Unsupervised Sentence Representation Learning with Syntactically Aligned Negative Samples

Zhilan Wang[1], Zekai Zhi[1], Rize Jin[1]*, Kehui Song[1], He Wang[2], Da-Jung Cho[3]

[1]School of Software, Tiangong University, Tianjin, China

[2]Beiqi Foton Motor Co., Ltd., Beijing, China

[3]Department of Software, Ajou University, Suwon, South Korea

{2330111275, 2330111274, jinrize, songkehui}@tiangong.edu.cn, wanghe10@foton.com.cn, dajungcho@ajou.ac.kr

Abstract

Sentence representation learning benefits from data augmentation strategies to improve model performance and generalization, yet existing approaches often encounter issues such as semantic inconsistencies and feature suppression. To address these limitations, we propose a method for generating Syntactically Aligned Negative (SAN) samples through a semantic importance-aware Masked Language Model (MLM) approach. Our method quantifies semantic contributions of individual words to produce negative samples that have substantial textual overlap with the original sentences while conveying different meanings. We further introduce Hierarchical-InfoNCE (HiNCE), a novel contrastive learning objective employing differential temperature weighting to optimize the utilization of both in-batch and syntactically aligned negative samples. Extensive evaluations across seven semantic textual similarity benchmarks demonstrate consistent improvements over state-of-the-art models[1].

1 Introduction

Sentence embeddings map sentences into fixed-length vectors using machine learning techniques, such as neural networks, making semantically similar sentences closer together in vector space. Sentence representation learning plays a crucial role in various natural language processing (NLP) tasks, including information retrieval (Le and Mikolov, 2014), semantic similarity comparison (Pennington et al., 2014), and translation quality evaluation (Zhang et al., 2020). With the rise of pre-trained language models (PLMs), researchers have achieved significant success by employing fine-tuning strategies based on PLMs. Specifically, BERT and its variants, such as RoBERTa (Liu et al., 1907), often use the [CLS] token from the final layer to represent sentence vectors. In recent years, with the emergence of large language models like GPT and LLaMA, researchers have started to explore the potential of these models in generating more expressive embedding representations. For example, Wang et al. (2024a) utilize Mistral-7B as an embedding model. However, such methods typically require substantial computational resources and processing power, while yielding only marginal performance improvements.

In sentence representation learning, model training typically employs either supervised or unsupervised learning approaches. For the semantic textual similarity (STS) task, supervised learning generally outperforms unsupervised methods. For instance, when using RoBERTa$_{\texttt{large}}$as the base model and trained, supervised SimCSE (Gao et al., 2022) achieved a top score of 83.76 - a score that no unsupervised model has yet surpassed. However, the supervised approach requires extensive manual annotation, which makes it difficult to scale. In contrast, unsupervised learning has become a mainstream approach due to the relative ease of acquiring unlabeled data. Nevertheless, due to the limitations of unsupervised data diversity, researchers have focused on developing various data augmentation methods to improve the performance of unsupervised models.

To overcome these challenges and enhance the efficiency of data utilization in unsupervised learning, researchers have proposed various data augmentation methods, such as cutoff, word repetition, random word deletion, noise injection, and dropout noise. However, these methods come with certain limitations. The augmented samples may alter the overall or local semantics of the original samples, potentially causing semantic inconsistencies between the original and augmented data. This, in turn, would result in inconsistent embedding distributions.

In the existing training paradigm of unsupervised

* Corresponding author

[1]Code, data, and resources are available for research purposes: https://github.com/bcai01/SAN.

Findings of the Association for Computational Linguistics:
NAACL 2025, pages 8262–8274

Original Sentences	Syntactically Aligned Negative Samples
The first series was confirmed on the 17 August 2017.	The last edition was held on the 20th May 2017.
His wife moved to Australia in 1956 along with a daughter.	They moved to london in 2013, with their son.
His first recording Artiste was the late Ebony Reign.	His first release on record was his first album release.
On Day 20, he was nominated to face the fourth eviction.	On may 20, she was nominated for being the first eviction.

Table 1: Examples of syntactically aligned negative samples, which share a significant amount of text with the original samples but have irrelevant sentence meanings.

contrastive learning, negative samples are derived from other sentences in the same mini-batch which share minimal textual overlap with anchor samples. In such cases, this can leads to feature suppression, a concept extensively discussed in the vision domain (Robinson et al., 2021) and explored in the NLP field by SNCSE (Wang et al., 2022). Feature suppression refers to a model's difficulty in distinguishing between textual similarity and semantic similarity. This can lead to inflated similarity scores for pairs with substantial lexical overlap, even when their underlying meanings diverge. To alleviate feature suppression, SNCSE introduced soft negative samples, which are constructed by adding negation to the original sentence. However, soft negative samples face a similar yet entirely opposite dilemma: these samples overlap too much with the original sentences at a literal level, differing only in the negation words, while being almost entirely semantically opposed. Such an augmentation method fails to enable the model to discern the subtle semantic variations caused by differences in lexical and grammatical structures. In this scenario, the effectiveness of soft negative samples in mitigating feature suppression is limited.

Is there a way to generate augmented samples that share substantial textual overlap with the original text while conveying distinct semantics? To better address the aforementioned issues, our work aims to explore the following two aspects: First, can we generate these samples that effectively capture the diversity, richness, and non-linearity of language concerning textual overlap and semantic relatedness? Second, how can these samples be effectively utilized within a contrastive learning framework? In contrast to existing approaches that employ rule-based methods or large language models for augmentation, we propose a novel sentence augmentation technique based on Masked Language Models (MLMs). This method generates new samples by predicting tokens that have been randomly masked. By controlling the masking ratio and the randomness of predictions, we can generate sentences that exhibit a wider range of diversity.

Specifically, we assess the semantic contribution of different words and use these values to determine the probability of masking them in the MLM task. Unlike previous negative samples, as demonstrated in Table 1, our constructed syntactically aligned negative samples maintain the original sentence's syntactic structure while altering the content words (such as nouns, verbs, adjectives, and adverbs). We refer to them as isomorphic negative samples.

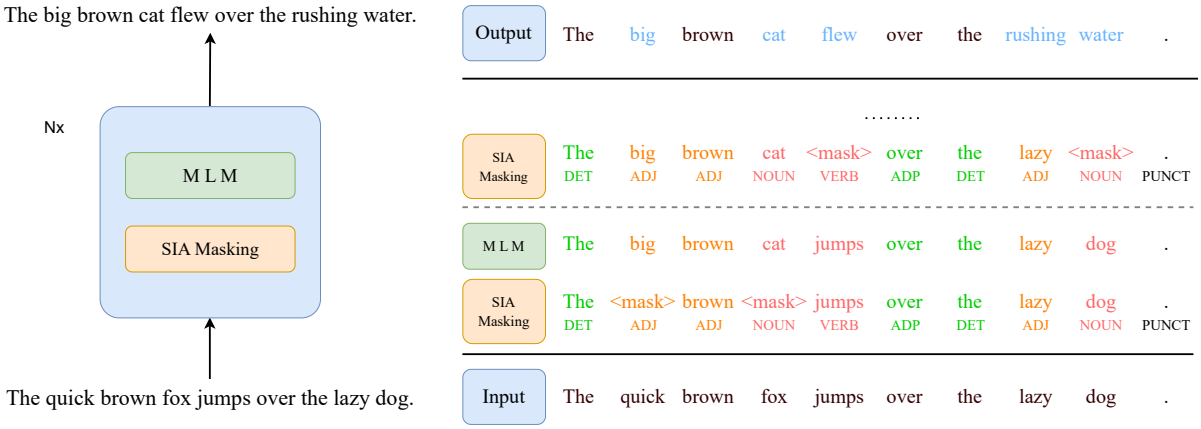
Syntactically aligned negative samples share substantial textual overlap with the original samples but differ in meaning, making it possible to disentangle textual similarity and semantic similarity in unsupervised contrastive learning.

We conduct a comprehensive evaluation across seven STS tasks, using the representative BERT and RoBERTa models as our backbones. We assess the performance of our approach against four baselines: SimCSE, SNCSE, RankCSE (Liu et al., 2023), and RLRD (Wang et al., 2024b). We improve the embedding of state-of-the-art models on all seven STS tasks, achieving superior results. A series of ablation studies further confirms that our approach successfully disentangles textual similarity from semantic similarity, thereby mitigating feature suppression.

2 Related Work

2.1 Sentence Representation Learning

Early sentence representation learning relied on methods like Bag of Words, TF-IDF, and Word2Vec. With the rise of PLMs such as BERT (Devlin, 2018), transformer-based models have become the standard for sentence embeddings, as seen in ConSERT (Yan et al., 2021), SimCSE (Gao et al., 2022), and PromptBERT (Jiang et al., 2022). PLMs typically use the [CLS] token or pooled representation, yielding higher-quality embeddings. Recent advances like RankCSE further refines sentence embeddings by incorporating



(a) The architecture of SAN

(b) An illustration of SAN data augmentation process

Figure 1: (a) The architecture of the Semantic Importance-Aware (SIA) MLM with multiple masking iterations. (b) An illustration of the SAN data augmentation process, where words are masked based on their semantic contributions. Red indicates the highest masking probability, followed by yellow, green, and black, representing progressively lower probabilities.

ranking consistency and distillation into contrastive learning with teacher models.

2.2 Data Augmentation Methods

The current trend in sentence embeddings primarily relies on PLMs such as BERT and RoBERTa, combined with contrastive learning for training and inference. Originally applied in computer vision, contrastive learning has been adopted in sentence representation learning and has proven to be an effective method. One of the key challenges in contrastive learning is the construction of positive and negative sample pairs. Supervised SimCSE utilizes the NLI (Natural Language Inference) dataset for training, achieving better performance compared to unsupervised methods. However, labeled datasets are often difficult to obtain. Therefore, designing and employing data augmentation methods to construct positive and negative sample pairs, as well as applying unsupervised contrastive learning, have become mainstream approaches.

ConSERT (Yan et al., 2021) employs adversarial attacks, token shuffling, truncation, and dropout strategies on the token embedding matrix to create positive samples. SNCSE (Wang et al., 2022) introduces the concept of soft negative samples and designs the Bidirectional Margin Loss (BML), demonstrating better performance than SimCSE on both BERT and RoBERTa. Existing data augmentation methods include word repetition, random word deletion, word shuffling, and cutoff. However, these methods share a common limitation. First, they are primarily applied to construct positive samples, but the randomness introduced by these operations can alter the local or overall semantics of the sentence, leading to semantic inconsistencies. Second, soft negative sentence pairs share a large portion of the text or are even completely identical, which can hinder the model's ability to correctly interpret sentence meaning and impair its capacity to effectively learn sentence representations.

3 Method

3.1 Syntactically Aligned Data Augmentation

In existing contrastive learning approaches for STS, the design of positive and negative sample pairs can lead to feature suppression, where the model overly relies on superficial features, such as sentence length and lexical overlap, while neglecting deeper semantic nuances. This overreliance not only adversely affects the model's generalization ability in complex linguistic tasks but also results in the degradation of the model's representational space, causing an inconsistent distribution of embeddings.

In linguistics, verbs and nouns are often considered the core elements in constructing sentence semantics, while the semantic contributions of modifying words such as adjectives and adverbs are relatively minor. This perspective is supported by various linguistic theories, including Transformational-Generative Grammar (TGG) and Systemic Functional Grammar (SFG). Based on this theoretical

POS	Score	POS	Score	POS	Score
NOUN	9	VERB	9	ADJ	8
PROPN	8	ADV	7	PRON	7
CCONJ	6	DET	6	ADP	5
CONJ	5	SCONJ	5	AUX	4
NUM	4	PART	3	INTJ	2
PUNCT	1	SYM	1	X	1

Table 2: Human-annotated importance scores indicating the semantic impact of different parts of speech (POS).

foundation, we propose a method for generating syntactically aligned negative samples by modifying words that contribute significantly to meaning, thus creating syntactically similar pairs with high textual overlap but opposite or irrelevant semantics.

Our study follow three steps: (1) assessing the magnitude of semantic effects of various parts of speech; (2) conducting parts-of-speech tagging on the training dataset; (3) replacing words that significantly influence meaning. The initial step involves quantifying the semantic influence of parts of speech, where the "en_core_web_sm" model of spaCy[2] will be utilized as the tagging tool. This approach not only automates the tagging process but also allows for precise quantification of the semantic impact of different parts of speech through manual review.

Related research has shown that nouns and verbs contribute most significantly to the overall semantics of sentences (Pollard and Sag, 1994; Bresnan, 2001; Chomsky, 1957; Matthiessen and Halliday, 2009). Once these parts of speech are removed, the original meaning of the sentence becomes difficult to discern, leading to an assigned importance score of 9 for these categories. In contrast, adjectives, adverbs, and pronouns primarily contribute to semantic nuances, resulting in scores ranging from 6 to 8. Auxiliary verbs and prepositions play a role in grammatical structure comprehension but have a lesser direct semantic contribution, receiving scores of 3 to 5. Words that are difficult to classify are marked as "X," typically indicating a lower semantic importance. Detailed quantification results are presented in Table 2.

We employ an MLM to perform token replacement, leveraging its ability to generate suitable replacements based on context, ensuring semantic consistency and diversity. By masking the target words and feeding them into the MLM, we obtain sentences with appropriate substitutions.

[2]The model and tool are available at: https://spacy.io and https://huggingface.co/spacy/en_core_web_sm.

We adopt a static tagging approach by first compiling a lexicon and assigning parts-of-speech tags to each entry. In subsequent data processing, we assign a masking probability to each word based on its part of speech and semantically importance score. In the experiments below, we use the semantically importance score divided by 20 to determine the masking probability for each part of speech. Further experiments related to the determination of the masking probability can be found in Appendix D. Verbs and nouns are assigned the highest masking probabilities, while punctuation and "X" receive the lowest. After masking, the masked sentence is fed into the MLM to generate replacements. This mask-predict cycle is repeated several times to create syntactically aligned negative samples. Notably, in each iteration, the overall masking probabilities are kept low to avoid distorting the output by masking too many words at once. Through multiple iterations, even with low masking probabilities per iteration, we are able to generate diverse augmented data, as shown in Figure 1.

Our method performs well on longer sentences; however, it encounters issues when applied to shorter sentences or phrases. In such cases, the process can lead to the replacement of most words with punctuation marks or other meaningless tokens. For instance, the sentence "Stafford acting General Secretary." might be augmented into "-., p." after applying data augmentation, which results in nonsensical token combinations that do not align with the definition of syntactically aligned negative samples and could negatively impact model training.

3.2 Hierarchical-InfoNCE

We first present the mathematical formulation for InfoNCE, as shown in Equation 1. In InfoNCE, negative samples are the other examples in the mini-batch excluding the anchor itself, and these negative samples generally share little textual similarity with the anchor. The objective of InfoNCE is to maximize the similarity between the anchor and the positive sample while minimizing the similarity between the anchor and the negative samples.

$$\mathcal{L}_{\text{InfoNCE}} = -\sum_{i=1}^{N} \log \frac{e^{\text{sim}(h_i, h_i^+)/\tau}}{\sum_{j=1}^{N} e^{\text{sim}(h_i, h_j)/\tau}}, \quad (1)$$

where τ is a temperature hyperparameter and $\text{sim}(h_1, h_2)$ is the cosine similarity $\frac{h_1^\top h_2}{\|h_1\| \cdot \|h_2\|}$. In

this work, we encode input sentences using a pre-trained language model such as BERT or RoBERTa: $h = f_\theta(x)$, and then fine-tune all the parameters.

Based on the discussion in Section 3.1, the similarity between syntactically aligned negative samples and the anchor should be lower than the similarity between the anchor and the positive samples. To further optimize this process, we propose adjusting the temperature parameter to influence the model's behavior. By increasing the temperature between syntactically aligned negative samples, we can thus decouple textual similarity from semantic similarity. This adjustment prevents the model from focusing excessively on syntactically aligned negative samples and neglecting other types of negative samples. While this approach may make the model more sensitive to certain negative samples, it risks reducing its ability to distinguish between others. Therefore, we increase the temperature between anchor and syntactically aligned negative samples, as reflected in our improved HiNCE, shown in Equation 2.

$$\mathcal{L}_{\text{HiNCE}} = -\sum_{i=1}^{N} \log \frac{e^{\text{sim}(h_i, h_i^+)/\tau_1}}{\sum_{j=0}^{N}(e^{\text{sim}(h_i, h_j)/\tau_1} + e^{\text{sim}(h_i, h_j^-)/\tau_2})}, \quad (2)$$

where τ_1 represents the temperature hyperparameter for the similarity between positive and negative sample pairs, while τ_2 is the temperature for the syntactically aligned negative sample pairs. h_i^- denotes the sentence embedding of the syntactically aligned negative sample produced by the encoder.

3.3 Combination with Other Methods

Syntactically aligned negative samples can be integrated into most existing methods within the same domain using HiNCE or other loss functions. We combine syntactically aligned negative samples with SimCSE (Gao et al., 2022), SNCSE, RankCSE (Liu et al., 2023), and RLRD (Wang et al., 2024b).

Combination with SimCSE The unsupervised SimCSE utilizes the standard InfoNCE loss function. To apply our method within the unsupervised SimCSE framework, we replace it with the HiNCE loss function, which is expressed in Equation 3.

$$\mathcal{L}_{\text{SimCSE+SAN}} = \mathcal{L}_{\text{HiNCE}}. \quad (3)$$

Combination with SNCSE. SNCSE constructs soft negative samples through explicit negation and trains using both the InfoNCE and BML loss. We replace the InfoNCE loss function with HiNCE while keeping the other components in their original configuration. The final loss function is expressed in Equation 4.

$$\mathcal{L}_{\text{SNCSE+SAN}} = \mathcal{L}_{\text{HiNCE}} + \lambda \mathcal{L}_{\text{BML}}. \quad (4)$$

Combination with RankCSE. RankCSE integrates ranking consistency and ranking distillation with contrastive learning into a unified framework by introducing teacher models.

The $\mathcal{L}_{\text{rank}}$ term can take one of two forms: ListMLE (Xia et al., 2008) or ListNet (Cao et al., 2007). We choose ListMLE loss as the baseline for improvement. During training, the value of ListMLE loss is usually three orders of magnitude larger than that of InfoNCE loss in in the RankCSE framework. Given the minimal contribution of the InfoNCE loss term within the total RankCSE loss, we apply syntactically aligned negative samples to both HiNCE and ListMLE. ListMLE, a machine learning algorithm for list ranking tasks, optimizes ranking outcomes by maximizing the pairwise ranking probabilities of all items in the list. We replace its inputs with anchor samples and corresponding syntactically aligned negative samples to derive the ListMLE+SAN loss term, as indicated in Equation 5. The final loss function combining RankCSE with our method is presented in Equation 6.

$$\mathcal{L}_{\text{ListMLE+SAN}} = -\sum_{i=1}^{N} \log P(\pi_i^T | S^{'}(x_i), \tau_3), \quad (5)$$

where $S^{'}(x_i) = \{\text{sim}(h_i, h_j^-)/\tau_3\} \sum_{j=1}^{N}$, denoting the similarity list between the anchor samples obtained from the encoder and the embedding vectors of the syntactically aligned negative samples within the current batch. π_i^T represents the sorted indices of the similarity scores calculated by the teacher model.

It is noteworthy that low-quality samples within syntactically aligned negative samples can significantly impact the performance of ListMLE. To address this issue, we introduce a post-processing step where we replace these low-quality samples with soft negative samples from SNCSE. This refinement enhances the overall quality of our training data and mitigates potential performance degradation. $P(\pi|S, \tau)$ is the permutation probability $\prod_{i=1}^{n} \frac{e^{S\pi(i)/\tau}}{\sum_{j=i}^{n} e^{S\pi(j)/\tau}}$.

$$\begin{aligned}\mathcal{L}_{\text{RankCSE+SAN}} = \beta \mathcal{L}_{\text{consistency}} + \gamma \mathcal{L}_{\text{ListMLE}} \\ + \mathcal{L}_{\text{HiNCE}} + \lambda \mathcal{L}_{\text{ListMLE+SAN}},\end{aligned} \quad (6)$$

Methods	STS12	STS13	STS14	STS15	STS16	STS-B	SICK-R	Avg.
SimCSE-BERT$_{\text{base}}$	68.59	82.62	74.60	81.87	78.67	78.06	72.04	76.64
+SAN	70.92	81.77	75.20	82.57	78.88	78.66	71.33	77.05↑
SNCSE-BERT$_{\text{base}}$	70.80	84.72	76.81	83.44	79.73	80.82	74.26	78.65
+SAN	71.12	84.48	77.12	83.77	80.68	81.52	**75.24**	79.13↑
RankCSE-BERT$_{\text{base}}$	74.89	85.85	77.86	84.82	**81.78**	81.95	73.94	80.16
+SAN	75.34	86.03	78.26	84.72	81.14	81.84	74.97	80.33↑
RLRD-BERT$_{\text{base}}$	75.44	86.31	79.15	85.81	81.34	82.91	75.12	80.87
+SAN	**75.68**	**86.38**	**79.47**	**85.93**	81.43	**83.05**	75.15	**81.01**↑
SimCSE-BERT$_{\text{large}}$	69.01	80.40	73.50	82.71	77.89	77.18	73.95	76.38
+SAN	70.40	84.68	77.04	84.16	78.86	78.88	75.13	78.45↑
SNCSE-BERT$_{\text{large}}$	71.48	86.25	78.12	85.24	80.17	81.84	75.15	79.75
+SAN	72.39	86.50	78.38	85.41	80.45	82.11	75.08	80.05↑
RankCSE-BERT$_{\text{large}}$	74.68	86.00	78.59	85.27	80.98	81.15	74.15	80.12
+SAN	74.99	86.04	78.92	85.22	80.38	81.17	74.96	80.24↑
RLRD-BERT$_{\text{large}}$	75.47	86.68	79.22	86.13	80.79	**82.62**	75.25	80.88
+SAN	**75.65**	**86.80**	**79.26**	**86.20**	**81.03**	82.46	**75.68**	**81.01**↑
SimCSE-RoBERTa$_{\text{base}}$	68.99	81.46	73.53	81.85	81.14	80.52	69.31	76.69
+SAN	69.14	81.54	73.09	81.61	81.97	81.08	70.14	76.94↑
SNCSE-RoBERTa$_{\text{base}}$	69.03	83.40	75.56	84.01	80.21	81.31	71.39	77.84
+SAN	70.85	83.90	**76.50**	**84.61**	81.24	82.42	**72.29**	78.83↑
RankCSE-RoBERTa$_{\text{base}}$	**73.34**	84.11	75.65	83.97	82.71	**82.89**	70.67	79.05
+SAN	73.25	**84.44**	75.99	84.25	**82.72**	82.68	70.84	**79.17**↑
SimCSE-RoBERTa$_{\text{large}}$	71.24	84.07	76.27	84.79	82.14	82.53	71.02	78.87
+SAN	71.62	84.32	75.94	84.68	81.38	82.61	71.75	78.90↑
SNCSE-RoBERTa$_{\text{large}}$	72.04	**85.98**	**79.32**	86.39	**82.45**	**83.95**	**76.88**	80.86
+SAN	73.17	85.57	79.22	**86.72**	82.05	83.61	76.83	**81.02**↑
RankCSE-RoBERTa$_{\text{large}}$	74.05	84.59	77.14	85.62	81.87	83.20	71.25	79.67
+SAN	**74.32**	84.59	76.86	85.39	82.39	83.19	71.69	79.78↑

Table 3: Main results of various contrastive learning methods on seven semantic textual similarity (STS) datasets. Each method is evaluated on full test sets by Spearman's correlation, "all" setting. Bold marks the best result among all competing methods under the same backbone model.

where β, γ, λ are hyperparameters that control the contribution of each loss component.

Combining with RLRD. The RLRD method enhances RankCSE by incorporating reinforcement learning techniques and additional loss functions, specifically InfoNCE, BML, and ListMLE. We replace the InfoNCE with HiNCE and introduce a new loss term, as presented in Equation 7.

$$\begin{aligned}\mathcal{L}_{\text{RLRD+SAN}} = \mu\mathcal{L}_{\text{BML}} + \lambda\mathcal{L}_{\text{ListMLE}} \\ + \zeta\mathcal{L}_{\text{HiNCE}} + \beta\mathcal{L}_{\text{ListMLE+SAN}}.\end{aligned} \tag{7}$$

4 Experiment

4.1 Datasets

We adopt a method similar to SimCSE, randomly selecting one million sentences from Wikipedia for unsupervised training. Ultimately, we compute similarity scores for sentence pairs in the STS dataset and calculate Spearman's correlation with human ratings as the final evaluation metric. We evaluate our method on the Semantic Textual Similarity (STS) dataset, which consists of seven subtasks, including STS12-STS16 (Agirre et al., 2012, 2013, 2014, 2015, 2016), STSbenchmark (STS-B) (Cer et al., 2017), and SICK-Relatedness (SICK-R) (Marelli et al., 2014).

4.2 Implementation Details

We retrained four unsupervised sentence embedding models—SimCSE (Gao et al., 2022), SNCSE (Wang et al., 2022), RankCSE (Liu et al., 2023), and RLRD (Wang et al., 2024b)—integrating them with our proposed syntactically aligned augmentation approach. We utilized pre-trained models including BERT-base-

uncased, BERT-large-uncased, RoBERTa-base, and RoBERTa-large as encoders across all experiments. We employed the Wiki dataset from SimCSE (Gao et al., 2022) as our self-supervised training dataset. All experiments were conducted on a single A100 40G GPU with mixed precision (FP16) training. We reproduced the results using the original settings from the corresponding papers. Further details on the training can be found in Appendix C.

4.3 Main Results

Table 3 presents the results of a single run, showing the performance of our method when integrated with various existing works. As demonstrated, our approach surpasses baseline scores in most cases. Notably, when combined with SimCSE and SNCSE, our method achieves significant improvements over the original methods. Specifically, the average Spearman's correlation coefficient for SimCSE-BERT$_{\texttt{base}}$increased from 76.64 to 77.05, while for SNCSE-BERT$_{\texttt{large}}$, it rose from 78.65 to 79.75. Additionally, when applied to RankCSE, our method yielded even more pronounced results, with the average Spearman's correlation coefficient for RankCSE-BERT$_{\texttt{base}}$improving from 80.16 to 80.33, and for RankCSE-BERT$_{\texttt{large}}$, from 80.12 to 80.24. Furthermore, among methods utilizing BERT$_{\texttt{base}}$, the combination with RLRD achieved the highest result of 81.01. These confirm that integrating our method with existing approaches stably improves the quality of sentence representation.

4.4 Downstream Tasks

To assess the generalization capability of our method, we conducted extensive experiments across multiple task types: reranking, retrieval, and classification. For classification tasks, we evaluated our method on ten diverse tasks: AmazonCounterfactual (O'Neill et al., 2021), AmazonReviews (Keung et al., 2020), Banking77 (Casanueva et al., 2020), Emotion (Saravia et al., 2018), MassiveIntent (FitzGerald et al., 2022), MassiveScenario (FitzGerald et al., 2022), MTOPDomain (Li et al., 2021), MTOPIntent (Li et al., 2021), ToxicConversations (cjadams, 2019) and TweetSentimentExtraction (Maggie, 2020). We report the classification accuracy as the main metric. We further evaluated our method on four reranking and retrieval tasks: AskUbuntuDupQuestions (Lei et al., 2016), SciDocsRR (Cohan et al., 2020), StackOverflowDupQuestions (Liu et al., 2018) and QuoraRetrieval (Thakur et al., 2021). We report the mean MRR@1 and MAP@1 as the main results. As shown in Figure 2, both SimCSE+SAN and RankCSE+SAN outperformed their corresponding baseline models in reranking and retrieval tasks, with RankCSE+SAN achieving the highest scores in both tasks. Additionally, we assessed our method's performance on sentence-level classification tasks, with Table 4 presenting the average accuracy across ten sentence-level classification tasks. The results indicate that our method enhances the performance of certain baseline models on these tasks.

Model	Avg. Classification Accuracy
SimCSE	0.6068
+SAN	0.6113↑
SNCSE	<u>0.6170</u>
+SAN	0.6106
RankCSE	0.6169
+SAN	**0.6290**↑
RLRD	0.6145
+SAN	0.6072

Table 4: Performance evaluation on classification tasks. Bold values highlight the highest classification accuracy, while underlined values denote the second-highest accuracy.

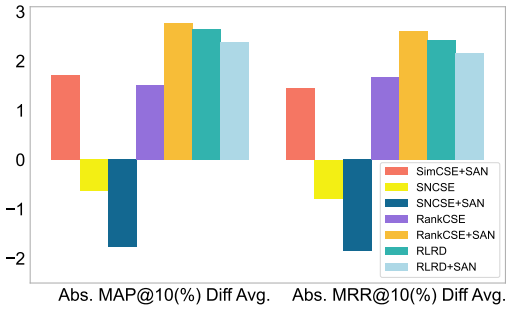


Figure 2: Absolute performance difference on reranking and retrieval tasks compared to SimCSE.

5 Ablation Study

5.1 Influence of Hyperparameters in HiNCE Loss

The performance of HiNCE is influenced by two hyperparameters: the dropout probabilities for syntactically aligned negative samples and the temperature hyperparameter associated with the similarity of anchor samples and syntactically aligned negative embeddings. This section examines the effects of these parameters on model performance. The

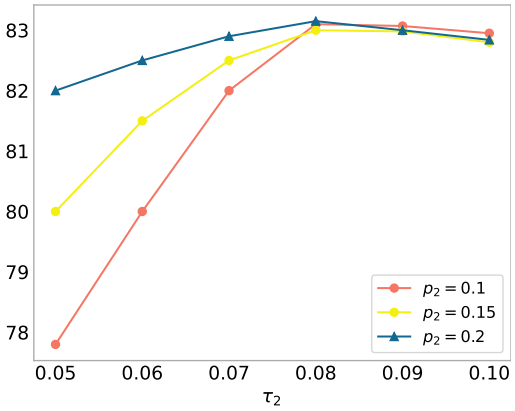


Figure 3: Influence of hyperparameters of HiNCE loss. τ_2 is the temperature hyperparameter corresponding to the similarity of sentence embeddings for anchor samples and syntactically aligned negative samples. p_2 is the dropout probability of the model for syntactically aligned negative samples.

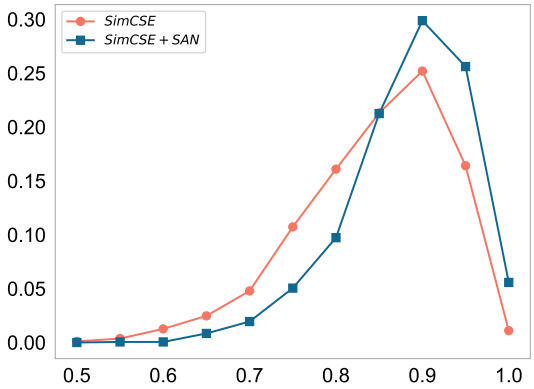


Figure 4: Cosine similarity distribution on different pairs.

dropout probabilities for syntactically aligned negative samples were adjusted within the range of [0.1, 0.15, 0.2], while the temperature hyperparameter was varied between [0.05, 0.06, 0.07, 0.08, 0.09, 0.1]. As shown in Figure 3, when the dropout probabilities remain constant, model performance improves with increasing values of τ_2. Notably, when τ_2 is greater than or equal to 0.08, models with different dropout probabilities exhibit relatively high and consistent performance. These findings indicate that the temperature hyperparameter τ_2 is a critical factor, and its selection is essential for optimizing model performance. In contrast, while the dropout probabilities do impact performance to some extent, the variation is not as pronounced.

5.2 Alleviation of Feature Suppression

To conduct a more comprehensive analysis of our method's impact on feature suppression, we selected 1,900 sentence pairs from the STS task dataset that have a similarity score of at least 4 and a lexical overlap ratio below 0.6. These sentence pairs, while exhibiting low textual similarity, exhibit high semantic similarity, making them challenging for discrimination.

As illustrated in Figure 4, both SimCSE+SAN and SimCSE peak around 0.9; however, the distribution of SimCSE+SAN is more concentrated near this value. In contrast, SimCSE exhibits greater variability in its similarity estimates across a wider range of sentence pairs. These results highlight the strength of our proposed method, as SimCSE+SAN achieves a more stable and dependable performance, particularly for challenging cases.

5.3 Comparison with LLM-based Data Augmentation Methods

To further investigate the generalizability of our approach, we compared it with data augmentation methods based on LLMs. In the experiments, the baseline used the NLI-partial dataset (Zhang et al., 2023), which is constructed from anchor samples in the NLI dataset, with positive and hard negative samples generated by GPT-3.5. We conducted the comparative analysis using the following two strategies: (1) directly using syntactically aligned negative samples as hard negative samples, and (2) integrating syntactically aligned negative samples into the training process via the HiNCE loss term. The results are shown in Table 5.

The experimental results indicate that directly using syntactically aligned negative samples as hard negative samples yields inferior performance compared to the original method. This is primarily because the design of syntactically aligned negative samples focuses on enhancing the model's ability to distinguish subtle semantic differences, which differs from the primary objective of hard negative samples. However, when syntactically aligned negative samples are combined with the hard negative sampling method, the model's performance improves. This outcome not only demonstrates the effectiveness of our approach but also further validates its generalizability across different scenarios.

6 Conclusion

In this paper, we propose a method that leverages the semantic impact of different parts of speech to generate syntactically aligned negative samples using MLM. We also introduce HiNCE loss, which

Loss	InfoNCE	NL	HiNCE
STS-B	84.99	83.07	**85.83**

Table 5: Comparison of different sample combination strategies (STS-B development set, Spearman's correlation). 'NL' refers to using syntactically aligned negative samples as hard negative samples directly, while 'HiNCE' indicates integrating syntactically aligned negative samples into the training process through the HiNCE loss term.

utilizes these syntactically aligned negative samples to improve sentence embedding performance. Furthermore, our methods demonstrate improved performance when combined with other existing methods.

We evaluate our method across multiple tasks, including semantic textual similarity, re-ranking, retrieval, and classification, showcasing its excellent performance and generalization capability. Additionally, we conduct detailed ablation studies to analyze the mechanisms of syntactically aligned negative samples and HiNCE loss, and to evaluate their impact on the model's performance.

7 Limitations

Our study has several limitations. First, semantic impact values for different parts of speech were manually assigned and need further refinement. Second, our model's input size increased by 50%, extending training time. Third, syntactically aligned negative samples were only applied to HiNCE and ListMLE. Lastly, all experiments were conducted on English datasets, with no multilingual evaluation.

STS technology also poses risks, including privacy concerns, potential biases in training data, and misuse for generating misleading content.

Future work will focus on refining semantic importance scoring, developing more efficient algorithms, and expanding the use of syntactically aligned negative samples.

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.

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, and Aitor Gonzalez-Agirre. 2012. Semeval-2012 task 6: A pilot on semantic textual similarity. in* sem 2012: The first joint conference on lexical and computational semantics–volume 1: Proceedings of the main conference and the shared task, and volume 2: Proceedings of the sixth international workshop on semantic evaluation (semeval 2012). *Association for Computational Linguistics. URL http://www. aclweb. org/anthology/S12-1051*.

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.

Joan Bresnan. 2001. *Lexical-Functional Syntax*. Blackwell Publishers, Oxford, UK.

Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. 2007. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th International Conference on Machine Learning*, ICML '07, page 129–136, New York, NY, USA. Association for Computing Machinery.

Iñigo Casanueva, Tadas Temčinas, Daniela Gerz, Matthew Henderson, and Ivan Vulić. 2020. Efficient intent detection with dual sentence encoders. In *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*, pages 38–45, Online. 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 cross-lingual focused evaluation. *arXiv preprint arXiv:1708.00055*.

Noam Chomsky. 1957. *Syntactic Structures*. Mouton, The Hague.

inversion Jeffrey Sorensen Lucas Dixon Lucy Vasserman nithum cjadams, Daniel Borkan. 2019. Jigsaw unintended bias in toxicity classification.

Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel S. Weld. 2020. Specter: Document-level representation learning using citation-informed transformers. In *ACL*.

Jacob Devlin. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Jack FitzGerald, Christopher Hench, Charith Peris, Scott Mackie, Kay Rottmann, Ana Sanchez, Aaron Nash, Liam Urbach, Vishesh Kakarala, Richa Singh, Swetha Ranganath, Laurie Crist, Misha Britan, Wouter Leeuwis, Gokhan Tur, and Prem Natarajan. 2022. Massive: A 1m-example multilingual natural language understanding dataset with 51 typologically-diverse languages. *Preprint*, arXiv:2204.08582.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2022. Simcse: Simple contrastive learning of sentence embeddings. *Preprint*, arXiv:2104.08821.

Ting Jiang, Jian Jiao, Shaohan Huang, Zihan Zhang, Deqing Wang, Fuzhen Zhuang, Furu Wei, Haizhen Huang, Denvy Deng, and Qi Zhang. 2022. Promptbert: Improving bert sentence embeddings with prompts. *arXiv preprint arXiv:2201.04337*.

Phillip Keung, Yichao Lu, György Szarvas, and Noah A. Smith. 2020. The multilingual Amazon reviews corpus. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4563–4568, Online. Association for Computational Linguistics.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 1188–1196, Bejing, China. PMLR.

Tao Lei, Hrishikesh Joshi, Regina Barzilay, Tommi Jaakkola, Kateryna Tymoshenko, Alessandro Moschitti, and Lluís Màrquez. 2016. Semi-supervised question retrieval with gated convolutions. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1279–1289, San Diego, California. Association for Computational Linguistics.

Haoran Li, Abhinav Arora, Shuohui Chen, Anchit Gupta, Sonal Gupta, and Yashar Mehdad. 2021. MTOP: A comprehensive multilingual task-oriented semantic parsing benchmark. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2950–2962, Online. Association for Computational Linguistics.

Jiduan Liu, Jiahao Liu, Qifan Wang, Jingang Wang, Wei Wu, Yunsen Xian, Dongyan Zhao, Kai Chen, and Rui Yan. 2023. Rankcse: Unsupervised sentence representations learning via learning to rank. *Preprint*, arXiv:2305.16726.

Xueqing Liu, Chi Wang, Yue Leng, and ChengXiang Zhai. 2018. Linkso: a dataset for learning to retrieve similar question answer pairs on software development forums. In *Proceedings of the 4th ACM SIGSOFT International Workshop on NLP for Software Engineering*, pages 2–5.

Y Liu, M Ott, N Goyal, J Du, M Joshi, D Chen, O Levy, M Lewis, L Zettlemoyer, and V Stoyanov. 1907. Roberta: A robustly optimized bert pretraining approach. arxiv [preprint](2019). *arXiv preprint arXiv:1907.11692*.

Wei Chen Maggie, Phil Culliton. 2020. Tweet sentiment extraction.

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).

Christian MIM Matthiessen and Michael Alexander Kirkwood Halliday. 2009. Systemic functional grammar: A first step into the theory.

James O'Neill, Polina Rozenshtein, Ryuichi Kiryo, Motoko Kubota, and Danushka Bollegala. 2021. I wish I would have loved this one, but I didn't – a multilingual dataset for counterfactual detection in product review. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7092–7108, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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

Carl Pollard and Ivan A. Sag. 1994. *Head-Driven Phrase Structure Grammar*. The University of Chicago Press, Chicago.

Joshua Robinson, Li Sun, Ke Yu, Kayhan Batmanghelich, Stefanie Jegelka, and Suvrit Sra. 2021. Can contrastive learning avoid shortcut solutions? In *Advances in Neural Information Processing Systems*, volume 34, pages 4974–4986. Curran Associates, Inc.

Elvis Saravia, Hsien-Chi Toby Liu, Yen-Hao Huang, Junlin Wu, and Yi-Shin Chen. 2018. CARER: Contextualized affect representations for emotion recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3687–3697, Brussels, Belgium. Association for Computational Linguistics.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Hao Wang, Yangguang Li, Zhen Huang, Yong Dou, Lingpeng Kong, and Jing Shao. 2022. Sncse: Contrastive learning for unsupervised sentence embedding with soft negative samples. *Preprint*, arXiv:2201.05979.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024a. Improving text embeddings with large language models. *Preprint*, arXiv:2401.00368.

Xintao Wang, Rize Jin, and Shibo Qi. 2024b. Reinforced multi-teacher knowledge distillation for unsupervised sentence representation. In *Artificial Neural Networks and Machine Learning – ICANN 2024*, pages 320–332, Cham. Springer Nature Switzerland.

Fen Xia, Tie-Yan Liu, Jue Wang, Wensheng Zhang, and Hang Li. 2008. Listwise approach to learning to rank: theory and algorithm. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, page 1192–1199, New York, NY, USA. Association for Computing Machinery.

Yuanmeng Yan, Rumei Li, Sirui Wang, Fuzheng Zhang, Wei Wu, and Weiran Xu. 2021. Consert: A contrastive framework for self-supervised sentence representation transfer. *Preprint*, arXiv:2105.11741.

Junlei Zhang, Zhenzhong Lan, and Junxian He. 2023. Contrastive learning of sentence embeddings from scratch. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3916–3932, Singapore. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. *Preprint*, arXiv:1904.09675.

A Static tagging and dynamic tagging

In Section 3.1, we introduce two parts-of-speech tagging approaches: dynamic and static tagging. Dynamic tagging uses spaCy to tag each corpus sample, ensuring higher accuracy but with increased processing time. Static tagging pre-assigns part of speech to all vocabulary words using a predefined dictionary and tags the corpus via a mapping table. To balance accuracy and efficiency, we adopt static tagging with $\text{BERT}_{\texttt{base}}$'s vocabulary as the dictionary in our experiments.

	$\text{BERT}_{\texttt{base}}$			
	SimCSE +SAN	**SNCSE +SAN**	**RankCSE +SAN**	**RLRD +SAN**
τ_2	0.08	0.1	0.05	0.1
p_2	0.2	0.2	0.2	0.2
$\mathcal{L}_{\text{HiNCE}}$	Y	Y	Y	Y
Post-processing	-	-	Y	Y
$\mathcal{L}_{\text{ListMLE+SAN}}$	-	-	0.4	0.2

Table 6: Hyperparameters of SAN.

B Comparison with Existing Methods

We compared various methods using $\text{BERT}_{\texttt{base}}$as the backbone, with results from original papers (Table 9). Our method improves SimCSE, SNCSE, and RLRD. For RankCSE, it achieves 80.33, surpassing the reproduced 80.16 but slightly below the original 80.36. This gap may stem from using a community implementation, as RankCSE's official code is unavailable, leading to potential differences in code details and hyperparameters.

C Hyperparameters of HiNCE

We re-trained four previous unsupervised sentence embedding methods on STS tasks – SimCSE, SNCSE, RankCSE and RLRD – with our novel method, SAN. All embedding training models used $\text{BERT}_{\texttt{base}}$(110M), $\text{BERT}_{\texttt{large}}$(340M), $\text{RoBERTa}_{\texttt{base}}$(125M) or $\text{RoBERTa}_{\texttt{large}}$(355M) as a starting checkpoint. We replicated the results using the original settings from the corresponding papers, adjusting only the relevant parameters for the additional loss terms in the experiments that combined our method as shown in Table 6. We carry out grid-search of $\tau_2 \in \{0.05, 0.06, 0.07, 0.08, 0.09, 0.1\}$, $p_2 \in \{0.1, 0.15, 0.2\}$ and the weight of $\mathcal{L}_{\text{ListMLE+SAN}} \in \{0.1, 0.15, 0.2\}$, where τ_2 is the temperature for the syntactically aligned negative sample pairs. p_2 is the dropout probabilities of the model for syntactically aligned negative samples. post-processing is lower-quality samples in the syntactically aligned negative samples dataset are replaced with soft negative samples generated from SNCSE. $\mathcal{L}_{\text{ListMLE+SAN}}$ is the ListMLE loss function using the similarity list of anchor samples and syntactically aligned negative samples as input. Following Gao et al. (2022), we used the validation performance on STS-B (one of the seven STS benchmark datasets) to select the best models.

Scaling Factor	10	20	30	40
Avg.	76.94	**77.05**	75.82	76.55

Table 7: Performance on STS Test Set with Different Masking Probability Scaling Factors (Semantically Importance Score divided by {10, 20, 30, 40}).

D Experiments on Masking Probability Assignment

We evaluate different masking probabilities for each part of speech by dividing the semantically importance score by 10, 20, 30, and 40. Experiments are conducted on $\text{BERT}_{\text{base}}$with the Wiki1M dataset and HiNCE loss, using hyperparameters consistent with the experimental section and Appendix C. Table 7 reports the model's average Spearman correlation on the STS test set, with the best results observed when dividing by 20. This may be because a low masking probability leads to insufficient differences between syntactically aligned negative samples and the original samples, limiting the training signal, while a high masking probability causes excessive differences that disrupt contextual information, both negatively impacting performance.

E Computational Cost

All experiments were conducted on a single A100 (40GB) GPU with the same initialization and random seed. $\text{BERT}_{\text{base}}$was used to generate syntactically aligned negative samples, with an average training time of 2.5 hours. As shown in Table 10, incorporating our method increased training times for all baseline models, averaging 25% longer due to the additional negative samples introduced by our approach.

F Case Study

We present several examples from the STS dataset with their similarity scores in Table 11, where the similarity scores of the sentence pairs exceed 4.8 (with a maximum score of 5) and the lexical overlap is below 0.6. These indicate that, after incorporating our method, SimCSE generates more effective similarity scores that are closer to the ground truth. This further demonstrates that while SimCSE primarily captures high-level semantic information through contrastive learning, our approach enhances its ability to capture more nuanced semantic details.

G Using LLM to generate Syntactically Aligned Negative Samples

In addition to using MLM to generate syntactically aligned negative samples, we also explored the use of an LLM (GPT-4o mini) to generate them. In this approach, we used the first 200k instances from the Wiki1M dataset and generated samples based on the prompt shown in Table 12, which were then used for training. The experimental results, as shown in Table 8, demonstrate that the SAN-based method achieves improvements across various metrics, further validating the effectiveness of our approach. Moreover, the syntactically aligned negative samples generated by the LLM outperform those generated by the MLM in terms of overall performance. This suggests that LLM-generated negative samples offer data quality and diversity, providing more substantial support for the optimization of contrastive learning.

Use SAN	Generation Method	Avg.
$\text{SimCSE-BERT}_{\text{base}}$		
N	-	76.74
Y	MLM	77.05
Y	LLM	**77.57**
$\text{SNCSE-RoBERTa}_{\text{large}}$		
N	-	80.86
Y	MLM	81.02
Y	LLM	**81.53**

Table 8: Comparison of training results using syntactically aligned negative samples generated by different methods. Syntactically aligned negative samples are introduced into the training process via HiNCSE, and the LLM used refers to GPT-4o mini. The reported results represent the average performance on the STS task test set.

H Scientific Artifacts and Licensing

In this study, we used several open-source artifacts, including BERT and RoBERTa (Apache 2.0 and MIT License, respectively), and datasets from SimCSE (MIT License). We also incorporated code and pre-trained models from SNCSE, RLRD, and RankCSE, all licensed under the MIT License. All artifacts were used in accordance with their respective licenses. Artifacts we created will be released under the MIT License to encourage open sharing and further research.

Methods	STS12	STS13	STS14	STS15	STS16	STS-B	SICK-R	Avg.
ConSERT	64.64	78.49	69.07	79.72	75.95	73.97	67.31	72.74
SimCSE	68.40	82.41	74.38	80.91	78.56	76.85	72.23	76.25
SAN-SimCSE	70.92	81.77	75.20	82.57	78.88	78.66	71.33	77.05
PromptBERT	71.56	84.58	76.98	84.47	80.60	81.60	69.87	78.54
SNCSE	70.67	84.79	76.99	83.69	80.51	81.35	74.77	78.97
SAN-SNCSE	71.12	84.48	77.12	83.77	80.68	81.52	**75.24**	79.13
SAN-RankCSE	75.34	86.03	78.26	84.72	81.14	81.84	74.97	80.33
RankCSE	75.66	86.27	77.81	84.74	81.10	81.80	75.13	80.36
RLRD	**75.84**	86.28	79.22	85.80	**81.43**	82.90	74.79	80.89
SAN-RLRD	75.68	**86.38**	**79.47**	**85.93**	81.43	**83.05**	75.15	**81.01**

Table 9: Sentence embedding performance on STS tasks (Spearman's correlation). We highlight the best performance among models with the same pre-trained encoder.

	SimCSE		SNCSE		RankCSE		RLRD	
	Baseline	+SAN	Baseline	+SAN	Baseline	+SAN	Baseline	+SAN
Epochs	1	1	1	1	1	1	4	4
Time	50min	63min	75min	99min	127min	143min	673min	793min
Time per epoch	50min	63min	75min	99min	127min	143min	168min	198min

Table 10: Training minutes for different models on $\text{BERT}_{\texttt{base}}$. Baseline denotes the training minutes of a method without adding any additional methods or components.

Sentence1	Sentence2	Label	SimCSE	SimCSE+SAN
ahmadinejad is embarking on an adventure ; bernanke is not.	ahmadinejad board in an adventure, not bernanke	5	0.7689	0.9200
The motocross rider is wearing blue and black pants	Blue and black pants are being worn by the motocross rider	5	0.7856	0.9096
bulb a and b are still contained within closed paths	bulbs a and b are in a closed path	5	0.7855	0.9003
the european union has got to do something and do it quickly.	the european union must be involved and do so quickly.	4.8	0.7967	0.9007
iran, atomic agency in first talks since rowhani election	iaea, iran to hold first nuclear talks since rohani election	4.8	0.7967	0.9007
Broccoli are being cut by a woman	A woman is cutting broccoli	4.8	0.8218	0.9220

Table 11: Examples from the STS dataset with their similarity scores. The label scores are from human annotations. The SimCSE and SimCSE+SAN similarity scores are from the model predictions respectively. It can be seen that our method generates more accurate similarity scores than SimCSE.

Prompt
Task Description: Please modify the given sentence according to the following requirements:Only change the nouns, verbs, adjectives, and adverbs, while keeping the syntactic structure of the sentence unchanged. Input: Original sentence: {origin-sent} Modification requirements: Replace all nouns, verbs, adjectives, and adverbs with antonyms, but do not change their parts of speech. Output: Modified sentence: {your-outputs} Notes: Only modify nouns, verbs, adjectives, and adverbs. Keep the syntactic structure of the original sentence unchanged. If there are no suitable antonyms, the original words can be replace with semantically unrelated vocabularies. The modified sentence should still be a grammatically correct sentence.

Table 12: Prompt used for generating syntactically aligned negative samples.