On Isotropy, Contextualization and Learning Dynamics of Contrastive-based Sentence Representation Learning

Chenghao Xiao   Yang Long   Noura Al Moubayed
Department of Computer Science
Durham University
{chenghao.xiao,yang.long,noura.al-moubayed}@durham.ac.uk

Abstract

Incorporating contrastive learning objectives in sentence representation learning (SRL) has yielded significant improvements on many sentence-level NLP tasks. However, it is not well understood why contrastive learning works for learning sentence-level semantics. In this paper, we aim to help guide future designs of sentence representation learning methods by taking a closer look at contrastive SRL through the lens of isotropy, contextualization and learning dynamics. We interpret its successes through the geometry of the representation shifts and show that contrastive learning brings isotropy, and drives high intra-sentence similarity: when in the same sentence, tokens converge to similar positions in the semantic space. We also find that what we formalize as "spurious contextualization" is mitigated for semantically meaningful tokens, while augmented for functional ones. We find that the embedding space is directed towards the origin during training, with more areas now better defined. We ablate these findings by observing the learning dynamics with different training temperatures, batch sizes and pooling methods.

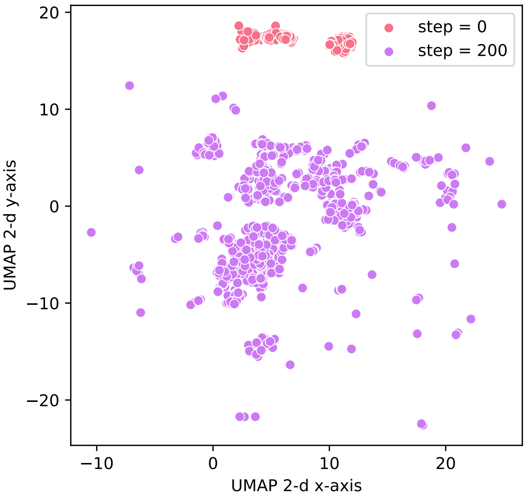

Figure 1: Expanded semantic space produced by contrastive learning (CL), visualized with UMAP. At the beginning of training, all embeddings occupied a narrow cone. After 200 steps of fine-tuning with a contrastive loss, they spread out to define a larger semantic space.

1 Introduction

Since vanilla pre-trained language models do not perform well on sentence-level semantic tasks, Sentence Representation Learning (SRL) aims to fine-tune pre-trained models to capture semantic information (Reimers and Gurevych, 2019; Li et al., 2020; Gao et al., 2021). Recently, it has gradually become *de facto* to incorporate contrastive learning objectives in sentence representation learning (Yan et al., 2021; Giorgi et al., 2021; Gao et al., 2021; Wu et al., 2022).

Representations of pre-trained contextualized language models (Peters et al., 2018; Devlin et al., 2019; Liu et al., 2019) have long been identified not to be isotropic, i.e., they are not uniformly distributed in all directions but instead occupying a narrow cone in the semantic space (Ethayarajh, 2019). This property is also referred to as the representation degeneration problem (Gao et al., 2019), limiting the expressiveness of the learned models. The quantification of this characteristic is formalized, and approaches to mitigate this phenomenon are studied in previous research (Mu and Viswanath, 2018; Gao et al., 2019; Cai et al., 2020).

The concept of learning dynamics focuses on what happens during the continuous progression of fine-tuning pre-trained language models. This has drawn attention in the field (Merchant et al., 2020; Hao et al., 2020), with some showing that fine-tuning mitigates the anisotropy of embeddings (Rajaee and Pilehvar, 2021), to different extent according to the downstream tasks. However, it is argued that the performance gained in fine-tuning is not due to its enhancement of isotropy in the em-

Findings of the Association for Computational Linguistics: ACL 2023, pages 12266–12283
July 9-14, 2023 ©2023 Association for Computational Linguistics

bedding space (Rajaee and Pilehvar, 2021). Moreover, little research is conducted on isotropy of sentence embedding models, especially contrastive learning-based sentence representations.

Vanilla Transformer models are known to underperform on sentence-level semantic tasks even compared to static embedding models like Glove (Pennington et al., 2014; Reimers and Gurevych, 2019), whether using the [cls] token or averaging word embeddings in the output layer. Since Reimers and Gurevych (2019) proposed SBERT, it has become the most popular Transformers-based framework in sentence representation tasks. The state-of-the-art is further improved by integrating contrastive learning objectives (Yan et al., 2021; Gao et al., 2021; Wu et al., 2022). The other line of works concern post-processing of embeddings in vanilla language models (Li et al., 2020; Su et al., 2021; Huang et al., 2021) to attain better sentence representations.

Learning dynamics in fine-tuning was previously investigated, revealing isotropy shifts in the process (Rajaee and Pilehvar, 2021; Gao et al., 2021), but few studies have systematically investigated relevant pattern shifts in sentence representation models, and none has drawn connections between these metrics and the performance gains on sentence-level semantic tasks. While some implicitly studied this problem by experimenting on NLI datasets (Rajaee and Pilehvar, 2021; Merchant et al., 2020; Hao et al., 2020), we argue that a more extensive study on the geometry change during fine-tuning SOTA sentence embedding models with contrastive objectives is neccessary.

In this work, we demystify the mechanism of why contrastive fine-tuning works for sentence representation learning.[1] Our main findings and contributions are as follows:

- Through measuring isotropy and contextualization-related metrics, we uncover a previously unknown pattern: contrative learning leads to extremely high intra-sentence similarity. Tokens converge to similar positions when given the signal that they appear in the same sentence.

- We find that functional tokens fall back to be the "entourage" of semantic tokens, and follow wherever they travel in the semantic space. We argue that the misalignment of the "spurious contextualization change" between semantic and functional tokens may explain how CL helps capturing semantics.

- We ablate all findings by analyzing learning dynamics through the lens of temperature, batch size, and pooling method, not only to validate that the findings are not artifacts to certain configurations, but also to interpret the best use of these hyperparamaters.

Our study offers fundamental insights into using contrastive objectives for sentence representation learning. With these, we aim to shed light on future designs of sentence representation learning methods.

2 Isotropy and Contextualization Analysis of Contrastive-based Sentence Embedding models

2.1 Preliminary

Anisotropy of token embeddings produced by pretrained language models has drawn attention in the field, and been validated both theoretically and empirically (Gao et al., 2019; Ethayarajh, 2019; Cai et al., 2020; Timkey and van Schijndel, 2021).

For an anisotropic model, the embeddings it encodes have a high expected value of pair-wise cosine similarity: $\mathbb{E}_{u,v \in S} cos(u,v) >> 0$, where u and v are contextualized representations of tokens randomly sampled from corpus S.

A contrastive learning objective to fine-tune a PLM on datasets that consist of sentence/document pairs is defined as follows:

$$\ell_i = -\log \frac{e^{sim(e_i, e_i^+)/\tau}}{\sum_{j=1}^{N} e^{sim(e_i, e_j^+)/\tau}}, \quad (1)$$

where e_i and e_i^+ denote embeddings of a sentence/document pair, whose cosine similarity is to be maximized, while all e_j^+ in a same training batch when $j \neq i$ is to be pushed further from e_i.

The central question posed in this paper revolves around the mechanism involved in the contrastive learning process that diminishes anisotropy, leading to an isotropic model. If anisotropy is neutralized, we would observe a new mathematical expectation of cosine similarity, represented by $\mathbb{E}_{u,v \in S} cos(u,v) \approx 0$. However, the precise process and the underlying mechanism that facilitate this transition remain the key questions we aim to address.

[1] Our code is publicly available.

Therefore, metrics such as self-similarity of same tokens in different contexts, and intra-sentence similarity of different tokens in the same context, are pertinent. More importantly, we could further trace the contextualization shift that brings mitigated anistropy to word type, i.e., are functional words and semantic words less/more contextualized after contrastive learning? We show that, this finding could potentially attribute to the performance gain on sentence-level semantic tasks brought by contrastive fine-tuning.

2.2 Metrics

We adopt the metrics defined in Ethayarajh (2019), who studied the extent to which word representations in pre-trained ELMo, BERT, and GPT-2 are contextualized, taking into consideration their anisotropy baselines. We reimplement the computation on self-similarity, intra-sentence similarity, and anisotropy baselines. We then break the similarity measures down into dimension level to inspect whether certain rogue dimensions (Timkey and van Schijndel, 2021) dominate these metrics and therefore making the similarity measures only artifacts of a small set of dimensions.

Self Similarity: Self similarity measures the similarity among different contextualized representations of a token across different contexts. Higher self-similarity indicates less contextualization. Given a token x, we denote the set of token embeddings of x contextualized by different contexts in corpus S as $S_{\vec{x}}$. Self similarity is then defined as the empirical mean of pair-wise cosine similarity of contextualized embeddings of token x in all these contexts:

$$selfsim(x) \triangleq \mathbb{E}_{u,v \in S_{\vec{x}}}[c\vec{o}s(u,v)] \tag{2}$$

Intra-sentence Similarity: By contrast, intra-sentence similarity measures the similarity across tokens in the same context.

Given a sentence s with n tokens $x_{i \in \{1,2,\ldots,n\}}$, we first attain sentence representation $\vec{s}$ by mean-pooling, i.e., averaging all token embeddings $\vec{x_i}$. Intra-sentence similarity is then defined as the average cosine similarity between token representations $\vec{x_i}$ and the sentence representation $\vec{s}$.

$$\begin{aligned} \vec{s} &\triangleq \frac{1}{n} \sum_{x_i \in s} \vec{x_i} \\ intrasim(s) &\triangleq \frac{1}{n} \sum_{n} cos(\vec{x_i}, \vec{s}) \end{aligned} \tag{3}$$

Intra-sentence similarity provides a quantitative measure of the extent to which tokens in the same sentence are similar, allowing us later to derive insights on: whether token representations would converge in the semantic space only because they appear in a same sentence.

Anisotropy Baselines: While self and intra-sentence similarity are computed given the restrictions of respectively 1) same word in different contexts 2) different words in the same context, these values are not reflective of the general distribution across different words and different contexts.

In line with Ethayarajh (2019), we adjust the above two metrics by substracting the anisotropy baseline of a model from them, i.e., average cosine similarity between randomly sampled tokens from different contexts as defined in preliminary.

Dimension-level Inspection of the Metrics Due to the fact that cosine similarity is highly sensitive to outlier dimensions, we inspect whether the outcomes of the above measurements are only artifacts of these dimensions, i.e. rogue dimensions (Timkey and van Schijndel, 2021).

Formally, the cosine similarity of two embeddings is defined as: $cos(u,v) = \frac{u \cdot v}{\|u\|\|v\|}$, where u and v are two embeddings to measure against. Since the term $u \cdot v$ is just a sum of the element-wise dot product of the i^{th} dimension of the embeddings, it is convenient to inspect the contribution each dimension makes to the global similarity: $cos(u,v) = \sum_{i=1}^{d} \frac{u_i v_i}{\|u\|\|v\|}$.

Given a set S that consists of n randomly sampled representations, the expected contribution of the i^{th} dimension in a model to a similarity metric could be approximated as:

$$cos_i = \mathbb{E}_{u,v \in S} \frac{u_i v_i}{\|u\| \, \|v\|}, \tag{4}$$

By breaking the global metrics down to dimension level, whether the output of a metric is a global property of all embeddings in the language model or is only dominated by a set of rogue dimensions D could be inspected by whether $\sum_{i \in D} cos_i >> \frac{\|D\|}{d} \mathbb{E}_{u,v \in S} cos(u,v)$, with d being the dimensionality of word embeddings.

Nonetheless, we could mathematically derive that, dominating dimensions dominate corpus-level similarity metric computations mostly because of their high average distances to the origin at the corresponding dimensions. However, if the values in these dimensions do not have high variation, then eliminating the top $\|D\|$ of these dimensions from the embeddings would not significantly bring semantic shifts to the original representations and therefore would not affect the corresponding relative similarity relationship between sentence pairs.

Therefore, we will also need to inspect whether there is a misalignment between the existence of the rogue dimensions, and their actual impact on informativity (Timkey and van Schijndel, 2021). Given a $f(t, k)$ that maps a token t to its representation, with top k rogue dimensions eliminated, we could compare the correlation between similarity measures yielded by the original representatations and those with top-k rogue dimensions removed. Formally, given:

$$cos_{original}(\mathcal{O}) = cos_{x,y \in \mathcal{O}}(f(x, 0), f(y, 0)) \quad (5)$$

$$cos_{post}(\mathcal{O}) = cos_{x,y \in \mathcal{O}}(f(x, k), f(y, k)), \quad (6)$$

we compute: $r = Corr[cos_{original}, cos_{post}]$, which is an indicator of the "authenticity" of the representations left without these rogue dimensions.

With the corresponding dimension-level inspections of the three metrics, we could take a step further to investigate whether fine-tuning a vanilla language model to sentence embedding tasks with the contrastive objective mitigates the dominance of rogue dimensions.

2.3 Models

We analyze two models that achieve SOTA performances on sentence embedding tasks and semantic search tasks, *all-mpnet-base-v2* [2] and *all-MiniLM-L6-v2*. [3] They have both been fine-tuned with a contrastive loss on 1B+ document pairs, with the goal of predicting the right match to a document d_i given its ground-true match d_i^+ and the rest of the in-batch d_j^+ as natural negative examples. The prediction is conducted again reversely with d_i^+, d_i and other in-batch d_j. The loss is averaged for these two components for every batch. The representation of each document d is by default the mean-pooled embedding of each token.

We compare the results to their vanilla versions, *mpnet-base* (Song et al., 2020) and *MiniLM*[4] (Wang et al., 2020) to get a closer look to the initial state of their corresponding pre-trained counterparts, and how the metrics change after fine-tuning on the goal of getting better sentence and document-level representations.

2.4 Data

We use STS-B (Cer et al., 2017), which comprises a selection of datasets from the original SemEval datasets between 2012 and 2017. We attain the dataset through Hugging Face Datasets[5]. Notably, the models that we are looking at were not exposed to these datasets during their training. Therefore, the pattern to be found is not reflective of any overfitting bias to their training process.

We use the test set and only use sentence 1 of each sentence pair to prevent the potential doubling effect on self-similarity measure, i.e., providing tokens with one more sentence where they are in the similar contexts. Following the description, 1359 sentences are selected as inputs.

2.5 Result

We show that after fine-tuning with contrastive loss, the anisotropy is almost eliminated in the output layer of both models, and is mitigated in the middle layers to different levels. This empirically validates the theoretical promise of uniformity brought by contrastive learning (Wang and Isola, 2020; Gao et al., 2021) in the context of sentence representation learning (Figure 2).

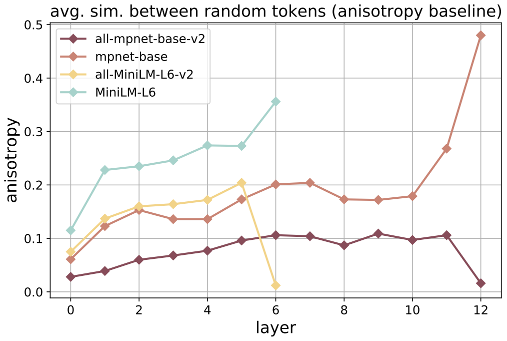

Figure 2: Anisotropy baseline of models

Complementing the enhanced isotropy, the average L2 norm of the randomly sampled token representations is also measured, showing a similar

[2] https://huggingface.co/sentence-transformers/all-mpnet-base-v2

[3] https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

[4] https://huggingface.co/nreimers/MiniLM-L6-H384-uncased. Notably, we use a 6-layer version.

[5] https://huggingface.co/datasets/stsb_multi_mt

drastic shift in mostly the output layer of both models. Geometrically, the embeddings of tokens are pushed toward the origin in the output layer of a model, compressing the dense regions in the semantic space toward the origin, making the embedding space more defined with concrete examples of words (see also Figure 1), instead of leaving many poorly-defined areas (Li et al., 2020). This property potentially contributes to models' performance gains on sentence embedding tasks.

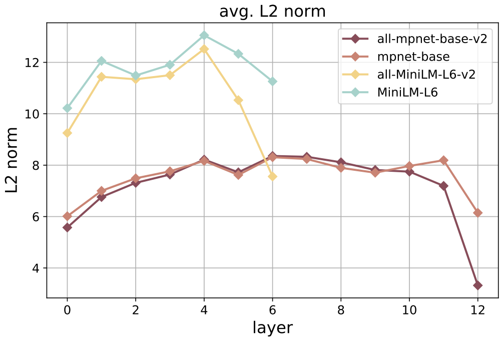

Figure 3: Avg. L2 norm of embeddings

Figure 4 and Figure 5 present respectively the self similarity and intra-sentence similarity of models adjusted (subtracted) by their anisotropy baselines (Unadjusted measures in Appendix C).

As for the adjusted self similarity, we can see that the fine-tuned models generally show higher self similarities across contexts (meaning tokens are less contextualized after fine-tuning) in all layers, except for the output layer of the fine-tuned mpnet. However, in general there does not exist a large difference on this metric (See why in Section 3).

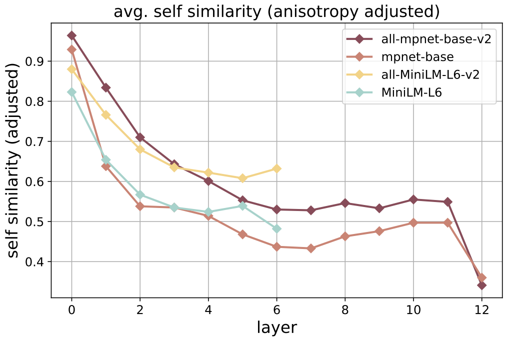

Figure 4: Adjusted self similarity of tokens: each self similarity is adjusted by the anisotropy of the corresponding model

We observe that intra-sentence similarity dramatically goes up in the output layer after contrastive fine-tuning. In the output layer of fine-tuned mpnet, the intra-sentence similarity reaches 0.834 (adjusted), meaning that tokens are 83.4% similar to one another if they appear in a same sentence. Since this pattern does not exist in the vanilla pre-trained models, the pattern is a unique behavior that accompanies the performance gain brought by contrastive learning. We argue that given contrastive examples and the goal of distinguishing between similar and non-similar in each batch, the model learns to provide more intense cross-attention among elements inside an input, and thus could better assign each example (sentence/document) to a unique position in the semantic space. With mean-pooling and positive pairs, the model learns to decide important tokens in a document d_i, in order to align with its paired document d_i^+, and other secondary tokens are likely to **imitate** the embeddings of these important tokens because they need to provide an average embedding together to match with their counterpart (In Appendix G we conduct an ablation study with other pooling methods). Further, with limited space in the now compressed space, inputs have now learned to converge to one another to squeeze to a point while keeping its semantic relationship to other examples. Therefore, we reason that, the unique behavior of this "trained intra-sentence similarity" is highly relevant to the models' enhanced performance on sentence-level semantic tasks.

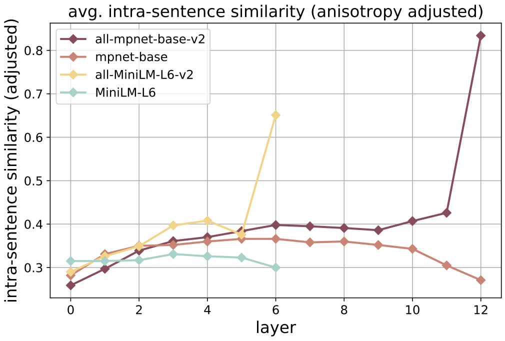

Figure 5: Adjusted intra-sentence similarity of tokens: each intra-sentence similarity is adjusted by the anisotropy of the corresponding model

Complementing the global properties found above, we present in Table 1 the dimension-level inspection on the measures. The analysis is conducted on self similarity. In line with previous work (Timkey and van Schijndel, 2021), there exists a significantly unequal contribution among dimen-

Model	Top 1	Top 2	Top 3
mpnet$_{\text{vanilla}}$	.548	.723	.741
mpnet$_{\text{fine-tuned}}$	.005	.010	.014
minilm$_{\text{vanilla}}$	.081	.129	.163
minilm$_{\text{fine-tuned}}$	.008	.014	.020
	10%	**20 %**	**50%**
mpnet$_{\text{vanilla}}$	1	1	1
mpnet$_{\text{fine-tuned}}$	28	64	209
minilm$_{\text{vanilla}}$	2	5	31
minilm$_{\text{fine-tuned}}$	19	40	121

Table 1: Dimension-wise inspection on vanilla and contrastive learning-based fine-tuned sentence representation models (last/output layer only). The upper part of the table presents the contribution percentage of the top 1 to 3 dominating dimensions. The lower part provides the number of top dimensions needed to account for {10,20,50}% of similarity metric computation.

sions. This inequality is most pronounced in the vanilla mpnet, with the top 1 dimension (out of the total 768) contributing to almost 55% of the similarity computation. After contrastive fine-tuning, this phenomena is largely removed, with dominating dimensions greatly "flattened" (Gao et al., 2021). For the fine-tuned mpnet, it now requires 209 (out of 768, 27.2%) dimensions to contribute to 50% of the metric computation, and for fine-tuned minilm, this number is 121 (out of 384, 31.5%).

In Appendix F, we present the informativity analysis by removing top-k dominating dimensions, we see a reallocation of information after contrastive fine-tuning and a misalignment between dominance toward similarity computation and informativity.

3 Connecting to Frequency Bias

The imbalance of word frequency has long been identified to be relevant to the anisotropy of trained embeddings (Gao et al., 2019). This has been also empirically observed in pre-trained Transformers like BERT (Li et al., 2020). Li et al. (2020) draw connection between frequency bias and the unideal performance of pre-trained language models on STS tasks, through deriving individual words as connections of contexts, concluding that rare words fail to play the role of connecting context embeddings. Rajaee and Pilehvar (2021) show that when fine-tuning pre-trained language models under the setting of Siamese architecture on STS-b datasets, the frequency bias is largely removed, with less significant frequency-based distribution of embeddings. However, it is also pointed out that these trained models are still highly anisotropic, which as we showed in Section 2.5, does not hold in the context of contrastive training, which, with sufficient data, has theoretical promise toward uniformity (Wang and Isola, 2020; Gao et al., 2021).

Therefore, it is of interest to see the corresponding behaviors of frequency bias shifts in the context of contrastive learning, and more importantly, how this correlates with our surprising finding on intra-sentence similarity.

3.1 How Self Similarities Change for Frequent Words?

Since word frequency has produced many problematic biases for pre-trained Transformer models, we would like to know whether contrastive learning eases these patterns. Thus, how the self-similarity measurement manifests for frequent words after the models are fine-tuned with the contrastive objective? Are they more/less contextualized now?

Validity of Measuring Self-Similarity Change We first define Self-Similarity Change and prove that this measurement is not prone to stochasticity in the training process.

The top 400 frequent tokens are first extracted from the constructed STS-b subset. Then, we measure the avg. self-similarity before and after fine-tuning for each word, adjusted for their anisotropy baseline. Formally, we define Self-Similarity Change (SSC) of a token as:

$$ssc = (ss_f - ani_f) - (ss_v - ani_v), \qquad (7)$$

where ss_f, ss_v, ani_f and ani_v stand for self-similarity and anisotropy baseline of fine-tuned and vanilla models respectively.

To validate that this measurement is not a product of stochasticity occurs in training but a common phenomenon that comes with contrastive learning, we compute the Self-Similarity Change for every token using both *mpnet* (vanilla & fine-tuned) and *MiniLM* (vanilla & fine-tuned). If the statistics produced by both models show high correlation, then there exists a pattern that would affect how self-similarity changes for different tokens during contrastive fine-tuning. Otherwise, the changes are a product of randomness.

We iterate $n = 1$ to 400 to compute the Pearson correlation of SSCs of the top n tokens produced by both *mpnet* and *MiniLM* and find the position where these statistics correlate the most, which is:

$\arg\max_n(corr(ssc_{mpnet}[:n], ssc_{MiniLM}[:n]))$. Throughout the iteration, the top 204 frequent tokens give the highest Pearson correlation, which reaches a surprisingly high number of 0.857, validating the universal pattern for similarity shifts of frequent words. After inspection, we find that these are tokens that appear more than 9 times in the 1359 sentences. Notably, even the full set of 400 tokens gives a correlation of over 0.8, again proving the robustness of this pattern for frequent words (Refer to Appendix H for the full statistics of the validation).

3.2 Reaching to the connection

Table 2 provides a glimpse of the top 10 tokens (among the top 400 frequent tokens) that are now most more contextualized (with top negative self-similarity changes) and most less contextualized (with top positive self-similarity changes).

	mpnet		minilm	
	SS (↓)	SS (↑)	SS (↓)	SS (↑)
0	has	onion	[SEP]	hands
1	is	piano	.	fire
2	,	unfortunately	;	run
3	’	cow	?	house
4	are	chair	)	japan
5	that	potato	the	hat
6	been	read	an	ukraine
7	while	dow	-	jumping
8	was	guitar	/	coffee
9	with	drums	a	points

Table 2: Top Self-Similarity Changes

After contrastive fine-tuning, tokens that contribute more to the semantics (tokens that have POS like nouns and adjectives) are now more reflective of their real-world limited connotations - tokens like "onion" and "piano" are not supposed to be that different in different contexts as they are in pre-trained models. We formalize this as **"Spurious Contextualization"**, and establish that **contrastive learning actually mitigates this phenomena for semantically meaningful tokens**. We speculate that these tokens are typically the ones that provide aligning signals in positive pairs and contrastive signals in negative pairs.

By contrast, however, the spurious contextualization of stopwords is even augmented after contrastive learning. "Has" is just supposed to be "has" - as our commonsense might argue - instead of having n meanings in n sentences. We speculate that, **stopwords fall back to be the "entourage" of a document after contrastive learning**, as they are likely the ones that do not reverse the semantics and thus do not provide contrastive signals in the training. Connecting this to our finding on high intra-sentence similarity, we observe that given a sentence/docuemnt-level input, certain semantic tokens drive the embeddings of all tokens to converge to a position, while functional tokens follow wherever they travel in the semantic space.

4 Ablation Analysis

In this section, we provide a derivation to interpret the role of temperature in CL, inspiring the searching method of its optimal range. We also show that contrastive frameworks are less sensitive to batch size at optimal temperature for SRL, unlike in visual representation learning.

4.1 Rethinking Temperature

Given a contrastive learning objective: $\ell_i = -\log \frac{e^{sim(e_i,e_i^+)/\tau}}{e^{sim(e_i,e_i^+)/\tau}+\sum_{j=1}^{N}\mathbb{1}_{\{j\neq i\}}e^{sim(e_i,e_j^+)/\tau}}$, we first look at its denominator, where the goal is to minimize the similarity between the anchor e_i and negative pairs e_j when $j \neq i$:

$$e^{sim(e_i,e_j^+)/\tau} \in (\frac{1}{e}^{\frac{1}{\tau}}, e^{\frac{1}{\tau}}) \tag{8}$$

Let x be $e^{sim(e_i,e_i^+)}$ we get:

$$e^{sim(e_i,e_j^+)/\tau} = x^{1/\tau}, x \in (\frac{1}{e}, e) \tag{9}$$

If $\tau << 1$, as long as $x < 1$, $x^{1/\tau}$ shrinks exponentially. While when $x > 1$, $x^{1/\tau}$ explodes exponentially. Therefore, $x = 1$, or $sim(e_i, e_j^+) = 0$ when $i \neq j$ is an important threshold when negative pairs are to decide whether or not to further push away, and this "thrust", is exactly what temperature provides: In-batch negatives are not motivated to be too dissimilar under a lower temperature, since once the similarity reaches below 0, the exponent $1/\tau$ is already doing the job of making them exponentially vanishing in the denominator.

We analyze the upper bound and lower bound of $sim(e_i, e_j^+)$ under 0, giving us $sim(e_i, e_j^+) = 0$ and $sim(e_i, e_j^+) = -1$ for every $sim(e_i, e_j^+)$ in batch when $i \neq j$. For both cases we pair them with $sim(e_i, e_i^+) \rightarrow 1^-$ since positive pairs are drawn closer regardless. Therefore,

$$\begin{aligned}\ell_{upperbound}(\tau) &= -\log \frac{e^{sim(e_i,e_i^+)/\tau}}{e^{sim(e_i,e_i^+)/\tau} + \sum^{n-1} e^{0/\tau}} \\ &= -\log \frac{e^{sim(e_i,e_i^+)/\tau}}{e^{sim(e_i,e_i^+)/\tau} + (n-1)},\end{aligned} \tag{10}$$

while given $sim(e_i, e_i^+) \to 1^-$,

$$\begin{aligned}\ell_{lowerbound}(\tau) &= -\log \frac{e^{sim(e_i,e_i^+)/\tau}}{e^{sim(e_i,e_i^+)/\tau} + \sum^{n-1} e^{-1/\tau}} \\ &= -\log \frac{e^{(sim(e_i,e_i^+)+1)/\tau}}{e^{(sim(e_i,e_i^+)+1)/\tau} + (n-1)} \\ &\approx -\log \frac{e^{2*sim(e_i,e_i^+)/\tau}}{e^{2*sim(e_i,e_i^+)/\tau} + (n-1)}\end{aligned} \tag{11}$$

Therefore, $\ell_{lowerbound}(2\tau) \approx \ell_{upperbound}(\tau)$.

We find that temperature affects making embeddings isotropic: to push in-batch negatives to the lower bound, the temperature needs to be twice as large than to push them to the upper bound. For example, if when temperature $= 0.05$, two sentences are pushed in training to have -1 cosine similarity, now given temperature $= 0.025$, the gradient is only around enough to push them to have 0 cosine similarity with each other.

The findings suggest that searching for the optimal value of this hyperparameter using a base of 10, as empirically shown in previous research (Gao et al., 2021), may not be the most efficient approach. Instead, we argue that a base of 2 would be more appropriate, and even to conduct finer-grained searching when a range of upper bound temperature that is twice the lower bound temperature is found to provide adequate performance.

Our analysis serves as a complementation to Wang and Liu (2021), who show that a lower temperature tends to punish hard-negative examples more (especially at the similarity range of $(0.5, 1)$), while a higher temperature tends to give all negative examples gradients to a same magnitude. This provides more theoretical justification to our approximation, since at the similarity range of $(-1, 0)$, all negative examples have gradients to the same magnitude (Wang and Liu, 2021) regardless. We suggest that this range plays a main role in making the entire semantic space isotropic.

4.2 Experiment Setup

We use a vanilla mpnet-base (Song et al., 2020) as the base model, and train it on a concatenation of SNLI (Bowman et al., 2015) and MNLI datasets (Williams et al., 2018). In accordance with our analysis, for the temperature τ subspace we deviate from the commonly adopted exponential selection with a base of 10 (e.g., Gao et al. (2021)), but we analyze around the best value found empirically, with a base of 2, i.e., $\{0.025, 0.05, 0.1\}$. We provide the same analysis on $\{0.001, 0.01, 0.05, 0.1, 1\}$ in Appendix D for comparison. To better illustrate the effect of temperature, we only use entailment pairs as positive examples, under supervised training setting. We do not consider using contradiction as hard negatives to distract our analysis, nor unsupervised settings using data augmentation methods such as standard dropout. We use all instances of entailment pairs as training set, yielding a training set of 314k. We truncate all inputs with a maximum sequence length of 64 tokens. All models are trained using a single NVIDIA A100 GPU. We train the models with different temperatures for a single epoch with a batch size of 64, yielding 4912 steps each, with 10% as warm-up. We save the models every 200 steps and use them to encode the subset of STS-B we have constructed.

4.3 Results

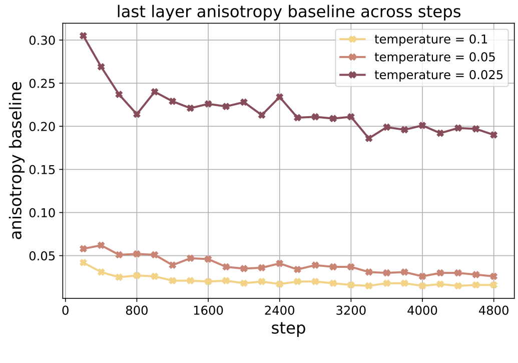

Figure 6: Anisotropy changes throughout training under different temperatures

Firstly, we present the centered property we are measuring, anisotropy. Figure 6 shows the last-layer anisotropy change throughout steps. The trend is in line with our hypothesis about temperature being a "thurst". Knowing that the vanilla model starts from encoding embeddings to be stuck in a narrow angle, temperature serves as the power to push them further through forcing negative pairs

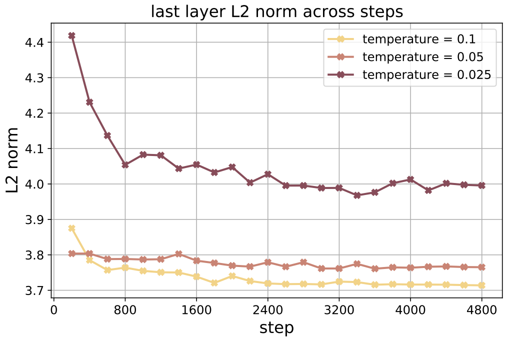

Figure 7: L2-norm under different temperatures

to be different. With a higher temperature, the cosine similarity between negative pairs has to be lower to reach a similar loss. Figure 7 further validates this through showing that higher temperatures compress the semantic space in general, pushing instances to the origin.

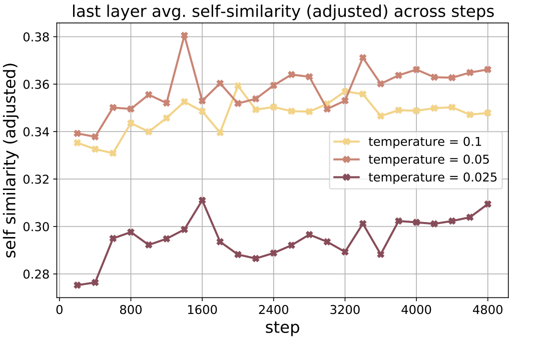

Figure 8: Self similarity under different temperature, adjusted by anisotropy baseline

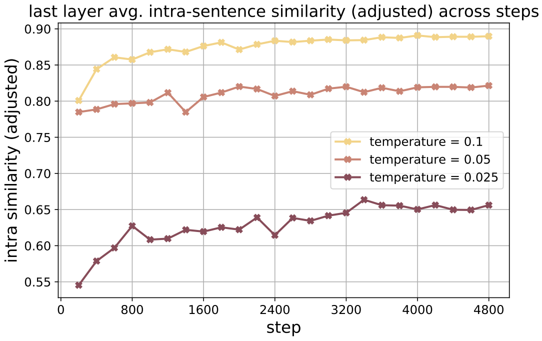

Figure 9: Intra-sentence similarity under different temperature, adjusted by anisotropy baseline

Figure 8 and Figure 9 present the adjusted self and intra-sentence similarity. Following the closer look at the contradicted pattern for frequency bias analyzed in Section 3, the behavior here becomes self-explainable. We could see that under the temperature of 0.1, the self similarity stays at a lower level compared to 0.05 in the last steps. This matches with the opposite result in intra-sentence similarity. According to our analysis in Section 3, it is the less meaningful tokens that drag down the self-similarity, and because they learn to follow the semantically meaningful tokens wherever their embeddings go in the semantic space, the corresponding intra-sentence similairty would become much higher. We speculate that, while a high intra similarity explains the performance gain of models trained with contrastive loss on semantic tasks, its being too intense (as shown when $\tau = 0.1$) might also account for the performance drop, making semantic meaningful tokens too dominating compared to auxiliary/functional tokens. Therefore, it again justifies the importance of selecting **a moderate temperature** that provides enough gradients, but not over-intensifying the attention leaning toward dominating tokens.

In Appendix E, we provide the analysis on batch size, revealing that batch size plays a less significant role, if given a relatively optimal temperature. This is the opposite of what is commonly found in visual representation learning. Appendix G compares the three commonly used pooling methods, showing that the found patterns are not just artifacts of a certain pooling method (mean pooling), but consistent across pooling methods.

5 Conclusion

In this paper, we demystify the successes of using contrastive objectives for sentence representation learning through the lens of isotropy and learning dynamics. We showed the theoretical promise of uniformity brought by contrastive learning through measuring anisotropy, complemented by showing the flattened domination of top dimensions. We then uncovered a very interesting yet under-covered pattern: contrastive learning learns to converge tokens in a same sentence, bringing extremely high intra-sentence similarity. We then explained this pattern by connecting it to frequency bias, and showed that semantically functional tokens fall back to be the by products of semantically meaningful tokens in a sentence, following wherever they travel in the semantic space. Lastly, we ablate all findings through temperature, batch size and pooling method, providing a closer look at these patterns through different angles.

6 Limitations

This paper only considers analyzing contrastive learning in the fine-tuning stage, but we note that with isotropy being a desiderata for pre-trained language models (Ethayarajh, 2019), recent works have considered incorporating contrastive objectives in the pre-training stage (Izacard et al., 2022; Su et al., 2022). We leave analysis on this line of research for future work.

We further note that the analysis in this work focuses on theoretical properties occurred during contrastive SRL (e.g., high intra-sentence similarity), thus only focuses on semantic textual similarity (STS) data as a proof of concept. However, with the growing attention on contrastive learning, we argue that the typical STS-B is perhaps no longer sufficient for revealing the full ability of models trained with newer contrastive SRL frameworks. We call for a standard practice that the performance of contrastive SRL should be assessed on both semantic textual similarity and information retrieval tasks (e.g., Thakur et al. (2021)). We leave analysis on information retrieval tasks leveraging our analysis pipeline for future studies. For example, how high intra-sentence similarity is related to the learned attention towards tokens that enable document retrieval with better performance.

References

Samuel 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.

Xingyu Cai, Jiaji Huang, Yuchen Bian, and Kenneth Church. 2020. Isotropy in the contextual embedding space: Clusters and manifolds. 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.

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.

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.

Jun Gao, Di He, Xu Tan, Tao Qin, Liwei Wang, and Tieyan Liu. 2019. Representation degeneration problem in training natural language generation models. In *International Conference on Learning Representations*.

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.

John Giorgi, Osvald Nitski, Bo Wang, and Gary Bader. 2021. Declutr: Deep contrastive learning for unsupervised textual representations. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 879–895.

Yaru Hao, Li Dong, Furu Wei, and Ke Xu. 2020. Investigating learning dynamics of bert fine-tuning. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 87–92.

Junjie Huang, Duyu Tang, Wanjun Zhong, Shuai Lu, Linjun Shou, Ming Gong, Daxin Jiang, and Nan Duan. 2021. Whiteningbert: An easy unsupervised sentence embedding approach. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 238–244.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022. Unsupervised dense information retrieval with contrastive learning. *Transactions on Machine Learning Research*.

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.

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*.

Amil Merchant, Elahe Rahimtoroghi, Ellie Pavlick, and Ian Tenney. 2020. What happens to bert embeddings during fine-tuning? In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 33–44.

Jiaqi Mu and Pramod Viswanath. 2018. All-but-the-top: Simple and effective postprocessing for word representations. In *International Conference on Learning Representations*.

Jeffrey Pennington, Richard Socher, and Christopher D 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.

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.

Sara Rajaee and Mohammad Taher Pilehvar. 2021. How does fine-tuning affect the geometry of embedding space: A case study on isotropy. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3042–3049, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. *Advances in Neural Information Processing Systems*, 33:16857–16867.

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

Yixuan Su, Fangyu Liu, Zaiqiao Meng, Tian Lan, Lei Shu, Ehsan Shareghi, and Nigel Collier. 2022. TaCL: Improving BERT pre-training with token-aware contrastive learning. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 2497–2507, Seattle, United States. Association for Computational Linguistics.

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

William Timkey and Marten van Schijndel. 2021. All bark and no bite: Rogue dimensions in transformer language models obscure representational quality. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4527–4546.

Feng Wang and Huaping Liu. 2021. Understanding the behaviour of contrastive loss. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2495–2504.

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.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in Neural Information Processing Systems*, 33:5776–5788.

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.

Xing Wu, Chaochen Gao, Zijia Lin, Jizhong Han, Zhongyuan Wang, and Songlin Hu. 2022. Infocse: Information-aggregated contrastive learning of sentence embeddings. *arXiv preprint arXiv:2210.06432*.

Yuanmeng Yan, Rumei Li, Sirui Wang, Fuzheng Zhang, Wei Wu, and Weiran Xu. 2021. Consert: A contrastive framework for self-supervised sentence representation transfer. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5065–5075.

Yuhao Zhang, Hongji Zhu, Yongliang Wang, Nan Xu, Xiaobo Li, and Binqiang Zhao. 2022. A contrastive framework for learning sentence representations from pairwise and triple-wise perspective in angular space. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4892–4903.

A Top Self Similarity Change (SSC): Token Examples

Table 2 presents top 10 positive and negative self similarity change of frequent tokens, before and after contrastive fine-tuning.

Although function tokens are found to be highly contextualized in pre-trained language models (Ethayarajh, 2019), this phenomenon is even intensified after contrastive fine-tuning. While for semantic tokens, the spurious contextualization is alleviated to a great extent.

B Expanded semantic space (Eased Anisotropy)

We provide a visualization of embedding geometry change in Figure 1.

We first use the vanilla mpnet to encode the STS-B subset we have constructed. During fine-tuning, we save the models every 200 steps and use them to encode the subset, We find that with optimal hyperparameters, the representations go through less change after 200 steps. We perform UMAP dimensionality reduction on embeddings provided by models up to 1000 step to preserve better global structure, and visualize only vanilla and 200-step embeddings.

C Unadjusted measures of Section 2.5

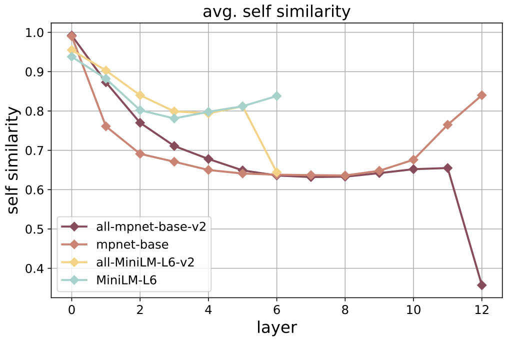

Figure 10: Unadjusted self similarity of tokens

Figure 10 and Figure 11 display respectively the unadjusted avg. self similarity and intra-sentence similarity. These values as we elucidated in previous sections, however, are likely to be artifact of anisotropy, and therefore are supposed to be adjusted by the anisotropy baseline of each model, based on the computation on randomly sampled token pairs.

As shown in main sections, to offset the effect of each model's intrinsic non-uniformity, we adjust them by the degree of anisotropy of each model, based on pair-wise average similarity among 1000 token representations that we randomly sample from each of the 1000 sentences (to avoid the sampling to bias toward long sentences).

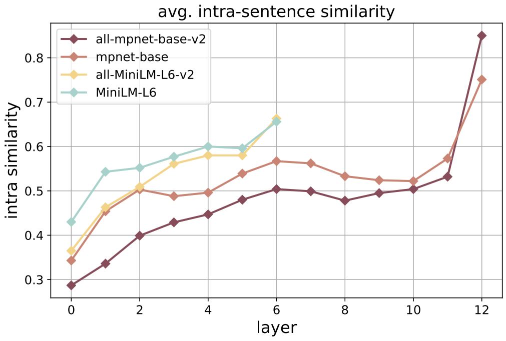

Figure 11: Unadjusted intra-sentence similarity of tokens

D Temperature Search: why searching to the order of magnitude by 10 is not optimal?

We have also run the search range of temperature in previous research, which is carried out to the order of magnitude by 10. We compare the metrics on the models run with these temperatures with the vanilla mpnet model's performance.

It is shown that, not all values of temperature push the metrics from the vanilla baseline toward a same direction. Therefore, there exists a relatively optimal range to search, which is empirically implemented in a few works (Yan et al., 2021; Zhang et al., 2022), but few seems to have discussed why the range should not be that large, while we show this through the math analysis in Section 4 and their contradicted performance on our studied metrics here.

Specifically, for anisotropy baseline, temperature being too low even augments the vanilla model's unideal behavior, and the same applies for L2-norm, by that temperature being too low actually pushes the embeddings even further from the origin.

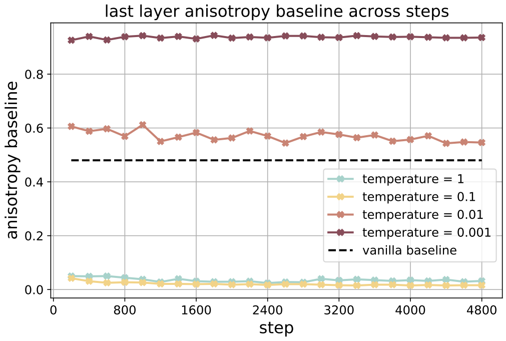

Figure 12: Anisotropy changes throughout training under different temperatures

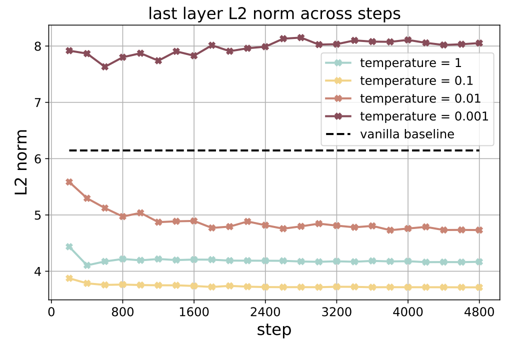

Figure 13: L2-norm under different temperatures

For the adjusted self similarity and intra-sentence similarity, the metrics for low temperature are largely offset by anisotropy, meaning that for these temperature (especially $\tau = 0.001$), tokens are not more similar to itself in different contexts, nor to other tokens they share contexts with, compared to just with a random token in whatever context.

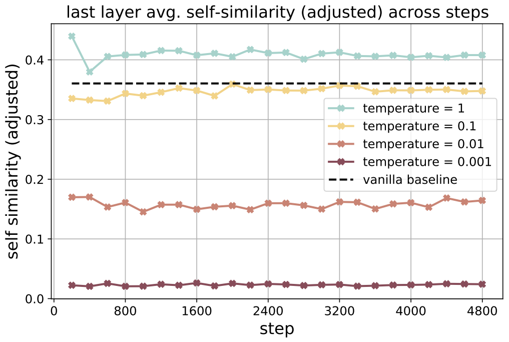

Figure 14: Self similarity under different temperatures, adjusted by anisotropy baseline

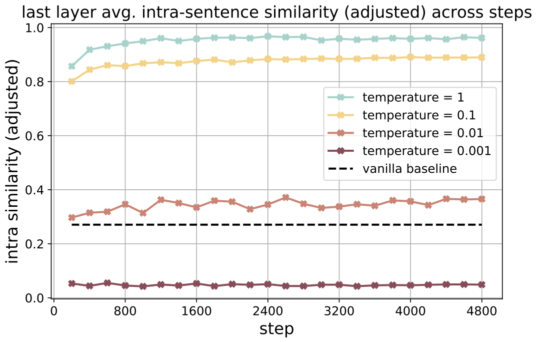

Figure 15: Intra-sentence similarity under different temperatures, adjusted by anisotropy baseline

Gao et al. (2019) and Gao et al. (2021) take a singular spectrum perspective in understanding regularization to anisotropy. Gao et al. (2019) propose a regularization term to the original log-likelihood loss in training machine translation model to mitigate the representation degeneration problem (or anisotropy). The regularization is proportional to $Sum(WW^T)$, where W is the stack of normalized word embeddings. If all elements are positive, then minimizing $Sum(WW^T)$ is equivalent to minimizing the upper bound for the largest top eigenvalue of $Sum(WW^T)$. Therefore, this regularization term shows theoretical promise to flatten the singular spectrum and make the representation more uniformly distributed around the origin. Gao et al. (2021) extend this analysis to show the same theoretical promise brought by the uniformity loss proposed by Wang and Isola (2020), by deriving that uniformity loss is in fact greater or equal to $\frac{1}{\tau m^2}\sum_{i=1}^{m}\sum_{j=1}^{m} h_i^T h_j$, which is also equivalent to flattening the spectrum of the similarity matrix. Our results show that despite the intuition reached by singular spectrum perspective, the assumption could probably only hold on a relatively optimal temperature. Thus, the effect of temperature should be considered using this perspective, which is beyond the scope of this paper.

E Batch size

Batch size on the other hand, does not produce impact as significant as temperature. We have run three models with the optimal $\tau = 0.05$ paired with a batch size range of $\{16, 64, 256\}$.

The metrics yielded by different batch sizes all stay in small range at the end of the epoch, albeit showing different rates and stability of convergence.

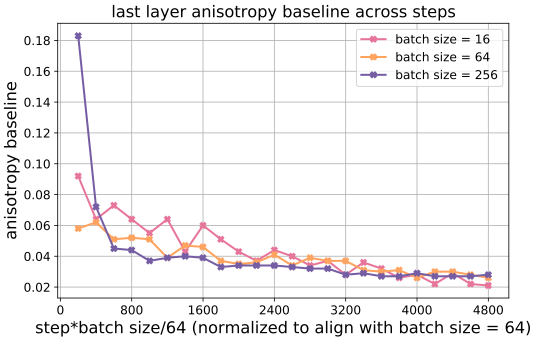

Figure 16: Anisotropy changes throughout training under different batch sizes

Model	k = 1	k = 2	k = 3	k = 5	k = 10	k = 20	k = 50	k = 100	k = 300	k = 700
$\text{mpnet}_{\text{vanilla}}$	.386	.338	.210	.169	.168	.182	.201	.195	.175	.040
$\text{mpnet}_{\text{fine-tuned}}$	**.999**	**.998**	**.996**	**.994**	**.990**	**.983**	**.960**	**.922**	**.783**	**.229**
$\text{minilm}_{\text{vanilla}}$	.993	**.980**	**.970**	**.947**	**.886**	.796	.559	.543	**.375**	/
$\text{minilm}_{\text{fine-tuned}}$	**.998**	.846	.836	.830	.817	**.805**	**.768**	**.690**	.285	/

Table 3: r^2 between the similarity matrices of sampled token embeddings, before and after removing the same top-k rogue dimensions from every token embedding.

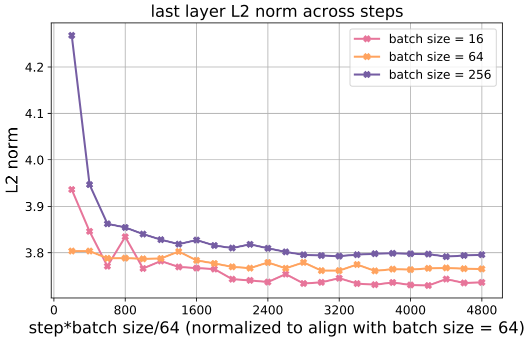

Figure 17: L2-norm under different batch sizes

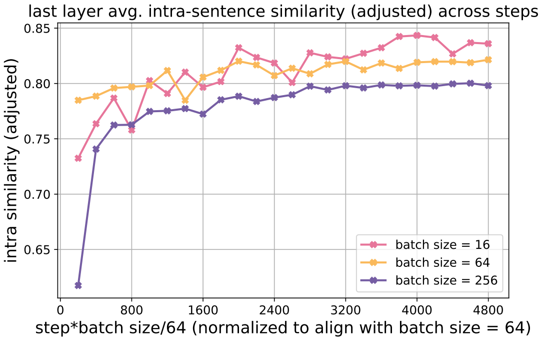

Figure 19: Intra-sentence similarity under different batch sizes, adjusted by anisotropy baseline

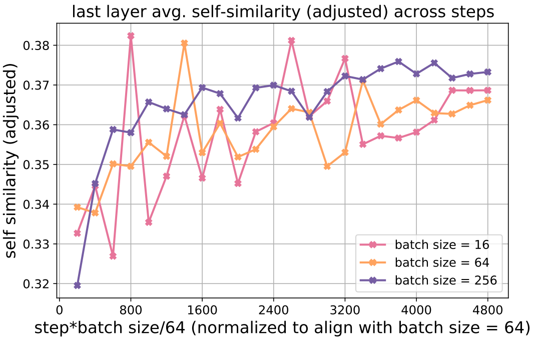

Figure 18: Self similarity under different batch sizes, adjusted by anisotropy baseline

F Informativity

In this section we present the informativity analysis outlined in Section 2. Specifically, after we identify how dominant are the top rogue dimensions, to what degree is semantics affected with these rogue dimensions removed? Do these dimensions only have large mean but do not contribute to large variance? We sample 1k token embeddings to compute their pair-wise similarity. After removing top-k dimensions from every embedding, we compute the similarity matrix again, and compute the Pearson Correlation r between flattened lower triangles of the matrices of the two excluding their diagonals. We then report the r^2 which represents the proportion of variance in the original similarity matrix explained by the post-processed matrix.

At a high level, Table 3 shows that dominance $\neq$ informativity. Specifically, MiniLM presents a misalignment between dominance toward similarity computation and the actual information stored in these dimensions. For instance, removing the top 1 dominant dimension of $\text{minilm}_{\text{finetuned}}$ seems to not affect the embeddings' relative similarity to one another at all, preserving an r^2 of .998. Also, recall from Section 2 that contributions of dimensions from $\text{minilm}_{\text{vanilla}}$ to similarity computation are relatively flatter than $\text{mpnet}_{\text{vanilla}}$, the results show that along with the even more flattened contributions after fine-tuning, the informativity seems to have been reallocated. For instance, from removing $k = 100$ to $k = 300$, the explainable variance goes down from .690 to .285, meaning this range of dimensions store a lot more information compared with the vanilla version. In general, that $\text{minilm}_{\text{vanilla}}$ and $\text{minilm}_{\text{fine-tuned}}$ take turn to yield higher r^2 with top-k removed demonstrates that there is generally no strong correlation between dominance and informativity, but it is rather random - especially when the dominance is already quite evenly distributed in the vanilla model.

G Pooling Method

In line with previous analysis, this section presents the measurement on different pooling methods. We follow the same setting in Section 4 to also investigate whether the patterns found in Section 2 are only attributable to mean pooling. We compare mean pooling with [cls] pooling and max pooling. Albeit the different performance on the metrics, contrastive learning in general presents consistent behaviors across pooling methods, such as eased anisotropy and enhanced intra-sentence similarity For anisotropy, we observe that [cls] pooling shows a slow convergence on producing isotropy. At the end of the epoch, it is still on a decreasing trend. By contrast, mean pooling and max pooling demonstrate a faster convergence, with mean pooling being most promising on isotropy. Their performance on L2-norm is also well-aligned, again showing strong correlation between isotropy and L2-norm in the training process utilizing contrastive loss. And this correlation seems agnostic to pooling methods. The following analysis focuses on their differences:

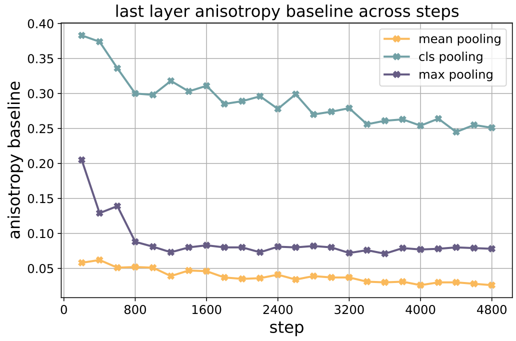

Figure 20: Anisotropy changes throughout training under different pooling methods

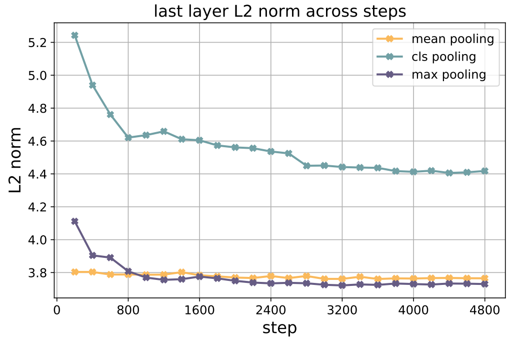

Figure 21: L2-norm under different pooling methods

For self similarity, [cls] pooling and mean pooling show a similar performance, which max pooling deviates from.

Max pooling presents an "unacceptably" high intra-sentence similarity. Although intra-sentence similarity is a potentially ideal property uniquely brought by contrastive learning, this metric could not be over-intensified, as also shown in Section 4, Appendix D, and Appendix E. There exists an ideal range for intra-sentence similarity, compatible to a model's performance on other metrics.

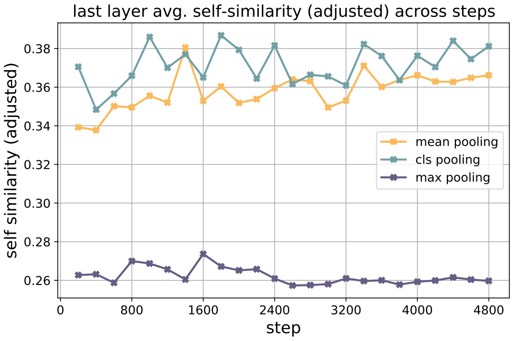

Figure 22: Self similarity under different pooling methods, adjusted by anisotropy baseline

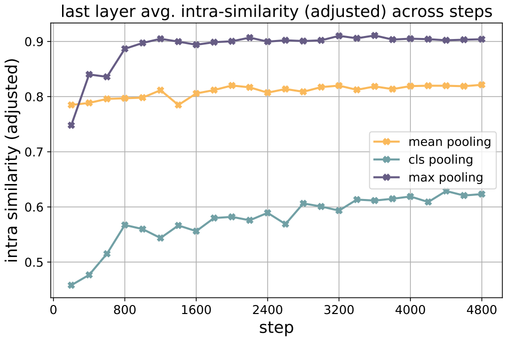

Figure 23: Intra-sentence similarity under different pooling methods, adjusted by anisotropy baseline

H Self Similarity Change and Correlation across Models

In Figure 24 we plot the Self Similarity Change (SSC) across models (mpnet and MiniLM), for the top 400 frequent tokens of the SST-b subset we construct.

The Pearson correlation between the two accumulated lists of the first $[: n]$ tokens is also plotted. The perfect correlation at the beginning is ignored because the most frequent words at the top are the [pad], [cls] and [sep] tokens. Excluding these, the correlation reaches the peak at 204 as mentioned in the main section, before which the correlation has been slowly stabilized with more tokens considered, while starting to drop after. This shows that the pattern mostly holds for tokens that are above

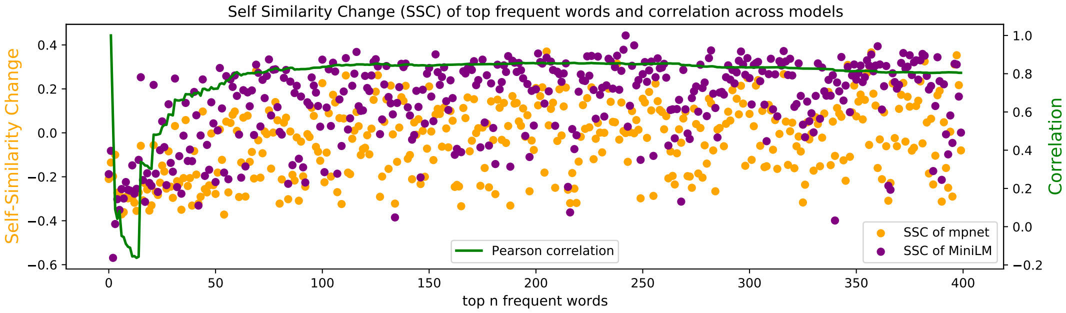

Figure 24: Self Similarity Change

certain frequency, which again provides empirical ground for our analysis on drawing connection of self and intra-sentence similarity to frequency bias.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
6

☒ A2. Did you discuss any potential risks of your work?
We haven't identified any risks associated with our work because it focuses on studying why a specific training framework works.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
1

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

2,3,4

☑ B1. Did you cite the creators of artifacts you used?
2,4

☒ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
The datasets we used are de facto for the task (sentence embedding learning) we're studying, i.e., STS-B, MNLI and SNLI with nothing specific to discuss.

☒ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
The datasets are standard for the task we're studying (sentence embedding learning) without specific different/inconsistent intent of usage.

□ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Not applicable. Left blank.

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
2,4

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Left blank.

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

C ☑ Did you run computational experiments?

2,3,4

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
2,4

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
3,4

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
2,3,4

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
2

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

□ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

□ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

□ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

□ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

□ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.