Geometry First: Shaping Sentence Embeddings with Controlled Singular Value Regularization

Anonymous ACL submission

Abstract

Contrastive learning is widely used for sentence embedding training, yet it provides limited control over the global geometry of the learned embedding space. Sentence embeddings often exhibit spectral concentration, where a small set of directions dominates the representation while many others remain weakly active. We present singular value regularization (SVR), a batch-level framework for shaping sentence embedding spaces by directly controlling the singular value spectrum during training. SVR enables explicit control over both the spectral floor and the number of regularized directions. Experiments show consistent improvements across encoder backbones, pooling strategies, English evaluation tasks, and multilingual settings. Our analysis further shows that the best regularization range is not full-dimensional flattening, but a controlled active-dimensional region aligned with the naturally active subspace of pretrained representations. Moreover, the strongest results are obtained with a large SVR weight, suggesting that spectral shaping is not a minor auxiliary constraint but a primary component of sentence embedding learning. These findings support a geometry-first view: effective contrastive learning benefits from first establishing a well-conditioned embedding space with sufficient active dimensions.

1 Introduction

Sentence embedding models map text into continuous representation spaces where semantic similarity and relatedness can be measured and compared (Reimers and Gurevych, 2019; Wang et al., 2024). In most widely used formulations, each input is represented as a single fixed-dimensional vector, enabling efficient similarity search, retrieval, clustering, classification, and reranking. A common training strategy is contrastive learning (van den Oord et al., 2019; Gao et al., 2021), which encourages semantically related texts to have nearby representations while pushing unrelated texts farther apart. While effective, contrastive learning mainly shapes local neighborhood relations among examples and provides limited direct control over the global geometry of the embedding space.

This limitation is important because prior work has shown that neural embedding spaces are often anisotropic (Ethayarajh, 2019), with representations tending to concentrate in a relatively narrow cone rather than being distributed uniformly across directions. From a spectral perspective, this means that variation is concentrated in a limited set of dominant directions, leaving much of the representation space weakly utilized. This issue is particularly relevant for sentence embeddings, which are often used as general-purpose semantic representations across diverse downstream settings. When variation is overly concentrated in a small number of dominant directions, the embedding space may underuse its available capacity and become less capable of preserving fine-grained semantic distinctions.

Empirically, we observe that spectral concentration persists in both raw pretrained representations and sentence embedding models trained with standard contrastive learning, as shown in Figure 1. This suggests that existing objectives do not adequately regulate how variation is distributed across singular directions. To address this, we propose singular value regularization (SVR), a batch-level framework that directly shapes the singular value spectrum of the embedding matrix. SVR imposes an explicit spectral floor to discourage excessive concentration of variation in a few dominant singular directions and provides interpretable control through a singular value budget. Experiments across diverse sentence embedding benchmarks show consistent improvements, and our analysis further suggests that the most effective regularization range is not full spectral flattening, but controlled expansion near the naturally active dimensions of pretrained representations.

Our contributions are as follows. First, we propose **singular value regularization (SVR)**, a batch-level framework for directly controlling the spectral geometry of sentence embedding spaces during contrastive training. SVR provides explicit control over both the spectral floor and the number of regularized directions, and yields consistent gains across different backbones, pooling strategies, English benchmarks, and multilingual evaluations. Second, we provide empirical evidence for a **geometry-first view** of sentence embedding learning. While contrastive loss defines which examples should be close or far apart, effective alignment depends on the geometry of the space in which these relations are expressed. Rather than treating geometry as a post-hoc diagnostic or a secondary auxiliary constraint, we argue that the **spectral structure of the embedding space should be treated as a primary object of control** during sentence embedding training. Our results point to a broader view of sentence embedding learning: effective representations require not only semantic supervision, but also **explicit control over the active geometry of the embedding space**. Our analysis further shows that the best controlled dimensionality is not the full embedding dimension, but lies near the naturally active range of pretrained representations.

2 Related Work

Sentence embeddings and contrastive learning. Modern sentence embedding research is largely built on Transformer encoders trained with contrastive objectives (van den Oord et al., 2019). In supervised settings, Sentence-BERT established the siamese fine-tuning paradigm for semantic similarity and retrieval (Reimers and Gurevych, 2019). In self-supervised settings, SimCSE showed that strong sentence representations can be learned from dropout-based contrastive learning alone (Gao et al., 2021). Subsequent work further improved sentence embeddings through instruction- or task-aware supervision (Su et al., 2023) and large-scale weakly supervised text-pair training (Wang et al., 2024). While these approaches differ in data source and training setup, they primarily optimize semantic similarity relationships between examples rather than directly constraining the geometric structure of the batch embedding space during training.

Geometry and anisotropy of embedding spaces. A parallel line of work studies the geometry of learned representations. Prior work has shown that contextual and sentence representations are often anisotropic, with variation concentrated in a small number of dominant directions (Ethayarajh, 2019; Li et al., 2020). This has motivated post-hoc correction methods such as removing dominant principal components or whitening embedding spaces to improve isotropy and retrieval quality (Mu et al., 2018; Su et al., 2021). These approaches diagnose or repair geometry after representations are produced. In contrast, our method regularizes the singular value spectrum of sentence embeddings directly during training, allowing spectral structure to be shaped jointly with the contrastive objective.

Regularization against representational collapse. A broader line of work addresses representational collapse by regularizing the geometry of features. DirectCLR studies dimensional collapse in contrastive learning and advocates operating directly on representation space (Jing et al., 2022). VICReg prevents collapse through variance and covariance regularization, while MCR^2 encourages feature diversity through a coding-rate objective (Bardes et al., 2022; Yu et al., 2020). SVMax (Taha et al., 2021) maximizes the mean singular value of the feature embedding matrix, encouraging spectral expansion through a global objective. However, it does not specify which directions should be regularized, how much spectral mass should be redistributed, or when the spectrum is sufficiently expanded. In contrast, SVR defines an explicit target geometry: it regularizes a top-N spectral range, derives the singular value floor from a closed-form singular value budget, and uses the leakage rate to control the allowed spectral concentration. Thus, SVR provides a targeted control mechanism for embedding geometry rather than an unconstrained pressure to increase overall singular value magnitude.

3 Preliminary

3.1 Contrastive Training Objective

Given a query q, its positive text d^+, and negatives $\{d_i^-\}_{i=1}^n$, we use the standard contrastive loss

$$L_{\text{cl}} = -\log \frac{e^{s(q,d^+)/\tau}}{e^{s(q,d^+)/\tau} + \sum_{i=1}^n e^{s(q,d_i^-)/\tau}},$$

where τ is the temperature and $s(q,d)$ is cosine similarity between sentence embeddings:

$$s(q,d) = \frac{E(q)^\top E(d)}{\|E(q)\|\,\|E(d)\|}.$$

We use in-batch negatives, treating the positives paired with other queries in the same batch as negatives.

3.2 Singular Value Decomposition

For a matrix $\mathbf{X} \in \mathbb{R}^{n\times d}$, let $k = \min(n, d)$. Its compact SVD is

$$\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top,$$

where $\mathbf{U} \in \mathbb{R}^{n\times k}$, $\mathbf{V} \in \mathbb{R}^{d\times k}$, and $\mathbf{\Sigma} \in \mathbb{R}^{k\times k}$ is diagonal with singular values

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_k \geq 0.$$

Larger singular values correspond to more dominant directions in $\mathbf{X}$, while the associated singular vectors define the corresponding orthonormal directions.

4 Method

In this section, we present our singular value regularization framework for sentence embedding training. We first introduce the regularization objective on the singular values of the batch embedding matrix, then describe how the margin is derived from an explicit singular value budget. Finally, we present a top-N variant that restricts the regularization to a selected subset of singular directions.

4.1 Singular Value Regularization

Let a training batch contain b sentence embeddings, each of dimension d, with $b > d$. Denote the embedding of the i-th sample by $\mathbf{e}_i \in \mathbb{R}^d$. We first normalize each embedding by its ℓ_2 norm:

$$\tilde{\mathbf{e}}_i = \frac{\mathbf{e}_i}{\|\mathbf{e}_i\|_2}.$$

The normalized embeddings are then stacked row-wise into a matrix

$$\mathbf{X} = \begin{bmatrix} \tilde{\mathbf{e}}_1^\top \\ \tilde{\mathbf{e}}_2^\top \\ \vdots \\ \tilde{\mathbf{e}}_b^\top \end{bmatrix} \in \mathbb{R}^{b\times d},$$

so that each row of $\mathbf{X}$ has unit norm.

We apply **compact** SVD to $\mathbf{X}$:

$$\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top,$$

where $k = \min(b, d)$, $\mathbf{U} \in \mathbb{R}^{b\times k}$, $\mathbf{V} \in \mathbb{R}^{d\times k}$, and

$$\mathbf{\Sigma} = \mathrm{diag}(\sigma_1, \ldots, \sigma_k) \in \mathbb{R}^{k\times k}.$$

The singular values are ordered as

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_k \geq 0.$$

In our setting, we assume $b > d$. Since each row of $\mathbf{X}$ is ℓ_2-normalized, the Frobenius norm of the batch matrix is fixed:

$$\|\mathbf{X}\|_F^2 = \sum_{i=1}^{b} \|\tilde{\mathbf{e}}_i\|_2^2 = b.$$

Because $b > d$, the compact SVD contains d singular values, and by the standard SVD identity,

$$\sum_{i=1}^{d} \sigma_i^2 = \|\mathbf{X}\|_F^2 = b.$$

To prevent the spectrum from becoming overly concentrated, we impose a lower margin m on the singular values and penalize those falling below it using a squared ReLU margin loss:

$$\mathcal{L}_{\mathrm{sv}} = \frac{1}{d}\sum_{i=1}^{d} \big[\max(0, m - \sigma_i)\big]^2.$$

This regularizer leaves singular values above the margin unchanged and only pushes up weaker directions. In this way, it encourages broader use of the available dimensions without explicitly flattening the entire spectrum.

4.2 Margin from a Singular Value Budget

We set the margin m from an explicit singular value budget. Let $\alpha \in [0, 1]$ be the **leakage rate**, which denotes the fraction of total squared singular value mass that may remain concentrated above the imposed floor. The remaining fraction, $1 - \alpha$, is treated as the minimum singular value budget to be distributed across all d singular directions.

Since the sum of squared singular values is b, the target distributed amount is

$$b(1 - \alpha).$$

If this amount is evenly assigned across the d singular directions, the corresponding per-direction squared magnitude is

$$\frac{b(1 - \alpha)}{d}.$$

We therefore define the singular value margin as

$$m = \sqrt{\frac{b(1 - \alpha)}{d}}.$$

For example, if 20% of the sum of squared singular values is allowed to remain concentrated in dominant directions, then $\alpha = 0.2$, and the margin becomes

$$m = \sqrt{\frac{0.8b}{d}}.$$

This parameterization makes the regularization directly interpretable. A larger α allows stronger concentration in a small number of dominant directions, while a smaller α enforces a higher minimum floor on the spectrum and encourages broader spectral usage.

4.3 Top-N Singular Value Regularization

Instead of regularizing all singular values, we restrict the regularization to the first N singular directions. Since singular values are sorted in descending order, this defines a controlled active subspace of size N. The goal is not to force all d embedding dimensions to be active, but to ensure that the leading N directions maintain sufficiently large singular values.

Let

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_d \geq 0$$

denote the singular values of the batch matrix, and let $N \leq d$. The top-N regularization loss is defined as

$$L_{\text{top-}N} = \frac{1}{N} \sum_{i=1}^{N} \max(0, m_N - \sigma_i)^2,$$

where m_N is the margin used for the top-N setting. This margin imposes a spectral floor on the controlled active subspace: singular values already above the margin are left unchanged, while underactive directions within the first N directions are pushed upward. Directions beyond N are not regularized.

Following the same singular value budget as in the full-spectrum case, the margin is computed by distributing the target squared singular value amount over the selected N directions:

$$m_N = \sqrt{\frac{b(1-\alpha)}{N}}.$$

Thus, N directly controls the intended active dimensionality of the embedding space, while α controls the spectral floor within that active range.

4.4 Final Training Objective

The proposed singular value regularization is used together with the standard contrastive training objective. Let $\mathcal{L}_{\text{cl}}$ denote the contrastive loss, and let $\mathcal{L}_{\text{reg}}$ denote either the full-spectrum regularization loss $\mathcal{L}_{\text{sv}}$ or the top-N variant $\mathcal{L}_{\text{top-}N}$. The final training objective is defined as

$$\mathcal{L} = \mathcal{L}_{\text{cl}} + \lambda \mathcal{L}_{\text{reg}},$$

where λ is a scalar hyperparameter controlling the strength of the regularization term. In this way, the model is trained not only to preserve semantic similarity under the contrastive objective, but also to maintain a less concentrated and more broadly utilized embedding space.

5 Experiments

In this section, we describe the experimental setup in detail, including the comparison baselines, the curated training datasets, the training configuration, the evaluation benchmarks, and the final experimental results. For fair comparison, unless otherwise specified, all baselines are trained under the same data mixture, optimization settings, and evaluation protocol as our method.

5.1 Baselines

We compare against three baselines corresponding to widely used pooling strategies: vanilla special-token pooling (i.e., [CLS] pooling), mean pooling, and weighted mean pooling over the last n layers, with $n = 6$ in our experiments and learnable weights.

5.2 Training Data

We construct the training corpus from 13 publicly available datasets/pair sources covering diverse domains and task types, including question answering, retrieval, paraphrase detection, summarization-style text matching, scientific text matching, and general-domain review pairs.[1] Further details of the training datasets are provided in Appendix B. The resulting training mixture contains 8,327,617 text pairs in total. For each dataset, if the number of available samples is below 1M, we retain the full dataset. Otherwise, we randomly sample 1M text pairs from that source. This sampling scheme reduces over-representation from very large datasets and yields a more balanced multi-source training

[1] We use the publicly available embedding dataset collection: dataset page.

corpus. Before training, we perform light preprocessing to improve data quality. In particular, we remove samples containing suspicious encoding artifacts and discard texts with more than three abnormal control characters.

5.3 Settings

For the main English evaluation, we train GTE-en-mlm-base [2] and ModernBERT-base[3]. For multilingual evaluation, we train EuroBERT-210m [4] on the multilingual mixture described in Section 6. All models are trained for one epoch with an effective batch size of 1024 and temperature 0.1. Additional optimization and implementation details are provided in Appendix A.

The SVR checkpoints reported in Table 1 use the same hyperparameter configuration across all backbone models and pooling strategies: leakage rate $\alpha = 0$, active dimensionality $N = 384$, and regularization weight $\lambda = 80$. We use this configuration for both CLS pooling and mean pooling on GTE-en-mlm-base and ModernBERT-base.

5.4 Evaluation

We evaluate the learned sentence embeddings on a subset of benchmarks from the Massive Text Embedding Benchmark (MTEB) (Muennighoff et al., 2022; Enevoldsen et al., 2025). Following the official MTEB evaluation pipeline, we report the benchmark-defined `main_score` for each task. Table 2 summarizes the evaluated tasks and their corresponding main metrics. Higher scores indicate better performance. For the main English evaluation, we use twelve MTEB tasks covering classification, pair classification, reranking, retrieval, and semantic textual similarity. For the multilingual evaluation, we use five multilingual MTEB tasks and report task-level averages over the evaluated language subsets.

5.5 Results

Table 1 reports the main English evaluation results across GTE-en-mlm-base and ModernBERT-base. Overall, singular value regularization consistently improves sentence embedding quality across both backbone models and pooling strategies. For GTE-en-mlm-base, applying SVR to CLS pooling improves the average score from 0.495 to 0.567, while applying SVR to mean pooling improves the average from 0.526 to 0.588. Similar gains are observed for ModernBERT-base, where SVR improves CLS pooling from 0.492 to 0.573 and mean pooling from 0.507 to 0.577. These improvements show that the proposed regularization is not tied to a specific backbone or pooling method. Additional ModernBERT-large results in Appendix Table 6 show that SVR also scales to a larger encoder backbone, improving the average score from 0.5476 with CLS pooling to 0.5992 with SVR.

The gains are largest on retrieval, reranking, and duplicate-question tasks. On SprintDuplicateQuestions, for example, SVR substantially improves both GTE-en-mlm-base and ModernBERT-base under CLS and mean pooling, with similar gains on SciFact, ArguAna, AskUbuntuDupQuestions, and StackOverflowDupQuestions. This suggests that SVR is especially useful when embeddings must form reliable neighborhoods, where spectral concentration can obscure fine-grained distinctions.

Overall, these results support our central hypothesis: contrastive learning improves semantic alignment but does not sufficiently control global spectral structure. By regularizing batch singular values, SVR encourages broader use of the representation space and yields stronger sentence embeddings. Additional comparisons with covariance regularization and SVMax are provided in Appendix D.1, where SVR substantially outperforms both SVMax and covariance regularization under identical training and evaluation settings.

6 Analysis and Discussion

We conduct three additional experiments to better understand the effect of SVR.

First, we study the effect of each SVR hyperparameter: the regularized dimensionality N, the leakage rate α, and the regularization weight λ. Unless otherwise specified, we use $N = 384$, $\alpha = 0$, and $\lambda = 80$ as the default setting. We vary one hyperparameter at a time while keeping the other two fixed. Specifically, we evaluate leakage rates $\alpha \in \{0.0, 0.2, 0.4, 0.6, 0.8\}$, regularized dimensionalities $N \in \{256, 384, 512, 640, 768\}$, and regularization weights $\lambda \in \{1, 10, 20, 50, 80\}$. The results are shown in Figure 2.

Second, we test whether the effect transfers beyond English. To this end, we construct a multilingual training setting by translating 4M English training examples into Italian, Spanish, French, and

[2] Alibaba-NLP/gte-en-mlm-base
[3] answerdotai/ModernBERT-base
[4] EuroBERT/EuroBERT-210m

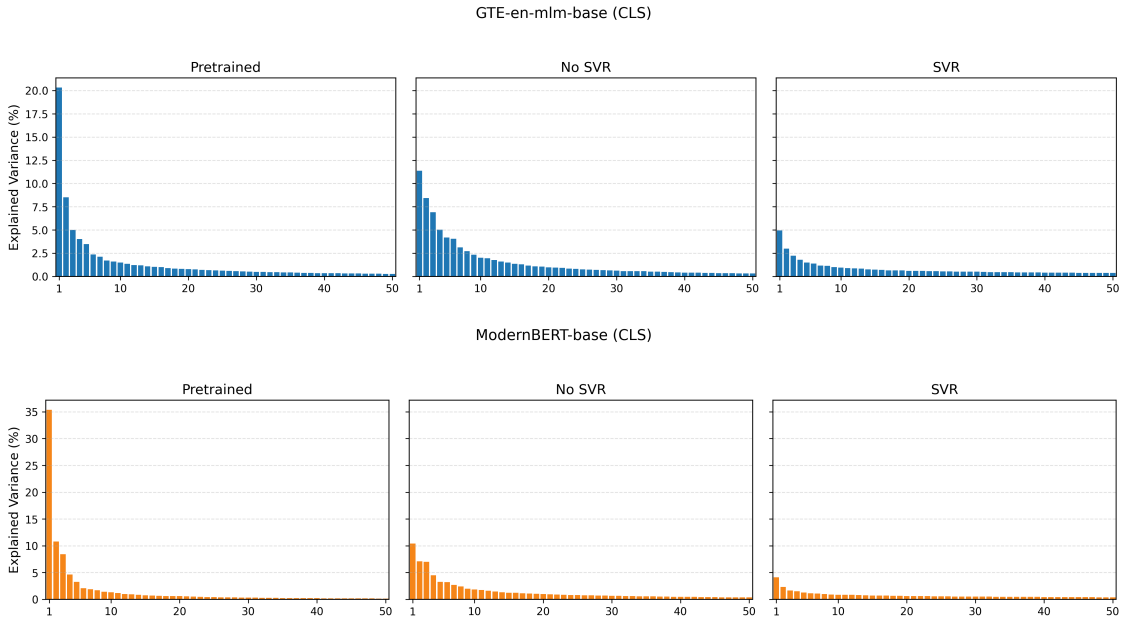

Figure 1: Explained variance spectra of the first 50 principal components for CLS representations. The top row shows GTE-en-mlm-base, and the bottom row shows ModernBERT-base. The x-axis denotes the principal component index, and the y-axis reports the percentage of total variance explained by each component. SVR produces a flatter spectrum across both backbones, indicating that variance is distributed across more principal components rather than concentrated in a few dominant directions.

German, with 1M examples per language, using GPT-4.1. We then train EuroBERT-base on this multilingual mixture and evaluate it on multilingual benchmarks shown in Table 2. The results are shown in Table 3.

Third, we perform spectral analysis. We randomly sample 100K sentences from the training corpus and compute PCA over the resulting sentence embeddings. This allows us to measure how many principal components are needed to explain different fractions of total variance and to visualize whether SVR reduces spectral concentration. The results are shown in Figure 1 and Table 4.

SVR reduces spectral concentration. Figure 1 and Table 4 show that SVR substantially reduces spectral concentration. Across both backbones and pooling strategies, SVR uses many more principal components to explain the same amount of variance. For example, with CLS pooling, the number of components required to explain 90% of the variance increases from 113 to 376 on GTE-en-mlm-base, and from 151 to 414 on ModernBERT-base. The explained variance spectra further confirm this pattern: SVR flattens the spectrum and reduces the dominance of the leading principal components.

Optimal controlled dimensionality. Combining the dimensionality ablation in Figure 2 with the PCA statistics in Table 4, we observe a concrete pattern: the best SVR range lies between the numbers of principal components needed to explain 95% and 99% of the pretrained model's variance, rather than at the full embedding dimension. For GTE-en-mlm-base CLS, the pretrained 95–99% range is 294–556 components, and the best setting appears around 384–512 dimensions. For ModernBERT-base CLS, the corresponding range is 173–462 components, and the best setting is around 256 dimensions. This suggests that SVR should not fully flatten the 768-dimensional space. Instead, it should expand the naturally active subspace to a middle region between 95% and 99% explained variance, providing enough, but not excessive, active directions for contrastive learning to organize semantic information.

Spectral shaping as a primary optimization signal. The λ ablation suggests that SVR is not a minor auxiliary regularizer: the best results require a large spectral weight. This effect is not simply due to a scale mismatch between the two loss terms. At the beginning of training, the unweighted contrastive loss and SVR loss are on a comparable numerical scale, typically both falling in the range of approximately 1–10. Thus, the large optimal value of λ indicates that strong spectral shaping provides a substantial optimization signal rather

Task	GTE-en-mlm-base					ModernBERT-base				
	6L-WMean	CLS		Mean		6L-WMean	CLS		Mean	
	Base	Base	SVR	Base	SVR	Base	Base	SVR	Base	SVR
BIOSSES	0.523	0.505	0.568	0.520	0.596	0.592	0.612	**0.695**	0.594	0.691
SciFact	0.235	0.254	0.426	0.288	**0.473**	0.295	0.251	0.426	0.255	0.436
STSBenchmark	0.659	0.635	0.690	0.678	**0.706**	0.660	0.636	0.695	0.653	0.695
ArguAna	0.369	0.347	0.428	0.394	**0.452**	0.406	0.274	0.377	0.378	0.426
EmotionClassification.v2	0.321	0.319	0.358	0.328	**0.360**	0.313	0.282	0.359	0.283	0.350
Banking77Classification.v2	0.735	0.714	0.777	0.745	0.786	0.750	0.733	**0.788**	0.721	0.778
SprintDuplicateQuestions	0.580	0.502	0.766	0.606	**0.836**	0.593	0.498	0.778	0.541	0.776
TwitterSemEval2015	0.541	0.504	0.524	0.537	**0.556**	0.470	0.454	0.460	0.477	0.482
SciDocsRR	0.729	0.726	0.754	0.740	**0.761**	0.718	0.716	0.742	0.715	0.729
AskUbuntuDupQuestions	0.511	0.506	0.548	0.522	0.545	0.531	0.515	**0.556**	0.518	0.549
StackOverflowDupQuestions	0.359	0.341	0.389	0.360	0.401	0.380	0.367	**0.426**	0.380	**0.426**
XNLI	0.586	0.587	0.573	**0.591**	0.579	0.589	0.567	0.572	0.569	0.583
Avg.	0.512	0.495	0.567	0.526	**0.588**	0.525	0.492	0.573	0.507	0.577

Table 1: Main results on English sentence embedding benchmarks. We compare different pooling strategies across GTE-en-mlm-base and ModernBERT-base. **Base** denotes the corresponding pooling baseline without singular value regularization, while **SVR** denotes the same pooling strategy trained with singular value regularization.

Category	Task	Metric
	English evaluation	
Classification	EmotionClassification.v2 (Saravia et al., 2018)	Accuracy
	Banking77Classification.v2 (Casanueva et al., 2020)	Accuracy
PairClassification	SprintDuplicateQuestions (Shah et al., 2018)	Euclidean AP
	TwitterSemEval2015 (Xu et al., 2015)	Euclidean AP
	XNLI (Conneau et al., 2018)	Cosine AP
Reranking	SciDocsRR (Cohan et al., 2020)	MAP@100
	AskUbuntuDupQuestions (Wang et al., 2021)	MAP@100
	StackOverflowDupQuestions (Liu et al., 2018)	MAP@100
Retrieval	SciFact (Wadden et al., 2020)	nDCG@10
	ArguAna (Thakur et al., 2021)	nDCG@10
STS	BIOSSES (Soğancıoğlu et al., 2017)	Spearman
	STSBenchmark (May, 2021)	Spearman
	Multilingual evaluation	
Classification	MassiveScenarioClassification (FitzGerald et al., 2022)	Accuracy
Clustering	SIB200ClusteringS2S (Adelani et al., 2023)	V-measure
PairClassification	XNLI (Conneau et al., 2018)	Cosine AP
Reranking	WikipediaRerankingMultilingual (Foundation)	MAP@100
STS	STSBenchmarkMultilingualSTS (May, 2021)	Spearman

Table 2: MTEB tasks used in our evaluation.

Task	6L-WMean	CLS		Mean	
	Base	Base	SVR	Base	SVR
STS-Multi.	**0.649**	0.646	0.644	0.644	0.646
MassiveScenario	0.626	0.623	**0.635**	0.622	0.624
WikiRerank	0.646	0.638	**0.764**	0.636	**0.764**
SIB200	0.416	0.414	**0.494**	0.417	0.477
XNLI	0.571	0.575	**0.581**	0.575	0.578
Avg.	0.581	0.579	**0.624**	0.579	0.618

Table 3: Multilingual evaluation results on EuroBERT. Full language-level results are provided in Table 7 in the Appendix.

than merely compensating for a numerically small regularization term. Contrastive loss defines which examples should be close or far apart, but these relations are difficult to express if most variance is compressed into a few dominant directions. SVR expands the active subspace first, giving contrastive learning a better-conditioned space in which to organize semantic relations.

Leakage rate as controlled regularization. Although the best setting in our experiments is $\alpha = 0$, we do not treat this as universal. The leakage rate explicitly controls how much spectral concentration the regularizer permits. This flexibility is useful because some features may naturally carry more important information than others, so enforcing a perfectly uniform spectrum may be unnecessarily restrictive. For the models studied here, stronger enforcement of the spectral floor is most beneficial, while larger leakage weakens the effect. Other architectures, larger backbones, or domain-specific settings may require different leakage rates.

Task and multilingual effects. The English and multilingual results show a consistent task-level pattern. SVR brings larger gains on retrieval, reranking, duplicate-question detection, and clustering tasks, while gains on STS and XNLI are smaller. This pattern is consistent with the view that reducing spectral concentration helps organize inputs at the cluster or candidate-set level. In contrast, isolated pairwise similarity and semantic classification appear less sensitive to this type of geometric improvement.

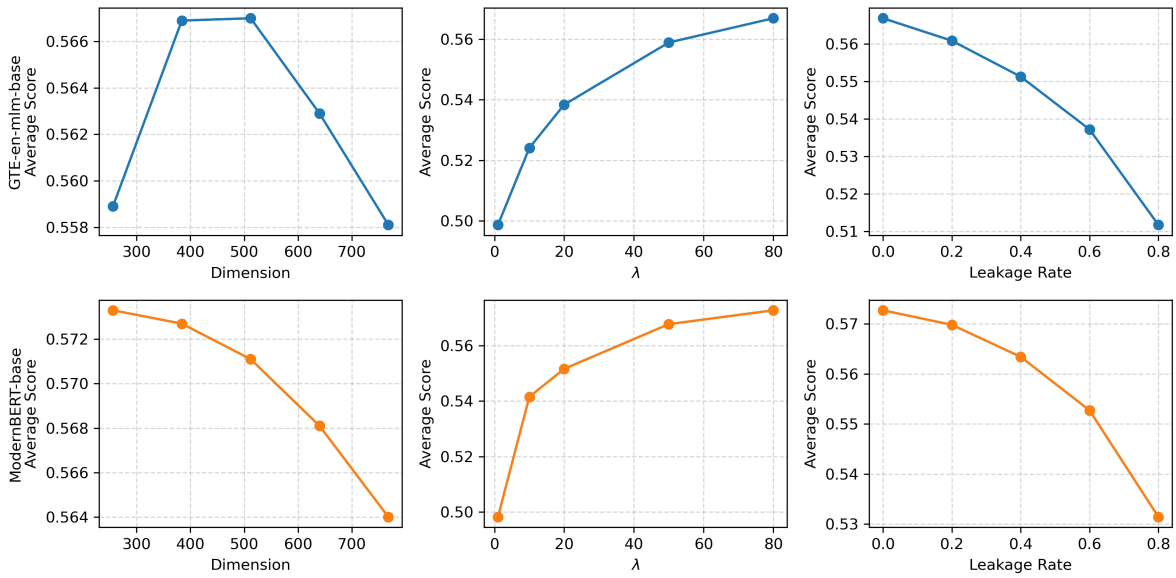

Figure 2: Ablation trends for singular value regularization across GTE-en-mlm-base and ModernBERT-base for CLS pooling. Rows correspond to backbones, and columns correspond to dimension, regularization strength λ, and leakage rate. Increasing λ consistently improves performance, whereas increasing leakage consistently degrades performance. The optimal active dimension varies across backbones.

Expl. Var.	GTE-en-mlm-base						ModernBERT-base					
	CLS			Mean			CLS			Mean		
	Pre.	No SVR	SVR	Pre.	No SVR	SVR	Pre.	No SVR	SVR	Pre.	No SVR	SVR
10%	1	1	3	1	2	5	1	1	5	1	1	5
20%	1	3	11	2	3	17	1	3	16	3	3	15
30%	3	4	27	5	5	38	1	5	33	6	5	30
40%	5	7	50	11	8	66	2	8	58	12	9	52
50%	10	10	81	23	12	103	3	14	90	24	15	82
60%	19	17	123	43	19	148	5	24	134	47	24	120
70%	38	28	177	78	29	204	10	39	192	87	40	173
80%	78	50	252	136	48	279	23	70	276	155	70	249
90%	177	113	376	246	103	394	74	151	414	283	144	380
95%	294	211	487	366	190	499	173	260	530	414	246	497
99%	556	477	656	608	443	659	462	523	690	641	501	673

Table 4: PCA explained variance concentration analysis. Each entry reports the minimum number of principal components required to explain a given fraction of total variance. Larger values indicate that variance is distributed across more dimensions. "Pre." denotes pretrained representations before contrastive fine-tuning.

7 Conclusion

We proposed singular value regularization (SVR), a batch-level framework for shaping sentence embedding spaces during contrastive training. SVR directly controls the singular value spectrum by imposing a spectral floor, derived from an explicit singular value budget, over a controlled set of active directions, thereby reducing spectral concentration and encouraging broader use of the embedding space. Experiments show consistent gains across backbones, pooling strategies, and English evaluation tasks, with additional multilingual results showing that the effect transfers beyond English.

Our analysis highlights two main findings. First, effective regularization does not require full-dimensional flattening: the strongest settings lie in a controlled active-dimensional region near the pretrained representation's 95% to 99% explained variance thresholds. This suggests that sentence embedding training should expand the naturally active subspace without enforcing complete spectral uniformity. Second, the best performance requires a large SVR weight, indicating that spectral shaping is not merely a minor auxiliary regularizer but a primary training target. Contrastive loss defines semantic relations between examples, but these relations are best expressed when the representation space has sufficient active and well-conditioned directions. Together, these results support a geometry-first view of sentence embedding learning: effective representations require not only semantic supervision, but also explicit control over the active geometry of the embedding space.

Limitation

This work currently focuses on encoder-only models, while a broad class of sentence embedding models is built from decoder-only language models. Although our method is formulated at the embedding-matrix level and is therefore not specific to a particular architecture, decoder-only models may exhibit different spectral behavior because their representations are shaped by autoregressive training, causal attention, and often different pooling choices such as last-token or EOS pooling. Future work should examine whether the same controlled active-dimensional range applies to decoder-based embedding models.

A second limitation is that SVR introduces additional computation because it requires an SVD over the batch embedding matrix during training. In our experiments, this cost is manageable because the regularizer is applied only to the sentence-level embedding matrix rather than token-level hidden states. However, the overhead may become more significant for very large embedding dimensions, very large batches, or multi-vector embedding models. More efficient approximations to the singular value spectrum could make the method easier to scale.

Finally, the analysis in this paper should be interpreted as empirical rather than universal. We observe that SVR reduces spectral concentration, that stronger enforcement of the spectral floor performs best in our setting, and that the most effective controlled dimensionality often lies between the pretrained representation's 95% and 99% explained variance thresholds. These findings are based on the backbones, pooling strategies, datasets, and benchmarks studied here. Our goal is to establish controlled spectral shaping as a useful training principle for sentence embeddings, rather than to exhaustively compare other geometry regularizers or model families. Broader validation across larger instruction-tuned embedding models, decoder-based embedding models, additional languages, domain-specific datasets, alternative training regimes, and other spectral or anti-collapse regularizers remains an important direction for future work.

We used ChatGPT for final writing polish.

References

David Ifeoluwa Adelani, Hannah Liu, Xiaoyu Shen, Nikita Vassilyev, Jesujoba O Alabi, Yanke Mao, Haonan Gao, and Annie En-Shiun Lee. 2023. Sib-200: A simple, inclusive, and big evaluation dataset for topic classification in 200+ languages and dialects. *arXiv preprint arXiv:2309.07445*.

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, Mir Rosenberg, Xia Song, Alina Stoica, Saurabh Tiwary, and Tong Wang. 2018. Ms marco: A human generated machine reading comprehension dataset. *Preprint*, arXiv:1611.09268.

Adrien Bardes, Jean Ponce, and Yann LeCun. 2022. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. *Preprint*, arXiv:2105.04906.

Iñigo Casanueva, Tadas Temcinas, Daniela Gerz, Matthew Henderson, and Ivan Vulic. 2020. Efficient intent detection with dual sentence encoders. In *Proceedings of the 2nd Workshop on NLP for ConvAI - ACL 2020*. Data available at https://github.com/PolyAI-LDN/task-specific-datasets.

Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel Weld. 2020. SPECTER: Document-level representation learning using citation-informed transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2270–2282, Online. Association for Computational Linguistics.

Alexis Conneau, Guillaume Lample, Ruty Rinott, Adina Williams, Samuel R. Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. Xnli: Evaluating cross-lingual sentence representations. *Preprint*, arXiv:1809.05053.

DataCanary, hilfialkaff, Lili Jiang, Meg Risdal, Nikhil Dandekar, and tomtung. 2017. Quora question pairs. `https://kaggle.com/competitions/quora-question-pairs`. Kaggle.

Matthew Dunn, Levent Sagun, Mike Higgins, V. Ugur G"uney, Volkan Cirik, and Kyunghyun Cho. 2017. Searchqa: A new q&a dataset augmented with context from a search engine. *CoRR*, abs/1704.05179.

Kenneth Enevoldsen, Isaac Chung, Imene Kerboua, Márton Kardos, Ashwin Mathur, David Stap, Jay Gala, Wissam Siblini, Dominik Krzemiński, Genta Indra Winata, Saba Sturua, Saiteja Utpala, Mathieu Ciancone, Marion Schaeffer, Gabriel Sequeira, Diganta Misra, Shreeya Dhakal, Jonathan Rystrøm, Roman Solomatin, and 67 others. 2025. Mmteb: Massive multilingual text embedding benchmark. *arXiv preprint arXiv:2502.13595*.

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings. In

Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 55–65, Hong Kong, China. Association for Computational Linguistics.

Anthony Fader, Luke Zettlemoyer, and Oren Etzioni. 2014. Open Question Answering Over Curated and Extracted Knowledge Bases. In *KDD*.

Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. 2019. ELI5: Long form question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3558–3567, Florence, Italy. Association for Computational Linguistics.

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

Wikimedia Foundation. Wikimedia downloads.

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, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Mansi Gupta, Nitish Kulkarni, Raghuveer Chanda, Anirudha Rayasam, and Zachary C Lipton. 2019. Amazonqa: A review-based question answering task. *Preprint*, arXiv:1908.04364.

Li Jing, Pascal Vincent, Yann LeCun, and Yuandong Tian. 2022. Understanding dimensional collapse in contrastive self-supervised learning. *Preprint*, arXiv:2110.09348.

Daniel Khashabi, Amos Ng, Tushar Khot, Ashish Sabharwal, Hannaneh Hajishirzi, and Chris Callison-Burch. 2021. GooAQ: Open question answering with diverse answer types. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 421–433, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Mahnaz Koupaee and William Yang Wang. 2018. Wikihow: A large scale text summarization dataset. *Preprint*, arXiv:1810.09305.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

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

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

Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel Weld. 2020. S2ORC: The semantic scholar open research corpus. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4969–4983, Online. Association for Computational Linguistics.

Philip May. 2021. Machine translated multilingual sts benchmark dataset.

Jiaqi Mu, Suma Bhat, and Pramod Viswanath. 2018. All-but-the-top: Simple and effective post-processing for word representations. *Preprint*, arXiv:1702.01417.

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

Jianmo Ni, Jiacheng Li, and Julian McAuley. 2019. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. 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 188–197, Hong Kong, China. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *Preprint*, arXiv:1606.05250.

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, Hong Kong, China. Association for Computational Linguistics.

Elvis Saravia, Hsien-Chi Toby Liu, Yen-Hao Huang, Junlin Wu, and Yi-Shin Chen. 2018. CARER: Contextualized affect representations for emotion recognition. In *Proceedings of the 2018 Conference on*

Empirical Methods in Natural Language Processing, pages 3687–3697, Brussels, Belgium. Association for Computational Linguistics.

Darsh Shah, Tao Lei, Alessandro Moschitti, Salvatore Romeo, and Preslav Nakov. 2018. Adversarial domain adaptation for duplicate question detection. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1056–1063, Brussels, Belgium. Association for Computational Linguistics.

Gizem Soğancıoğlu, Hakime "Ozt"urk, and Arzucan "Ozg"ur. 2017. Biosses: a semantic sentence similarity estimation system for the biomedical domain. *Bioinformatics*, 33(14):i49–i58.

Hongjin Su, Weijia Shi, Jungo Kasai, Yizhong Wang, Yushi Hu, Mari Ostendorf, Wen-tau Yih, Noah A. Smith, Luke Zettlemoyer, and Tao Yu. 2023. One embedder, any task: Instruction-finetuned text embeddings. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1102–1121, Toronto, Canada. Association for Computational Linguistics.

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

Ahmed Taha, Alex Hanson, Abhinav Shrivastava, and Larry Davis. 2021. Svmax: A feature embedding regularizer. *Preprint*, arXiv:2103.02770.

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)*.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2019. Representation learning with contrastive predictive coding. *Preprint*, arXiv:1807.03748.

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. 2020. Fact or fiction: Verifying scientific claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7534–7550, Online. Association for Computational Linguistics.

Kexin Wang, Nils Reimers, and Iryna Gurevych. 2021. Tsdae: Using transformer-based sequential denoising auto-encoderfor unsupervised sentence embedding learning. *arXiv preprint arXiv:2104.06979*.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2024. Text embeddings by weakly-supervised contrastive pre-training. *Preprint*, arXiv:2212.03533.

Wei Xu, Chris Callison-Burch, and Bill Dolan. 2015. SemEval-2015 task 1: Paraphrase and semantic similarity in Twitter (PIT). In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 1–11, Denver, Colorado. Association for Computational Linguistics.

Yaodong Yu, Kwan Ho Ryan Chan, Chong You, Chaobing Song, and Yi Ma. 2020. Learning diverse and discriminative representations via the principle of maximal coding rate reduction. *Preprint*, arXiv:2006.08558.

Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *NIPS*.

A Experimental Setup Details

For the main English evaluation, we use two encoder backbones: GTE-en-mlm-base and ModernBERT-base. For multilingual evaluation, we additionally train EuroBERT-base on the multilingual mixture described in Section 6. All experiments are conducted on a single NVIDIA L40S GPU. During training, we set the temperature to 0.1 and use a learning rate of 2×10^{-5} for the backbone parameters. Optimization is performed using AdamW, with 5% of the total training steps used for warmup, a weight decay of 0.01, and a linear learning-rate scheduler. The gradient clipping threshold is set to 1. The maximum sequence length is set to 1024, and all experiments are trained for one epoch.

We use an effective batch size of 1024 while computing gradients over micro-batches of 32 pairs at a time. After all 1024 pairs in the batch have been processed, we perform the final backpropagation step over the full batch.

B Training datasets

Our training corpus is constructed from 13 datasets and derived pairs covering a broad range of domains and task types. These include question answering and retrieval datasets: ELI5 (Fan et al., 2019), GooAQ (Khashabi et al., 2021), MS MARCO (Bajaj et al., 2018), Natural Questions (Kwiatkowski et al., 2019), SearchQA (Dunn et al., 2017), SQuAD (Rajpurkar et al., 2016), and Amazon QA (Gupta et al., 2019); paraphrase and duplicate detection datasets: Quora Question Pairs (DataCanary et al., 2017) and WikiAnswers (Fader et al., 2014); summarization and compression-style datasets: WikiHow (Koupaee and Wang, 2018) and

AG News (Zhang et al., 2015); scientific text matching data derived from S2ORC (Lo et al., 2020), including citation-title pairs and title-abstract pairs; and general-domain review text pairs from Amazon Reviews 2018 (Ni et al., 2019).

C Additional Preliminary

C.1 Sentence Embeddings

Given an input text $x = \{t_1, t_2, \ldots, t_N\}$, a transformer-based encoder with L layers produces final-layer token representations

$$H^{(L)} = \{\mathbf{h}_1^{(L)}, \mathbf{h}_2^{(L)}, \ldots, \mathbf{h}_N^{(L)}\}, \quad \mathbf{h}_i^{(L)} \in \mathbb{R}^d.$$

A sentence embedding is obtained by applying a pooling function $P(\cdot)$ over these token representations:

$$\mathbf{s} = P(H^{(L)}) \in \mathbb{R}^d.$$

Common choices of $P(\cdot)$ include mean pooling, which averages token representations across the sequence, and special-token pooling, which uses the final hidden state of a designated token such as `[CLS]` or `[EOS]`.

D Additional Experimental Results

D.1 Comparison with other Geometry Regularizers

To further examine whether the gains of SVR come from generic geometric regularization or from the proposed controlled spectral formulation, we compare SVR with two alternative geometry-based regularizers: covariance regularization and SVMax. Covariance regularization penalizes correlations between embedding dimensions, while SVMax encourages larger singular values by maximizing the mean singular value of the feature embedding matrix. We conduct this comparison on GTE-en-mlm-base with CLS pooling under the same evaluation setting.

Table 5 shows that both alternative regularizers remain close to the unregularized CLS baseline. The best average score among covariance-regularized models is 0.496, and the best SVMax result is 0.497, compared with 0.495 for the CLS baseline. In contrast, SVR reaches an average score of 0.567. This suggests that the improvement does not come merely from adding a generic geometry regularizer. Instead, the gains are tied to the controlled design of SVR, which explicitly specifies both the spectral floor and the active regularization range.

For SVMax, we only report smaller weights because its objective directly maximizes the mean singular value, typically by subtracting this term from the task loss. With a large coefficient such as $\lambda = 80$, the total loss can become dominated by the negative singular value term and is no longer comparable to the contrastive training objective. This differs from SVR, whose regularization term is a non-negative margin loss. SVR can therefore use a large weight to strongly enforce a specified spectral floor, while the loss naturally becomes inactive once the controlled singular directions satisfy the margin.

Task	CLS	Covariance Reg.				SVMax		SVR
	Baseline	$\lambda = 0.1$	$\lambda = 1$	$\lambda = 10$	$\lambda = 80$	$\lambda = 0.1$	$\lambda = 1$	$N = 384, \alpha = 0, \lambda = 80$
BIOSSES	0.505	0.506	0.506	0.512	0.499	0.504	0.508	0.568
SciFact	0.254	0.258	0.260	0.241	0.232	0.258	0.259	0.426
STSBenchmark	0.635	0.634	0.634	0.634	0.644	0.634	0.635	0.690
ArguAna	0.347	0.347	0.346	0.341	0.346	0.346	0.347	0.428
EmotionClassification.v2	0.319	0.320	0.319	0.320	0.314	0.320	0.321	0.358
Banking77Classification.v2	0.714	0.714	0.715	0.713	0.723	0.716	0.717	0.777
SprintDuplicateQuestions	0.502	0.504	0.504	0.486	0.485	0.504	0.505	0.766
TwitterSemEval2015	0.504	0.505	0.504	0.502	0.508	0.504	0.504	0.524
SciDocsRR	0.726	0.726	0.726	0.725	0.724	0.726	0.727	0.754
AskUbuntuDupQuestions	0.506	0.506	0.507	0.506	0.518	0.508	0.508	0.548
StackOverflowDupQuestions	0.341	0.340	0.341	0.339	0.339	0.340	0.341	0.389
XNLI	0.587	0.586	0.586	0.590	0.589	0.586	0.587	0.573
Avg.	0.495	0.495	0.496	0.492	0.493	0.496	0.497	0.567

Table 5: Comparison with alternative geometry regularizers on GTE-en-mlm-base with CLS pooling. Covariance regularization and SVMax remain close to the unregularized CLS baseline, while SVR yields substantially larger gains under the same evaluation setting. This suggests that the benefit of SVR comes from explicitly controlling the spectral floor and active regularization range, rather than from applying a generic geometry regularizer.

Task	CLS	Mean	SVR	SVR				
	Base	**Base**	$\mathbf{N = 384}$	$\mathbf{N = 512}$				
			$\lambda = 80$	$\lambda = 20$	$\lambda = 40$	$\lambda = 80$	$\lambda = 100$	$\lambda = 200$
BIOSSES	0.6102	0.5536	0.7336	0.6913	0.7125	0.7263	0.7308	**0.7525**
SciFact	0.3912	0.3577	0.4930	0.4495	0.4765	0.5024	**0.5115**	0.5081
STSBenchmark	**0.7040**	0.6608	0.7002	0.6957	0.6956	0.7024	0.7023	0.6989
ArguAna	0.3948	0.4087	0.4352	0.4049	0.4203	0.4340	0.4388	**0.4439**
EmotionClassification.v2	0.3296	0.3093	0.3601	0.3512	0.3601	0.3654	**0.3654**	0.3618
Banking77Classification.v2	0.7741	0.7498	0.7987	0.7883	0.7939	0.7979	0.7990	**0.8051**
SprintDuplicateQuestions	0.5413	0.5042	0.7284	0.6540	0.6992	0.7394	0.7497	**0.7620**
TwitterSemEval2015	0.5192	0.5061	0.4979	**0.5366**	0.5245	0.5067	0.5014	0.4790
SciDocsRR	0.7690	0.7658	**0.7859**	0.7773	0.7815	0.7839	0.7839	0.7812
AskUbuntuDupQuestions	0.5575	0.5491	0.5733	0.5689	0.5716	0.5759	0.5770	**0.5783**
StackOverflowDupQuestions	0.3769	0.3726	0.4279	0.4033	0.4188	0.4269	0.4308	**0.4369**
XNLI	**0.6033**	0.5982	0.5875	0.5994	0.5958	0.5910	0.5888	0.5824
Avg.	0.5476	0.5280	0.5935	0.5767	0.5875	0.5960	0.5983	**0.5992**

Table 6: Additional results on ModernBERT-large. We compare CLS and mean pooling baselines with several SVR configurations using CLS pooling. SVR consistently improves over both pooling baselines, with the best average score obtained at $N = 512$ and $\lambda = 200$.

Task	Language	6L-WMean	CLS		Mean	
		Base	Base	SVR	Base	SVR
STSBenchmarkMultilingualSTS	English	0.676	0.677	0.684	0.677	**0.692**
	German	0.641	0.638	0.633	0.637	**0.643**
	Spanish	0.637	**0.639**	0.634	0.637	0.632
	French	**0.652**	0.647	0.642	0.644	0.640
	Italian	**0.639**	0.627	0.628	0.625	0.625
MassiveScenarioClassification	German	0.610	0.617	**0.628**	0.612	0.621
	English	0.687	0.676	**0.688**	0.679	0.676
	Spanish	0.608	0.607	**0.617**	0.603	0.605
	French	0.626	0.622	**0.634**	0.623	0.619
	Italian	0.598	0.592	**0.610**	0.595	0.599
WikipediaRerankingMultilingual	German	0.684	0.681	**0.753**	0.678	0.750
	English	0.649	0.645	**0.807**	0.640	0.804
	Italian	0.604	0.590	0.734	0.589	**0.739**
SIB200ClusteringS2S	German	0.399	0.397	**0.510**	0.391	0.475
	English	0.454	0.436	**0.470**	0.443	0.459
	French	0.426	0.447	**0.501**	0.427	0.490
	Italian	0.387	0.381	**0.487**	0.400	0.479
	Spanish	0.414	0.407	**0.500**	0.423	0.481
XNLI	German	0.565	0.573	**0.576**	0.575	0.575
	English	0.575	0.577	**0.587**	0.577	0.581
	Spanish	0.561	0.572	**0.579**	0.564	0.569
	French	0.582	0.579	0.583	0.585	**0.588**
Avg.		0.555	0.553	**0.604**	0.553	0.595

Table 7: Full language-level multilingual evaluation results on EuroBERT trained with multilingual data. We compare mean pooling over the last six layers, standard CLS, SVR applied to CLS, standard mean pooling, and SVR applied to mean pooling.

Task	Dimension ($\lambda = 80$)					Regularization Strength ($N = 384$)					Leakage Rate ($N = 384, \lambda = 80$)				
	256d	384d	512d	640d	768d	λ=1	λ=10	λ=20	λ=50	λ=80	0.0	0.2	0.4	0.6	0.8
BIOSSES	0.551	0.568	0.567	0.550	0.551	0.505	0.527	0.522	0.546	0.568	0.568	0.549	0.537	0.532	0.525
SciFact	0.411	0.426	0.430	0.427	0.416	0.264	0.311	0.343	0.413	0.426	0.426	0.414	0.382	0.337	0.276
STSBenchmark	0.693	0.690	0.686	0.682	0.677	0.638	0.665	0.674	0.686	0.690	0.690	0.687	0.681	0.675	0.658
ArguAna	0.428	0.428	0.422	0.421	0.415	0.352	0.383	0.404	0.423	0.428	0.428	0.423	0.415	0.401	0.367
EmotionClassification.v2	0.355	0.358	0.356	0.355	0.354	0.322	0.342	0.354	0.358	0.358	0.358	0.361	0.357	0.350	0.330
Banking77Classification.v2	0.772	0.777	0.777	0.774	0.769	0.720	0.747	0.759	0.772	0.777	0.777	0.773	0.768	0.755	0.735
SprintDuplicateQuestions	0.753	0.766	0.765	0.754	0.732	0.513	0.592	0.657	0.732	0.766	0.766	0.743	0.703	0.641	0.546
TwitterSemEval2015	0.503	0.524	0.533	0.532	0.529	0.505	0.515	0.520	0.526	0.524	0.524	0.527	0.528	0.527	0.513
SciDocsRR	0.755	0.754	0.752	0.750	0.748	0.728	0.739	0.745	0.751	0.754	0.754	0.752	0.748	0.742	0.733
AskUbuntuDupQuestions	0.540	0.548	0.547	0.546	0.545	0.508	0.522	0.530	0.543	0.548	0.548	0.544	0.540	0.534	0.517
StackOverflowDupQuestions	0.379	0.389	0.390	0.384	0.380	0.342	0.360	0.367	0.379	0.389	0.389	0.382	0.375	0.366	0.351
XNLI	0.567	0.573	0.578	0.580	0.581	0.587	0.587	0.585	0.578	0.573	0.573	0.576	0.581	0.586	0.590
Avg.	0.559	**0.567**	**0.567**	0.563	0.558	0.499	0.524	0.538	0.559	**0.567**	**0.567**	0.561	0.551	0.537	0.512

Table 8: Ablation results in task-level for **GTE-en-mlm-base** with batch size 1024. We vary the embedding dimension, the regularization strength, and the leakage rate.

Task	Dimension ($\lambda = 80$)					Regularization Strength ($N = 384$)					Leakage Rate ($N = 384, \lambda = 80$)				
	256d	384d	512d	640d	768d	λ=1	λ=10	λ=20	λ=50	λ=80	0.0	0.2	0.4	0.6	0.8
BIOSSES	0.675	0.695	0.713	0.709	0.704	0.616	0.654	0.655	0.691	0.695	0.695	0.691	0.677	0.663	0.634
SciFact	0.434	0.426	0.412	0.405	0.399	0.261	0.371	0.383	0.412	0.426	0.426	0.417	0.395	0.372	0.342
STSBenchmark	0.697	0.695	0.690	0.686	0.683	0.641	0.681	0.687	0.692	0.695	0.695	0.692	0.691	0.688	0.683
ArguAna	0.388	0.377	0.366	0.365	0.358	0.285	0.321	0.348	0.374	0.377	0.377	0.377	0.369	0.354	0.314
EmotionClassification.v2	0.357	0.359	0.359	0.353	0.348	0.289	0.331	0.341	0.355	0.359	0.359	0.357	0.352	0.344	0.323
Banking77Classification.v2	0.788	0.788	0.787	0.784	0.780	0.740	0.776	0.781	0.787	0.788	0.788	0.788	0.786	0.782	0.767
SprintDuplicateQuestions	0.785	0.778	0.783	0.775	0.759	0.513	0.658	0.690	0.746	0.778	0.778	0.760	0.737	0.691	0.621
TwitterSemEval2015	0.452	0.460	0.460	0.463	0.465	0.457	0.470	0.476	0.469	0.460	0.460	0.467	0.475	0.478	0.469
SciDocsRR	0.746	0.742	0.737	0.735	0.734	0.719	0.731	0.736	0.740	0.742	0.742	0.740	0.738	0.735	0.728
AskUbuntuDupQuestions	0.555	0.556	0.556	0.553	0.551	0.516	0.540	0.542	0.550	0.556	0.556	0.550	0.548	0.543	0.537
StackOverflowDupQuestions	0.432	0.426	0.422	0.417	0.412	0.374	0.398	0.410	0.424	0.426	0.426	0.425	0.419	0.409	0.388
XNLI	0.570	0.572	0.570	0.572	0.573	0.569	0.569	0.571	0.573	0.572	0.572	0.573	0.575	0.574	0.570
Avg.	**0.573**	0.573	0.571	0.568	0.564	0.498	0.542	0.552	0.568	**0.573**	**0.573**	0.570	0.563	0.553	0.531

Table 9: Ablation results in task-level for **ModernBERT-base** with batch size 1024. We vary the embedding dimension, the regularization strength, and the leakage rate.