A Reproduction Study: The Kernel PCA Interpretation of Self-Attention Fails Under Scrutiny

Karahan Sarıtaş
University of Tübingen
karahan.saritas@student.uni-tuebingen.de

Çağatay Yıldız
University of Tübingen
Tübingen AI Center

Abstract

In this reproduction study, we revisit recent claims that self-attention implements kernel principal component analysis (KPCA) (Teo and Nguyen, 2024), positing that (i) value vectors V capture the eigenvectors of the Gram matrix of the keys, and (ii) that self-attention projects queries onto the principal component axes of the key matrix K in a feature space. Our analysis reveals three critical inconsistencies: (1) No alignment exists between learned self-attention value vectors and what is proposed in the KPCA perspective, with average similarity metrics (optimal cosine similarity ≤ 0.32, linear CKA (Centered Kernel Alignment) ≤ 0.11, kernel CKA ≤ 0.32) indicating negligible correspondence; (2) Reported decreases in reconstruction loss J_{proj}, arguably justifying the claim that the self-attention minimizes the projection error of KPCA, are misinterpreted, as the quantities involved differ by orders of magnitude ($\sim 10^3$); (3) Gram matrix eigenvalue statistics, introduced to justify that V captures the eigenvector of the gram matrix, are irreproducible without undocumented implementation-specific adjustments. Across 10 transformer architectures, we conclude that the KPCA interpretation of self-attention lacks empirical support.

1 Introduction

Transformers (Vaswani et al., 2023; Dehghani et al., 2019) dominate tasks spanning computer vision (Dosovitskiy et al., 2021; Liu et al., 2021; Caron et al., 2021; Esser et al., 2021; Parmar et al., 2018), natural language processing (Devlin et al., 2019; Brown et al., 2020; Raffel et al., 2023), and beyond (Chen et al., 2021; Huang et al., 2018; Schwaller et al., 2019). At their core lies the attention mechanism, which recent works reinterpret through kernel methods (Tsai et al., 2019; Choromanski et al., 2022; Chen et al., 2023; Teo and Nguyen, 2024; Chowdhury et al., 2022). This perspective bridges transformers with classical kernel techniques, leveraging their interpretability (Ponte and Melko, 2017) and computational efficiency via the kernel trick (Vankadara and Ghoshdastidar, 2019).

Recent work by Teo and Nguyen (2024) reframes self-attention through the lens of kernel principal component analysis (KPCA), proposing that self-attention implicitly projects query vectors onto the principal component axes of the key matrix in a feature space. The authors further assert that the value matrix V converges to encode the eigenvectors of the Gram matrix formed by the key vectors. While theoretical proofs for such convergence under stochastic gradient descent training remain challenging due to non-convex optimization dynamics, they provide empirical justifications for their claims. This theory, *if empirically validated*, offers significant potential to enhance the interpretability and efficiency of state-of-the-art methods in Computer Vision, NLP, and related domains. By reducing the quadratic complexity of transformers through scalable kernel methods (Choromanski et al., 2022), it can unlock practical improvements in resource-intensive applications.

In this reproduction study, we empirically validate the core claims of the KPCA interpretation proposed by Teo and Nguyen (2024) Our findings challenge the validity of the KPCA analogy, revealing inconsistencies the empirical justifications proposed that question the robustness of the original claims. Specifically, we evaluate (1) the correspondence between attention-learned value vectors and the KPCA correspondence, (2) reconstruction loss and its true interpretation, and (3) the eigenvalue justification of proposed KPCA framework. Further analysis indicates that key visualizations in the prior work relied on misleading log-scale representations and non-reproducible inconsistent results, suggesting their conclusions may not hold under rigorous empirical scrutiny.

2 A Quick Overview: Kernel PCA Analysis of Attention

Self-Attention: For input $X \in \mathbb{R}^{N\times d}$ (sequence length N, embedding dim. d), compute:

$$Q = XW_Q^\top, \quad K = XW_K^\top, \quad V = XW_V^\top \quad (1)$$

with weight matrices $W_Q, W_K \in \mathbb{R}^{d_q\times d}$, $W_V \in \mathbb{R}^{d_v\times d}$. Let $q_i := Q[i,:]$, $k_i := K[i,:]$, and $v_i := V[i,:]$ denote the query/key/value vectors for position i (row vectors). The output h_i is then:

$$h_i = \sum_{j=1}^{N} \underbrace{\sigma\left(\frac{q_i K^\top}{\sqrt{d_q}}\right)_j}_{\text{attention weight } \alpha_{ij}} v_j, \quad \sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{N} e^{z_j}} \quad (2)$$

where σ applies row-wise softmax normalization to the scaled attention score matrix $QK^\top/\sqrt{d_q}$. Output vector $h_i \in \mathbb{R}^{d_v}$ is the convex combination of value vectors v_j, weighted by α_{ij}.

Kernel PCA Derivation: Let $\{k_j\}_{j=1}^N \subset \mathbb{R}^{d_q}$ be mapped through $\varphi(k_j) := \phi(k_j)/g(k_j)$ and scaling $g(k_j) = \sum_{j'} k(k_j, k_{j'})$. Centered key features $\tilde{\varphi}(k_j) = \varphi(k_j) - \frac{1}{N}\sum_{j'} \varphi(k_{j'})$ yield covariance:

$$C = \tfrac{1}{N}\sum_j \tilde{\varphi}(k_j)\tilde{\varphi}(k_j)^\top \quad (3)$$

Eigenvectors of C are denoted by u_d with eigenvalue λ_d, which can be expressed as a weighted sum of the keys $u_d = \sum_{j=1}^N a_{dj}\tilde{\varphi}(k_j)$. Weights a_{dj} are given by $a_{dj} = \frac{1}{N\lambda_d}\tilde{\varphi}(k_j)^\top u_d$. Then the kernel is set $k(x,y) = \exp\left(x^\top y/\sqrt{d_q}\right)$ to resemble the scaled softmax attention. Projection score h_{id} (d^{th} entry of the output vector $h_i \in \mathbb{R}^{d_v}$) of query q_i onto principal component u_d yields:

$$h_{id} = \varphi(q_i)^\top u_d = \sum_{j=1}^{N} \frac{k(q_i,k_j)}{g(q_i)}\dot{v}_{jd}$$

where $\dot{v}_{jd} := \frac{a_{dj}}{g(k_j)} - \frac{1}{N}\sum_{j'=1}^N \frac{a_{dj'}}{g(k_j)}$. Here comes one of the main claims of the paper, which suggests that the

Self-attention learned value vectors $v_j = W_V x_j$ converge to the KPCA term $\dot{v}_j$.

during training (see Section 2.2 in Teo and Nguyen (2024)), and therefore concluding that attention outputs are projections of the query vectors onto the principal components axes of the key matrix K in a feature space $\varphi(\cdot)$.

To determine coefficients $\{a_{dj}\}$, they define the centered Gram matrix $\tilde{K}_\varphi \in \mathbb{R}^{N\times N}$ where $\tilde{K}_\varphi(i,j) = \tilde{\varphi}(k_i)^\top\tilde{\varphi}(k_j)$, which can be calculated during the forward pass using key values. Substituting the eigenvector expansion $u_d = \sum_j a_{dj}\tilde{\varphi}(k_j)$ into $Cu_d = \lambda_d u_d$ gives:

$$\frac{1}{N}\sum_{j=1}^{N}\tilde{\varphi}(k_j)\tilde{\varphi}(k_j)^\top \sum_{j'=1}^{N} a_{dj'}\tilde{\varphi}(k_{j'}) = \lambda_d \sum_{j=1}^{N} a_{dj}\tilde{\varphi}(k_j) \quad (4)$$

Left-multiplying by $\tilde{\varphi}(k_i)^\top$ yields:

$$\tilde{K}_\varphi^2 a_d = \lambda_d N \tilde{K}_\varphi a_d \implies \tilde{K}_\varphi a_d = \lambda_d N a_d, \quad (5)$$

where $a_d = [a_{d1}, ..., a_{dN}]^\top$ are eigenvectors of $\tilde{K}_\varphi$. Defining $G := \text{diag}(\frac{1}{g(k_1)}, ..., \frac{1}{g(k_N)})$, $\mathbf{1}_N \in \mathbb{R}^{N\times N}$ consisting of $\frac{1}{N}$ in all entries, and $A := [a_1, ..., a_{d_v}] \in \mathbb{R}^{N\times d_v}$ consisting of d_v eigenvectors of the gram matrix, KPCA value matrix $\dot{V}_{\text{KPCA}} = [\dot{v}_1, ..., \dot{v}_N]^\top \in \mathbb{R}^{N\times d_v}$ can be expressed as follows:

$$\dot{V}_{\text{KPCA}} = GA - G\mathbf{1}_N A \quad (6)$$

$$\implies \hat{a}_d = (I - \mathbf{1}_N)^{-1}G^{-1}V[:,d] \quad (7)$$

Building on the hypothesis that self-attention's learned value vectors V converge to kernel PCA coefficients $\dot{V}_{\text{KPCA}}$, Teo and Nguyen (2024) assert that the value matrix encodes the eigenvectors of the Gram matrix derived from key vectors in a feature space. In Section 3, we empirically test their claims by analyzing their proposed evidence for eigenvector alignment and projection error minimization.

3 Experiments

Is self-attention learned $V \approx \dot{V}_{\text{KPCA}}$? We first assess whether attention-learned value matrices V align with theoretical kernel PCA counterparts $\dot{V}$, evaluating 10 vision transformers: 6 DeiT models (`tiny`, `small`, `base`, and their distilled variants (patch 16)) (Touvron et al., 2021) and 4 ViT variants (`tiny`/`small`/`base`/`large`) (Dosovitskiy et al., 2021), all trained on IMAGENET1K (Russakovsky et al., 2015) with image size 224×224. We analyze each attention head in each layer using a random selection of 100 images during inference. We calculate $\dot{V}_{\text{KPCA}}$ using Equation 6, where we first calculate the Gram matrix K_φ, center it, and

then extract its eigenvectors to achieve the matrix A. We use the top d_v eigenvectors of A to construct $\dot{V}_{\text{KPCA}}$.

We first compare matrix entries pairwise, checking if $|\text{input}_i - \text{other}_i| \leq 10^{-3} + 10^{-5} \times |\text{other}_i|$, using relatively higher error thresholds to avoid false negatives. Across all combinations of model $\times$ image $\times$ layer $\times$ head, we conduct 114,000 tests, none of which passes the check. As this criterion may be overly stringent, we proceed with the following relaxed approaches.

We compute cosine similarity between self-attention and KPCA value vectors. To satisfy $V \approx \dot{V}_{\text{KPCA}}$, we compare: (1) direct column-wise matches, and (2) optimal column alignment via `scipy`'s Jonker-Volgenant algorithm (Crouse, 2016) implementation using cosine distance costs to test if the hypothesis holds in the best scenario possible. Then, as a final approach to measure matrix similarity, we employ Centered Kernel Alignment (CKA) (Kornblith et al., 2019) - which was originally used to measure the similarity of neural network representations. All comparisons are conducted after normalizing vectors remove the sensitivity to vector magnitudes.

As illustrated in Table 1, all four similarity measures yield relatively low values across the examined models, failing to provide compelling evidence that $V \approx \dot{V}_{\text{KPCA}}$ at the conclusion of training. Even the most promising metric—Maximum Optimal Cosine Similarity with Jonker-Volgenant matching—reaches only 0.32 at its peak, suggesting limited alignment between the attention-learned value matrices and their theoretically proposed counterparts.

Having found no evidence that self-attention-learned V matrices converge to KPCA theoretical values, we now analyze the authors' empirical justifications for their hypothesis.

Does the decrease in J_{proj} imply convergence? We reproduce the projection error minimization plot from (Teo and Nguyen, 2024), where the error is defined as:

$$J_{\text{proj}} = \frac{1}{N} \sum_{i=1}^{N} \left\| \varphi(q_i) - \sum_{d=1}^{d_v} h_{id} u_d \right\|^2$$

While our implementation replicates the numerical results using the authors' code[1], critical discrepancies arise in implementation. The original work visualizes $\log(J_{\text{proj}})$ without explicitly stating this logarithmic scaling in their manuscript, obscuring the raw magnitude of the projection error. Furthermore, the omission of a $\sqrt{d_v}$ scaling factor for $\varphi(q_i)$ leads to inflated $\|\varphi(q_i)\|^2$ values resulting in values of e^{35} even after 300-epoch training. We train both ViT-Tiny and DeiT-Tiny on IMAGENET1K, and plot the minimization error in Figure 1 after correcting the normalization and adopting mean absolute error (see Appendix A).

[1] https://github.com/rachtsy/KPCA_code/blob/07e579a/Reconstruction/softmax.py#L73

Table 1: Similarity results between attention-learned value matrix V and proposed $\dot{V}_{\text{KPCA}}$ using the following metrics: MDC: Max Direct Cosine Similarity, MOC: Max Optimal Cosine Similarity using Jonker-Volgenant matching, LCKA: Linear CKA, KCKA: Kernel CKA

Model	**Similarity Measures**			
	MDC	MOC	LCKA	KCKA
ViT-Tiny	0.09	0.29	0.06	0.28
ViT-Small	0.11	0.30	0.05	0.27
ViT-Base	0.14	0.30	0.06	0.28
ViT-Large	0.13	0.30	0.06	0.25
DeiT-Tiny	0.15	0.31	0.11	0.31
DeiT-Small	0.11	0.31	0.08	0.28
DeiT-Base	0.12	0.32	0.10	0.29
DeiT-Tiny-D	0.11	0.31	0.11	0.32
DeiT-Small-D	0.11	0.31	0.09	0.29
DeiT-Base-D	0.11	0.32	0.10	0.28

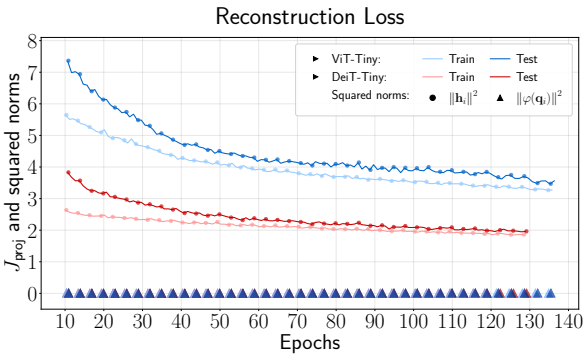

Figure 1: Reconstruction loss (J_{proj}) over training epochs for ViT-Tiny and DeiT-Tiny models, along with the values of the individual squared norms, shown with markers. Circle markers indicate average of squared output norms ($\|\mathbf{h}_i\|^2$) and triangle markers (extremely low values around 10^{-3}) show the average of squared feature map norms ($\|\varphi(\mathbf{q}_i)\|^2$).

At first, decreasing projection loss J_{proj} may seem to indicate a meaningful alignment between the quantities; however, analysis of individual squared norms reveals a more nuanced picture. As shown by the markers, $\|\varphi(q_i)\|^2$ values (around 10^{-3}) remain orders of magnitude smaller than

$\|h_i\|^2$ throughout training. In practice, the observed error reduction stems predominantly from decreasing $\|h_i\|^2$ magnitudes rather than genuine convergence between $\varphi(q_i)$ and the reconstruction. Our observations on vision transformers generalize to the language models with transformers (See Appendix A.2 for additional visualizations).

Do eigenvalues of $\tilde{K}_\varphi$ match with the reported results? The authors empirically verified the relationship $\frac{\tilde{K}_\varphi \hat{a}_d}{N\hat{a}_d} = \gamma = [\gamma_1, \ldots, \gamma_N]$, where $\gamma_1 = \cdots = \gamma_N =$ constant, which they interpret as confirmation that $\hat{a}_d$ is an eigenvector of $\tilde{K}_\varphi$ (with eigenvalue $N\gamma$).

Plots of the means and standard deviations of absolute differences $|\gamma_i - \gamma_j|$ in the vector $\mathbf{1}\lambda_d$ can be misleading, as small values may yield low differences without satisfying the eigenvalue constraint (Appendix B). Therefore we have to focus on reproducing the actual eigenvalues. The authors emphasize that the eigenvalues' magnitudes—averaged across all attention heads and layers—are substantially larger, with maximum, minimum, mean, and median values of 648.46, 4.65, 40.07, and 17.73, respectively, far exceeding $|\gamma_i - \gamma_j|$. Unfortunately, they provide no reproducible implementation for this claim. Our analysis of eigenvalues of $\tilde{K}_\varphi$ across 10 distinct transformer models demonstrates fundamental inconsistencies: the empirical eigenvalue distribution directly contradicts the reported values to justify their claims. We compute absolute eigenvalues across all attention heads and layers for each image, average them *by eigenvalue rank* (see Appendix B.1), then derive per-image statistics (max/min/mean/median) from these rank-wise averages. We report mean ± standard deviation over 25 randomly selected IMAGENET1K images.

Table 2 reveals eigenvalues of $\tilde{K}_\varphi$ on the order of 10^{-6}—orders of magnitude smaller than those reported in (Teo and Nguyen, 2024). This discrepancy not only challenges the reproducibility of their spectral analysis but also undermines the validity of the γ-difference plots to validate self-attention's convergence to KPCA value vectors.

Table 2: Eigenvalue Statistics for Vision Transformer Models ($\times 10^{-6}$)

Model	**Eigenvalue Statistics**			
	Max	Min	Mean	Median
ViT-Tiny	147 ± 11	17 ± 5	37 ± 7	30 ± 7
ViT-Small	181 ± 22	17 ± 4	36 ± 6	28 ± 5
ViT-Base	206 ± 30	15 ± 4	33 ± 6	25 ± 5
ViT-Large	177 ± 22	21 ± 5	42 ± 6	34 ± 6
DeiT-Tiny	325 ± 5	34 ± 10	65 ± 10	53 ± 11
DeiT-Small	306 ± 4	34 ± 9	66 ± 11	54 ± 11
DeiT-Base	259 ± 7	35 ± 9	64 ± 10	54 ± 10
DeiT-Tiny-D	205 ± 7	32 ± 9	61 ± 10	51 ± 10
DeiT-Small-D	224 ± 7	33 ± 9	63 ± 10	53 ± 10
DeiT-Base-D	226 ± 6	36 ± 9	67 ± 10	56 ± 10

4 Conclusion

In essence, the kernel PCA interpretation of self-attention proposed by Teo and Nguyen (2024) lacks empirical and theoretical robustness under detailed scrutiny. Our results extend to language models: the similarity between V and $\dot{V}_{\text{KPCA}}$ stays low, and the two norms diverge (see Appendix C). We emphasize that this critique neither disputes the viability of Robust PCA (RPCA) as an algorithm nor asserts that the self-attention cannot be interpreted as a projection—rather, it challenges the proposed framework's empirical and theoretical foundations. Specifically, the claim that the self-attention can be derived from kernel PCA (and therefore can be replaced with) by the proposed mechanism, is unsupported by reproducible evidence. We believe that the RPCA's improvements stem from its complementary role within the existing architecture, using the symmetric self-attention mechanism as a low-rank approximator in its Principal Component Pursuit (PCP) algorithm rather than replacing it outright. All of the experiments and analysis can be found at our Github repository[2].

The interpretation of self-attention has become a rapidly developing area, with numerous works proposing formulations from different mathematical perspectives (Chen et al., 2023; Choromanski et al., 2022; Tsai et al., 2019; Nguyen et al., 2024). However, such rapid progress risks false positives in research community. We hope our work helps researchers navigate this landscape more efficiently, focusing attention on evidence-based progress rather than superficially consistent narratives, mis-interpreted plots or undocumented, unconventional implementation practices. While the interpretation of self-attention mechanisms as projections of input, key, or query vectors remains an open research question, our empirical evidence directly refutes how this mechanism is characterized in (Teo and Nguyen, 2024).

[2] https://github.com/KarahanS/Reproduction-Study-KPCA

5 Limitations

Despite our extensive evaluation, several practical limitations should be acknowledged. First, we had to resort to a proxy reconstruction loss (similar to the original work (Teo and Nguyen, 2024) (e.g., MAE over squared norm differences) rather than an exhaustive permutation-based matching (see Appendix A.1). Secondly, the numerical instability in computing the eigenvalues of the centered Gram matrix $\tilde{K}_\varphi$ forced us to adopt pre-processing steps (Z-score normalization) that, although minimally impacting overall trends and conclusions, produces different eigenvalues. Lastly, we can compare the self-attention learned V with the KPCA counterpart $\dot{V}_{\text{KPCA}}$ through two directions: First, estimating eigenvectors $\hat{A} = (I - \mathbf{1}_N)^{-1} G^{-1} V$ to verify alignment with $\tilde{K}_\varphi$'s eigenvectors, but this approach is not feasible due to the singular centering matrix $I - \mathbf{1}_N$, which introduces numerical instability during inversion. Alternatively, we compute A directly from the eigenvectors of $\tilde{K}_\varphi$ and validate whether $GA - G\mathbf{1}_N A \approx V$ holds. Due to the numerical instability in the first method, we adopt the second approach in our analysis

References

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv:2004.05150*.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. *Preprint*, arXiv:2005.14165.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. 2021. Emerging properties in self-supervised vision transformers. *Preprint*, arXiv:2104.14294.

Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Michael Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. 2021. Decision transformer: Reinforcement learning via sequence modeling. *Preprint*, arXiv:2106.01345.

Yingyi Chen, Qinghua Tao, Francesco Tonin, and Johan A. K. Suykens. 2023. Primal-attention: Self-attention through asymmetric kernel svd in primal representation. *Preprint*, arXiv:2305.19798.

Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, David Belanger, Lucy Colwell, and Adrian Weller. 2022. Rethinking attention with performers. *Preprint*, arXiv:2009.14794.

Sankalan Pal Chowdhury, Adamos Solomou, Avinava Dubey, and Mrinmaya Sachan. 2022. On learning the transformer kernel. *Preprint*, arXiv:2110.08323.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. *Preprint*, arXiv:2003.10555.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *CoRR*, abs/1911.02116.

David F. Crouse. 2016. On implementing 2d rectangular assignment algorithms. *IEEE Transactions on Aerospace and Electronic Systems*, 52(4):1679–1696.

Mostafa Dehghani, Stephan Gouws, Oriol Vinyals, Jakob Uszkoreit, and Łukasz Kaiser. 2019. Universal transformers. *Preprint*, arXiv:1807.03819.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *Preprint*, arXiv:1810.04805.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. *Preprint*, arXiv:2010.11929.

Patrick Esser, Robin Rombach, and Björn Ommer. 2021. Taming transformers for high-resolution image synthesis. *Preprint*, arXiv:2012.09841.

Cheng-Zhi Anna Huang, Ashish Vaswani, Jakob Uszkoreit, Noam Shazeer, Ian Simon, Curtis Hawthorne, Andrew M. Dai, Matthew D. Hoffman, Monica Dinculescu, and Douglas Eck. 2018. Music transformer. *Preprint*, arXiv:1809.04281.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. 2019. Similarity of neural network representations revisited. *Preprint*, arXiv:1905.00414.

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

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. *Preprint*, arXiv:2103.14030.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric Villemonte de la Clergerie, Djamé Seddah, and Benoît Sagot. 2020. Camembert: a tasty french language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2016. Pointer sentinel mixture models. *Preprint*, arXiv:1609.07843.

Tan M. Nguyen, Tam Nguyen, Nhat Ho, Andrea L. Bertozzi, Richard G. Baraniuk, and Stanley J. Osher. 2024. A primal-dual framework for transformers and neural networks. *Preprint*, arXiv:2406.13781.

Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Łukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. 2018. Image transformer. *Preprint*, arXiv:1802.05751.

Pedro Ponte and Roger G. Melko. 2017. Kernel methods for interpretable machine learning of order parameters. *Physical Review B*, 96(20).

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2023. Exploring the limits of transfer learning with a unified text-to-text transformer. *Preprint*, arXiv:1910.10683.

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*. Association for Computational Linguistics.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. 2015. Imagenet large scale visual recognition challenge. *Preprint*, arXiv:1409.0575.

Philippe Schwaller, Teodoro Laino, Théophile Gaudin, Peter Bolgar, Christopher A. Hunter, Costas Bekas, and Alpha A. Lee. 2019. Molecular transformer: A model for uncertainty-calibrated chemical reaction prediction. *ACS Central Science*, 5(9):1572–1583.

Johan A.K. Suykens. 2016. Svd revisited: A new variational principle, compatible feature maps and nonlinear extensions. *Applied and Computational Harmonic Analysis*, 40(3):600–609.

Rachel S. Y. Teo and Tan M. Nguyen. 2024. Unveiling the hidden structure of self-attention via kernel principal component analysis. *Preprint*, arXiv:2406.13762.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. 2021. Training data-efficient image transformers & distillation through attention. *Preprint*, arXiv:2012.12877.

Yao-Hung Hubert Tsai, Shaojie Bai, Makoto Yamada, Louis-Philippe Morency, and Ruslan Salakhutdinov. 2019. Transformer dissection: A unified understanding of transformer's attention via the lens of kernel. *Preprint*, arXiv:1908.11775.

Leena Chennuru Vankadara and Debarghya Ghoshdastidar. 2019. On the optimality of kernels for high-dimensional clustering. *Preprint*, arXiv:1912.00458.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. Attention is all you need. *Preprint*, arXiv:1706.03762.

Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. 2020. Luke: Deep contextualized entity representations with entity-aware self-attention. In *EMNLP*.

Supplement to "A Reproduction Study: The Kernel PCA Interpretation of Self-Attention Fails Under Scrutiny"

A Calculation of J_{proj} and Practical Limitations

Main claim by Teo and Nguyen (2024) is that the output $h_i \in \mathbb{R}^{d_v}$ of self-attention is equivalent to the projection of query vector q_i onto the principal components of the key matrix $K \in \mathbb{R}^{N \times N}$ in a feature space $\varphi(\cdot)$. Projection scores can be expressed as $h_{id} = \varphi(q_i)^\top u_d$, where u_d is an eigenvector of the matrix C (Equation 3). If u_d is a unit eigenvector, then it is a *normalized* projection score, otherwise *unnormalized* which requires dividing by the scalar $u_d^\top u_d$ to normalize it.

To reconstruct the projected vector, we sum the projection scores along each principal component: $\hat{\varphi}(q_i) = \sum_{d=1}^{d_v} h_{id} u_d = \sum_{d=1}^{d_v} (u_d^\top \varphi(q_i)) u_d$, which gives us the reconstructed vector in the original embedding space.

If the eigenvectors are orthonormal (unit and orthogonal to each other), then the last equation reduces to the following squared norm difference (using $u_a^\top u_b = 0$ if $a \neq b$, otherwise 1): $\frac{1}{N} \sum_{i=1}^{N} \left(\|\varphi(q_i)\|^2 - \|h_i\|^2 \right)$. A very useful property of this equation is that it is an "eigenvector-invariant" computation, meaning we don't need to compute individual eigenvectors or assign them to corresponding rows of the output matrix $H \in \mathbb{R}^{N \times d_v}$. If eigenvectors aren't orthonormal, we must use the original equation for correct loss calculation. However, this introduces a technical challenge: *if the theory holds*, we do know each component h_{id} of output vector $h_i \in \mathbb{R}^{d_v}$ represents the projection score along eigenvector u_d – but we do not know which eigenvector of C corresponds to u_d. For $d_v = 64$, the combinatorial permutation alignment problem between d_v eigenvectors and components exhibits factorial computational complexity $\mathcal{O}(d_v!)$, fundamentally limiting practical verification. Due to this computational bottleneck, we used the squared norm difference (as in original work) to maintain eigenvector-invariant computation. However, $\|\varphi(q_i)\|^2 \geq \|h_i\|^2$ is not guaranteed without orthonormality, so we switched to Mean Absolute Error:

$$J_{\text{proj}} = \frac{1}{N} \sum_{i=1}^{N} \left| \|\varphi(q_i)\|^2 - \|h_i\|^2 \right|$$

A.1 Eigenvector Assignment Sensitivity in Projection Loss

With a toy example, we demonstrate that different selections of eigenvectors $\{u_d\}_{d=1}^{d_v}$ yield different projection loss values. While the first two terms in the projection loss, $\|\varphi(q_i)\|^2$ and $\|h_i\|^2$, are invariant to eigenvector selection, the critical cross-term $\sum_{m=1}^{d_v} \sum_{n=1}^{d_v} u_m^\top u_n h_{im} h_{in}$ exhibits high sensitivity to the specific assignment of eigenvectors.

Consider a minimal example where an output representation $h_i = [1, 2]$ is projected into a 2-dimensional space ($d_v = 2$). Given two eigenvectors $[1, 1]^\top$ and $[-1, 0]^\top$, cross-term can be evaluated for two different assignments. Under assignment $A_1 : u_1 = [1, 1]^\top, u_2 = [-1, 0]^\top$, we obtain $= 2 - 2 - 2 + 4 = 2$, whereas under assignment $A_2 : u_1 = [-1, 0]^\top, u_2 = [1, 1]^\top$, we obtain $1 - 2 - 2 + 8 = 5$, resulting in different loss values.

Simply permuting the assignment of identical eigenvectors can yield substantially different loss values. To compute the actual loss, would need to evaluate $d_v!$ different assignment permutations to identify the optimal configuration—rendering the approach computationally prohibitive. To avoid the excessive computation, we adopt the proxy MAE loss, which eliminates this assignment sensitivity.

A.2 Additional Details on Projection Error Minimization

We evaluate the projection error J_{proj} for ViT-Tiny and DeiT-Tiny models. Following the methodology of (Teo and Nguyen, 2024), the reconstruction loss is computed on the same batch of images coming from the training set, with results averaged across layers, attention heads, and batches to align with the original implementation.

The following plots in Figure 2 demonstrate that the theoretically calculated values of $\|\varphi(q_i)\|^2$, derived from a pretrained model, fail to align with the squared norms of the output vectors $\|h_i\|^2$ over different layers. While these measures occasionally converge to a similar scale in deeper layers, they remain distinct.

Plots in Figure 4 demonstrate that the relative error for $\|h_i\|^2$ remains near 1.0 during training, while the error for $\|\varphi(q_i)\|^2$ spans $\sim 10^6$—highlighting a stark magnitude disparity. This discrepancy underscores that the observed decrease in projection error J_{proj} does not imply convergence as initially suggested.

$$
\begin{aligned}
J_{\text{proj}} &= \frac{1}{N}\sum_{i=1}^{N}\left\|\varphi(q_i) - \sum_{d=1}^{d_v} h_{id}u_d\right\|^2 \\
&= \frac{1}{N}\sum_{i=1}^{N}\left(\varphi(q_i) - \sum_{d}^{d_v} h_{id}u_d\right)^\top \left(\varphi(q_i) - \sum_{d}^{d_v} h_{id}u_d\right) \\
&= \frac{1}{N}\sum_{i=1}^{N}\left(\varphi(q_i)^\top\varphi(q_i) - \sum_{d}^{d_v} h_{id}\varphi(q_i)^\top u_d - \sum_{d}^{d_v} h_{id}u_d^\top\varphi(q_i) + \sum_{m}^{d_v}\sum_{n}^{d_v} u_m^\top u_n h_{ia}h_{ib}\right) \\
&= \frac{1}{N}\sum_{i=1}^{N}\left(\|\varphi(q_i)\|^2 - \underbrace{\sum_{d}^{d_v} h_{id}^2}_{\|h_i\|^2} - \underbrace{\sum_{d}^{d_v} h_{id}^2}_{\|h_i\|^2} + \sum_{m}^{d_v}\sum_{n}^{d_v} u_m^\top u_n h_{ia}h_{ib}\right) \\
&= \frac{1}{N}\sum_{i=1}^{N}\left(\|\varphi(q_i)\|^2 - 2\,\|h_i\|^2 + \underbrace{\sum_{m}^{d_v}\sum_{n}^{d_v} u_m^\top u_n h_{ia}h_{ib}}_{\|h_i\|^2 \text{ if orthonormal eigenvectors}}\right)
\end{aligned}
$$

B Eigenvalue Analysis

To empirically demonstrate that visualizations resembling the original authors' results can emerge even without strict adherence to the eigenvector condition $\tilde{K}_\varphi \hat{a}_d = \lambda \hat{a}_d$, we generate a perturbed matrix A_{random} by adding standard Gaussian noise scaled by 0.1 to each entry of A, followed by QR-decomposition to re-orthogonalize its columns. In Figure 3, we show two cases where the initial type of plots can be misleading, whereas the second plots reveal the difference between them.

B.1 Eigenvalue Statistics Calculation

For each image, we compute the absolute eigenvalues of the attention mechanism for every head and layer. These eigenvalues are grouped by their *rank* (sorted position) across all heads and layers. We then compute the average eigenvalue value *for each rank position* (e.g., the mean of all 1st-largest eigenvalues, the mean of all 2nd-largest eigenvalues, etc.). From these rank-wise averages, we calculate four statistics— max, min, mean, and median—across all rank positions. Finally, we report the mean and standard deviation of these statistics over 25 randomly sampled images from IMAGENET1K.

For certain transformer architectures, direct eigenvalue computation exhibited numerical instability due to floating-point precision limitations. We resolved this by standardizing key (k) vectors (i.e., subtracting means and dividing by standard deviations per dimension) prior to covariance matrix construction. While standardization during inference risks severely degrading model performance, Table 3 reveals that its impact on the eigenvalues of the centered Gram matrix $\tilde{K}_\varphi$ is negligible. This confirms that discrepancies with (Teo and Nguyen, 2024) arise from undocumented methodological choices, not pre-processing steps.

Relative projection error J_{proj} plots reveal a fundamental flaw in the "reconstruction loss minimization" argument: During the training, $\|\varphi(q_i)\|^2$ (bottom) remains negligible ($\sim 10^{-3}$) compared to $\|h_i\|^2$ (top). This disparity confirms that decreasing J_{proj} arises not from alignment between $\varphi(q_i)$ and reconstructions, but from collapsing $\|h_i\|^2$ magnitudes. A similar inconsistency is observed for language models in Appendix C.

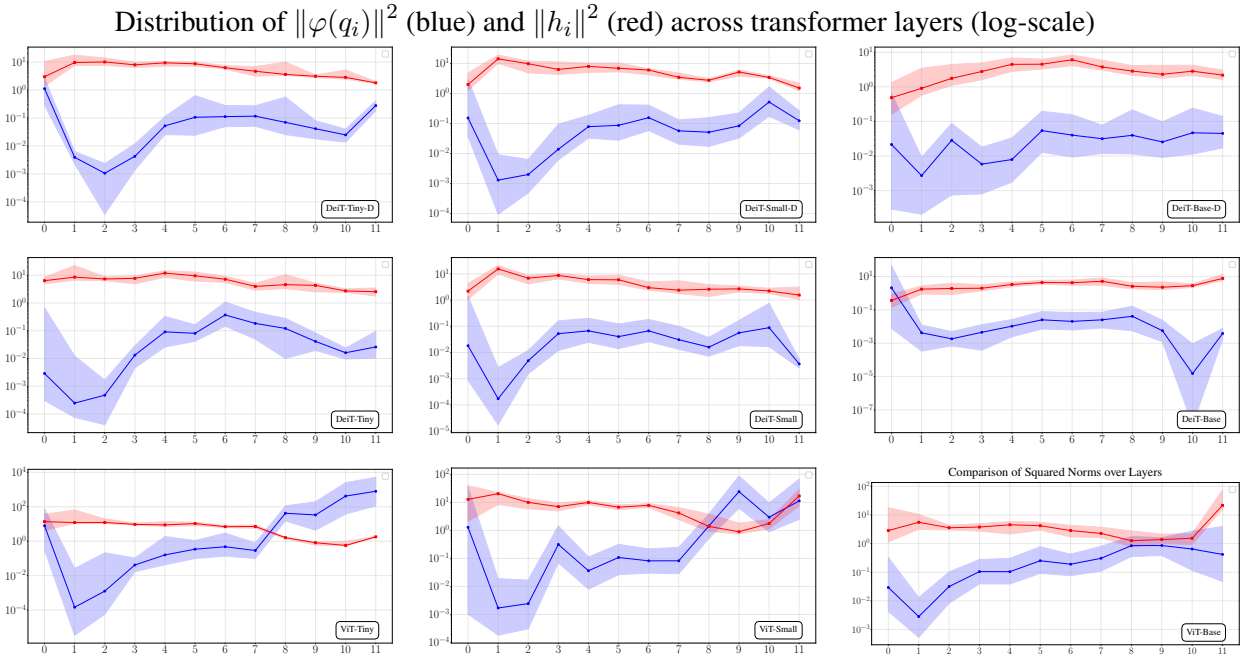

Figure 2: Comparison of squared norms across transformer layers. The plots show medians (solid lines) and 95% percentiles (shaded regions) of $\|\varphi(q_i)\|^2$ (blue) and $\|h_i\|^2$ (red) for 9 pre-trained transformer models for an input image. Values are displayed in log-scale due to the small magnitude of $\|\varphi(q_i)\|^2$. Log scaling highlights vanishing $\|\varphi(q_i)\|^2$ magnitudes. Notice the (1) high variance in $\varphi(q_i)$ projections vs. stable attention outputs, (2) no layer-wise convergence despite architectural scaling (DeiT/ViT, Tiny→Base)

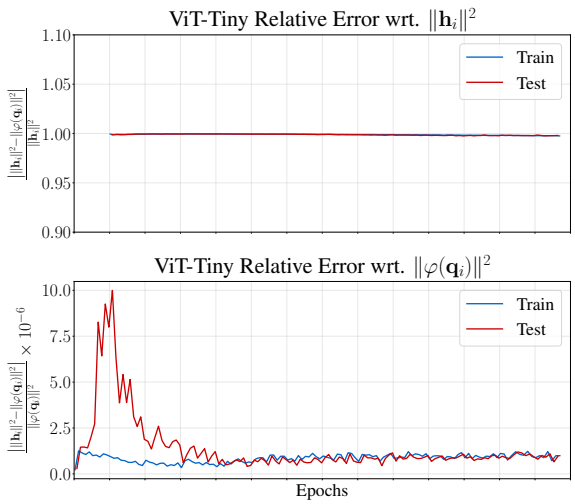

Figure 4: Relative absolute reconstruction train/test errors with respect to $\|\varphi(\mathbf{q}_i)\|^2$ and $\|h_i\|^2$ for ViT-Tiny. Errors with respect to $\|\varphi(\mathbf{q}_i)\|^2$ are in scale 10^{-6}. For clarity, the lower panel excludes the first 10 epochs to mitigate outlier effects and enhance trend visibility.

Table 3: Eigenvalue statistics with and without(*) query-key standardization ($\times 10^{-6}$)

Model	Eigenvalue Statistics			
	Max	Min	Mean	Median
ViT-Tiny	147 ± 11	17 ± 5	37 ± 7	30 ± 7
ViT-Tiny*	$178 \pm 19 \uparrow 21\%$	$4 \pm 2 \downarrow 76\%$	$16 \pm 3 \downarrow 57\%$	$9 \pm 3 \downarrow 70\%$
ViT-Large	177 ± 22	21 ± 5	42 ± 6	34 ± 6
ViT-Large*	$497 \pm 36 \uparrow 181\%$	$7 \pm 2 \downarrow 67\%$	$31 \pm 2 \downarrow 26\%$	$15 \pm 2 \downarrow 56\%$
DeiT-Tiny	325 ± 5	34 ± 10	65 ± 10	53 ± 11
DeiT-Tiny*	$1043 \pm 99 \uparrow 221\%$	$34 \pm 11 \uparrow 0\%$	$96 \pm 15 \uparrow 48\%$	$60 \pm 14 \uparrow 13\%$
DeiT-Small	306 ± 4	34 ± 9	66 ± 11	54 ± 11
DeiT-Small*	$1343 \pm 175 \uparrow 339\%$	$25 \pm 8 \downarrow 26\%$	$87 \pm 11 \uparrow 32\%$	$46 \pm 11 \downarrow 15\%$
DeiT-Tiny-D	205 ± 7	32 ± 9	61 ± 10	51 ± 10
DeiT-Tiny-D*	$796 \pm 95 \uparrow 288\%$	$27 \pm 8 \downarrow 16\%$	$78 \pm 12 \uparrow 28\%$	$49 \pm 11 \downarrow 4\%$
DeiT-Small-D	224 ± 7	33 ± 9	63 ± 10	53 ± 10
DeiT-Small-D*	$1153 \pm 151 \uparrow 415\%$	$23 \pm 7 \downarrow 30\%$	$78 \pm 10 \uparrow 24\%$	$43 \pm 10 \downarrow 19\%$

B.2 Gram Matrix Eigenvalue Equation

In this subsection, we will derive the gram matrix eigenvalue equation explicitly. We begin with the following expressions:

$$u_d = \sum_{j=1}^{N} a_{dj} \tilde{\varphi}(k_j)$$

$$\frac{1}{N} \sum_{j=1}^{N} \tilde{\varphi}(k_j) \{\tilde{\varphi}(k_j)^\top u_d\} = \lambda_d u_d$$

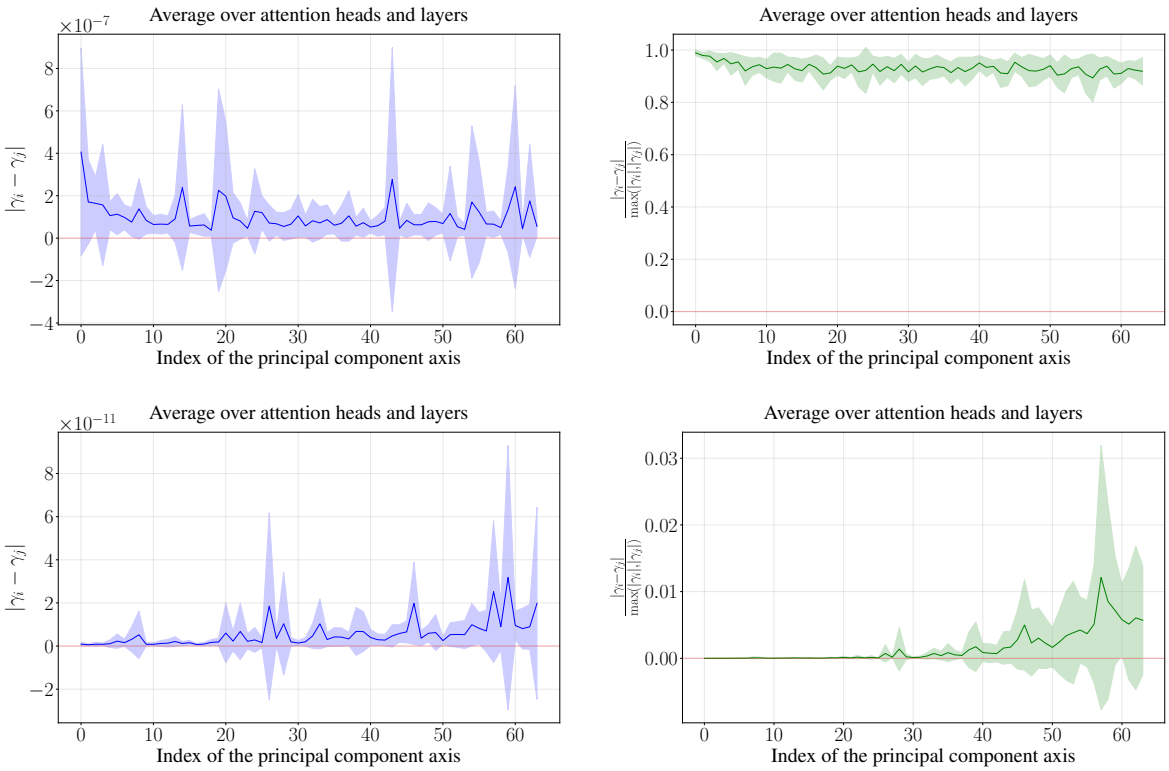

Figure 3: (ViT Tiny) Top row: Mean and standard deviation of the absolute differences of entries in the γ vector from true eigenvectors of $\tilde{K}_\varphi$ (matrix A). Bottom row: Corresponding results for random-direction eigenvectors (A_{random}) with matched row norms. Left panels initially suggest both satisfy $\frac{\tilde{K}_\varphi \hat{a}_d}{N\hat{a}_d} = \gamma = [\gamma_1, \ldots, \gamma_N]$ with $\gamma_1 = \cdots = \gamma_N = \text{const.}$; however, absolute differences reveal orders-of-magnitude deviation (10^{-7} vs. 10^{-11}). Right panels (relative error to $\max(|\gamma_i|, |\gamma_j|)$) demonstrate the condition violation more explicitly through significantly higher relative errors for A_{random}, showing small $\tilde{K}_\varphi$ eigenvalues permit visual resemblance despite failing the eigenvector criterion.

When $\tilde{K}_\varphi$ is invertible, the only solution is:

$$\tilde{K}_\varphi a_d = N\lambda_d a_d$$

which corresponds to the eigenvalue solution, where a_d are eigenvectors of $\tilde{K}_\varphi$ with corresponding eigenvalues $N\lambda_d$.

If $\tilde{K}_\varphi$ is singular, additional solutions exist in the form $\{a_d | \tilde{K}_\varphi a_d - N\lambda_d a_d \in \text{Null}(\tilde{K}_\varphi)\}$. However, since the Gram matrix is symmetric and positive semi-definite, it can only be singular if it has a zero eigenvalue. In practice, using 10 different transformer models in our experiments shows that $\tilde{K}_\varphi$ is typically invertible, allowing us to assume that the solutions a_d are eigenvectors.

C Language Models

Same experiments on encoder-only language models in Figure 5 reveals a similar pattern. As our models, we utilized `bert-base-uncased` (Devlin et al., 2018), `roberta-base` (Liu et al., 2019), `electra-small-discriminator`, `electra-base-discriminator` (Clark et al., 2020), `xlm-roberta-base` (Conneau et al., 2019), `longformer-base-4096` (Beltagy et al., 2020), `all-MiniLM-L6-v2` (Reimers and Gurevych, 2019), `camembert-base` (Martin et al., 2020), `luke-base` (Yamada et al., 2020).

Table 4 reveals that, across all examined language-model encoders, the similarity between the attention-learned value matrix V and its KPCA-based approximation $\dot{V}_{\text{KPCA}}$ remains disappointingly low, indicating that the proposed reconstruction is no more effective for NLP models than for their vision counterparts. We used 100 randomly sampled images from WikiText-103 dataset

We will be using the following matrix multiplication:

$$\tilde{K}_\varphi a_d = \begin{bmatrix} \tilde{\varphi}(k_1)^\top \tilde{\varphi}(k_1) & \tilde{\varphi}(k_1)^\top \tilde{\varphi}(k_2) & \cdots & \tilde{\varphi}(k_1)^\top \tilde{\varphi}(k_N) \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{\varphi}(k_N)^\top \tilde{\varphi}(k_1) & \tilde{\varphi}(k_N)^\top \tilde{\varphi}(k_2) & \cdots & \tilde{\varphi}(k_N)^\top \tilde{\varphi}(k_N) \end{bmatrix} \begin{bmatrix} a_{d1} \\ a_{d2} \\ \vdots \\ a_{dN} \end{bmatrix}$$

whose entries can be expressed as:

$$(\tilde{K}_\varphi a_d)_i = \sum_{j=1}^{N} \tilde{\varphi}(k_i)^\top \tilde{\varphi}(k_j) a_{dj}$$

Substituting u_d as the weighted combination of $\tilde{\varphi}(k_j)$ yields:

$$\frac{1}{N} \sum_{j=1}^{N} \tilde{\varphi}(k_j) \tilde{\varphi}(k_j)^\top \sum_{j'=1}^{N} a_{dj'} \tilde{\varphi}(k_{j'}) = \lambda_d \sum_{j=1}^{N} a_{dj} \tilde{\varphi}(k_j)$$

$$\frac{1}{N} \sum_{j=1}^{N} \tilde{\varphi}(k_j) \underbrace{\sum_{j'=1}^{N} \tilde{\varphi}(k_j)^\top a_{dj'} \tilde{\varphi}(k_{j'})}_{(\tilde{K}_\varphi a_d)_j} = \lambda_d \sum_{j=1}^{N} a_{dj} \tilde{\varphi}(k_j)$$

$$\frac{1}{N} \sum_{j=1}^{N} \tilde{\varphi}(k_i)^\top \tilde{\varphi}(k_j) \underbrace{\sum_{j'=1}^{N} \tilde{\varphi}(k_j)^\top a_{dj'} \tilde{\varphi}(k_{j'})}_{(\tilde{K}_\varphi a_d)_j} = \lambda_d \underbrace{\sum_{j=1}^{N} \tilde{\varphi}(k_i)^\top a_{dj} \tilde{\varphi}(k_j)}_{(\tilde{K}_\varphi a_d)_i}$$

$$\underbrace{\frac{1}{N} \sum_{j=1}^{N} \tilde{\varphi}(k_i)^\top \tilde{\varphi}(k_j) (\tilde{K}_\varphi a_d)_j}_{(\tilde{K}_\varphi (\tilde{K}_\varphi a_d))_i} = \lambda_d (\tilde{K}_\varphi a_d)_i$$

$$(\tilde{K}_\varphi (\tilde{K}_\varphi a_d))_i = N \lambda_d (\tilde{K}_\varphi a_d)_i$$

$$\tilde{K}_\varphi \tilde{K}_\varphi a_d = N \lambda_d \tilde{K}_\varphi a_d$$

$$\tilde{K}_\varphi (\tilde{K}_\varphi a_d - N \lambda_d a_d) = 0$$

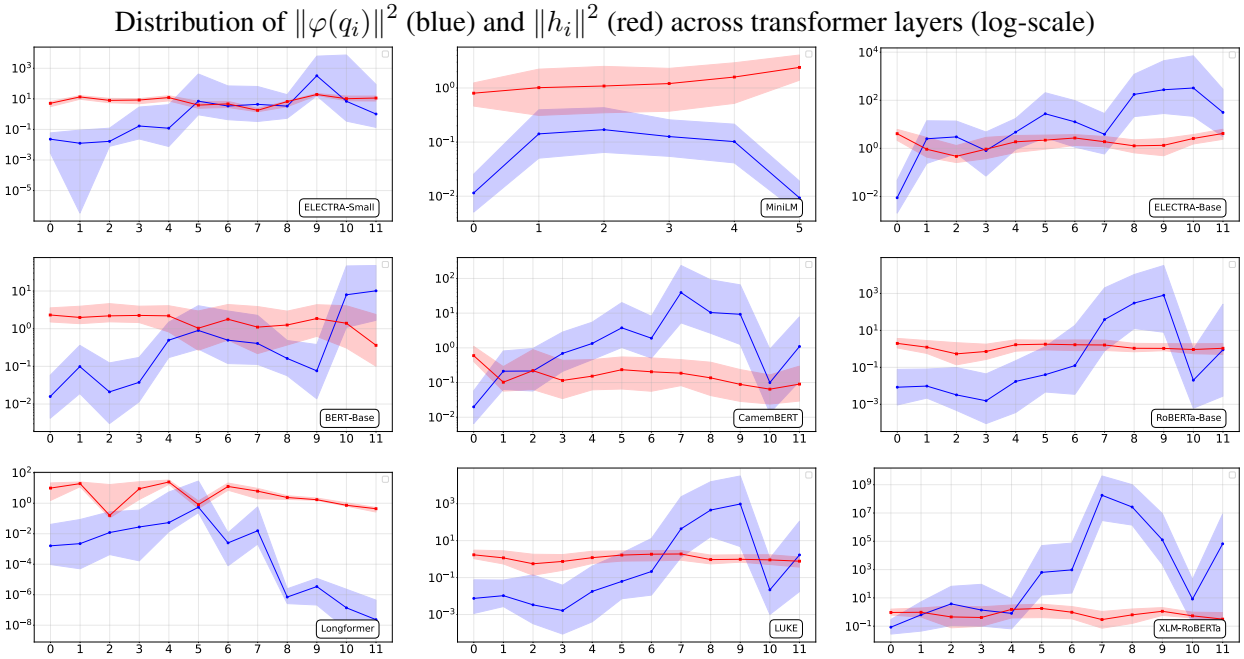

Figure 5: Distribution of $|\varphi(q_i)|^2$ (blue) and $|h_i|^2$ (red) across layers of nine pre-trained encoder-only language models (log scale) (ordered by the parameter count). Each plot shows the median (solid line) and 95th percentile (shaded region) of the squared norm values across tokens. Despite differences in architecture and scale, all models exhibit a similar pattern: large variability in $|\varphi(q_i)|^2$ compared to the more stable $|h_i|^2$, and no consistent convergence behavior across layers. This mirrors observations made in vision transformers.

(Merity et al., 2016).

Table 4: Similarity results between attention-learned value matrix V and proposed $\dot{V}_{\text{KPCA}}$ on a range of NLP encoder models. MDC: Max Direct Cosine Similarity; MOC: Max Optimal Cosine Similarity (Jonker–Volgenant matching); LCKA: Linear CKA; KCKA: Kernel CKA. Models are listed from the smallest to the largest (approximate) parameter count.

Model	**Similarity Measures**			
	MDC	MOC	LCKA	KCKA
ELECTRA-Small	0.22	0.40	0.07	0.28
MiniLM	0.40	0.57	0.13	0.38
BERT-Base	0.30	0.45	0.07	0.29
CamemBERT	0.30	0.46	0.09	0.30
ELECTRA-Base	0.27	0.46	0.05	0.29
RoBERTa-Base	0.15	0.30	0.05	0.35
Longformer	0.18	0.21	0.03	0.45
LUKE	0.14	0.30	0.05	0.34
XLM-RoBERTa	0.20	0.38	0.05	0.29

D Final Comments

KSVD *v.* KPCA perspectives While both KPCA and Kernel SVD (KSVD) interpret self-attention through kernel methods, they differ fundamentally in what they *guarantee*. The KPCA view of Teo and Nguyen (2024) claims that the canonical mechanism *by itself* drives the value matrix V towards the eigenvectors of the centred Gram matrix of the keys - which *fails* under our empirical scrutiny. In contrast, the KSVD formulation of Chen et al. (2023) finds a *resemblance* between vanilla self-attention output with the *dual* representation of an asymmetric-kernel SVD and makes no claim of spontaneous convergence.

The additional KSVD regulariser To realize the KSVD in practice, Chen et al. (2023) augment the task loss with a variance-maximisation term

$$\min_{\Theta} \; \mathcal{L}_{\text{task}}(\Theta) \; + \; \eta \sum_{l=1}^{L} J_l \tag{8}$$

where J_l is the KSVD loss of the l-th Primal-Attention layer, averaged over heads, and $\eta > 0$ is a hyper-parameter.[3] Solving (8) forces the dual variables $\{h_{rj}\}_{j=1}^{N}$ in $e(x_i) = \sum_{j=1}^{N} h_{rj}\, \kappa(x_i, x_j)$ to become orthonormal right singular vectors of the asymmetric kernel matrix $K_{ij} = \kappa(x_i, x_j)$ (Suykens, 2016).

[3] See Eqs. (6–7) in Chen et al. (2023) for the exact form of J_l.

Implication for canonical self-attention Without the regulariser ($\eta = 0$) canonical self-attention provides at most an *interpretive lens*: the value vectors can be *identified algebraically* with some set of dual coefficients, but they are *not* guaranteed to align with the principal right singular directions of K. Such alignment – and the attendant orthogonality/variance properties – emerges solely after optimising the joint objective (8). Hence, unlike the strong convergence asserted under the KPCA view, the KSVD lens remains descriptive unless that additional constraint is enforced during training.