Capturing the Relationship Between Sentence Triplets for LLM and Human-Generated Texts to Enhance Sentence Embeddings

Na Min An, Sania Waheed and **James Thorne**
KAIST AI

Abstract

Deriving meaningful sentence embeddings is crucial in capturing the semantic relationship between texts. Recent advances in building sentence embedding models have centered on replacing traditional human-generated text datasets with those generated by LLMs. However, the properties of these widely used LLM-generated texts remain largely unexplored. Here, we evaluate the quality of the LLM-generated texts from four perspectives (Positive Text Repetition, Length Difference Penalty, Positive Score Compactness, and Negative Text Implausibility) and find the limitation of only using LLM to build high-quality NLI datasets. Then, we attempt to improve each of these models either fine-tuned with human, LLM, or human+LLM-generated sentence triplets data with our proposed loss function that incorporates Positive-Negative sample Augmentation (PNA) within the contrastive learning objective. Our results demonstrate the effectiveness of PNA, especially in RoBERTa-large, by showing decreased cosine similarity for sentence triplets, mitigating the sentence anisotropy problem in Wikipedia corpus (-7% compared to CLHAIF), and improving the Spearman's correlation in standard Semantic Textual Similarity (STS) tasks (+1.47% compared to CLHAIF). Our code is available at https://github.com/xfactlab/eacl2024-pna.

1 Introduction

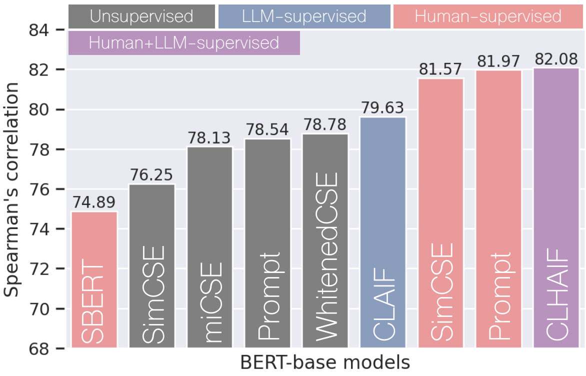

Figure 1: LLM-supervised models comparable to unsupervised models than the state-of-the-art human-supervised models. SBERT: Reimers and Gurevych, 2019; DINO: Schick and Schütze, 2021; SimCSE: Gao et al., 2021; miCSE: Klein and Nabi, 2023; Whitened-CSE: Zhuo et al., 2023; Prompt: Jiang et al., 2022; CLAIF/CLHAIF[1]: Cheng et al., 2023.

Sentence embeddings with contextual representations are more informative than static text embeddings for various natural language processing (NLP) tasks (Ethayarajh, 2019). Semantic similarity scoring has been an important fundamental testbed for understanding the quality of sentence embeddings (Dolan and Brockett, 2005; Wang et al., 2018). *Unsupervised* sentence embedding model employs data augmentation strategies such as dropout to create positive pairs (Gao et al., 2021; Yan et al., 2021; Zhuo et al., 2023; Klein and Nabi, 2023), but there is a limitation of creating diverse samples of semantically similar positives by modifying the embedding parameters in the latent space. Thus, *supervised* models which are fine-tuned with human-generated data (Gao et al., 2021; Jiang et al., 2022; Cheng et al., 2023) often surpass these unsupervised models. However, human subject experiments often take tremendous time and effort to manually create large-scale, high-quality data samples with few annotation artifacts (Gururangan et al., 2018).

The emergence of billion-scale generative large language models (LLMs), such as GPT-3 (Brown et al., 2020) and InstructGPT (Ouyang et al., 2022), has allowed many researchers to explore their capability in diverse settings, such as generating datasets in natural language inference (NLI) (Liu et al., 2022), reasoning (Ho et al., 2023), and text annotation (Huang et al., 2023; Gilardi et al.,

[1] CLHAIF refers to SimCSE w/ CLAIF from the original paper, and it is a human+LLM-supervised model since it uses human-generated NLI texts and GPT-3 similarity scores.

Findings of the Association for Computational Linguistics: EACL 2024, pages 624–638
March 17-22, 2024 ©2024 Association for Computational Linguistics

2023). Specifically in the context of semantic textual similarity (STS) (Agirre et al., 2012, 2013, 2014; Marelli et al., 2014; Agirre et al., 2015; Cer et al., 2017; Agirre et al., 2016), LLMs have been useful for generating positive and negative samples (defined in Section 2.1) (Schick and Schütze, 2021; Liu et al., 2022; Cheng et al., 2023) and obtaining LLM feedback score to assess the similarity of reference and positives (Cheng et al., 2023).

Despite the increasing utility of LLMs for data generation and model evaluation, numerous studies still use comparably smaller sized sentence embedding backbone models (Gao et al., 2021; Jiang et al., 2022; Zhong et al., 2022; Cheng et al., 2023; Klein and Nabi, 2023), such as BERT-base (110M) (Devlin et al., 2019), RoBERTa-large (355M) (Liu et al., 2019), and T5-large (800M) (Raffel et al., 2020) to build neural evaluators for STS tasks. It is necessary to fine-tune these million-scale pre-trained language models with human or LLM-generated positives and negatives to achieve a high correlation with human evaluations (Jiang et al., 2022) and to better understand how sentence embeddings are represented in a latent space (Ethayarajh, 2019; Gao et al., 2021), which cannot be done merely by prompting LLMs.

Based on the observation that LLM-supervised models consistently underperform when compared to models trained on human-annotated data, they are often compared with less challenging, unsupervised models (Schick and Schütze, 2021; Cheng et al., 2023) (Figure 1), we seek to study the following research questions: 1. What kinds of properties exist in LLM-generated positives/negatives that differ from human-generated texts for building sentence embedding models? 2. Are the standard contrastive training objective losses (*e.g.*, SimCSE (Gao et al., 2021) and CLHAIF (Cheng et al., 2023)) sufficient to learn the relationship between sentence triplets? Our main contributions are as follows:

- We compare embedded properties between human and LLM-generated texts used for fine-tuning sentence embedding models.
- We propose a new loss applicable to any sentence embedding models that are to be fine-tuned with sentence triplets to learn a more intuitive relationship.
- We conduct experiments on the effectiveness of our loss in terms of Spearman correlation and sentence anisotropy, showing more distinctive performances in larger models.

2 Related Works

2.1 Sentence Embeddings

To improve the sentence embedding representations, contrastive learning has been widely employed by minimizing the distance between a semantically similar pair (*alignment*) and maximizing the distance between a random pair (*uniformity*) (Gao et al., 2021). The former refers to a pair of reference text and positive sample (*i.e.*, positive), and the latter contains a reference text and negative sample (*i.e.*, hard-negative[2]). These pairs could be either generated with an unsupervised or supervised approach. In the unsupervised setting, a sentence embedding model (*e.g.*, BERT-base) is fine-tuned with positives constructed by data augmentation strategies such as dropout (Gao et al., 2021; Yan et al., 2021), adversarial attacks, token shuffling, cut-off (Yan et al., 2021), different prompt-based templates (Jiang et al., 2022). A more recent study, Deng et al., 2023 detects hard-negatives in in-batch negatives, and Zhuo et al., 2023 enhances the diversity of positives by performing whitening for embedding features in different subgroups. Finally, Klein and Nabi, 2023 enforces alignment of the attention tensors of positives. However, these unsupervised models still show lower performances on STS tasks than supervised models.

Supervised models leverage human-generated texts, especially natural language inference (NLI) datasets (SNLI: Bowman et al., 2015 and MNLI: Williams et al., 2018) since they are known to be most effective for training a sentence embedding model (Conneau et al., 2017; Reimers and Gurevych, 2019; Gao et al., 2021). Specifically, SBERT is BERT cast with a 3-way (entailment, neutral, and contradiction) classification task using siamese and triplet network structures (Reimers and Gurevych, 2019). On the other hand, Gao et al., 2021 regards only *entailed* and *contradicted* sentences with respect to reference texts from NLI datasets as *positives* and *hard-negatives*. Jiang et al., 2022 reformulates sentence embedding task to masked language task using the same human-generated NLI dataset as Gao et al., 2021 to improve the quality of predicted tokens. However,

[2] We use the term "negative" and "hard-negatives" interchangeably throughout this paper.

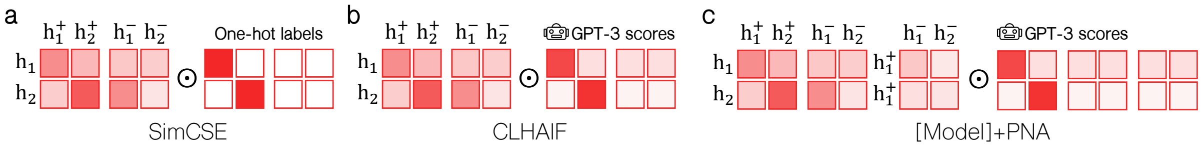

Figure 2: Comparison of log softmax of cosine similarity and labels between (a) Gao et al., 2021, (b) Cheng et al., 2023, and (c) ours. For simplicity, the batch size is two, and a warmer color indicates a higher value. h_i, h_i^+, and h_i^- ($i = 1, 2$) are encoded reference, positive, and negative, respectively, and $\odot$ denotes element-wise multiplication. Unlike (a) SimCSE and (b) CLHAIF, (c) PNA incorporates the cosine similarity of encoded positives and negatives.

these prior works do not focus on the relationship between positives and negatives.

2.2 Large Language Model

Shifting a data creation paradigm from relying only on human workers to combining both humans and LLMs improves the quality and diversity of the datasets (Liu et al., 2022) and reduces per-annotation cost (Gilardi et al., 2023). However, whether LLMs are truly helpful in making well-represented sentence embeddings has yet to be investigated. Although several sentence embedding models fine-tuned with datasets produced by pre-trained LLMs, such as DINO (Schick and Schütze, 2021) and CLAIF (Cheng et al., 2023) exhibit better performances than unsupervised models, they are still below sentence embedding models fine-tuned with human-generated NLI datasets like SimCSE (Gao et al., 2021).

3 Methods

Here we first present how we conduct a heuristic evaluation on human/LLM-generated datasets (3.1). Next, we propose a novel loss objective called Positive-Negative Augmentation (PNA) that can be applied to sentence embedding models that are to be fine-tuned with any type of sentence triplet datasets either generated with human, LLM, or both (3.2). The explanation of proposing PNA loss after the heuristic evaluation is stated in Section 6.

3.1 Heuristic evaluation on texts/scores generated by humans/LLM

We capture different aspects of properties in human or LLM-generated texts that are used for building sentence embedding models by examining four perspectives: 1. Positive Text Repetition (PTR), 2. Positive Score Compactness (PSC), 3. Length Difference Penalty (LDP), and 4. Negative Text Implausibility (NTI). We normalize each of these four perspectives of scores across datasets to be summed as one to make a distribution.

PTR measures the overlapping n-grams between reference and positive excluding the subject[3] with BLEU-1 (Papineni et al., 2002). This score assesses how many diverse wordings humans or LLM use to make positives, not relying on the superficial clues of words or phrases that already appeared in reference texts (Kavumba et al., 2019).

PSC score is a reciprocal of the variance of similarity scores for positive pair (*i.e.*, reference and positive). This metric captures a wide range of similarity scores since even within positive pairs, some pairs might have a higher similarity (score: 0.9), while others might have less semantically similar meaning (score: 0.7). A lower PSC score indicates more various levels of scores between references and positives. It should be noted that datasets with similarity scores can be evaluated with PSC scores.

LDP score is penalized if there is a large difference between the length of reference and the positive. Hence, a lower LDP suggests that humans or LLM produce positive with a length very close to the reference length.

NTI scores the implausibility of hard-negatives by prompting GPT-3.5-turbo to answer in a binary mode whether each human or LLM-generated positive can happen in real life (Appendix A). We calculate the ratio of negative answers out of valid generated outputs to define NTI. Note that this measure can be applied to datasets containing hard negatives.

3.2 PNA objective definition

We present a training loss, namely PNA, that can be integrated with other sentence embedding models such as SimCSE (Gao et al., 2021) and CLHAIF

[3] We use the Python package, spacy (Explosion, 2017) to identify the subject in a sentence.

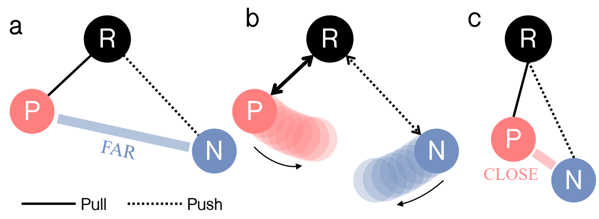

Figure 3: Different cases of the relationship among reference, positive, and negative. Aligning positives to references and distancing negatives from references either leads to positives and negatives (a) far apart or (b, c) become close together.

(Cheng et al., 2023) by incorporating the cosine similarity between positives and negatives (Figure 2). Whereas previous models only learn the relationship between a reference and a positive or reference and a negative, our PNA loss also allows the model to learn the relationship between embedded positives and negatives. In other words, the objectives of SimCSE and CLHAIF are to pull the reference-positive pair together and push the reference-negative pair apart, which does not guarantee the ideal “far” distance between the positive and negative (Figure 3). Also, whereas the human-generated positives are weighted equally with one-hot labels in SimCSE, we use label smoothing using GPT-3 scores, inspired by CLHAIF (smooth-all version) (Cheng et al., 2023). Here is a proposed Positive Negative Augmentation (PNA) loss that includes the relationship between positives and negatives:

$$L_i = y_i^+ \log \frac{e^{\cos(\mathrm{h}_i,\mathrm{h}_i^+)/\tau}}{S} + y_i^- [\sum_{j=1,j\neq i}^{N} \log \frac{e^{\cos(\mathrm{h}_i,\mathrm{h}_j^+)/\tau}}{S} + \sum_{j=1}^{N} (\log \frac{e^{\cos(\mathrm{h}_i,\mathrm{h}_j^-)/\tau}}{S} + \log \frac{e^{\cos(\mathrm{h}_i^+,\mathrm{h}_j^-)/\tau}}{S})]$$

$$S = \sum_{j=1}^{N} (e^{\cos(\mathrm{h}_i,\mathrm{h}_j^+)/\tau} + e^{\cos(\mathrm{h}_i,\mathrm{h}_j^-)/\tau} + e^{\cos(\mathrm{h}_i^+,\mathrm{h}_j^-)/\tau})$$

$$y_i^+ = \mathrm{SimScore}(\mathrm{x}_i, \mathrm{x}_i^+)$$

$$y_i^- = \frac{1 - y_i^+}{3N - 1}$$

In the above equations, L_i is the proposed PNA loss function for each sample from a batch containing N positives and N negatives, and h_i, h_i^+, and h_i^- are sentence encodings of reference (x_i), positive (x_i^+), and negative (x_i^-). y_i^+ is a similarity score between reference and positive. This can be computed by the GPT-3 score for CLHAIF (Cheng et al., 2023) or randomly generated from the uniform distribution ranging from 0 to 1 for SimCSE (Gao et al., 2021). y_i^- is a uniformly divided score from the rest of the probability minus the target label score (y_i^+). τ indicates a temperature, which we set to a fixed value of 0.05.

4 Experiments

4.1 LLM-generated dataset analysis

Datasets We conduct an analysis to investigate what properties make LLM-supervised models perform lower than human-supervised models by comparing four sets of datasets: DINO (Schick and Schütze, 2021), CLAIF (Cheng et al., 2023), NLI (Gao et al., 2021), and DINO$_{\text{GPT-3.5}}$, which include positives/negatives generated by prompting GPT-3.5-turbo for a randomly sampled 100k references from the NLI dataset (Appendix A).

DINO dataset contains pairs of GPT2-XL (Radford et al., 2019)-generated sentences with three levels of similarity[4] (Schick and Schütze, 2021). We manually assign positives for the datasets with a similarity score close to 1 (n =20,013).

CLAIF dataset consists of sentence pairs and similarity scores that are generated by prompting GPT-3 to fill out the masked sentences and to label a similarity score ranging from 0 to 1, respectively (Cheng et al., 2023). We select positives as samples that have GPT-3 similarity scores higher than 0.5 (n = 53,041).

NLI dataset is the only human-generated dataset consisting of sentence triplets (Bowman et al., 2015; Williams et al., 2018). We use the GPT-3 similarity scores for each triplet provided by Cheng et al., 2023 to select positives as the samples with GPT-3 score higher than 0.5 (n =198,479).

DINO$_{\text{GPT-3.5}}$ is a relabeled DINO (Schick and Schütze, 2021) dataset using GPT-3.5-turbo to examine the effect of stronger LLM baseline (Appendix B). Since it does not contain a corresponding similarity score, we select instances from the datasets with the same indices as the selected NLI dataset (n =198,479).

4.2 PNA implementation

PNA-applicable models We implement PNA loss, which can be applied to any sentence embedding model fine-tuned using triplet data, such

[4] 0: completely different, 0.5: somewhat similar, 1: same

as SimCSE (Gao et al., 2021), CLHAIF (Cheng et al., 2023), and DINO$_{\text{GPT-3.5}}$. To ensure fairness, we reproduce these models with and without PNA and always extract the average ("avg") of the hidden state in the last layer for each token for making sentence embeddings[5]. The fine-tuning/evaluation details are stated in Appendix B.

Model categorization The models mentioned in this paper fall into one of the following categories: 1. *Static token embeddings* (BERT static avg. from Jiang et al., 2022), 2. *Pre-trained-only* (BERT last avg. from Jiang et al., 2022), 3. *Human-supervised* (SBERT/SRoBERTa from Reimers and Gurevych, 2019; supervised SimCSE from Gao et al., 2021), 4. *LLM-supervised*[6] (DINO from Schick and Schütze, 2021; DINO$_{\text{GPT-3}}$ and CLAIF from Cheng et al., 2023), and 5. *Human+LLM-supervised* (SimCSE w/ CLHAIF from Cheng et al., 2023). Our backbone models are BERT-base (Devlin et al., 2019) and RoBERTa-base/large (Liu et al., 2019).

False negative elimination strategy We additionally implement false negative elimination method inspired by Huynh et al., 2022 for three PNA-applicable models: DINO$_{\text{GPT}-3.5}$, SimCSE, and CLHAIF) and SimCLHAIF. This approach discards one in-batch negative sample with the highest cosine similarity. In-batch negatives for each sample refer to one hard-negative pair and all the other implicit negatives, such as positives and negatives of other samples within the same batch. For instance, in-batch negatives for h_1 in Figure 2 are h_1^- (hard-negative), h_2^+ (positive of the other sample, h_2), and h_2^- (negative of the other sample, h_2).

Tasks We assess the alignment between the sentence embedding model and human-annotated ranking scores by computing Spearman's correlation on STS tasks, consisting of STS 2012-2016 (Agirre et al., 2012, 2013, 2014, 2015, 2016) STS-Benchmark (Cer et al., 2017), and SICK-Relatedness (Marelli et al., 2014). Furthermore, we evaluate how much random sentence embeddings are uniformly distributed in the latent space. We compute a sentence anisotropy defined as cosine similarity between two embeddings from all combinations of 100k sentence pairs sampled from Wikipedia corpus (Jiang et al., 2022). It is crucial to reduce the sentence anisotropy or to maximize the distance of random sentence pairs in the latent space to avoid representation collapse (Gao et al., 2021; Ethayarajh, 2019).

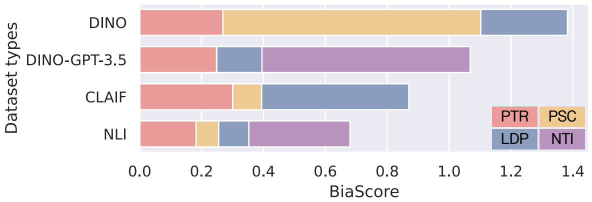

Figure 4: Comparison of PTR, PSC, LDP, and NTI scores across datasets (lower the better). NLI achieves the lowest scores in terms of four perspectives: 1. Positive Text Repetition (PTR), 2. Length Difference Penalty (LDP), 3. Positive Score Compactness (PSC), and 4. Negative Text Implausibility (NTI).

Model	**Layer**	**Spearman correlation ↑**	**Sentence anisotropy ↓**
Static token embeddings			
BERT-b$^\diamond$	First	56.02	0.8250
RoBERTa-b$^\diamond$	First	55.88	0.5693
RoBERTa-l*	First	55.47	0.9100
Pre-trained-only			
BERT-b*	Last	52.58↓	0.4859↓
RoBERTa-b$^\diamond$	Last	53.49↓	0.9554↑
RoBERTa-l*	Last	52.80↓	0.9911↑
Human-supervised (SimCSE+PNA)			
BERT-b	Last	80.48↑	0.3770↓
RoBERTa-b	Last	79.01↑	0.7911↑
RoBERTa-l	Last	81.63↑	0.4051↓
Human+LLM-supervised (CLHAIF+PNA)			
BERT-b	Last	81.01↑	0.3936↓
RoBERTa-b	Last	80.71↑	0.7964↑
RoBERTa-l	Last	82.91↑	0.3959↓

Table 1: Average Spearman's correlation on STS tasks and sentence anisotropy on Wikipedia corpus. Fine-tuning a sentence embedding model with human/LLM-generated texts is needed to improve Spearman's correlation and allay sentence anisotropy issues. $\diamond$: Jiang et al., 2022; $*$: reproduced results (Appendix B).

5 Results

Inherent differences between human and LLM-generated texts In Figure 4, the *human-generated* NLI dataset scores the lowest PTR, PSC, LDP, and NTI scores compared to the other *LLM-generated* datasets such as DINO (Schick and Schütze, 2021), DINO$_{\text{GPT-3.5}}$, and CLAIF (Cheng

[5] The pooler type for the original CLHAIF is "cls" ([CLS] representation with MLP pooler) for BERT-b and "avg" for RoBERTa-b., and SimCSE reports "cls."

[6] We exclude CLAIF$_{\text{scaled}}$ (Cheng et al., 2023) because it is intentionally built to use four times larger fine-tuning dataset size than the other models using STS-B and NLI datasets.

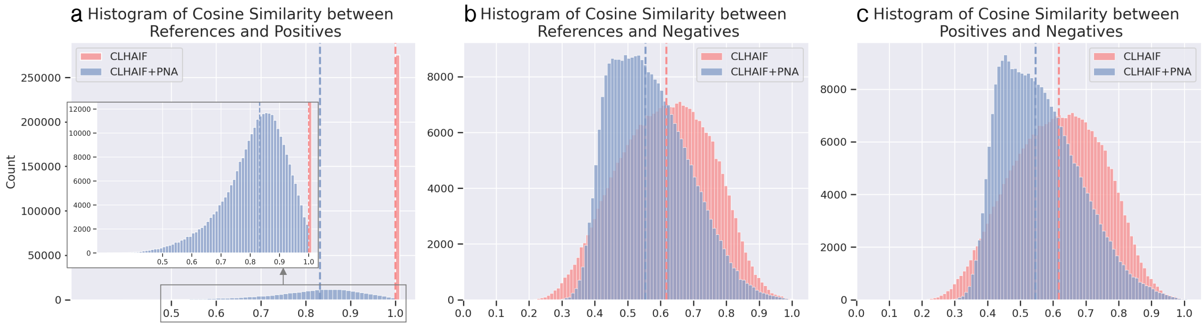

Figure 5: The distribution of cosine similarity between references, positives, and negatives from the training NLI dataset. CLHAIF+PNA (backbone: BERT-b) assigns (a) different levels of similarity score (≤ 1.0) between reference and positive pairs and (b, c) lower similarity scores for reference/positives and negative pairs than CLHAIF.

et al., 2023), showing the inherent, irreducible differences between LLM and human-generated datasets. Specifically, we observe the lowest amount of positive text repetitions (PTR) in the NLI dataset, suggesting that humans use more diverse wordings to write positive samples. The NLI dataset also shows the lowest positive score compactness (PSC), implying that it has a wide scale of scores between a reference and a positive pair (0.094 for CLAIF and 0.073 for NLI). Whereas CLAIF produces positives with a length different from that of references (LDP ↑), NLI and DINO$_{\text{GPT-3.5}}$ have more similar lengths for references and positives. Lastly, DINO$_{\text{GPT-3.5}}$ contains more non-realistic samples (NTI ↑) compared to the NLI dataset. Overall, the resulting heuristic scores suggest that it is challenging to generate high-quality positive and hard-negative pairs for NLI dataset instances with LLM to be on par with human-generated positives and hard negatives.

Necessity of fine-tuning Although the Spearman's correlation performance of pre-trained language models degrades using the averaged embeddings from the last layer compared to the static input embeddings (Jiang et al., 2022), as can be seen in Table 1, we observe that Spearman's correlation increases significantly (at least more than 23%) than static token embeddings for fine-tuned models - SimCSE+PNA and CLHAIF+PNA. At the same time, fine-tuning alleviates the sentence anisotropy problem since our models overall show lower sentence anisotropy than static token embeddings[7]. Hence, fine-tuning overall helps the baseline models attain a high Spearman correlation and prevents arbitrary sentence embeddings from being clustered together.

Reduced cosine similarity among references, positives, and negatives Pushing positives and negatives apart in the fine-tuning process allows the sentence embedding model to capture different levels of similarity score between the embedded references and positives (Figure 5). It is crucial to note that CLHAIF without PNA also uses GPT-3 feedback scores with a smooth-all setting, but it shows a similarity score of 1.0 for almost all the samples. That means, without PNA, the model only learns to locate embedded references and positives as close to each other, not considering the relationship between positives and negatives. In addition, the overall cosine similarity between references/positives and negatives decreases using CLHAIF/SimCSE+PNA compared to CLHAIF/SimCSE, showing better fine-tuning results (Figures 5 and 10).

Spearman correlation improvement Implementing PNA on the representative human-supervised model, SimCSE, and human+LLM-supervised model, CLHAIF helps to improve the Spearman's correlations for most STS tasks, especially for RoBERTa-l, achieving 3.14% and 1.47% higher results for SimCSE+PNA and CLHAIF+PNA compared to SimCSE and CLHAIF, respectively (Table 2). Even though using PNA may not always lead to significantly higher Spearman's correlation for STS tasks, it should be emphasized that PNA better captures different levels of similarity for references, positives, and negatives (Figure 5) and alleviates sentence anisotropy problem (Figure 7).

Comparison with false negative elimination strategy In figures 6 and 7, we use RoBERTa-

[7] The sentence anisotropy of RoBERTa-b is already very low in static token embeddings compared to the other models.

	Model	STS12	STS13	STS14	STS15	STS16	STS-B	SICK-R	Avg.
BERT-b	SBERT$^{\heartsuit}$	70.97	76.53	73.19	79.09	74.30	77.03	72.91	74.89
	DINO$_{\text{GPT-3}}$§	72.61	81.92	75.09	80.42	76.26	77.10	70.43	76.26
	DINO$_{\text{GPT-3.5}}$	70.66	82.14	74.06	80.00	78.05	78.73	72.99	76.66
	CLAIF§	70.62	81.51	76.29	85.05	81.36	**84.34**	78.22	79.63
	SimCSE*	**75.47**	82.39	76.78	85.36	80.72	82.68	80.24	80.52
	+PNA	72.40	83.91	78.86	85.49	80.63	82.69	79.37	80.48
	CLHAIF*	75.19	82.89	78.05	85.93	80.79	83.01	**81.21**	81.01
	+PNA	73.54	**84.83**	**79.96**	**86.26**	**81.37**	83.24	79.25	**81.21**↑
RoBERTa-b	SRoBERTa$^{\heartsuit}$	71.54	72.49	70.80	78.74	73.69	77.77	74.46	74.21
	DINO$^{\clubsuit}$	70.27	81.26	71.25	80.49	77.18	77.82	68.09	75.20
	DINO$_{\text{GPT-3}}$§	71.24	81.55	75.67	81.42	78.77	80.10	71.31	77.15
	DINO$_{\text{GPT-3.5}}$	72.58	82.65	75.01	78.80	80.60	80.22	72.25	77.44
	CLAIF§	68.33	82.26	77.00	**85.18**	**83.43**	**85.05**	78.02	79.90
	SimCSE*	77.26	73.80	75.14	83.44	81.10	81.59	78.06	78.63
	+PNA	74.65	78.27	78.24	84.12	81.26	80.95	75.56	79.01↑
	CLHAIF*	**78.48**	81.74	79.05	84.99	81.42	82.66	**78.72**	**81.01**
	+PNA	76.34	**82.78**	**80.60**	84.85	81.91	82.47	75.99	80.71
RoBERTa-l	SRoBERTa$^{\heartsuit}$	74.53	77.00	73.18	81.85	76.82	79.10	74.29	76.68
	DINO$_{\text{GPT-3.5}}$	71.36	81.40	75.55	80.82	80.93	81.15	74.60	77.97
	CLAIF*	71.86	83.69	78.81	86.04	**83.92**	**85.44**	**80.66**	81.49
	SimCSE*	77.45	75.48	77.10	82.64	81.75	82.61	72.43	78.49
	+PNA	76.07	84.43	81.62	86.28	82.39	84.09	76.52	81.63↑
	CLHAIF*	**77.81**	84.43	81.26	85.41	82.79	84.70	73.67	81.44
	+PNA	77.13	**87.08**	**83.27**	**87.13**	83.14	85.39	77.20	**82.91**↑

Table 2: Spearman's correlation performances of human (red), LLM (blue), and human+LLM (purple)-supervised sentence embedding models across STS tasks. Using PNA for fine-tuning SimCSE and CLHAIF enhances the correlation performances for most STS tasks, especially for RoBERTa-l. ♡: Reimers and Gurevych, 2019; §: Cheng et al., 2023; ♣: Schick and Schütze, 2021; ⋆: reproduced results (Appendix B). Bold and underlined texts indicate the first and the second best value for each backbone model and STS task.

l as the backbone model to observe the effect of PNA on both Spearman's correlation and sentence anisotropy. Although dropping false negative improves the averaged Spearman's correlation performances for DINO$_{\text{GPT-3.5}}$, SimCSE, and CLHAIF, adding PNA shows higher and more robust improvement for all four models in terms of Spearman's correlation (Figure 6) and sentence anisotropy (Figure 7). Between these two figures, in most cases, there exists a trade-off between Spearman's correlation and sentence anisotropy.

Scalability of sentence embedding models Varying the fine-tuning data size from 0 (corresponding to the pre-trained-only model from Table 1) to the full NLI dataset (n =275,601), CLHAIF+PNA shows the second highest performance starting from 10k data size among the models after SimCLHAIF+PNA (Figure 8)[8]. However, with insufficient training data (*e.g.*, < 10k), CLHAIF+PNA has the lowest performance. Although most models reach a similar rate of convergence for Spearman's correlation, PNA-based models exhibit later convergence of sentence anisotropy (Figure 9). The sentence anisotropy values also seem to be noisier than Spearman's correlations, and the best model in terms of Spearman's correlation, CLHAIF+PNA, is not the best in terms of sentence anisotropy.

6 Discussion

What is the motivation for proposing PNA loss after the heuristic evaluation? In this paper, we first explore why the LLM-generated dataset, while widely used and cost-efficient, is less beneficial than the human-generated dataset for fine-tuning a sentence embedding model and evaluate the existing human-generated dataset (NLI)

[8] SimCLHAIF+PNA shows the highest correlation even from the start since it is already fine-tuned on full NLI dataset, whereas other models are only pre-trained not fine-tuned.

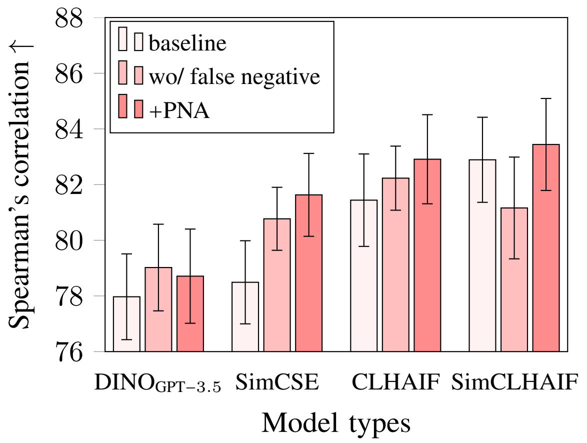

Figure 6: Effect of PNA on Spearman's correlation. The correlation increases for all four types of models (backbone: RoBERTa-l) with PNA compared to the baselines more than the models fine-tuned without false negatives. The error bar indicates standard error across seven STS tasks.

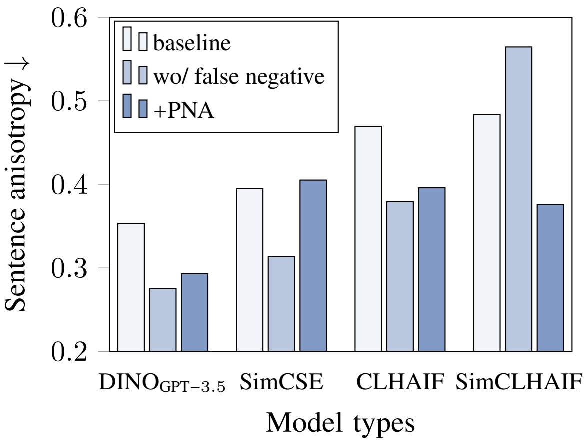

Figure 7: Effect of PNA on sentence anisotropy. The cosine similarity for arbitrary sentence pairs decreases for three out of four types of models (backbone: RoBERTa-l) with PNA compared to the baselines.

and LLM-generated datasets (DINO and CLAIF) and a newly introduced LLM-generated dataset (DINO-GPT-3.5) in four perspectives. The reason for this heuristic evaluation is that we originally wanted to show that it might be possible to outperform human-supervised SimCSE, which is the standard SOTA sentence embedding model without any prompt variations with the model fine-tuned with DINO-GPT-3.5. However, similarly to Schick and Schütze, 2021; Cheng et al., 2023, we find it difficult to generate high-quality texts to be on par with human-generated texts.

Hence, we instead delve into why a difference exists between LLM and human-generated datasets. After analyzing the difference with our heuristic evaluation approach, we acknowledge the limitation of only using LLM to build higher-quality datasets like NLI. Thus, rather than focusing on creating an LLM-generated dataset more like a human-generated dataset, which is possibly due to the limitation of the current LLM, we attempt to devise a way to improve any model, including the current SOTA sentence embedding model, which is human+LLM-supervised CLHAIF that uses sentence triplets as the fine-tuning dataset.

Why is it important to consider the relationship between sentence triplets? Although the CLHAIF model is fine-tuned to learn different levels of similarity between references and positives (Cheng et al., 2023), we unexpectedly observe most of the cosine similarity scores are skewed to the overconfident or maximum value, 1.0 in Figure 5. We hypothesize that as the training proceeds, the model mostly focuses on learning the relations across the data instances by pushing different instances apart from each other. Hence, the model seemingly forgets to learn the relations within each data instance, keeping reference and positive close together (Figure 5a) and the same for reference/positive and negative (Figure 5b-c).

However, humans can differentiate the subtle different levels of closeness for each sentence triplet (Gulordava and Baroni, 2011). For example, the sentence pairs "I love to explore NLP." and "I like to explore NLP." should show a slightly higher similarity score than the sentence pairs "I love to explore NLP in AI." and "I love to explore arts." if we are to regard "love" and "like" more similar than "NLP and "arts." For the reference/positive-negative pair, it is intuitively better to separate them, which adding the PNA loss helps to achieve.

Why do LLM-supervised models show lower performances than human-supervised models? Though LLMs show remarkable abilities in generating and evaluating text data (Liu et al., 2022, 2023), we find that it is still very challenging to produce *human-like* positives and hard-negatives for each NLI dataset instance. Thus, the performances of LLM-supervised sentence embedding models (*e.g.*, CLAIF) remain much lower than human-supervised models (*e.g.*, SimCSE). Here, we also attempt to make a newer version of DINO (Schick et al., 2021) called DINO$_{\text{GPT-3.5}}$, but it shows lower Spearman correlation performance

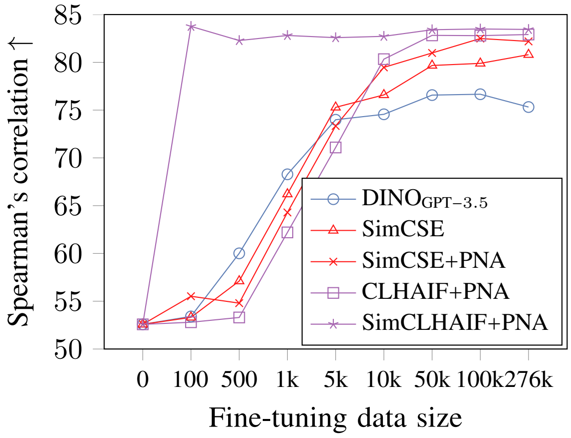

Figure 8: Effect of fine-tuning data size and PNA on Spearman's correlation. The performances of PNA-based models (backbone: RoBERTa-l) are lower than the other models when fine-tuned with less than 10k data, but they converge with much higher values. The error bar indicates standard error across seven STS tasks.

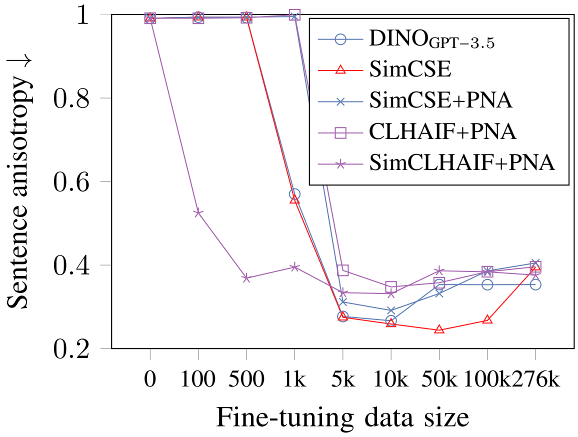

Figure 9: Effect of fine-tuning data size and PNA on sentence anisotropy. The performances of PNA-based models (backbone: RoBERTa-l) converge slower than the other models. SimCLHAIF+PNA, which attains the highest Spearman's correlation (Figure 8) does not produce the lowest sentence anisotropy using more than 10k fine-tuning data.

than human-supervised models (Table 2). One possible reason may be because LLM often constructs unhelpful hard negatives, which are quantified by NTI score (Figure 4; Appendix F). To reduce the biases from LLM-generated texts, we could implement an auxiliary supervised model that helps to revise LLM-generated sentences using human-generated texts as labels.

Is it fair to compare LLM-supervised models with unsupervised models? Throughout this paper, we make a comparison of LLM-supervised models with human-supervised models, whereas these models are generally compared with less challenging, unsupervised models (Schick and Schütze, 2021; Zhang et al., 2023; Cheng et al., 2023). However, this comparison may not be entirely fair since models fine-tuned on LLM-generated data can be viewed as weakly supervised rather than truly unsupervised since LLMs are pre-trained with a large-scale dataset generated by humans or human feedback (Ouyang et al., 2022). Hence, LLM-generated texts could be viewed as the product of weakly-supervised human-generated texts, justifying our stricter comparison criterion. Nevertheless, we leave for future work to discuss this open research question further.

7 Conclusion

We study why LLM-generated texts hinder a sentence embedding model from producing less semantically meaningful sentence representations compared to human-generated texts by analyzing their embedded properties. Then, for the models fine-tuned with human-generated sentence triplets and feedback similarity scores for positive pairs, we enhance the sentence representations with our PNA loss. Not only does PNA help the model to achieve high Spearman's correlation and low sentence anisotropy, but it also captures a wide range of similarity scores between references and positives and returns lower cosine similarity between references/positives and negatives. We hope our work will catalyze efforts in exploring different aspects of LLM-generated texts for various downstream tasks.

Limitations

Although our method effectively reduces sentence anisotropy while maintaining or enhancing SOTA performance on STS tasks, it is important to note that the PNA loss is designed for use with sentence triplets and may not be directly applicable to methods that solely rely on positive sample augmentations during fine-tuning. Furthermore, our evaluation primarily focuses on STS tasks, leaving the performance of PNA loss in other text-embedding tasks largely unexplored. Further research is required to establish its versatility in such cases.

Ethics Statement

When generating positive and negative sentences for DINO$_{\text{GPT-3.5}}$, GPT-3.5-turbo might unintentionally produce harmful content. However, all the reference texts that are used in prompting GPT-3.5-turbo are extracted from NLI datasets. Hence, unless a reference sentence itself or its contradictory sentence addresses a risky topic, we expect almost no harm in our LLM-generated texts.

Acknowledgements

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2019-0-00075, Artificial Intelligence Graduate School Program (KAIST)). We thank Noah Lee and Andrew Wan Ju Kang for providing us with helpful and constructive feedback on the final version of the manuscript.

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)*, pages 385–393.

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.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Daniel Cer, Mona Diab, Eneko E Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. Semeval-2017 task 1: Semantic textual similarity multilingual and cross-lingual focused evaluation. In *The 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14.

Qinyuan Cheng, Xiaogui Yang, Tianxiang Sun, Linyang Li, and Xipeng Qiu. 2023. Improving contrastive learning of sentence embeddings from AI feedback. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 11122–11138, Toronto, Canada. Association for Computational Linguistics.

Alexis Conneau and Douwe Kiela. 2018. Senteval: An evaluation toolkit for universal sentence representations. *arXiv preprint arXiv:1803.05449*.

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.

Jinghao Deng, Fanqi Wan, Tao Yang, Xiaojun Quan, and Rui Wang. 2023. Clustering-aware negative sampling for unsupervised sentence representation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8713–8729, Toronto, Canada. Association for Computational Linguistics.

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 and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Third International Workshop on Paraphrasing (IWP2005)*.

M ELLEN. 2000. Building a question answering test collection. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*.

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.

AI Explosion. 2017. spacy-industrial-strength natural language processing in python. *URL: https://spacy.io*.

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.

Fabrizio Gilardi, Meysam Alizadeh, and Maël Kubli. 2023. Chatgpt outperforms crowd workers for text-annotation tasks.

Kristina Gulordava and Marco Baroni. 2011. A distributional similarity approach to the detection of semantic change in the Google Books ngram corpus. In *Proceedings of the GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics*, pages 67–71, Edinburgh, UK. Association for Computational Linguistics.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A Smith. 2018. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112.

Namgyu Ho, Laura Schmid, and Se-Young Yun. 2023. Large language models are reasoning teachers. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14852–14882, Toronto, Canada. Association for Computational Linguistics.

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.

Fan Huang, Haewoon Kwak, and Jisun An. 2023. Is chatgpt better than human annotators? potential and limitations of chatgpt in explaining implicit hate speech. In *Companion Proceedings of the ACM Web Conference 2023*, pages 294–297.

Tri Huynh, Simon Kornblith, Matthew R Walter, Michael Maire, and Maryam Khademi. 2022. Boosting contrastive self-supervised learning with false negative cancellation. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 2785–2795.

Ting Jiang, Jian Jiao, Shaohan Huang, Zihan Zhang, Deqing Wang, Fuzhen Zhuang, Furu Wei, Haizhen Huang, Denvy Deng, and Qi Zhang. 2022. Prompt-bert: Improving bert sentence embeddings with prompts. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8826–8837.

Pride Kavumba, Naoya Inoue, Benjamin Heinzerling, Keshav Singh, Paul Reisert, and Kentaro Inui. 2019. When choosing plausible alternatives, clever hans can be clever. In *Proceedings of the First Workshop on Commonsense Inference in Natural Language Processing*, pages 33–42.

Tassilo Klein and Moin Nabi. 2023. miCSE: Mutual information contrastive learning for low-shot sentence embeddings. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6159–6177, Toronto, Canada. Association for Computational Linguistics.

Alisa Liu, Swabha Swayamdipta, Noah A. Smith, and Yejin Choi. 2022. WANLI: Worker and AI collaboration for natural language inference dataset creation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6826–6847, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. Gpteval: Nlg evaluation using gpt-4 with better human alignment. *arXiv preprint arXiv:2303.16634*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Marco Marelli, Stefano Menini, Marco Baroni, Luisa Bentivogli, Raffaella Bernardi, Roberto Zamparelli, et al. 2014. A sick cure for the evaluation of compositional distributional semantic models. In *Lrec*, pages 216–223. Reykjavik.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. *arXiv preprint cs/0409058*.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. *arXiv preprint cs/0506075*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992.

Timo Schick and Hinrich Schütze. 2021. Generating datasets with pretrained language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6943–6951.

Timo Schick, Sahana Udupa, and Hinrich Schütze. 2021. Self-diagnosis and self-debiasing: A proposal for reducing corpus-based bias in NLP. *Transactions of the Association for Computational Linguistics*, 9:1408–1424.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y 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.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355.

Janyce Wiebe, Theresa Wilson, and Claire Cardie. 2004. Annotating expressions of opinions and emotions in. *To appear in Language Resources and Evaluation*, 1:2.

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.

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

Junlei Zhang, Zhenzhong Lan, and Junxian He. 2023. Contrastive learning of sentence embeddings from scratch. *arXiv preprint arXiv:2305.15077*.

Ming Zhong, Yang Liu, Da Yin, Yuning Mao, Yizhu Jiao, Pengfei Liu, Chenguang Zhu, Heng Ji, and Jiawei Han. 2022. Towards a unified multi-dimensional evaluator for text generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2023–2038.

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.

A Prompt templates

DINO$_{\text{GPT-3.5}}$ We prompt GPT-3.5-turbo to generate positives and negatives for fine-tuning DINO$_{\text{GPT-3.5}}$ with the temperature set to 1.0 using the templates in Table 3. DINO$_{\text{GPT-3.5}}$ is fine-tuned using the same model architecture as supervised SimCSE with hard-negatives (Gao et al., 2021). We randomly sample 100k references from the NLI datasets to fine-tune the model. For BERT-b, we report the evaluation results of princeton-nlp/sup-simcse-bert-base-uncased (Gao et al., 2021) with the pooler type of "avg," and for RoBERTa-b and RoBERTa-l, we fine-tune the pretrained roberta-base and roberta-large (Liu et al., 2019). DINO$_{\text{GPT-3.5}}$ attains higher averaged Spearman correlation performances than DINO$_{\text{GPT-3}}$ (Cheng et al., 2023) for BERT-b and RoBERTa-b in STS tasks (Table 2) and transfer learning tasks (Table 6).

NTI We instruct GPT-3.5-turbo with the temperature set to 0.0 to answer whether the given sentence that is either human-generated or LLM-generated is plausible or not (Table 4). We consider the generated outputs as valid answers if the output contains either "1," "2," or "3."

B Implementation details

Below, we lay out how we fine-tune and evaluate reproduced models used in Tables 2 and 6 and Figures 5, 6, 7, 8, 9, and 10 using one NVIDIA RTX A6000 for BERT-b and RoBERTa-b and two NVIDIA RTX A6000s for RoBERTa-l:

SimCSE

- BERT-b is evaluated on fine-tuned princeton-nlp/sup-simcse-bert-base-uncased with the pooler type of "avg."
- RoBERTa-b and RoBERTa-l are fine-tuned on roberta-base and roberta-large using 276,501 NLI datasets for three epochs with a batch size of 128 per GPU and a learning rate of 5e-5 (Gao et al., 2021). The models are validated every 125 training steps using Spearman's correlation on the STS-B task.

CLAIF

- The evaluation results of BERT-b and RoBERTa-b are from Cheng et al., 2023.
- RoBERTa-l is fine-tuned on roberta-large using 276,501 NLI datasets with a smooth-all option (Cheng et al., 2023) and the same training implementation as SimCSE (above).

CLHAIF

- BERT-b is evaluated on fnlp/clhaif-simcse-bert-base with the pooler type of "avg."
- RoBERTa-b and RoBERTa-l are fine-tuned on roberta-base and roberta-large using 276,501 NLI datasets and GPT-3 similarity scores with a smooth-all option (Cheng et al., 2023) and the same training implementation as SimCSE.

SimCLHAIF

- BERT-b, RoBERTa-b, and RoBERTa-l are fine-tuned on princeton-nlp/sup-simcse-[model] using the same training process as CLHAIF (above).

C The distribution of cosine similarity

The histograms of cosine similarity for references, positives, and negatives embedded using SimCSE and SimCSE+PNA are visualized in Figure 10. Similar to CLHAIF+PNA from Figure 5, SimCSE+PNA shows reduced cosine similarity than SimCSE for all three cases (Figure 10a-c).

D Full Spearman's correlation performances

We lay out the Spearman's correlations across all STS tasks for static token embeddings and the pre-trained-only model from Table 1 (Table 5). Full performances of human-supervised and human+LLM-supervised models are listed in Table 2.

E Transfer learning task results

PNA-based models do not always show higher Spearman correlation performances than non-PNA-based models on seven transfer learning tasks (Conneau and Kiela, 2018): MR (Pang and Lee, 2005), CR (Hu and Liu, 2004), SUBJ (Pang and Lee, 2004), MPQA (Wiebe et al., 2004), SST-2 (Socher et al., 2013), TREC (ELLEN, 2000), and MRPC (Dolan and Brockett, 2005) (Table 6).

F Hard-negative examples in NLI and DINO$_{\text{GPT-3.5}}$ datasets

Reference: Three people are on a white surface in front of a fenced in area.

Hard-negative (NLI): Two men work on cars.
Hard-negative (DINO$_{\text{GPT-3.5}}$): The three people are swimming in a pool of chocolate syrup.

Reference: A man in a gray suit is talking to another man in a black suit.

Hard-negative (NLI): A man stares at the girls.
Hard-negative (DINO$_{\text{GPT-3.5}}$): The man in the gray suit is actually a robot disguised as a human, having a conversation with an alien in a black suit.

Reference: Four children hold hands and jump into a pool.

Hard-negative (NLI): The children are riding horses.
Hard-negative (DINO$_{\text{GPT-3.5}}$): The children hold hands and jump into a pool filled with sharks.

Reference: A dirt biker is riding through deep sand and dirt.

Hard-negative (NLI): the man is in a coma
Hard-negative (DINO$_{\text{GPT-3.5}}$): A dirt biker is riding through deep sand and dirt, while juggling chainsaws.

Write one sentence that is definitely correct about the situation or event in the following sentence: [reference]

Write one sentence that is definitely incorrect about the situation or event in the following sentence: [reference]

Table 3: Prompt templates for generating positives (top) and negatives (bottom) for DINO$_{\text{GPT-3.5}}$. We adopt the last sentence of prompts presented to the human annotators when making the MNLI dataset (Williams et al., 2018).

Question: Is the following sentence likely to happen in real life? If you answer 'yes,' please provide a reference.
Sentence: [human or LLM-generated negative]
1. Yes.
2. No.
3. I don't know.
Answer:

Table 4: A prompt template for labeling the plausibility of a given text generated by humans or LLM. GPT-3.5-turbo needs to also provide the reference if it answers "yes" to make sure it gives answers based on some evidence.

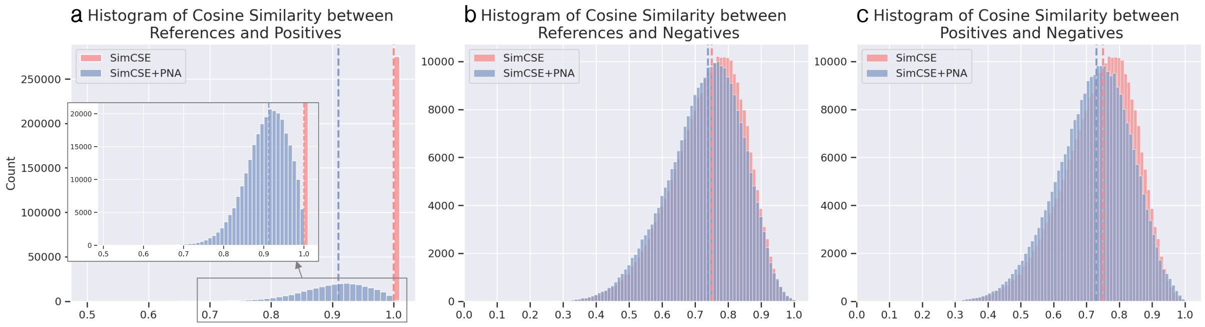

Figure 10: The distribution of cosine similarity between references, positives, and negatives from the training NLI dataset. SimCSE+PNA (backbone: RoBERTa-b) assigns (a) different levels of similarity score (≤ 1.0) between reference and positive pairs and (b, c) slightly lower similarity scores for reference/positives and negative pairs than SimCSE.

Model	STS12	STS13	STS14	STS15	STS16	STS-B	SICK-R	Avg.
Static token embeddings								
BERT-b$^{\diamond}$	42.37	56.74	50.60	65.08	62.39	56.82	58.15	56.02
RoBERTa-b$^{\diamond}$	44.80	57.96	51.24	7.41	59.40	52.17	58.16	55.88
RoBERTa-l*	43.33	58.83	52.09	64.51	58.28	54.14	57.08	55.47
pre-trained-only								
BERT-b*	30.87	59.90	47.73	60.29	63.74	47.29	58.22	52.58↓
RoBERTa-b$^{\diamond}$	32.11	56.33	45.22	61.35	61.98	55.39	62.03	53.49↓
RoBERTa-l*	33.61	57.23	45.66	62.99	61.17	50.56	58.39	52.80↓

Table 5: Full Spearman's correlation of the static token embeddings and unsupervised models from Table 1. There is not much of a difference between the input and the last embeddings (Jiang et al., 2022). $\diamond$: Jiang et al., 2022; $\star$: reproduced results (Appendix B).

	Model	MR	CR	SUBJ	MPQA	SST-2	TREC	MRPC	Avg.
BERT-b	SBERT$^{\heartsuit}$	**83.64**	**89.43**	94.39	89.86	**88.96**	89.60	76.00	**87.41**
	DINO$_{\text{GPT-3}}^{\S}$	79.96	85.27	93.67	88.87	84.29	88.60	69.62	84.33
	DINO$_{\text{GPT-3.5}}^{*}$	82.25	88.40	94.36	90.11	87.75	87.40	75.42	86.53
	CLAIF§	81.64	87.98	94.24	89.34	86.16	89.80	**77.16**	86.62
	SimCSE*	82.51	88.85	94.90	90.24	88.03	88.40	76.29	87.03
	+PNA	82.24	88.69	**94.95**	90.10	87.42	88.60	75.88	86.84
	CLHAIF*	82.15	88.95	94.79	**90.41**	85.94	**90.40**	76.17	86.97
	+PNA	82.30	88.59	94.50	90.00	87.59	90.20	76.00	87.03↑
RoBERTa-b	SRoBERTa$^{\diamond}$	84.91	90.83	92.56	88.75	90.50	88.60	**78.14**	87.76
	DINO$_{\text{GPT-3}}^{\S}$	82.31	88.66	93.95	88.72	87.53	88.20	73.74	86.16
	DINO$_{\text{GPT-3.5}}^{*}$	84.91	90.92	93.62	89.34	91.43	86.40	75.54	87.45
	CLAIF§	84.11	90.62	94.29	89.13	89.57	91.00	77.22	87.99
	SimCSE*	84.62	91.29	**94.86**	89.89	90.99	92.00	76.70	88.62
	+PNA	84.86	91.23	94.54	89.76	**92.09**	91.60	76.64	88.67↑
	CLHAIF*	84.65	91.23	94.53	**90.02**	90.66	**94.20**	77.80	**89.01**
	+PNA	**84.94**	**91.34**	94.63	89.97	91.76	91.60	77.45	88.80
RoBERTa-l	SRoBERTa$^{\heartsuit}$	84.88	90.07	94.52	90.33	90.66	87.40	75.94	87.69
	DINO$_{\text{GPT-3.5}}^{*}$	87.53	92.08	94.72	90.61	**92.37**	88.20	73.91	88.49
	CLAIF*	85.18	90.28	94.56	89.89	90.50	**93.80**	**76.00**	88.60
	SimCSE*	87.50	**92.27**	94.67	90.62	92.20	91.40	74.55	89.03
	+PNA	86.60	91.44	94.86	91.06	92.09	88.60	71.13	87.97
	CLHAIF*	**87.74**	92.18	**95.26**	90.84	91.87	93.20	75.59	**89.53**
	+PNA	87.00	91.55	94.19	**91.16**	92.26	91.40	75.88	89.06

Table 6: Spearman's correlation performances of human (red), LLM (blue), and human+LLM (purple)-supervised sentence embedding models across transfer learning tasks. PNA shows an improvement in some of the transfer learning tasks. $\heartsuit$: Reimers and Gurevych, 2019; §: Cheng et al., 2023; $\diamond$: Jiang et al., 2022; $\star$: reproduced results (Appendix B). Bold and underlined texts indicate the first and the second best value for each backbone model and STS task.