BeLLM: Backward Dependency Enhanced Large Language Model for Sentence Embeddings

Xianming Li [†], **Jing Li** [†,‡*]
[†] Department of Computing
[‡] Research Centre on Data Science & Artificial Intelligence
The Hong Kong Polytechnic University
xianming.li@connect.polyu.hk, jing-amelia.li@polyu.edu.hk

Abstract

Sentence embeddings are crucial in measuring semantic similarity. Most recent studies employed large language models (LLMs) to learn sentence embeddings. Existing LLMs mainly adopted autoregressive architecture without explicit backward dependency modeling. Therefore, we examined the effects of backward dependencies in LLMs for semantic similarity measurements. Concretely, we propose a novel model: backward dependency enhanced large language model (BeLLM). It learns sentence embeddings via transforming specific attention layers from uni- to bi-directional. We extensively experiment across various semantic textual similarity (STS) tasks and downstream applications. BeLLM achieves state-of-the-art performance in varying scenarios. It shows that auto-regressive LLMs benefit from backward dependencies for sentence embeddings. [1]

1 Introduction

Sentence embedding is fundamental in natural language processing (NLP). It captures essential semantics in text, benefiting various semantic similarity measurement scenarios (Gao et al., 2021), such as semantic matching (Lu et al., 2020) and clustering (Reimers and Gurevych, 2019).

In previous work, the primary efforts employed smaller-scale bi-directional models (Peters et al., 2018; Reimers and Gurevych, 2019; Gao et al., 2021) to extensively explore the context to learn sentence embeddings. However, in the paradigm revolution of LLMs and the increasingly large model scales, most advanced NLP models adopted autoregressive (decoder-only) architectures with forward dependency modeling only (Touvron et al., 2023). While some recent efforts used LLMs for sentence embeddings (Li and Li, 2023; Jiang et al., 2023), limited attention has been paid to studying how backward dependency affects sentence embedding learning in autoregressive architectures.

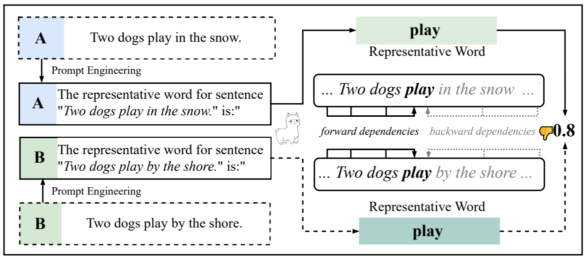

Figure 1: Two sample sentences A and B from STS-B dataset in dashed boxes. LLaMA predicted 0.8 similarity for A and B without backward dependency modeling (in grey). The ground-truth similarity is 0.5 because of differences in the playground in snow and shore.

To illustrate the potential help of backward dependencies in sentence embedding, Figure 1 illustrates two samples from STS-B (Cer et al., 2017) dataset. To enable LLaMA for sentence embeddings, we prompt it inspired by Jiang et al. (2023) and observe that LLaMA exhibits a very high similarity despite the different locations of events. The possible reason is that the uni-directional model LLaMA cannot extensively capture backward dependency, which indicates the relations between the representative word "play" and its different playgrounds. This observation highlights the potential benefits of engaging backward dependencies in LLMs for learning sentence embeddings.

To the best of our knowledge, *our work is the first to extensively investigate the effects of backward dependencies in autoregressive LLMs architectures for sentence embedding learning*.

We start our study with a pilot analysis to quantitatively examine the autoregressive capabilities of LLMs in capturing dependencies. It is observed that they are inferior to smaller-scale BERT in these capabilities. The results suggest the benefits of engaging backward dependencies in LLMs to en-

[*] Corresponding author
[1] The code is available at: https://github.com/4AI/BeLLM.

Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 792–804
June 16-21, 2024

hance their dependency-capturing capabilities.

To incorporate backward dependency into LLMs, we propose a novel model, BeLLM, for sentence embedding learning. Our core idea is to convert specific attention layers in the transformer decoder from uni- to bi-directional. We first conduct a degradation experiment to determine which attention layers should be converted bi-directional. It aims to explore the relations between transformer decoder layers and the performance of STS tasks (for semantic similarity measurement). It is observed that when uni-directional layers exceed a turning point, the STS performance will notably decrease. Furthermore, the turning point occurs at the penultimate layer for all LLMs we examined. We then convert the last layer bi-directional by removing their causal masks. In doing so, BeLLM involves both uni- and bi-directional layers to balance generation and dependency-capturing capabilities.

To train BeLLM, we first employ a representative word strategy to generate a representative word for a sentence via prompt engineering. Representative word embedding hence serves as the sentence embedding. Then, we apply contrastive learning (Gao et al., 2021) to pull embeddings of similar sentences close and push apart those not.

For experiments, we extensively evaluate sentence embeddings learned by BeLLM on various STS tasks. The main results indicate that BeLLM can significantly outperform previous SOTA on both the standard and the more challenging conditional STS tasks. For example, BeLLM achieves 49.74 Spearman's correlation compared to 47.50 from prior SOTA (Deshpande et al., 2023). It indicates that BeLLM is effective by introducing backward dependencies into LLMs. Then, our 7 downstream tasks experiments suggest that BeLLM's sentence embeddings can benefit various scenarios. Finally, a case study shows that BeLLM can better measure semantic similarities.

In summary, our contributions are three-fold:

- We explore the dependencies within LLMs and provide quantitative evidence that adding backward dependencies is helpful for sentence embeddings.
- We propose a novel backward dependency-enhanced large language model, BeLLM, to learn sentence embeddings in both uni- and bi-directions.
- Extensive experimental results demonstrate that BeLLM can obtain SOTA results across various STS tasks and downstream applications.

2 Quantitative Pilot Analysis

Before introducing BeLLM, we conduct a pilot analysis to explore how advanced LLMs capture dependencies in contexts. It is driven by the observations that mainstream LLMs (OpenAI, 2022; Touvron et al., 2023) employ an autoregressive architecture in an uni-directional manner. Intuitively, it lacks the ability to learn backward dependencies (Schuster and Paliwal, 1997), resulting in potential inferiority to capture dependencies compared to its bi-directional alternatives. Meanwhile, LLMs possess remarkable emergent abilities (Wei et al., 2022), implicitly benefit dependency capturing. We consequently conduct a pilot analysis to quantify whether explicit backward dependency modeling would benefit advanced autoregressive LLMs.

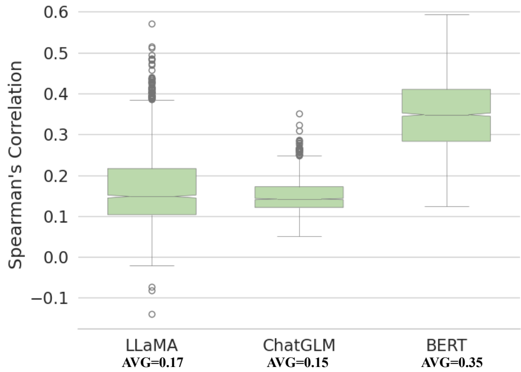

Figure 2: Box plot of the sentence-level Spearman correlation on the STS-B test set. The average sentence-level Spearman correlations for LLaMA, ChatGLM, and BERT are about 0.17, 0.15, and 0.35, respectively.

Our analysis is inspired by a phenomenon discovered by Ethayarajh (2019). The study qualitatively shows that bi-directional networks (e.g., ELMo (Peters et al., 2018) and BERT (Devlin et al., 2019)) demonstrate higher intra-sentence similarity (indicated by word similarities) than uni-directional networks (e.g., GPT-2). Here, we further explore this finding from a quantitative perspective.

Concretely, we experiment on the STS-B test set (Cer et al., 2017) and explore how models in uni- and bi-directional architectures capture dependencies. For the bi-directional model, we adopt BERT (base) and select the embedding of the "CLS" token as the pivot token to represent the sentence embedding. For the uni-directional, autoregressive architecture, we employ two representative LLMs: LLaMA (Touvron et al., 2023) (7B) and ChatGLM (Zeng et al., 2022) (6B). They offer two different implementations of autoregressive architectures. For them, we choose the last token as the pivot token following (Ethayarajh, 2019). Based on pivot tokens, we compute their Spearman correlation with the remaining tokens in a sentence to reflect the dependency-capturing capabilities. The results are shown in a box plot in Figure 2.

The results indicate that BERT shows a higher Spearman correlation, implying its better capability to capture dependencies compared to LLaMA and ChatGLM. Interestingly, BERT achieves an average score that is about *twice* as high as that of LLaMA and ChatGLM. It is possibly attributed to BERT's bi-directional architecture allowing both forward and backward dependency modeling. In contrast, autoregressive models focus on forward dependencies only. The results imply the potential benefits of adding backward dependency modeling to autoregressive models for sentence embeddings.

3 BeLLM

The above analysis has revealed the potential benefits of adding backward dependencies to autoregressive LLMs. Consequently, this section will describe our proposed model, BeLLM, with bi-directional attention layers. Figure 3 shows the overall framework of BeLLM. Subsequent sections are organized as follows: Firstly, we present a degradation experiment to examine how to add backward layers. Then, we introduce the architectures of BeLLM, followed by the training methods.

3.1 Degradation Experiment

To enable BeLLM to model backward dependencies, we adopt a straightforward way to turn some uni-directional layers of LLMs into bi-directional ones. However, uni-directional layers are crucial to LLMs' language generation capabilities (Kaplan et al., 2020), which may affect representative word prediction for sentence embeddings (Section 3.3).

To practically balance uni- and bi-directional layers, we conduct a degradation experiment to explore the effects of uni-directional layer number on STS performance. Here, we gradually removed layers from the last to the first [2] to reduce generation capabilities and show the Standard STS benchmark results in Figure 4. The results present a *S*-shaped curve. It indicates that uni-directional layers are generally helpful to sentence embeddings. However, there is a *turning point* at the penultimate layer, exceeding which we observe a consistent performance drop for all three LLMs.

Such observations might be attributed to the extreme anisotropy (common-word biases) in the last layer of autoregressive LLMs, also shown in GPT-2's experiments (Ethayarajh, 2019). Consequently, we convert the last attention layer from uni- to bi-directional to introduce backward dependencies, yet keeping most layers uni-directional.

3.2 Model Architecture

BeLLM exhibits layers of auto-regressive LLM and BiLLM (bi-directional LLM) as follows.

For an input sentence s, we first obtain its word embeddings with the following formula:

$$\begin{aligned} \mathbf{x} &= \text{Embedding}^{LLM}(t) \\ t &= \text{Tokenizer}^{LLM}(s). \end{aligned} \tag{1}$$

We then fetch the layers from the first up to the penultimate layer denoted as $\text{LLM}^{1:n-1}$ (n indicates the layer number). They keep autoregressive architectures to handle language generation effectively and their attention computation is as follows:

$$\text{Attn}_i^{LLM}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{SoftMax}(\frac{\mathbf{QK}}{\sqrt{d}} + \mathcal{M})\mathbf{V} \tag{2}$$

and

$$\mathcal{M} = \begin{bmatrix} 0 & -\infty & -\infty & -\infty & -\infty \\ 0 & 0 & -\infty & -\infty & -\infty \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & 0 & 0 & -\infty \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \tag{3}$$

where Attn_i^{LLM} is the i-th head of multi-head self attention (Vaswani et al., 2017) in LLM. $\mathbf{Q} = \mathbf{W}^q\mathbf{x} + b$, $\mathbf{K} = \mathbf{W}^k\mathbf{x} + b$, $\mathbf{V} = \mathbf{W}^v\mathbf{x} + b$. $\mathcal{M}$ denotes the causal mask. It ensures that the i-th token is not visible to the $i+1 \ldots L$ tokens (L is the input token length) during attention computation, which is crucial to training language generation.

Based on the experiments in Section 3.1, for the last layer (after the turning point), we detach and transform it from uni- to bi-directional to obtain $\text{BiLLM}^{n-1:n}$. Its attention is computed as follows:

$$\text{Attn}_i^{BiLLM}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{SoftMax}(\frac{\mathbf{QK}}{\sqrt{d}})\mathbf{V}, \tag{4}$$

To engage backward dependencies, we remove the casual mask to turn the last layer bi-directional inspired by Li et al. (2023).

[2] We did not remove it reversely, i.e., from first to last, as doing so would cause a malfunction in the generation ability.

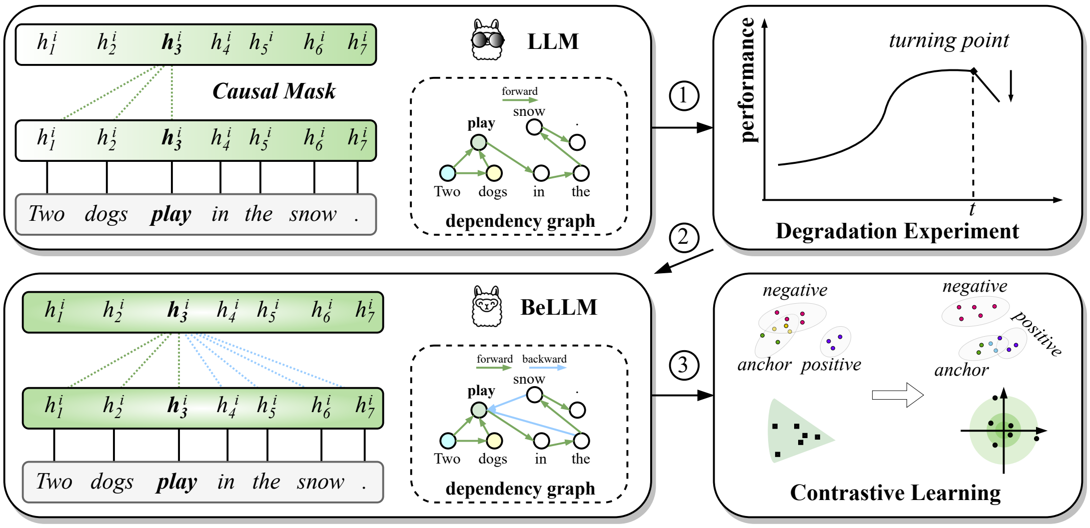

Figure 3: The overall framework of BeLLM. It includes three steps: 1) It first examines how to balance uni- and bi-directional layers with the degradation experiment and finds a turning point. 2) It transforms the attention layers after the turning point from uni- to bi-directional by removing the causal mask. 3) It employs contrastive learning to learn sentence embedding. Here, we visualize the dependencies of the representative word "play." LLM only captures the forward dependencies of "play" and BeLLM can capture both forward and backward dependencies.

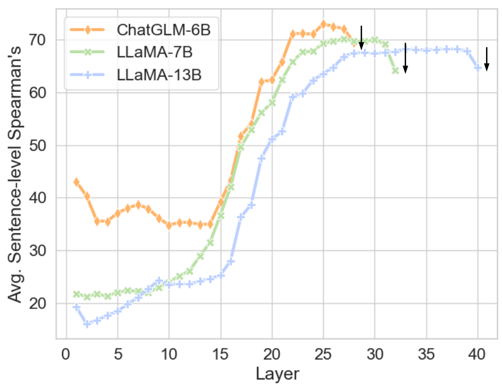

Figure 4: Degradation results on the Standard STS benchmark. X-axis: the number of uni-directional layers. Y-axis: the average Spearman's correlations computed with SentEval (Conneau and Kiela, 2018). The down arrow indicates a dramatic performance drop.

At last, we couple autoregressive LLM (retaining language generation capabilities) and BiLLM (the bi-directional last layer engaging forward and backward dependency modeling) components. The formula to represent BeLLM is as follows:

$$\mathbf{h} = \overrightarrow{\text{LLM}}^{1:n}(\mathbf{x}) + \overleftrightarrow{\text{BiLLM}}^{n-1:n}(\mathbf{x}). \quad (5)$$

3.3 Training Methods

In the training of BeLLM, we first generate the representative word of a given sentence as the pivot to learn its embedding. Concretely, we employ a prompt *The representative word for {sentence} is:"* for BeLLM to produce the representative word, where {sentence} is the placeholder for the actual sentence. Then, the embedding of the representative word serves as the sentence embedding.

Finally, to enable sentence similarity training, we adopt contrastive learning (Gao et al., 2021) to optimize the contrastive objective as follows:

$$\mathcal{L} = -\sum_i \log \frac{e^{cos(\mathbf{h}_i, \mathbf{h}_i^+)/\tau}}{\sum_{j=1}^{N} \left(e^{cos(\mathbf{h}_i, \mathbf{h}_j^+)/\tau} + e^{cos(\mathbf{h}_i, \mathbf{h}_j^-)/\tau} \right)}, \quad (6)$$

where N is the mini-batch size, $\mathbf{h}_i^+$ and $\mathbf{h}_i^-$ refer to the positive and negative samples of $\mathbf{h}_i$, respectively. τ is the temperature. $cos(a, b)$ is the cosine similarity function. This way, the embeddings of semantically similar sentences are pulled closer together while dissimilar ones are pushed apart.

Moreover, our training methods can potentially mitigate the common *anisotropy problem* in sentence embeddings, which constrains the embeddings' expressiveness. It happens because common words can bias sentence embeddings, rendering the learned embeddings to occupy a narrow cone in the vector space instead of distributing uniformly. The detailed discussions are shown in Appendix A.

4 Experimental Setup

Datasets. We evaluate sentence embeddings on STS tasks following the common practices. It includes the standard and conditional STS as follows.

Model	STS12	STS13	STS14	STS15	STS16	STS-B	SICR-R	Avg.
				Closed-Source Models				
openai-ada-002 ♢	69.80	83.27	76.09	86.12	85.96	83.17	80.60	80.72
				Unsupervised Models				
GloVe (avg.) †	55.14	70.66	59.73	68.25	63.66	58.02	53.76	61.32
BERT								
+flow ‡	58.40	67.10	60.85	75.16	71.22	68.66	64.47	66.55
+whitening ‡	57.83	66.90	60.90	75.08	71.31	68.24	63.73	66.28
+IS ‡	56.77	69.24	61.21	75.23	70.16	69.21	64.25	66.58
+CT ‡	61.63	76.80	68.47	77.50	76.48	74.31	69.19	72.05
+DiffCSE	72.28	84.43	76.47	83.90	80.54	80.59	71.23	78.49
+SimCSE	68.40	82.41	74.38	80.91	78.56	76.85	72.23	76.25
				Supervised Models				
InferSent †	52.86	66.75	62.15	72.77	66.87	68.03	65.65	65.01
USE †	64.49	67.80	64.61	76.83	73.18	74.92	76.69	71.22
SBERT †	70.97	76.53	73.19	79.09	74.30	77.03	72.91	74.89
AnglE								
+BERT$_{base}$ ♡	75.09	85.56	80.66	86.44	82.47	85.16	81.23	82.37
+LLaMA ♡	79.00	90.56	85.79	**89.43**	87.00	88.97	80.94	85.96
SimCSE								
+RoBERT$_{large}$	77.46	87.27	82.36	86.66	83.93	86.70	**81.95**	83.76
+LLaMA ♡	78.39	89.95	84.80	88.50	86.04	87.86	81.11	85.24
BeLLM (ours)	**79.39** $\pm$0.30	**91.04** $\pm$0.15	**86.52** $\pm$0.15	89.24 $\pm$0.60	**87.43** $\pm$0.3	**89.04** $\pm$0.73	81.14 $\pm$0.80	**86.26** $\pm$0.52

Table 1: Spearman's correlation on the standard STS benchmark datasets. Higher scores indicate better sentence embedding performance. Results marked with † are obtained from (Reimers and Gurevych, 2019), those with ‡ are from (Gao et al., 2021), those with ♡ are from (Li and Li, 2023). ♢ indicates the sentence embedding model released by OpenAI, and its results are from (Muennighoff et al., 2022). Other results are based on reimplementation. For BeLLM, we report the average scores in five runs with standard deviation (std) expressed as a percentage (%) after ±. BeLLM outperforms all baselines significantly on average (p-value < 1%, paired t-test).

Standard STS. The standard STS (S-STS) benchmark consists of seven STS tasks: STS 2012-2016 (Agirre et al., 2012, 2013, 2014, 2015, 2016), SICK-R (Marelli et al., 2014), and STS-B (Cer et al., 2017). They contain manual annotations of sentence similarities to test the effectiveness of sentence embeddings. For training and testing, we follow the prior work (Reimers and Gurevych, 2019; Gao et al., 2021) to train sentence embeddings on MNLI (Williams et al., 2018) and SNLI (Bowman et al., 2015) datasets. Then, the trained embeddings are evaluated on the S-STS benchmark datasets.

Conditional STS. To test sentence embeddings with a more challenging setup, we adopt the Conditional Semantic Textual Similarity (C-STS) benchmark (Deshpande et al., 2023). Here, the pairwise sentence similarity is labeled under varying conditions with details in Appendix B. We follow the setting of C-STS to train sentence embeddings and test them with the C-STS official evaluation API.

Evaluation Metrics. For the automatic evaluation of sentence embeddings, we follow previous studies to use Spearman's correlation. It compares the ranks of pairwise embeddings and assesses the ranked monotonic relations based on manual annotations. For the S-STS benchmark, we used the SentEval (Conneau and Kiela, 2018) toolkit and reported the results in the "all" setting following prior work. For C-STS, we reported Spearman's correlation returned by its official evaluation API.

Baselines and Comparisons. For **S-STS**, we considered widely adopted unsupervised and supervised sentence embedding baselines. 1) The unsupervised baselines include GloVe (Pennington et al., 2014) with average pooling , BERT-flow (Li et al., 2020), BERT-whitening (Su et al., 2021), and other BERT-based baselines trained with contrastive learning: IS-BERT (Zhang et al., 2020), CT-BERT (Carlsson et al., 2020), SimCSE (Gao et al., 2021), and DiffCSE (Chuang et al., 2022). 2) The supervised baselines include InferSent (Conneau et al., 2017), USE (Cer et al., 2018), SBERT (Reimers and Gurevych, 2019), AnglE (Li and Li, 2023), and supervised SimCSE. In addition to open-source models, we also compared with popular closed-source *openai-ada-002* (Muennighoff et al., 2022) embeddings.

For **C-STS**, we employed few-shot LLM of Flan-T5 (Chung et al., 2022), Tk-Instruct (Wang et al., 2022), GPT-3.5 (OpenAI, 2022), GPT-4 (OpenAI, 2023), and supervised SimCSE (fine-tuned) (Gao et al., 2021).

For baselines' benchmark results, we will report the scores from the original papers and prior work.

Model Settings. BeLLM employed LLaMA2-7B model (Touvron et al., 2023) as the backbone. For efficient training, we used the LoRA technique (Hu et al., 2021; Dettmers et al., 2024) for fine-tuning with $lora\_r = 32$, $lora\_alpha = 32$, and $lora\_dropout = 0.1$. In this setting, the trainable parameters are about 80M, smaller than BERT-base's 110M. We trained the model on 4 RTX 3090 Ti GPUs. We chose the batch size by searching on the values $\{16, 32, 64, 128\}$. The initial learning rate was set to $2e-4$ via grid search on validation data. For a fair comparison with baselines, we set the random seed to 42 following SimCSE (Gao et al., 2021) for main experiments. We also test using different random seeds to verify BeLLM's robustness in the ablation study (Section 5.3).

5 Experimental Results

The following will first present the main results with intrinsic evaluation (Section 5.1), followed by the extrinsic transfer task results in Section 5.2. To provide more insight, we will then analyze the ablation study and case study results in Section 5.3 and 5.4, respectively. At last, we further examine the intra-sentence dependency of BeLLM to supplement the pilot analysis in Section 2.

5.1 Main Comparison Results

We first discuss the main results in S-STS and the more challenging C-STS benchmarks.

Standard STS. We show S-STS benchmark results in Table 1 and draw the following observations. First, supervised models generally outperform unsupervised models; it suggests that sentence embedding cannot be effectively learned by shallow features, and human supervision can provide positive help. Second, the LLM-based models perform better than the BERT-based models, implying that larger model scales help capture sentence semantics. Third, our proposed BeLLM model performs the best in all S-STS datasets. It consistently outperforms AnglE$_{LLaMA}$ and SimCSE$_{LLaMA}$, with 0.3% and 1.02% improvements in average score, respectively. These results indicate the effectiveness of incorporating backward dependencies into LLMs for sentence embeddings.

Model	Spearman's Correlation
Flan-T5$_{XL}$ †	24.80
Flan-T5$_{XXL}$ †	29.20
Flan-UL2 †	23.20
Tk-Instruct$_{3B}$ †	4.90
Tk-Instruct$_{11B}$ †	17.10
GPT-3.5 †	15.50
GPT-4 †	43.60
SimCSE	
+RoBERTa$_{large}$ (prior SOTA) †	47.50
+LLaMA	48.64
BeLLM (ours)	**49.74** ± 0.25

Table 2: Results on the C-STS benchmark. † denotes results from (Deshpande et al., 2023). Prompts for LLMs can be retrieved from (Deshpande et al., 2023). BeLLM results are the average in five runs. The reported standard deviation (std) value is expressed as a percentage (%) after the ± symbol. BeLLM outperforms all baselines significantly (p-value < 1%, paired t-test).

Conditional STS. To allow sentence embedding evaluation in a more challenging setup, we further employ the C-STS dataset to assess semantic similarities in conditions. Here, we devise a prompt to enable LLM to summarize the given sentence in one representative word by putting the conditions into the context. The prompt is as follows: *Given the context {condition}, summarize the sentence {sentence} in one word:"*, where {condition} and {sentence} are placeholders for the actual input condition and sentence, respectively.

The results are presented in Table 2. We observe that LLMs, such as GPT-3.5 and GPT-4, yield inferior results in few-shot settings compared to supervised SimCSE with fine-tuning. It implies that C-STS is challenging and cannot be well solved by few-shot LLMs; fine-tuning can helpfully boost the results of a smaller-scale model. Among fine-tuned models, the results are consistent with S-STS: larger model sizes and backward dependencies both contribute positively. Combining their effects, BeLLM achieves the SOTA performance. It demonstrates a 1.10% improvement over SimCSE$_{LLaMA}$ without modeling backward dependency and a remarkable 2.24% gain over the prior SOTA model, SimCSE (RoBERTa$_{large}$).

Model	MR	CR	SUBJ	MPQA	SST2	TREC	MRPC	Avg.
GloVe †	77.25	78.30	91.17	87.85	80.18	83.00	72.87	81.52
Skip-thought ‡	76.50	80.10	93.60	87.10	82.00	92.20	73.00	83.50
Avg. BERT †	78.66	86.25	94.37	88.66	84.40	92.80	69.54	84.94
BERT-CLS †	78.68	84.85	94.21	88.23	84.13	91.40	71.13	84.66
IS-BERT ‡	81.09	87.18	94.96	88.75	85.96	88.64	74.24	85.83
DiffCSE ♢	82.82	88.61	94.32	87.71	88.63	90.40	**76.81**	87.04
SimCSE								
+RoBERTa ⋆	83.37	87.76	95.05	87.16	89.02	90.80	75.13	86.90
+LLaMA ♣	90.40	92.90	**96.88**	91.57	95.11	95.40	75.13	91.06
BeLLM (ours)	**90.79** $\pm_{0.28}$	**93.43** $\pm_{0.50}$	96.53 $\pm_{0.55}$	**92.01** $\pm_{0.32}$	**95.77** $\pm_{0.30}$	**95.45** $\pm_{0.67}$	75.48 $\pm_{0.59}$	**91.35** $\pm_{0.48}$

Table 3: Accuracy of transfer task results based on different sentence embeddings. †: results from Reimers and Gurevych (2019); ‡: results from Zhang et al. (2020); ⋆: results from Gao et al. (2021). ♢: results from Chuang et al. (2022); ♣: results from Jiang et al. (2023). BeLLM's results are the average of five runs. The reported standard deviation (std) value is expressed as a percentage (%) after the $\pm$ symbol. The paired t-test suggests that the average improvements of BeLLM compared to LLaMA are significant, with corresponding p-values of 4.06%.

5.2 Transfer Task Results

To assess the benefits of sentence embeddings in downstream tasks, we evaluate our model on seven transfer tasks: MR (Movie Reviews) (Pang and Lee, 2005), CR (Customer Reviews) (Hu and Liu, 2004), SUBJ (Subjectivity) (Pang and Lee, 2004), MPQA (Multi-Perspective Question Answering) (Wiebe et al., 2005), SST2 (Stanford Sentiment Treebank) (Socher et al., 2013), TREC (Text Retrieval Conference) (Voorhees and Tice, 2000), and MRPC (Microsoft Research Paraphrase Corpus) (Dolan et al., 2004). The transfer task results are measured by SentEval (Conneau and Kiela, 2018) toolkit. Following common practices, we train a logistic regression classifier using sentence embeddings as features and used default configurations of SentEval for a fair comparison.

The results are presented in Table 3. As can be seen, BeLLM achieves superior performance compared to the baselines, obtaining the best results on average and in 6 out of 7 tasks. These results indicate that incorporating backward dependencies to LLMs can helpfully learn sentence embeddings, which benefit downstream task performances.

5.3 Ablation Study

We have shown the overall effectiveness of BeLLM. Here, we further analyze the effects of its varying settings with an ablation study results in Table 4. From the *turning point* ablation results, we find bi-directional attention layers are useful to BeLLM, yet converting all attention layers into bi-directional results in deficient performance. It is possibly because uni-directional attention layers are crucial in language generation for representative word prediction. Consequently, finding a good balance of uni- and bi-directional attention layers plays a crucial role in sentence embeddings.

From the *model scale* results, we find that the 13B BeLLM outperforms the 7B BeLLM. This aligns with the finding that larger model scales are more effective in capturing sentence semantics, as indicated in the main findings.

From the *random seed* results, we follow Wu et al. (2023) to use different random seeds. It is observed that this yields even better performance than fixed ones, suggesting BeLLM is robust.

From the *bi-directional strategy* results, we observe that the "modification" strategy yields better results than the "addition" strategy. This implies that the increase in trainable parameters to LLMs by adding new attention layers will increase the difficulty of fine-tuning the model. It demonstrates that modifying the uni-directional attention layer by removing casual masks is effective.

5.4 Case Study

To better understand why BeLLM performs well, we conduct a text retrieval experiment on the test split of the flickr30k dataset (Young et al., 2014). It contains images, each with 5 captions. We used the first caption vector to retrieve the top 4 similar sentences using the faiss vector search framework (Johnson et al., 2019). For strict accuracy (correct cases only count for the top 4 retrieved captions exactly matching the 4 references), SimCSE$_{RoBERTa}$, SimCSE$_{LLaMA}$, and BeLLM obtains 16.4%, 18.4%, and 20.2%, respectively. It shows that BeLLM's sentence embeddings superi-

Ablation Models	Spearman's ρ
Turning Point	
BeLLM (*last bi-layer*) (default)	**86.26**
BeLLM (*no bi-layer*)	85.24
BeLLM (*all bi-layers*)	75.55
Model Scale	
BeLLM$_{7B}$	86.26
BeLLM$_{13B}$	**86.87**
Random Seed	
fixed random seed 42	86.26
different random seeds (3 runs)	**86.29**
Bi-directional Strategy	
Modification	**86.26**
Addition	84.35

Table 4: The results of BeLLM ablations with the average Spearman's correlation (Spearman's ρ) of S-STS. *Last bi-layer*: bi-directional layer at the last layer; no bi-layer and all bi-layers: the BeLLM ablation with no and all bi-directional layers. *Modification*: removing casual masks; *Addition*: adding new bi-directional layers.

orly reflect semantic similarities for retrieval.

We then examine a case in Table 5. In the query caption, "police officer" is the subject, and "female" is its subject modifier; their relations can be learned through forward dependency; "hat" and "sunglasses" are objects, and their relations to "police" should be indicated by backward dependency. SimCSE$_{RoBERTa}$ has a smaller model scale, which limits its ability to exploit global context. As a result, its top results even messed up the subject. Despite having a larger model scale, SimCSE$_{LLaMA}$ exhibits inferior results due to the lack of backward dependency modeling. For this reason, most of the retrieved results missed 1-2 objects. On the contrary, BeLLM performs the best, and the top two captions exactly match the references. The results indicate the superiority of BeLLM in exploiting context to learn high-quality sentence embeddings.

5.5 Discussion of Enhanced Dependency

Finally, we experiment on the intra-sentence dependencies for BeLLM to supplement Section 2. Figure 5 depicts the dependencies change from NLI-trained LLaMA to BeLLM. As can be seen, BeLLM exhibits much higher overall Spearman's correlation scores than LLaMA. This evidence again demonstrates that transforming the last attention layer bi-directional effectively enhances the capabilities of capturing dependency and is helpful in exploiting sentence context for embeddings.

Q:	A female police officer wears an officer's hat and sunglasses.
SimCSE$_{RoBERTa}$	
#1	An officer stands next to a car on a city street.
#2	A police woman smiling and wearing sunglasses and a hat.
#3	Young, smiling, blond female police officer from New York standing outside on a sidewalk.
#4	An officer in a black uniform and hat stands to the left of a large structure with other officers in the background.
SimCSE$_{LLaMA}$	
#1	A female police officer in a cap and navy uniform smiles while wearing sunglasses outside of a shop.
#2	Young, smiling, blond female police officer from New York standing outside on a sidewalk.
#3	An officer stands next to a car on a city street.
#4	An officer in a black uniform and hat stands to the left of a large structure with other officers in the background.
BeLLM	
#1	A female police officer in a cap and navy uniform smiles while wearing sunglasses outside of a shop.
#2	A police woman smiling and wearing sunglasses and a hat.
#3	Young, smiling, blond female police officer from New York standing outside on a sidewalk.
#4	An attractive young New York City police woman pauses on the sidewalk.

Table 5: The top 4 retrieved sentences by RoBERTa, LLaMA, and BeLLM from the flickr30k dataset. The words with green color represent the keywords.

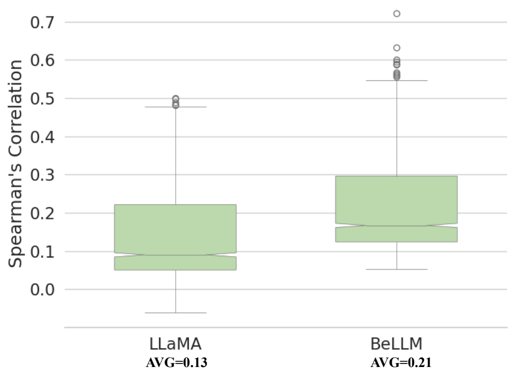

Figure 5: The sentence-level Spearman correlation plot for NLI-trained LLaMA and BeLLM on STS-B test set.

6 Related Work

BeLLM is in line with sentence embeddings. In contrast to early attempts focusing on embeddings on the word level (Mikolov et al., 2013), sentence embeddings capture sentence-level semantics to understand context better. Many previous studies adopted **unsupervised** approaches to utilize large-scale unlabeled text. Here, a popular method is to use BERT-alike transformers (Li et al., 2020; Su et al., 2021) to encode sentence embeddings. Based on that, contrastive learning (Zhang et al., 2020; Gao et al., 2021; Chuang et al., 2022; Zhuo et al., 2023) was further exploited to explore semantic similarities for sentence embeddings with self-supervision.

To better align sentence embeddings to human senses, other prior studies (Conneau et al., 2017; Cer et al., 2018) employed labeled data for **supervised** sentence embedding learning. Here, pre-trained language models typically worked as the backbone architectures (Reimers and Gurevych, 2019; Su, 2022). Following that, there is growing attention to the use of pre-trained LLMs for sentence embeddings (Li and Li, 2023; Jiang et al., 2023). However, existing works employed commonly used autoregressive LLMs, neglecting the potential benefits of backward dependencies to sentence embeddings. Viewing this gap, we extensively explore the effects of backward dependencies in LLMs for sentence embeddings.

7 Conclusion

Our work has pointed out the benefits of coupling forward and backward dependencies in LLMs for sentence embeddings. We have introduced BeLLM, a novel LLM with backward dependency-enhanced sentence embedding learning. In extensive experiments, BeLLM has achieved state-of-the-art performance across varying STS and downstream tasks.

Limitations

BeLLM has large-scale parameters, which can hinder its efficiency when applied to real-world applications. Addressing this challenge is an essential aspect of our future work. We plan to optimize the model, ensuring it remains practical and efficient for real-world applications.

Acknowledgements

This work is supported by the NSFC Young Scientists Fund (Project No. 62006203), a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. PolyU/25200821), the Innovation and Technology Fund (Project No. PRP/047/22FX), and PolyU Internal Fund from RC-DSAI (Project No. 1-CE1E).

Here, we sincerely thank the reviewers and ACs for their valuable input, which has greatly improved our work.

References

Eneko Agirre, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Iñigo Lopez-Gazpio, Montse Maritxalar, Rada Mihalcea, German Rigau, Larraitz Uria, and Janyce Wiebe. 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, Denver, Colorado. Association for Computational Linguistics.

Eneko Agirre, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Rada Mihalcea, German Rigau, and Janyce Wiebe. 2014. SemEval-2014 task 10: Multilingual semantic textual similarity. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 81–91, Dublin, Ireland. Association for Computational Linguistics.

Eneko Agirre, Carmen Banea, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Rada Mihalcea, German Rigau, and Janyce Wiebe. 2016. SemEval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 497–511, San Diego, California. 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)*, pages 385–393, Montréal, Canada. Association for Computational Linguistics.

Eneko Agirre, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, and Weiwei Guo. 2013. *SEM 2013 shared task: Semantic textual similarity. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity*, pages 32–43, Atlanta, Georgia, USA. Association for Computational Linguistics.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Fredrik Carlsson, Amaru Cuba Gyllensten, Evangelia Gogoulou, Erik Ylipää Hellqvist, and Magnus Sahlgren. 2020. Semantic re-tuning with contrastive tension. In *International conference on learning representations*.

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14, Vancouver, Canada. Association for Computational Linguistics.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder for English. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 169–174, Brussels, Belgium. Association for Computational Linguistics.

Yung-Sung Chuang, Rumen Dangovski, Hongyin Luo, Yang Zhang, Shiyu Chang, Marin Soljacic, Shang-Wen Li, Scott Yih, Yoon Kim, and James Glass. 2022. DiffCSE: Difference-based contrastive learning for sentence embeddings. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4207–4218, Seattle, United States. Association for Computational Linguistics.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Alexis Conneau and Douwe Kiela. 2018. SentEval: An evaluation toolkit for universal sentence representations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680, Copenhagen, Denmark. Association for Computational Linguistics.

Ameet Deshpande, Carlos E. Jimenez, Howard Chen, Vishvak Murahari, Victoria Graf, Tanmay Rajpurohit, Ashwin Kalyan, Danqi Chen, and Karthik Narasimhan. 2023. Csts: Conditional semantic textual similarity. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2024. Qlora: Efficient finetuning of quantized llms. *Advances in Neural Information Processing Systems*, 36.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Bill Dolan, Chris Quirk, and Chris Brockett. 2004. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 350–356, Geneva, Switzerland. COLING.

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 55–65, Hong Kong, China. Association for Computational Linguistics.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910. Association for Computational Linguistics.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177.

Ting Jiang, Shaohan Huang, Zhongzhi Luan, Deqing Wang, and Fuzhen Zhuang. 2023. Scaling sentence embeddings with large language models. *arXiv preprint arXiv:2307.16645*.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3):535–547.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Bohan Li, Hao Zhou, Junxian He, Mingxuan Wang, Yiming Yang, and Lei Li. 2020. On the sentence embeddings from pre-trained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9119–9130, Online. Association for Computational Linguistics.

Xianming Li and Jing Li. 2023. Angle-optimized text embeddings. *arXiv preprint arXiv:2309.12871*.

Zongxi Li, Xianming Li, Yuzhang Liu, Haoran Xie, Jing Li, Fu-lee Wang, Qing Li, and Xiaoqin Zhong. 2023. Label supervised llama finetuning. *arXiv preprint arXiv:2310.01208*.

Wenpeng Lu, Xu Zhang, Huimin Lu, and Fangfang Li. 2020. Deep hierarchical encoding model for sentence semantic matching. *Journal of Visual Communication and Image Representation*, 71:102794.

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).

Tomás Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *27th Annual Conference on Neural Information Processing Systems 2013.*, pages 3111–3119.

Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. 2022. Mteb: Massive text embedding benchmark. *arXiv preprint arXiv:2210.07316*.

OpenAI. 2022. Introducing chatgpt.

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 271–278, Barcelona, Spain.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 115–124, Ann Arbor, Michigan. Association for Computational Linguistics.

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.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, pages 3980–3990. Association for Computational Linguistics.

Mike Schuster and Kuldip K Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, 45(11):2673–2681.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

Jianlin Su. 2022. Cosent (1): A more effective sentence vector scheme than sentence bert.

Jianlin Su, Jiarun Cao, Weijie Liu, and Yangyiwen Ou. 2021. Whitening sentence representations for better semantics and faster retrieval. *arXiv preprint arXiv:2103.15316*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Ellen M Voorhees and Dawn M Tice. 2000. Building a question answering test collection. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 200–207.

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

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Kuntal Kumar Pal, Maitreya Patel, Mehrad Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Savan Doshi, Shailaja Keyur Sampat, Siddhartha Mishra, Sujan Reddy A, Sumanta Patro, Tanay Dixit, and Xudong Shen. 2022. Super-NaturalInstructions: Generalization via declarative instructions on 1600+ NLP tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5085–5109, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. 2022. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*.

Janyce Wiebe, Theresa Wilson, and Claire Cardie. 2005. Annotating expressions of opinions and emotions in language. *Language resources and evaluation*, 39:165–210.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Zhuofeng Wu, Chaowei Xiao, and VG Vydiswaran. 2023. Hicl: Hierarchical contrastive learning of unsupervised sentence embeddings. *arXiv preprint arXiv:2310.09720*.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. 2014. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78.

Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, et al. 2022. Glm-130b: An open bilingual pre-trained model. *arXiv preprint arXiv:2210.02414*.

Yan Zhang, Ruidan He, Zuozhu Liu, Kwan Hui Lim, and Lidong Bing. 2020. An unsupervised sentence embedding method by mutual information maximization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1601–1610, Online. Association for Computational Linguistics.

Wenjie Zhuo, Yifan Sun, Xiaohan Wang, Linchao Zhu, and Yi Yang. 2023. WhitenedCSE: Whitening-based contrastive learning of sentence embeddings. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12135–12148, Toronto, Canada. Association for Computational Linguistics.

A Discussion of Anisotropy Problem

Our proposed representative word strategy and contrastive learning can mitigate the anisotropy issue.

Firstly, our proposed representative word strategy can help alleviate this issue since representative words are typically not high-frequency words. For instance, in our experiment using the BeLLM model, the representative word for the sentence "I am unhappy because it is raining" is "unhappy", which is not as commonly used as words like "it" and "is". Our quantitative experiment in the STS-B dataset indicates that approximately 91.74% of representative words are not high-frequency words.

Furthermore, the use of contrastive learning can alleviate this problem further. Here, we give a theoretical explanation. Following Wang and Isola (2020), the contrastive learning objective of Eq. 6 can be formulated as follows:

$$\mathcal{L}_{contrastive} = \mathbb{E}_{(x,x^+)\sim p_{pos}}\left[-f(x)^T f(x^+)/\tau\right] + \underset{\substack{(x,x^+)\sim p_{pos}\\(x,x_i^-)\sim p_{data}}}{\mathbb{E}}\left[\log\left(e^{f(x)^T f(x^+)/\tau} + \sum_i e^{f(x_i^-)^T f(x)/\tau}\right)\right], \tag{7}$$

where x^+ and x^- are the positive and negative examples of x, respectively. p_{pos} denotes positive instances. p_{data} is uniform over finite samples $x_{i\,i=1}^{m}$. $f(x)^T f(x^+)/\tau$ measures the alignment (similarity). Since the term $\sum e^{f(x_i^-)^T f(x)/\tau}$ is always positive, the loss function inherently favors smaller values for $\mathbb{E}\left[-f(x)^T f(x^+)/\tau\right]$, i.e., it can be optimized well. Suppose the encoder is perfectly aligned, i.e., $\mathbb{P}[f(x) = f(x^+)] = 1$, then minimizing Eq. 7 equally minimizes following equation:

$$\underset{\substack{(x,x^+)\sim p_{pos}\\(x,x_i^-)\sim p_{data}}}{\mathbb{E}}\left[\log\left(e^{1/\tau} + \sum_i e^{f(x_i^-)^T f(x)/\tau}\right)\right], \tag{8}$$

To minimize it, all sentence embeddings should be pushed away from each other to form a *uniform* distribution. The uniform distribution of sentence embeddings can eliminate the effect of the common words and thus alleviate the anisotropy problem.

B Examples of CSTS

Table 6 shows an example of C-STS. Each sample in C-STS includes three fields: sentence 1, sentence 2, and condition. The sentence pair exhibits varying similarities based on different conditions. The same sentence pair can be classified as either high similarity or low similarity under different

Sentence 1: An older man holding a glass of wine while standing between two beautiful ladies.	
Sentence 2: A group of people gather around a table with bottles and glasses of wine.	
Condition	Similarity
The people's demeanor	5
The number of bottles	1

Table 6: An example from the C-STS validation set, where the same sentence pair has different similarities based on different conditions. A similarity score of 1 indicates dissimilar, while a score of 5 represents similar.

conditions. The scale ranges from 1 (dissimilar) to 5 (similar).