 500 sentences, with an average length of 10 words per sentence. Each sentence is associated with one coarse class label and one fine class label, with the label being a single word. We attempted to remove sentences from TREC that were shorter than eight and six words, respectively, and then re-evaluated PrdSimCSE; the results are shown in Table 12. After removing the shorter sentences, PrdSimCSE's performance improved by 6.23%, indirectly

confirming our hypothesis. Due to PrdSimCSE's use of repetitive, meaningless prefixes during training, the encoder reduces its attention to the tokens at the beginning of sentences, thus weakening its understanding of short texts compared with SimCSE.

Table 11. Performance on sentence embeddings for SICK-R with different sentence lengths.

Model	Dataset	Spearman
SimCSE	Original	72.23
	>8	70.22 (−2.01)
	≤8	73.69 (+1.46)
PrdSimCSE	Original	73.75
	>8	71.68 (−2.07)
	≤8	75.59 (+1.84)

Table 12. Performance on sentence embeddings in TREC dataset with different sentence lengths.

Dataset	Original	Remove Length < 6	Remove Length < 8
PrdSimCSE	80.91	84.86	87.14

6. Semantic Similarity Event and Prefix Data Augmentation

6.1. Semantic Similarity Event

We assume that all sentences A can be mapped to the vector space Rn; thus, any sentence pair (A_1, A_2) can be mapped to the vector space R2n, with the resulting vector denoted as x. Then, we define the sample space $\Omega(\{\Omega = \text{sim}(x)|x \in R^{2n}\})$, where $\text{sim}(x)$ represents the vector similarity of the sentence pair (A_1, A_2). The event space $\mathcal{F}$ is the set of combinations of any events (e.g., $\text{sim}(x) > 0.6$); $\mathcal{P}$ is the probability function mapping events to $[0, 1]$. We define a probability space $(\Omega, \mathcal{F}, \mathcal{P})$ based on the similarity of sentence vectors. As such, we can define the occurrence of semantically similar event E in the sample space as

$$E = \{\text{sim}(x) > \tau\} \tag{3}$$

where $\tau \in [0, 1]$ is a threshold parameter. Therefore, the probability of the occurrence of a semantically similar event is represented as

$$P(E) \tag{4}$$

In practice, determining whether two sentences are similar is based on specific contextual information. Therefore, only conditional semantic similarity events exist based on different contextual information, and their probability is

$$P(E|C) \tag{5}$$

where C is an event implying contextual information (e.g., $C = \text{sim}(x) > 0.5$, $E = \text{sim}(x) > 0.8$). In this paper, $P(E|C)$ represents our trained PrdSimCSE model. For any given pair of sentences, based on the training corpus, $P(E|C)$ can provide their semantic similarity score, thereby determining whether the pair belongs to a semantic similarity event. According to Bayes' theorem, the estimation of the probability of a semantic similarity event can be considered as

$$\hat{P}(E) = \frac{P(C)P(E|C)}{P(C|E)} \tag{6}$$

The alignment of the sentence vector space using positive samples strongly impacts contrastive learning. This is because the selection of positive samples is directly related to the quality of sentence pairs in the semantic similarity set. Additionally, Prd can expand

contextual information; namely, the precision of semantic similarity event probability estimation is improved by adding higher-quality negative samples.

6.2. Advantages of Positive Prefixes

The term "positive prefix" refers to a positive data augmentation method that uses meaningless modal particles as prefixes. This approach changes the position of tokens within the sentence without altering the original sentence's semantics. In contrast, dropout merely deactivates some positions in the sentence embedding. Hence, our PrdSimCSE does not incur errors due to positional information. From the perspective of semantic similarity events, semantic events derived through dropout only contain sentence pairs of the same length, which fails to provide an accurate estimation of semantic similarity event probabilities. The proposed PosPrd, by expanding the sentence pairs within semantic similarity events, increased the precision of estimating unconditional semantic similarity event probabilities. We adjusted the proportion of sentences with added prefixes in the training set, as shown in Table 13, to verify our hypothesis. By gradually increasing this proportion, the training results progressively improved. As illustrated in Figure 3, gradually increasing the proportion of sentences with added prefixes—equivalent to expanding the yellow area—resulted in a smaller probability estimation error for semantic similarity events.

Table 13. The performance of sentence embeddings with gradually increasing proportions of positive prefixes.

Model	STS-B
SimCSE	82.5
+20%	82.3
+40%	82.6
+60%	83.1
+80%	82.9
PrdSimCSE	83.5

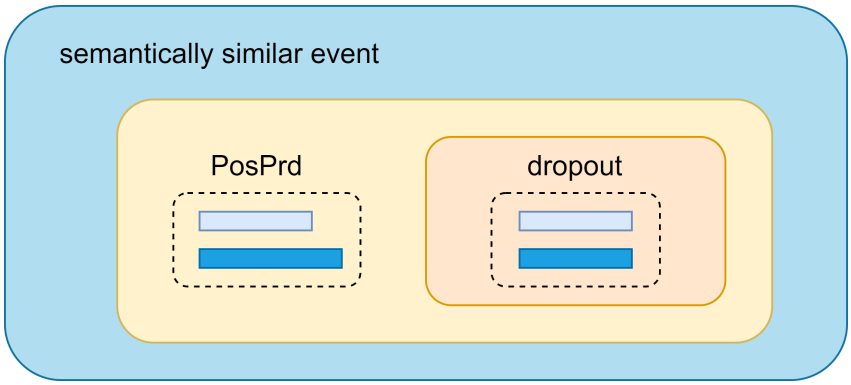

Figure 3. The difference between PosPrd and dropout in constructing positive samples, as well as the relationship between the two and the inclusion of semantically similar events.

6.3. Role of Negative Prefixes

"Negative prefix" refers to a negative data augmentation technique that uses semantic inversion prompts as prefixes. The goal of the negative prefix is to change the original sentence's meaning as much as possible, shifting the meaning to a semantically different category. When training the model using corpora, we were essentially estimating the probability of semantic similarity events using conditional semantic similarity events. During training, negative samples were considered the contextual information of conditional semantic similarity events; hence, we assumed that anchor and positive samples were similar. In Table 14, we attempted to replace sentences with negative prefixes with empty characters, finding that reducing the replacement proportion also improved the results.

This confirms that by adding negative prefixes, which enrich contextual information, the estimation of conditional semantic similarity probability becomes more accurate.

Table 14. The performance of sentence embeddings as the proportion of negative prefixes replaced with empty characters becomes smaller.

Model	STS-B
100%	75.2
80%	78.6
60%	79.3
40%	81.4
20%	81.9
PrdSimCSE	83.5

7. Conclusions

We developed a novel text augmentation method, prefix data augmentation. By using modal words as prefixes to construct positive samples, we avoided the positional information error that arises when SimCSE constructs positive samples. Constructing negative samples by employing specific semantic inversion prompts as prefixes effectively distinguishes negative samples. Based on these methods, we developed an unsupervised sentence embedding contrastive learning model enhanced using prefix data augmentation. The results of the experiments showed that compared with SimCSE, PrdSimCSE achieved comprehensive performance improvements on semantic similarity task sets, achieving a 2.01% increase on STS-B and a 1.08% increase on average.

We conducted a search on the form of prefixes. For positive prefixes, we found that as sentence length increases, adding more semantic word prefixes improves the model's performance. For negative prefixes, we discovered that semantic inversion cue sentences are suitable choices, where the shorter the cue sentence, the larger the performance loss of the model. Additionally, we found that in some transfer tasks, PrdSimCSE's understanding of short texts is not as strong as that of SimCSE. Through multiple ablation experiments, we observed that although PrdSimCSE avoids the semantic biases caused by positive samples of the same length, it loses some ability to comprehend short text. In summary, PrdSimCSE produced improvements compared with SimCSE on 16 tasks, also proving the feasibility and effectiveness of prefix data augmentation.

PrdSimCSE not only addresses the bias issues experienced with previous approaches but also provides a new approach to constructing negative samples for contrastive learning. We think that the method presented in this paper offers valuable references for researchers in the NLP field.

Author Contributions: Conceptualization, S.L. and C.W.; methodology, C.W.; software, C.W.; validation, S.L. and C.W.; formal analysis, S.L.; investigation, C.W.; resources, S.L.; data curation, C.W.; writing—original draft preparation, C.W.; writing—review and editing, S.L.; supervision, S.L. All authors have read and agreed to the published version of the manuscript.

Funding: This study received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available in [13].

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Kiros, R.; Zhu, Y.; Salakhutdinov, R.R.; Zemel, R.; Urtasun, R.; Torralba, A.; Fidler, S. Skip-thought vectors. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 3294–3302.
2. Hill, F.; Cho, K.; Korhonen, A. Learning distributed representations of sentences from unlabelled data. *arXiv* **2016**, arXiv:1602.03483.
3. Conneau, A.; Kiela, D.; Schwenk, H.; Barrault, L.; Bordes, A. Supervised learning of universal sentence representations from natural language inference data. *arXiv* **2017**, arXiv:1705.02364.
4. Logeswaran, L.; Lee, H. An efficient framework for learning sentence representations. *arXiv* **2018**, arXiv:1803.02893.
5. Cer, D.; Yang, Y.; Kong, S.Y.; Hua, N.; Limtiaco, N.; John, R.S.; Constant, N.; Guajardo-Cespedes, M.; Yuan, S.; Tar, C.; et al. Universal sentence encoder for English. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Brussels, Belgium, 31 October–4 November 2018; pp. 169–174.
6. Reimers, N.; Gurevych, I. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv* **2019**, arXiv:1908.10084.
7. Gao, J.; He, D.; Tan, X.; Qin, T.; Wang, L.; Liu, T.Y. Representation degeneration problem in training natural language generation models. *arXiv* **2019**, arXiv:1907.12009.
8. Ethayarajh, K. How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings. *arXiv* **2019**, arXiv:1909.00512.
9. Li, B.; Zhou, H.; He, J.; Wang, M.; Yang, Y.; Li, L. On the sentence embeddings from pre-trained language models. *arXiv* **2020**, arXiv:2011.05864.
10. Su, J.; Cao, J.; Liu, W.; Ou, Y. Whitening sentence representations for better semantics and faster retrieval. *arXiv* **2021**, arXiv:2103.15316.
11. Zhang, Y.; He, R.; Liu, Z.; Lim, K.H.; Bing, L. An unsupervised sentence embedding method by mutual information maximization. *arXiv* **2020**, arXiv:2009.12061.
12. Carlsson, F.; Gyllensten, A.C.; Gogoulou, E.; Hellqvist, E.Y.; Sahlgren, M. Semantic re-tuning with contrastive tension. In Proceedings of the International Conference on Learning Representations, Addis Ababa, Ethiopia, 26–30 April 2020.
13. Gao, T.; Yao, X.; Chen, D. Simcse: Simple contrastive learning of sentence embeddings. *arXiv* **2021**, arXiv:2104.08821.
14. Chuang, Y.S.; Dangovski, R.; Luo, H.; Zhang, Y.; Chang, S.; Soljačić, M.; Li, S.W.; Yih, W.T.; Kim, Y.; Glass, J. DiffCSE: Difference-based contrastive learning for sentence embeddings. *arXiv* **2022**, arXiv:2204.10298.
15. Wu, Z.; Wang, S.; Gu, J.; Khabsa, M.; Sun, F.; Ma, H. Clear: Contrastive learning for sentence representation. *arXiv* **2020**, arXiv:2012.15466.
16. Wang, T.; Isola, P. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In Proceedings of the International Conference on Machine Learning (PMLR), Virtual, 13–18 July 2020; pp. 9929–9939.
17. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.
18. Wang, H.; Dou, Y. Sncse: Contrastive learning for unsupervised sentence embedding with soft negative samples. In Proceedings of the International Conference on Intelligent Computing, Zhengzhou, China, 10–13 August 2023; Springer: Berlin/Heidelberg, Germany, 2023, pp. 419–431.
19. Nishikawa, S.; Ri, R.; Yamada, I.; Tsuruoka, Y.; Echizen, I. EASE: Entity-aware contrastive learning of sentence embedding. *arXiv* **2022**, arXiv:2205.04260.
20. Edunov, S.; Ott, M.; Auli, M.; Grangier, D. Understanding back-translation at scale. *arXiv* **2018**, arXiv:1808.09381.
21. Wang, X.; Pham, H.; Dai, Z.; Neubig, G. SwitchOut: An efficient data augmentation algorithm for neural machine translation. *arXiv* **2018**, arXiv:1808.07512.
22. Wei, J.; Zou, K. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. *arXiv* **2019**, arXiv:1901.11196.
23. Xie, Q.; Dai, Z.; Hovy, E.; Luong, T.; Le, Q. Unsupervised data augmentation for consistency training. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 6256–6268.
24. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient estimation of word representations in vector space. *arXiv* **2013**, arXiv:1301.3781.
25. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
26. Wang, S.; Fang, Y.; Sun, S.; Gan, Z.; Cheng, Y.; Jiang, J.; Liu, J. Cross-thought for sentence encoder pre-training. *arXiv* **2020**, arXiv:2010.03652.
27. Yang, Z.; Yang, Y.; Cer, D.; Law, J.; Darve, E. Universal sentence representation learning with conditional masked language model. *arXiv* **2020**, arXiv:2012.14388.
28. He, K.; Fan, H.; Wu, Y.; Xie, S.; Girshick, R. Momentum contrast for unsupervised visual representation learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 9729–9738.
29. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv* **2019**, arXiv:1907.11692.
30. Chen, T.; Kornblith, S.; Norouzi, M.; Hinton, G. Simclr: A simple framework for contrastive learning of visual representations. In Proceedings of the International Conference on Learning Representations, Addis Ababa, Ethiopia, 26–30 April 2020; Volume 2, p. 4.

31. Agirre, E.; Cer, D.; Diab, M.; Gonzalez-Agirre, A. Semeval-2012 task 6: A pilot on semantic textual similarity. In Proceedings of the * SEM 2012: The First Joint Conference on Lexical and Computational Semantics–Volume 1: Proceedings of the Main Conference and the Shared Task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012), Montréal, QC, Canada, 7–8 June 2012; pp. 385–393.
32. Agirre, E.; Cer, D.; Diab, M.; Gonzalez-Agirre, A.; Guo, W. * SEM 2013 shared task: Semantic textual similarity. In Proceedings of the Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity, Atlanta, GA, USA, 13–14 June 2013; pp. 32–43.
33. Agirre, E.; Banea, C.; Cardie, C.; Cer, D.; Diab, M.; Gonzalez-Agirre, A.; Guo, W.; Mihalcea, R.; Rigau, G.; Wiebe, J. Semeval-2014 task 10: Multilingual semantic textual similarity. In Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), Dublin, Ireland, 23–24 August 2014; pp. 81–91.
34. Agirre, E.; Banea, C.; Cardie, C.; Cer, D.; Diab, M.; Gonzalez-Agirre, A.; Guo, W.; Lopez-Gazpio, I.; Maritxalar, M.; Mihalcea, R.; et al. 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), Denver, CO, USA, 4–5 June 2015; pp. 252–263.
35. Agirre, E.; Banea, C.; Cer, D.; Diab, M.; Gonzalez Agirre, A.; Mihalcea, R.; Rigau Claramunt, G.; Wiebe, J. Semeval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation. In Proceedings of the SemEval-2016: 10th International Workshop on Semantic Evaluation, San Diego, CA, USA, 16–17 June 2016; ACL (Association for Computational Linguistics): Stroudsburg, PA, USA, 2016; pp. 497–511.
36. Bentivogli, L.; Bernardi, R.; Marelli, M.; Menini, S.; Baroni, M.; Zamparelli, R. SICK through the SemEval glasses. Lesson learned from the evaluation of compositional distributional semantic models on full sentences through semantic relatedness and textual entailment. *Lang. Resour. Eval.* **2016**, *50*, 95–124. [CrossRef]
37. Cer, D.; Diab, M.; Agirre, E.; Lopez-Gazpio, I.; Specia, L. Semeval-2017 task 1: Semantic textual similarity-multilingual and cross-lingual focused evaluation. *arXiv* **2017**, arXiv:1708.00055.
38. Pang, B.; Lee, L. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. *arXiv* **2005**, arXiv:cs/0506075v1.
39. Hu, M.; Liu, B. Mining and summarizing customer reviews. In Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seattle, WA, USA, 22–25 August 2004; pp. 168–177.
40. Pang, B.; Lee, L. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. *arXiv* **2004**, arXiv:cs/0409058.
41. Wiebe, J.; Wilson, T.; Cardie, C. Annotating expressions of opinions and emotions in language. *Lang. Resour. Eval.* **2005**, *39*, 165–210. [CrossRef]
42. Socher, R.; Perelygin, A.; Wu, J.; Chuang, J.; Manning, C.D.; Ng, A.Y.; Potts, C. Recursive deep models for semantic compositionality over a sentiment treebank. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, Seattle, CA, USA, 18–21 October 2013; pp. 1631–1642.
43. Voorhees, E.M.; Tice, D.M. Building a question answering test collection. In Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Athens, Greece, 24–28 July 2000; pp. 200–207.
44. Dolan, B.; Brockett, C. Automatically constructing a corpus of sentential paraphrases. In Proceedings of the Third International Workshop on Paraphrasing (IWP2005), Jeju Island, Republic of Korea, 14 October 2005.
45. Chen, J.; Chen, Q.; Liu, X.; Yang, H.; Lu, D.; Tang, B. The bq corpus: A large-scale domain-specific chinese corpus for sentence semantic equivalence identification. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018; pp. 4946–4951.
46. Liu, X.; Chen, Q.; Deng, C.; Zeng, H.; Chen, J.; Li, D.; Tang, B. Lcqmc: A large-scale chinese question matching corpus. In Proceedings of the 27th International Conference on Computational Linguistics, Santa Fe, NM, USA, 21–25 August 2018; pp. 1952–1962.
47. Pennington, J.; Socher, R.; Manning, C.D. Glove: Global vectors for word representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; pp. 1532–1543.