Isotropy, Clusters, and Classifiers

Timothee Mickus[♡] **Stig-Arne Grönroos**[♡♠] **Joseph Attieh**[♡]

[♡] University of Helsinki, [♠] Silo.AI, Finland
firstname.lastname@helsinki.fi

Abstract

Whether embedding spaces use all their dimensions equally, i.e., whether they are isotropic, has been a recent subject of discussion. Evidence has been accrued both for and against enforcing isotropy in embedding spaces. In the present paper, we stress that isotropy imposes requirements on the embedding space that are not compatible with the presence of clusters—which also negatively impacts linear classification objectives. We demonstrate this fact both empirically and mathematically and use it to shed light on previous results from the literature.

1 Introduction

Recently, there has been much discussion centered around whether vector representations used in NLP do and should use all dimensions equally. This characteristic is known as isotropy: In an isotropic embedding model, every direction is equally probable, ensuring uniform data representation without directional bias. At face value, such a characteristic would appear desirable: Naively, one could argue that an anisotropic embedding space would be overparametrized, since it can afford to use some dimensions inefficiently.

The debate surrounding isotropy was initially sparked by Mu and Viswanath (2018), who highlighted that isotropic static representations fared better on common lexical semantics benchmarks, and Ethayarajh (2019), who stressed that contextual embeddings are anisotropic. Since then, evidence has been accrued both for and against enforcing isotropy on embeddings.

In the present paper, we demonstrate that this conflicting evidence can be accounted for once we consider how isotropy relates to embedding space geometry. Strict isotropy, as assessed by IsoScore (Rudman et al., 2022), requires the absence of clusters, and thereby also conflicts with linear classification objectives. This echoes previous empirical studies connecting isotropy and cluster structures (Ait-Saada and Nadif, 2023, a.o.). In the present paper, we formalize this connection mathematically in Section 2. We then empirically verify our mathematical approach in Section 3, discuss how this relation sheds light on earlier works focusing on anisotropy in Section 4, and conclude with directions for future work in Section 5.

2 Some conflicting optimization objectives

We can show that isotropy—as assessed by IsoScore (Rudman et al., 2022)—impose requirements that conflict with cluster structures—as assessed by silhouette scores (Rousseeuw, 1987)—as well as linear classifier objectives.

Notations. In what follows, let $\mathcal{D}$ be a multiset of points in a vector space, Ω a set of labels, and $\ell : \mathcal{D} \to \Omega$ a labeling function that associates a given data-point in $\mathcal{D}$ to the relevant label. Without loss of generality, let us further assume that $\mathcal{D}$ is PCA-transformed. Let us also define the following constructs for clarity of exposition:

$$\mathcal{D}_\omega = \{\mathbf{d} \;:\; \ell(\mathbf{d}) = \omega\}$$

$$\mathrm{sign}(\omega, \omega') = \begin{cases} -1 & \text{if } \omega = \omega' \\ +1 & \text{otherwise} \end{cases}$$

Simply put, $\mathcal{D}_\omega$ is the subset of points in $\mathcal{D}$ with label ω, whereas the sign function helps delineate terms that need to be maximized (inter-cluster) vs. terms that need to be minimized (intra-cluster).

2.1 Silhouette objective for clustering

We can consider whether the groups as defined by ℓ are in fact well delineated by the Euclidean distance, i.e., whether they form natural clusters. This is something that can be assessed through silhouette scores, which involve a *separation* and a *cohesion* score for each data-point. The cohesion score consists in computing the average distance

Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 75–84
August 11-16, 2024 ©2024 Association for Computational Linguistics

between the data-point and other members of its group, whereas separation consists in computing the minimum cohesion score the data-point could have received with any other label than the one it was assigned to. More formally, let:

$$\mathrm{cost}(\mathbf{d}, \mathcal{S}) = \frac{1}{|\mathcal{S}|} \sum_{\mathbf{d}' \in \mathcal{S}} \sqrt{\sum_i \left(\mathbf{d}_i - \mathbf{d}'_i\right)^2}$$

then we can define the silhouette for one sample as

$$\begin{aligned} \mathrm{coh}(\mathbf{d}) &= \mathrm{cost}\left(\mathbf{d}, \mathcal{D}_{\ell(\mathbf{d})} \setminus \{\mathbf{d}\}\right) \\ \mathrm{sep}(\mathbf{d}) &= \min_{\omega' \in \Omega \setminus \{\ell(\mathbf{d})\}} \mathrm{cost}\left(\mathbf{d}, \mathcal{D}_{\omega'}\right) \\ \mathrm{silhouette}(\mathbf{d}) &= \frac{\mathrm{sep}(\mathbf{d}) - \mathrm{coh}(\mathbf{d})}{\max\{\mathrm{sep}(\mathbf{d}), \mathrm{coh}(\mathbf{d})\}} \end{aligned}$$

Or in other words, the silhouette score is maximized when separation cost (sep) is maximized and cohesion cost (coh) is minimized. Hence, to maximize the silhouette score across the whole dataset $\mathcal{D}$, one needs to (i) maximize all inter-cluster distances, and (ii) minimize all intra-cluster distances.

We can therefore define a maximization objective for the entire set $\mathcal{D}$:

$$\sum_{\mathbf{d} \in \mathcal{D}} \sum_{\mathbf{d}' \in \mathcal{D}} \mathrm{sign}(\ell(\mathbf{d}), \ell(\mathbf{d}')) \sqrt{\sum_i \left(\mathbf{d}_i - \mathbf{d}'_i\right)^2}$$

which, due to the monotonicity of the square root in $\mathbb{R}^+$, will have the same optimal argument $\mathcal{D}^*$ as the simpler objective $\mathcal{O}_\mathrm{S}$

$$\mathcal{O}_\mathrm{S} = \sum_{\mathbf{d} \in \mathcal{D}} \sum_{\mathbf{d}' \in \mathcal{D}} \mathrm{sign}(\ell(\mathbf{d}), \ell(\mathbf{d}')) \sum_i \left(\mathbf{d}_i - \mathbf{d}'_i\right)^2 \tag{1}$$

2.2 Incompatibility with IsoScore

How does the objective in (1) conflict with isotropy requirements? Assessments of isotropy such as IsoScore generally rely on the variance vector. As we assume $\mathcal{D}$ to be PCA transformed, the covariance matrix is diagonalized, and we can obtain variance for each individual component through pairwise squared distances (Zhang et al., 2012):

$$\mathbb{V}(\mathcal{D})_i = \frac{1}{2|\mathcal{D}|^2} \sum_{\mathbf{d} \in \mathcal{D}} \sum_{\mathbf{d}' \in \mathcal{D}} \left(\mathbf{d}_i - \mathbf{d}'_i\right)^2$$

In IsoScore, this variance vector is then normalized to the length of the $\vec{1}$ vector of all ones, before computing the distance between the two:

$$\sqrt{\sum_i \left(\frac{\|\vec{1}\|_2}{\|\mathbb{V}(\mathcal{D})\|_2} \mathbb{V}(\mathcal{D})_i - 1\right)^2}$$

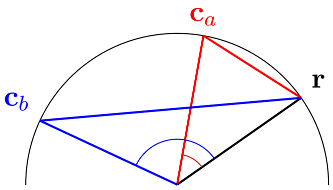


Figure 1: Relation between angle and chord.

This distance is taken as an indicator of isotropy defect, i.e., isotropic spaces will minimize it.

Given the normalization applied to the variance vector, the defect is computed as the distance between two points on a hyper-sphere. Hence it is conceptually simpler to think of this distance as an *angle* measurement: Remark that as the cosine between $\mathbb{V}(\mathcal{D})$ and $\vec{1}$ increases, the isotropy defect decreases. A diagram illustrating this relation is provided in Figure 1: For a given reference point $\mathbf{r}$ and two comparison points $\mathbf{c}_a$ and $\mathbf{c}_b$, we can observe that the shortest chord (from $\mathbf{r}$ to $\mathbf{c}_a$) also corresponds to the smallest angle.

More formally, let $\tilde{\mathbf{v}} = \frac{\|\vec{1}\|_2}{\|\mathbb{V}(\mathcal{D})\|_2} \mathbb{V}(\mathcal{D})$ be the renormalized observed variance vector. We can note that both $\tilde{\mathbf{v}}$ and the ideal variance vector $\vec{1}$ are points on the hyper-sphere centered at the origin and of radius $\|\vec{1}\|_2$. As such, the defect is then equal to the distance between two points on a circle, i.e., the length of the chord between the renormalized observed variance vector and the ideal variance vector—which can be computed by simple trigonometry means, as $2\|\vec{1}\|_2 \sin\left(\alpha/2\right)$, with α the angle between $\tilde{\mathbf{v}}$ and $\vec{1}$. This can be converted to the more familiar cosine by applying a trigonometry identity (given that $0 \leq \alpha \leq \pi/4$):

$$\begin{aligned} \|\tilde{\mathbf{v}} - \vec{1}\|_2 &= 2\|\vec{1}\|_2 \sqrt{1 - \cos^2(\alpha/2)} \\ \frac{1}{4d}\|\tilde{\mathbf{v}} - \vec{1}\|_2^2 - 1 &= -\cos^2(\alpha/2) \end{aligned}$$

where d is the dimension of the vectors in our point cloud. Hence we can exactly relate the isotropic defect (squared) to the cosine (squared) of the angle between ideal and observed variance vectors.

By monotonicity arguments, we can simplify this as follows: To maximize isotropy, we have to maximize the objective $\mathcal{O}_\mathrm{I}$

$$\begin{aligned} \mathcal{O}_\mathrm{I} &= \cos\left(\vec{1}, \mathbb{V}\left(\mathcal{D}\right)\right) \\ &\propto \sum_{\mathbf{d} \in \mathcal{D}} \sum_{\mathbf{d}' \in \mathcal{D}} \sum_i \left(\mathbf{d}_i - \mathbf{d}'_i\right)^2 \end{aligned} \tag{2}$$

This intuitively makes sense: Ignoring vector norms, we have to maximize all distances between every pair of data-points to ensure all dimensions are used equally, i.e., spread data-points out evenly on a hyper-sphere. However, in the general case, it is not possible to maximize both the isotropy objective in (2) and the silhouette score objective in (1): Intra-cluster pairwise distances must be minimized for optimal silhouette scores, but must be maximized for optimal isotropy scores. In fact, the two objectives can only be jointly maximized in the degenerate case where no two data-points in $\mathcal{D}$ are assigned the same label.[1]

2.3 Relation to linear classifiers

Informally, latent representations need to form clusters corresponding to the labels in order to optimize a linear classification objective. Consider that in classification problems (i) any data-point $\mathbf{d}$ is to be associated with a particular label $\ell(\mathbf{d}) = \omega_i$ and dissociated from other labels $\Omega \setminus \{\ell(\mathbf{d})\}$, and (ii) association scores are computed using a dot product between the latent representation to be classified and the output projection matrix, where each column vector $\mathbf{c}^\omega$ corresponds to a different class label ω. As such, for any point $\mathbf{d}$ to be associated with its label $\ell(\mathbf{d})$, one has to maximize

$$\langle \mathbf{d}, \mathbf{c}^{\ell(\mathbf{d})} \rangle = \tfrac{1}{2} \left(\|\mathbf{d}\|_2^2 + \|\mathbf{c}^{\ell(\mathbf{d})}\|_2^2 - \|\mathbf{d} - \mathbf{c}^{\ell(\mathbf{d})}\|_2^2 \right)$$

In other words, one must either augment the norm of $\mathbf{d}$ or $\mathbf{c}^{\ell(\mathbf{d})}$, or minimize the distance between $\mathbf{d}$ and $\mathbf{c}^{\ell(\mathbf{d})}$. Note however that this does not factor in the other classes $\omega' \in \Omega \setminus \{\ell(\mathbf{d})\}$ from which $\mathbf{d}$ should be dissociated, i.e., where we must minimize the above quantity. To account for the other classes, the global objective $\mathcal{O}_\mathrm{C}$ to maximize can be defined as

$$\begin{aligned} \mathcal{O}_\mathrm{C} &= - \sum_{\mathbf{d} \in \mathcal{D}} \sum_{\omega \in \Omega} \operatorname{sign}\left(\omega, \ell\left(\mathbf{d}\right)\right) \langle \mathbf{d}, \mathbf{c}^\omega \rangle \\ &= - \sum_{\mathbf{d} \in \mathcal{D}} \frac{|\Omega| - 2}{2} \|\mathbf{d}\|_2^2 - \sum_{\omega \in \Omega} \frac{|\mathcal{D}| - 2|\mathcal{D}_\omega|}{2} \|\mathbf{c}^\omega\|_2^2 \\ &\quad + \frac{1}{2} \sum_{\mathbf{d} \in \mathcal{D}} \sum_{\omega \in \Omega} \operatorname{sign}\left(\omega, \ell\left(\mathbf{d}\right)\right) \sum_i \left(\mathbf{d}_i - \mathbf{c}_i^\omega\right)^2 \end{aligned} \tag{3}$$

where the weights $|\Omega| - 2$ and $|\mathcal{D}| - 2|\mathcal{D}_\omega|$ stem from counting how many other vectors a given data or class vector is associated with or dissociated from: we have one label to associate with any data-point $\mathbf{d}$, and $|\Omega| - 1$ to dissociate it from; whereas a class vector $\mathbf{c}^\omega$ should be associated with the corresponding subset $\mathcal{D}_\omega$ and dissociated from the rest of the dataset (viz. $\mathcal{D} \setminus \mathcal{D}_\omega$).[2]

Focusing on the last line of Equation (3), we find that maximizing classification objectives entails minimizing the distance between a latent representation $\mathbf{d}$ and the vector for its label $\mathbf{c}^{\ell(d)}$, and maximizing its distance to all other class vectors. It is reminiscent of the silhouette score in Equation (1): In particular any optimum for $\mathcal{O}_\mathrm{C}$ is an optimum for $\mathcal{O}_\mathrm{S}$, since it entails $\mathcal{D}^*$ such that

$$\forall \mathbf{d}, \mathbf{d}' \in \mathcal{D}^* \quad \ell(\mathbf{d}) = \ell(\mathbf{d}') \iff \mathbf{d} = \mathbf{d}' \tag{4}$$

Informally: The cluster associated with a label should collapse to a single point. Therefore the isotropic objective $\mathcal{O}_\mathrm{I}$ in Equation (2) is equally incompatible with the learning objective $\mathcal{O}_\mathrm{C}$ of a linear classifier.

In summary, (i) point clouds cannot both contain well-defined clusters and be isotropic; and (ii) linear classifiers should yield clustered and thereby anisotropic representations.

3 Empirical confirmation

To verify the validity of our demonstrations in Section 2, we can optimize a set of data-points for a classification task using a linear classifier: We should observe an increase in silhouette scores, and a decrease in IsoScore. Note that we are therefore evaluating the behavior of parameters as they are optimized; i.e., we do not intend to test whether silhouettes and IsoScore behave as expected on held-out data. This both allows us to precisely test the argument laid out in Section 2 and cuts down computational costs significantly.

3.1 Methodology

We consider four setups: (i) optimizing SBERT sentence embeddings (Reimers and Gurevych, 2019)[3] on the binary polarity dataset of Pang and Lee (2004); (ii) optimizing paired SBERT embeddings[3] on the validation split of SNLI (Bowman et al., 2015); (iii) optimizing word2vec embeddings[4] on

[1] Hence some NLP applications and tasks need not be impeded by isotropy constrains, e.g., linear analogies that rely on vector offsets are *a prima facie* compatible with isotropy.

[2] The corresponding two sums can be understood as probabilistic priors over the data: The objective entails that the norm of a class vector $\mathbf{c}^\omega$ should be proportional to the number of data-points with this label ω, whereas one would expect a uniform distribution for vectors $\mathbf{d}$. These terms cancel out for balanced, binary classification tasks.

[3] `all-MiniLM-L6-v2`

[4] `http://vectors.nlpl.eu/repository/`, model 222, trained on an English Wikipedia dump of November 2021.

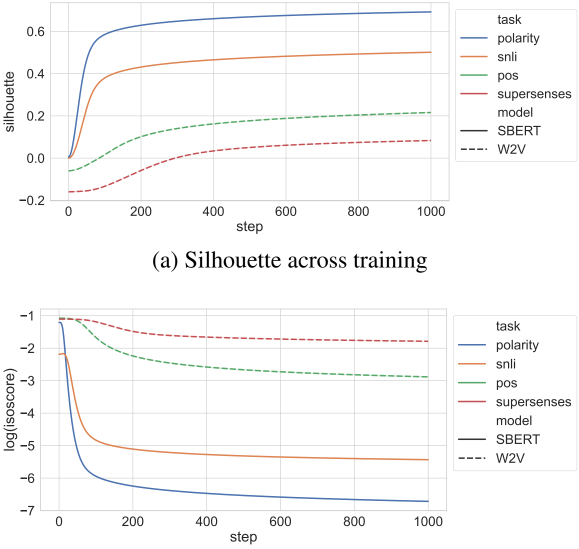
Dataset	N. items	N. params.
Pang and Lee (2004) through `nltk` (Bird and Loper, 2004)	10 662	4 094 976
Bowman et al. (2015) from `nlp.stanford.edu`	9 842	4 987 395
Mickus et al. (2022b) from `codwoe.atilf.fr`	11 462	4 341 004
Fellbaum (1998) from `github.com/altsoph`	2 275	690 326

Table 1: Dataset vs. number of datapoints (N. items) and corresponding number of trainable parameters (N. params.).

POS-tagging multi-label classification using the English CoDWoE dataset (Mickus et al., 2022b); and (iv) optimizing word2vec embeddings[4] for WordNet supersenses multi-label classification (Fellbaum, 1998; pre-processed by Tikhonov et al., 2023). All these datasets and models are in English and CC-BY or CC-BY-SA.[5] Relevant information is available in Table 1; remark we do not split the data as we are interested on optimization behavior. We also replicate and extend these experiments on GLUE in Appendix A.

For (i) and (ii), we directly optimize the output embeddings of the SBERT model rather than update the parameters of the SBERT model. In all cases, we compute gradients for the entire dataset, and compute silhouette scores with respect to the target labels and IsoScore over 1000 updates. In multi-label cases (iii) and (iv), we consider distinct label vectors as distinct target assignments when computing silhouette scores. Models are trained using the Adam algorithm (Kingma and Ba, 2014);[6] in cases (i) and (ii) we optimize cross-entropy, in cases (iii) and (iv), binary cross-entropy per label. Remark that setups (ii), (iii) and (iv) subtly depart from the strict requirements laid out in Section 2.

Training per model requires between 10 minutes and 1 hour on an RTX3080 GPU; much of which is in fact devoted to CPU computations for IsoScore and silhouette scores values. Hyperparameters listed correspond to default PyTorch values (Paszke et al., 2019), no hyperparameter search was carried out. IsoScore is computed with the pip package `IsoScore` (Rudman et al., 2022) on unpaired embeddings, silhouette scores with `scikit-learn` (Pedregosa et al., 2011).



(a) Silhouette across training

(b) Log-normalized IsoScore across training

Figure 2: Evolution of silhouette score and IsoScore across classification optimization (avg. of 5 runs).

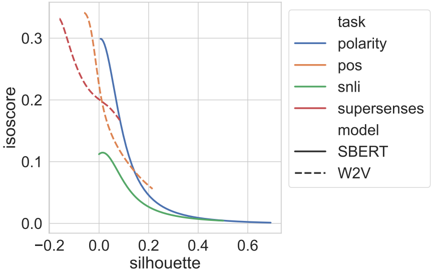


Figure 3: Relationship between silhouette scores and IsoScore (avg. of 5 runs).

3.2 Results

Results of this empirical study are displayed in Section 3.1. Performances with five different random initialization reveal negligible standard deviations (maximum at any step < 0.0054, on average < 0.0008). Our demonstration is validated: Across training to optimize classification tasks, the datapoints become less isotropic and better clustered. We can also see a monotonically decreasing relationship between IsoScore and silhouette scores, which is better exemplified in Figure 3: We find correlations with Pearson's r of -0.808 for the polarity task, -0.878 for SNLI, -0.947 for POS-tagging and -0.978 for supersense tagging; Spearman's ρ are always below -0.998.

In summary, we empirically confirm that isotropy requirements conflict with silhouette scores and linear classification objectives.

[5] Our use is consistent with the intended use of these resources. We trust the original creators of these resources that they contain no personally identifying data.

[6] Learning rate of 0.001, β of $(0.9, 0.999)$.

4 Related works

How does the connection between clusterability and isotropy that we outlined shed light on the growing literature on anisotropy?

While there is currently more evidence in favor of enforcing isotropy in embeddings, the case is not so clear cut that we can discard negative findings, and a vast majority of the positive evidence relies on improper techniques for quantifying isotropy (Rudman et al., 2022). Ethayarajh (2019) stressed that contextual embeddings are effective yet anisotropic. Ding et al. (2022) provides experiments that advise against using isotropy calibration on transformers to enhance performance in specific tasks. Rudman and Eickhoff (2023) finds that anisotropy regularization in fine-tuning appears to be beneficial on a large array of tasks. Lastly, Rajaee and Pilehvar (2021a) find that the contrasts encoded in dominant dimensions can, at times, capture linguistic knowledge.

On the other hand, the original study of Mu and Viswanath (2018) found that enforcing isotropy on static embeddings improved performances on semantic similarity, both at the word and sentence level, as well as word analogy. Subsequently, a large section of the literature has focused on this handful of tasks (e.g., Liang et al., 2021; Timkey and van Schijndel, 2021). Isotropy was also found to be helpful beyond these similarity tasks: Haemmerl et al. (2023) report that isotropic spaces perform much better on cross-lingual tasks, and Jung et al. (2023) stress its benefits for dense retrieval.

These are all applications that require graded ranking judgments, and therefore are generally hindered by the presence of clusters—such clusters would for instance introduce large discontinuities in cosine similarity scores. To take Haemmerl et al. (2023) as an example, note that language-specific clusters are antithetical to the success of cross-lingual transfer applications. It stands to reason that isotropy can be found beneficial in such cases, although the exact experimental setup will necessarily dictate whether it is boon or bane: For instance Rajaee and Pilehvar (2021b) tested fine-tuning LLMs as Siamese networks to optimize performance on sentence-level similarity, and found enforcing isotropy to hurt performances—here, we can conjecture that learning to assign inputs to specific clusters is a viable solution in their case.

The literature has previously addressed the topic of isotropy and clustering. Rajaee and Pilehvar (2021a) advocated for enhancing the isotropy on a cluster-level rather than on a global-level. Cai et al. (2021) confirmed the presence of clusters in the embedding space with local isotropy properties. Ait-Saada and Nadif (2023) investigated the correlation between isotropy and clustering tasks and found that fostering high anisotropy yields high-quality clustering representations. The study presented here provides a mathematical explanation for these empirical findings.

5 Conclusion

We argued that isotropy and cluster structures are antithetical (Section 2), verified that this argument holds on real data (Section 3), and used it to shed light on earlier results (Section 4). This result however opens novel and interesting directions of research: If anisotropic spaces implicitly entail cluster structures, then what is the structure we observe in our modern, highly anisotropic large language models? Prior results suggest that this structure is in part linguistic in nature (Rajaee and Pilehvar, 2021a), but further confirmation is required.

Another topic we intend to pursue in future work concerns the relation between non-classification tasks and isotropy: Isotropy constraints have been found to be useful in problems that are not well modeled by linear classification, e.g. word analogy or sentence similarity. Our present work does not yet offer a thorough theoretical explanation why.

Acknowledgments

This work is part of the FoTran project, funded by the European Research Council (ERC) under the EU's Horizon 2020 research and innovation program (agreement № 771113). We also thank the CSC-IT Center for Science Ltd., for computational resources.

Limitations

The present paper leaves a number of important problems open.

Idealized conditions. Our discussion in Section 2 points out optima that are incompatible, but says nothing of the behavior of models trained until convergence on held out data. In fact, enforcing isotropy could be argued to be a reasonable regularization strategy in that it would lead latent representations to not be tied to a specific classification structure.

Relatedly, a natural point of criticism to raise is whether our reasoning will hold for deep classifiers with non-linearities: Most (if not all) modern deep-learning classification approaches rely on non-linear activation functions across multiple layers of computations. The present demonstration has indeed yet to be expanded to account for such more common cases.

Insofar neural architectures trained on classification objectives are concerned, we strongly conjecture their output embeddings would tend to be anisotropic. The anisotropy of inner representations appears to be a more delicate question: For Transformers, there has been extensive work showcasing that their structure is for the most part additive (Ferrando et al., 2022a,b; Modarressi et al., 2022; Mickus et al., 2022a; Oh and Schuler, 2023; Yang et al., 2023; Mickus and Vázquez, 2023), and we therefore expect anisotropy to spread to bottom layers to some extent. For architectures based on warping random distributions such as normalizing flows (Kobyzev et al., 2021), GANs (Goodfellow et al., 2014), or diffusion models (Ho et al., 2020), the fact that (part of) their input is random and isotropic likely limits how anisotropic their inner representations are.

Thoroughness of the mathematical framework. The mathematical formalism is not thorough. For the sake of clarity and given page limitations, we do not include a formal demonstration that the linear classification optimum necessarily satisfies the clustering objective. Likewise, when discussing isotropy in Equation (2), we ignore the cosine denominator.

Choice of objectives. Our focus on silhouette scores and linear classifier objectives may seem somewhat restrictive. Our use of the silhouette score in the present derivation is motivated by two facts. First, our interest is in how the point cloud will cluster along the provided labels—this rules out any external evaluation metric comparing predicted and gold label, such as ARI (Hubert and Arabie, 1985) or purity scores. Second, we can also connect silhouette scores to a broader family of clustering metrics such as the Dunn index (Dunn, 1974), the Caliński–Harabasz index (Caliński and Harabasz, 1974) or the Davies–Bouldin index (Davies and Bouldin, 1979). Silhouette scores have the added benefit of not relying on centroids in their formulation, making their relation to the variance vector $\mathbb{V}(\mathcal{D})$ more immediate. We conjecture that these other criteria could be accounted for by means of triangular inequalities, as they imply the same optimum layout $\mathcal{D}^*$ as Equation (4).

As for our focus on the linear classifier objective, we stress this objective is a straightforward default approach; but see Appendix B for a discussion of triplet loss within a similar framework as sketched here.

References

Mira Ait-Saada and Mohamed Nadif. 2023. Is anisotropy truly harmful? a case study on text clustering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1194–1203, Toronto, Canada. Association for Computational Linguistics.

Steven Bird and Edward Loper. 2004. NLTK: The natural language toolkit. In *Proceedings of the ACL Interactive Poster and Demonstration Sessions*, pages 214–217, Barcelona, Spain. Association for Computational Linguistics.

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

Xingyu Cai, Jiaji Huang, Yuchen Bian, and Kenneth Church. 2021. Isotropy in the contextual embedding space: Clusters and manifolds. In *International Conference on Learning Representations*.

Tadeusz Caliński and Jerzy Harabasz. 1974. A dendrite method for cluster analysis. *Communications in Statistics*, 3(1):1–27.

David L. Davies and Donald W. Bouldin. 1979. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-1(2):224–227.

Yue Ding, Karolis Martinkus, Damian Pascual, Simon Clematide, and Roger Wattenhofer. 2022. On isotropy calibration of transformer models. In *Proceedings of the Third Workshop on Insights from Negative Results in NLP*, pages 1–9, Dublin, Ireland. Association for Computational Linguistics.

J. C. Dunn. 1974. Well-separated clusters and optimal fuzzy partitions. *Journal of Cybernetics*, 4(1):95–104.

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.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. Bradford Books.

Javier Ferrando, Gerard I. Gállego, Belen Alastruey, Carlos Escolano, and Marta R. Costa-jussà. 2022a. Towards opening the black box of neural machine translation: Source and target interpretations of the transformer. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8756–8769, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Javier Ferrando, Gerard I. Gállego, and Marta R. Costa-jussà. 2022b. Measuring the mixing of contextual information in the transformer. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8698–8714, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc.

Katharina Haemmerl, Alina Fastowski, Jindřich Libovický, and Alexander Fraser. 2023. Exploring anisotropy and outliers in multilingual language models for cross-lingual semantic sentence similarity. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 7023–7037, Toronto, Canada. Association for Computational Linguistics.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851. Curran Associates, Inc.

Lawrence Hubert and Phipps Arabie. 1985. Comparing partitions. *Journal of Classification*, 2(1):193–218.

Euna Jung, Jungwon Park, Jaekeol Choi, Sungyoon Kim, and Wonjong Rhee. 2023. Isotropic representation can improve dense retrieval. In *Advances in Knowledge Discovery and Data Mining*, pages 125–137, Cham. Springer Nature Switzerland.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization.

Ivan Kobyzev, Simon J.D. Prince, and Marcus A. Brubaker. 2021. Normalizing flows: An introduction and review of current methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(11):3964–3979.

Yuxin Liang, Rui Cao, Jie Zheng, Jie Ren, and Ling Gao. 2021. Learning to remove: Towards isotropic pre-trained BERT embedding. In *Artificial Neural Networks and Machine Learning – ICANN 2021*, pages 448–459, Cham. Springer International Publishing.

Timothee Mickus, Denis Paperno, and Mathieu Constant. 2022a. How to dissect a Muppet: The structure of transformer embedding spaces. *Transactions of the Association for Computational Linguistics*, 10:981–996.

Timothee Mickus, Kees Van Deemter, Mathieu Constant, and Denis Paperno. 2022b. Semeval-2022 task 1: CODWOE – comparing dictionaries and word embeddings. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 1–14, Seattle, United States. Association for Computational Linguistics.

Timothee Mickus and Raúl Vázquez. 2023. Why bother with geometry? on the relevance of linear decompositions of transformer embeddings. In *Proceedings of the 6th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pages 127–141, Singapore. Association for Computational Linguistics.

Ali Modarressi, Mohsen Fayyaz, Yadollah Yaghoobzadeh, and Mohammad Taher Pilehvar. 2022. GlobEnc: Quantifying global token attribution by incorporating the whole encoder layer in transformers. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 258–271, Seattle, United States. Association for Computational Linguistics.

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

Byung-Doh Oh and William Schuler. 2023. Token-wise decomposition of autoregressive language model hidden states for analyzing model predictions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10105–10117, Toronto, Canada. Association for Computational Linguistics.

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

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: an imperative style, high-performance deep learning library. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, Red Hook, NY, USA. Curran Associates Inc.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Sara Rajaee and Mohammad Taher Pilehvar. 2021a. A cluster-based approach for improving isotropy in contextual embedding space. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 575–584, Online. Association for Computational Linguistics.

Sara Rajaee and Mohammad Taher Pilehvar. 2021b. 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. 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.

Peter J. Rousseeuw. 1987. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65.

William Rudman and Carsten Eickhoff. 2023. Stable anisotropic regularization.

William Rudman, Nate Gillman, Taylor Rayne, and Carsten Eickhoff. 2022. IsoScore: Measuring the uniformity of embedding space utilization. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3325–3339, Dublin, Ireland. Association for Computational Linguistics.

Alexey Tikhonov, Lisa Bylinina, and Denis Paperno. 2023. Leverage points in modality shifts: Comparing language-only and multimodal word representations. In *Proceedings of the 12th Joint Conference on Lexical and Computational Semantics (*SEM 2023)*, pages 11–17, Toronto, Canada. Association for Computational Linguistics.

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

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

Sen Yang, Shujian Huang, Wei Zou, Jianbing Zhang, Xinyu Dai, and Jiajun Chen. 2023. Local interpretation of transformer based on linear decomposition. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10270–10287, Toronto, Canada. Association for Computational Linguistics.

Yuli Zhang, Huaiyu Wu, and Lei Cheng. 2012. Some new deformation formulas about variance and covariance. In *2012 Proceedings of International Conference on Modelling, Identification and Control*, pages 987–992.

A Supplementary experiments on GLUE

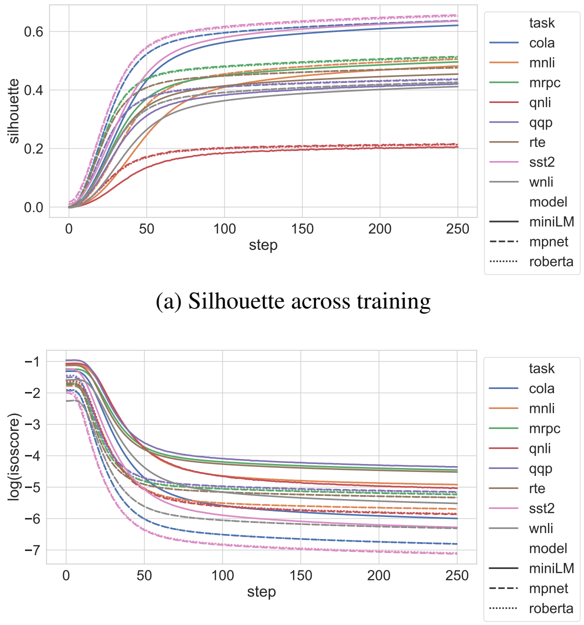
We reproduce experiments described in Section 3 on GLUE tasks (Wang et al., 2018).[7] We train our models on the provided training sets—hence we only consider tasks for which there is a training set (all but `ax`) and that correspond to a classification problem (all but `stsb`, a regression task); we remove all datapoints where no label is provided. Given our earlier results, we limit training to 250 updates; we directly update sentence-bert output embeddings by computing gradients for the entire training set all at once. We compute IsoScore and silhouette scores after every update; to alleviate computational costs, they are evaluated on random samples of $20,000$ items whenever the training set is larger than this (samples are performed separately for each update). We test three different publicly available pretrained SBERT models: `all-mpnet-base-v2` (referred to as “`mpnet`” in what follows), `all-distilroberta-v1` (viz. “`roberta`”) and `all-MiniLM-L6-v2` (viz. “`miniLM`”). Training details otherwise match those of Section 3; see Table 2 for further information on the number of datapoints and parameter counts of all models considered.

Corresponding results are depicted in Figure 4. While there is some variation across models and

[7] From `huggingface.co`.

Dataset	N. items	N. params.		
		miniLM	mpnet	roberta
cola	8 551	3 277 058	6 554 114	6 554 114
mnli	392 702	199 380 483	398 760 963	398 760 963
mrpc	3 668	2 709 506	5 419 010	5 419 010
qnli	104 743	42 617 090	85 234 178	85 234 178
qqp	363 846	189 649 154	379 298 306	379 298 306
rte	2 490	1 738 370	3 476 738	3 476 738
sst2	67 349	25 720 322	51 440 642	51 440 642
wnli	635	356 738	713 474	713 474

Table 2: Supplementary experiments on GLUE: Dataset vs. number of datapoints (N. items) and corresponding number of trainable parameters (N. params.).



(a) Silhouette across training

(b) Log-normalized IsoScore across training

Figure 4: Supplementary experiments on GLUE: Evolution of silhouette score and IsoScore across classification optimization (avg. of 5 runs).

GLUE tasks, all the setups considered display the same trend: Silhouette score increases and IsoScore decreases across training. We can quantify this trend by computing correlation scores between IsoScore and silhouette scores. Corresponding correlations are listed in Table 3: As is obvious, we find consistent and pronounced anti-correlations in all setups, with Pearson's r always below -0.838 and Spearman's ρ always below -0.966. This further consolidates our earlier conclusions in Section 3.

B Relation to triplet loss

To underscore some of the limitations of our approach, we can highlight a connection with the triplet loss, which is often used to learn clusters.

setup		r	ρ
miniLM	cola	−0.882 91	−0.999 96
	mnli	−0.852 17	−0.999 38
	mrpc	−0.939 73	−0.996 62
	qnli	−0.911 88	−0.985 88
	qqp	−0.928 90	−0.996 66
	rte	−0.926 48	−0.999 85
	sst2	−0.845 51	−0.999 97
	wnli	−0.896 90	−0.999 87
mpnet	cola	−0.872 99	−0.999 98
	mnli	−0.844 58	−0.999 20
	mrpc	−0.924 56	−0.999 70
	qnli	−0.905 06	−0.966 50
	qqp	−0.915 83	−0.995 04
	rte	−0.913 48	−0.999 80
	sst2	−0.838 64	−0.999 95
	wnli	−0.890 77	−0.999 94
roberta	cola	−0.871 37	−0.999 99
	mnli	−0.838 65	−0.999 20
	mrpc	−0.918 83	−0.998 49
	qnli	−0.899 18	−0.969 38
	qqp	−0.911 15	−0.994 24
	rte	−0.915 15	−0.999 41
	sst2	−0.841 03	−0.999 95
	wnli	−0.890 20	−0.999 91

Table 3: Supplementary experiments on GLUE: Correlations (Pearson's r and Spearman's ρ) of IsoScore and silhouette scores in GLUE task

It is defined for a triple of points $\mathbf{d}^a, \mathbf{d}^p, \mathbf{d}^n$ where $\ell(\mathbf{d}^a) = \ell(\mathbf{d}^p) \neq \ell(\mathbf{d}^n)$ as

$$\begin{aligned}\mathcal{L}_{apn} &= \max\left(\|\mathbf{d}^a - \mathbf{d}^p\|_2 - \|\mathbf{d}^a - \mathbf{d}^n\|_2, 0\right)\\ &= \max\left(\|\mathbf{d}^a - \mathbf{d}^p\|_2, \|\mathbf{d}^a - \mathbf{d}^n\|_2\right) - \|\mathbf{d}^a - \mathbf{d}^n\|_2\\ &\geq \|\mathbf{d}^a - \mathbf{d}^p\|_2 - \|\mathbf{d}^a - \mathbf{d}^n\|_2\\ &= \sum_{\mathbf{d}_c \in \{\mathbf{d}^p, \mathbf{d}^n\}} -\mathrm{sign}\left(\ell\left(\mathbf{d}^a\right), \ell\left(\mathbf{d}_c\right)\right) \|\mathbf{d}^a - \mathbf{d}_c\|_2\end{aligned}$$

The objective across the entire dataset $\mathcal{D}$ is thus:

$$\begin{aligned}\mathcal{O}_{\mathrm{T}} &= \sum_{\omega\in\Omega} \sum_{\mathbf{d}^a\in\mathcal{D}_\omega} \sum_{\mathbf{d}^p\in\mathcal{D}_\omega\setminus\{\mathbf{d}^a\}} \sum_{\mathbf{d}^n\in\mathcal{D}\setminus\mathcal{D}_\omega} -\mathcal{L}_{apn}\\ &\leq \sum_{\omega\in\Omega} \sum_{\mathbf{d}^a\in\mathcal{D}_\omega} \sum_{\mathbf{d}^p\in\mathcal{D}_\omega\setminus\{\mathbf{d}^a\}} \sum_{\mathbf{d}^n\in\mathcal{D}\setminus\mathcal{D}_\omega}\\ &\qquad \sum_{\mathbf{d}_c\in\{\mathbf{d}^p,\mathbf{d}^n\}} \mathrm{sign}\left(\ell\left(\mathbf{d}^a\right), \ell\left(\mathbf{d}^c\right)\right) \|\mathbf{d}^a - \mathbf{d}^c\|_2\\ &= \sum_{\mathbf{d}\in\mathcal{D}} \sum_{\mathbf{d}'\in\mathcal{D}} \mathrm{sign}_{\mathrm{wgt}}\left(\ell\left(\mathbf{d}\right), \ell\left(\mathbf{d}'\right)\right) \|\mathbf{d} - \mathbf{d}'\|_2\end{aligned} \tag{5}$$

using a weighted variant of our original sign function:

$$\mathrm{sign}_{\mathrm{wgt}}(\omega, \omega') = \begin{cases} |\mathcal{D}_\omega| - |\mathcal{D}| & \text{if } \omega = \omega' \\ |\mathcal{D}_\omega| - 1 & \text{otherwise} \end{cases}$$

Remark that this is in fact an upper bound on both the silhouette objective as defined in Equation (1) and the triplet objective $\mathcal{O}_{\mathrm{T}}$. However, as they are to be maximized, the above does not entail that models trained with a triplet loss will necessarily develop anisotropic representations.