# Sparse Latents for CLIP Spatial Reasoning Performance

Alexander Kyimpopkin
*University of Pennsylvania*
alxkp@seas.upenn.edu

***Abstract*—**

**We perform mechanistic affinity matrix analysis of CLIP latents to determine whether spectral clustering techniques can be used to emphasize key tokens for compositional and spatial reasoning tasks, a category which CLIP has historically struggled with, such as Winoground VQA. We then construct sparsity losses to encourage this behavior in training time, and find that counterintuitively, rather than relying on the "hinge" words such as "to" and "in", it is the intermediate, sparse connections in the affinity map that enable use to answer these spatial and compositional reasoning tasks. We present this as a mechanistic interpretability result towards learning better representations - denser intermediate affinity maps may be more appropriate for compositional and spatial reasoning tasks, rather than the intuitive sparser, higher magnitude maps to answer short, but very order dependent questions that might otherwise be intuitive. We present this as a direction for future latent design work towards creating more meaningful latents that can improve spatial reasoning performance downstream for VLMs and vision embeddings.**

**Our code is on Github: https://github.com/alxkp/cis680_clip_study**

## I. Introduction

CLIP [1] has become a standard and core method for all vision language models, such as Llava [2], [3], where the image encoder is entirely derived from a clip model, and specifically already frozen.

Notably, recent work has focused on improving the ability to reason for VLMs, given their challenges related to the quality of vision embeddings into text token space via a linear layer [4]. This remains an active area of research, in embedding the existing CLIP latents into the text token space via a variety of different methods, ranging from a simple matrix multiplication [2], [3], to transformer based methods in [5].

All these methods are bottlenecked by their ability to construct information dense and **complete** mapping of the image token space to the text token space, in a way that concentrates the information content into a small number of high density tokens that can function as consistent attention targets during generation.

## II. Background

CLIP [1] is a model which contrastively learns to align image and text embeddings using a similarity score from special tokens added to the end of each sequence. This implicitly aligns the prior text and image embeddings, as the learned representations in the embedding models are aligned via their end tokens and backpropagation. This, however, presents a challenge for spatial and compositional questions with specific orderings, such as [6]. Intuitively, we can see the challenge by considering the nature of the learning objective tokens. From our image embedding model, we have a `<EOS>` token, which is a single vector, and from our text embedding model, we have a `<CLS>` token, also a single vector. We cannot encode combinatorial relationships between objects ex ante via a single vector. Simply, we can see this by the dimensionality. If we have a vocabulary of size $v$, we then have our clip score computed as $\langle \text{<EOS>}, \text{<CLS>}\ v\rangle$. Each of these vectors is in $\mathbb{R}^v$, so we know that unless we can fit both our combinatorial relationships and our text and image semantic meanings into a single vector of size $v$, we cannot capture the full relationship between the image and text.

In information terms, we have introduced an extremely tight information bottleneck [7], where we are fitting the semantic information and the combinatorial information into the same vocabulary size $v$. Mathematically, if we want to motivate this, let us assume that we have some representation capacity $h$ in our vocabulary space, and we have a sequence of length $s$ in text space, and an image of $w \times w$ pixel patches. We then basically are trying to fit $s \times w^2$ relationships on top of our existing semantic information $h$ into the same representation space.

While it is possible that our vocabulary is sufficiently large to fit our semantic information and our combinatorial relationships, this implies that existing models have a very highly unoptimized vocabulary for image tokens. This runs counter to the shape optimization work in siglip400m-so [8], which actively optimizes all the shape values for more efficient image and language representation alignment.

This motivates more recent work in the field, expanding the loss space to be over the full sequence of embedded image patches and text tokens, rather than simply the `<EOS>` and `<CLS>` tokens. We see this with the Dense Cosine Similarity Map (DCSM) in [9], which constructs a single matrix as the loss objective for each image and text pairing. The challenge here is that this DCSM map assumes that we want dense connections between all terms, rather than encouraging sparsity.

Notably, we see from [10] that when we measure a trained CLIP model, we find that the second-order effects of the neurons are far more sparse, and thus we ought to encourage sparsity in order to capture stronger, more interpretable representations, based on these second-order effects results from a trained clip model. The ideal case here would be to prune out all the superfluous dense connections, and keep only the sparse, high significance connections over this entire joined space, making it easier to capture the full relationship between all image and text patches without noise from superfluous connections.

Further, if CLIP latents can become sparse with specific structure, we could see large gains, up to 100x in performance via sparsity [11] at a hardware level when we use a deliberately sparse clip set of output embeddings.

## III. Method

In order to encourage sparsity in the CLIP latents, we ablated over several different losses and measured their effects on performance, as well as validating that we were able to learn a meaningful representation in the latent space. We considered several different formulations, all designed to encourage sparsity in the latent space.

For context, the standard clip score is computed as:

$$\text{clip\_score}(x, y) = \sum_{i=1}^{n} \text{cls}_i \cdot \text{eos}_i \tag{1}$$

as noted above, we see that this is fairly unaware of the specific structure of the image and text patches, and only considers the overall semantic similarity between the image and text.

From this, we then constructed several different loss formulations all in order to encourage sparsity in the latent space. More granularly, our goal was to construct latents which were comprised of a small number of high magnitude eigenvectors, and penalize low magnitude eigenvectors, with the assumption that the true representation lies on a low dimensional manifold.

We theorized that our true representation was $r \in \mathcal{M}^n$ such that $n \ll s$, ie that our sequence and image space could be compressed, but not before constructing their interactions and pushing the information to their appropriate basis eigenvectors.

Doing this through the standard CLIP loss attempts to push this information inside the embedding networks, rather than letting it flow through relatively unmodified and then compressing it. It is this distinction about when, rather than how much we compress the information which we emphasize here as a mechanism to encourage sparsity without loss of richness and fidelity, the way that standard clip latents would provide.

We considered several different formulations, all designed to encourage sparsity in the latent space.

Specifically, we tested several different losses in order to confirm that our sparsity behaviors would not be determined by the loss formulation, and rather were a property of the sparsity itself only.

Generally, we treat the image and text features as separate vectors, zero padding them to $\max(\text{image}, \text{text})$ in order to construct a variety of affinity matrices. Below we give our formulations which we tested, but we note that the general form is simply more granular and different ways to decompose and structure

$\Lambda \in \mathcal{I} \otimes \mathcal{T}$, where $\mathcal{I}$ and $\mathcal{T}$ are the image and text feature spaces respectively, and we are looking for their eigenvectors, or measuring the matrix norm of their affinity - $\|\mathcal{I} \otimes \mathcal{T}\|_p$ for some arbitrary $p$-norm.

This leads us to our specific formulations below:

### A. Centered Kernel Alignment (CKA) Loss

We formulated the Centered Kernel Alignment (CKA) loss under the idea that the Hilbert-Schmidt Independence Criterion (HSIC) could be used to measure the variability of the alignment of the image and text features, and thus their independence.

We then wanted to orthogonalize the image and text feature spaces to reduce them to their specific independent bases, and then measure the alignment in that space, rather than in more general unaligned space to make sure that the loss was well formulated.

We then used the Frobenius norm to compute the magnitude of the alignment as this generalized version of the clip score over the entire affinity matrix.

Given image features $X \in \mathbb{R}^{n \times d_1}$ and text features $Y \in \mathbb{R}^{n \times d_2}$:

$$K = XX^T, \quad L = YY^T \tag{2}$$

We then center the Gram matrices to remove the mean of the features and focus only on their variance.

$$H = I_n - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^T \tag{3}$$

$$\tilde{K} = HKH, \quad \tilde{L} = HLH \tag{4}$$

As discussed above, we want to then compute the Hilbert-Schmidt Independence Criterion (HSIC) as the inner product of the centered Gram matrices:

$$\text{HSIC}(K, L) = \langle \tilde{K}, \tilde{L} \rangle_F = \sum_{i,j} \tilde{K}_{ij}\tilde{L}_{ij} \tag{5}$$

Finally, our complete CKA loss is:

$$\text{CKA}(X, Y) = \frac{\text{HSIC}(K, L)}{\sqrt{\text{HSIC}(K, K) \times \text{HSIC}(L, L)}} \tag{6}$$

We then minimize this loss to maximize alignment in this sparse regime.

$$\mathcal{L}_{\text{CKA}} = -\text{CKA}(X, Y) \tag{7}$$

We discuss our results in the experimental results section.

### B. Cross-Covariance SVD Loss

Similarly to the CKA, we once again want to encourage a small number of highly sparse directions in the image and text feature spaces, and we can do this by encouraging a small number of highly sparse singular values in the cross-covariance matrix - ie by penalizing the spectral gap after the

$k$-th singular value, and then promoting the magnitude of the top $k$ singular values.

We first compute the cross-covariance matrix (with normalized features):

$$C = \hat{X}^T \hat{Y} \in \mathbb{R}^{d_1 \times d_2}, \quad \text{where } \hat{X}_i = \frac{X_i}{\|X_i\|} \tag{8}$$

We then perform a singular value decomposition to get the singular values $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_{\min(d_1,d_2)}$.

We then encourage a spectral gap after the $k$-th singular value:

$$\mathcal{L}_{\text{gap}} = -(\sigma_k - \sigma_{k+1}) \tag{9}$$

We then add a reward, the higher the magnitude of the top $k$ singular values, the better:

$$\mathcal{L}_{\text{mag}} = -\sum_{i=1}^{k} \sigma_i \tag{10}$$

We then combine these two losses to form our complete SVD loss:

$$\mathcal{L}_{\text{SVD}} = \alpha \mathcal{L}_{\text{gap}} + \beta \mathcal{L}_{\text{mag}} \tag{11}$$

### C. CLIP + SVD Loss

The challenge we encountered during testing here is that we lose all of our original clip score performance that drove much of the original performance of the model.

Without the clip score, it becomes trivial to cheat the performance, simply by tuning the SVD loss to have generically high magnitude singular values without any meaningful alignment. We add back in the clip score as a way to ensure we retain alignment, after seeing catastrophic performance collapse when we use only either the SVD or the CKA loss without the joint CLIP score.

As before, we give the CLIP score (mean pairwise cosine similarity):

$$\text{CLIP}(X, Y) = \frac{1}{n} \sum_{i=1}^{n} \langle \hat{X}_i, \hat{Y}_i \rangle \tag{12}$$

And then combine this with our SVD loss to form our complete CLIP + SVD loss:

$$\mathcal{L}_{\text{CLIP-SVD}} = \underbrace{-\gamma \cdot \text{CLIP}(X, Y)}_{\text{pairwise alignment}} + \underbrace{\alpha \mathcal{L}_{\text{gap}} + \beta \mathcal{L}_{\text{mag}}}_{\text{structured alignment}} \tag{13}$$

We give our results below:

## IV. Experimental Results

First, in order to validate our idea that there existed some low dimensional manifold for the data to lie on, we used aligned NCUT on the generic outer product affinity matrix to find the eigenvectors that captured the most meaningful semantics of the data. [12]. We found that the eigenvector cutting procedure revealed smaller subspaces of high magnitude eigenvectors that captured distinct regions of the data, as shown below in Fig. 1.

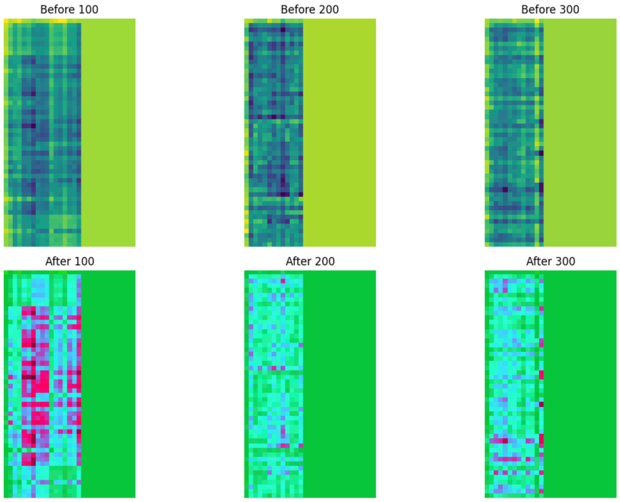


Fig. 1. Distinct regions of semantic data in clustered ncut representations of the data.

As we can see from clustering our latents via NCUT, we are able to capture the distinct regions of the data and see that that there are small, very dense regions of the affinity map that should be the able to be clustered into more meaningful regions.

We ran further tests to see that after clustering, we were able to capture meaningfully different directions of understanding, as seen below in Fig. 2.

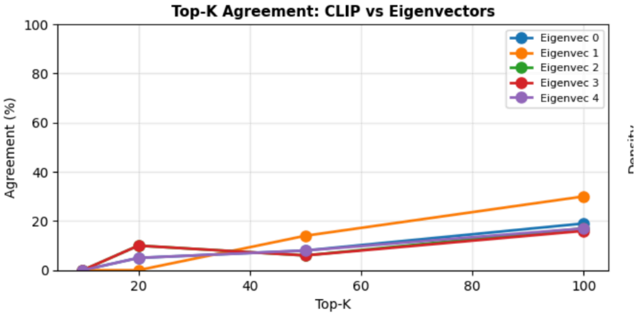
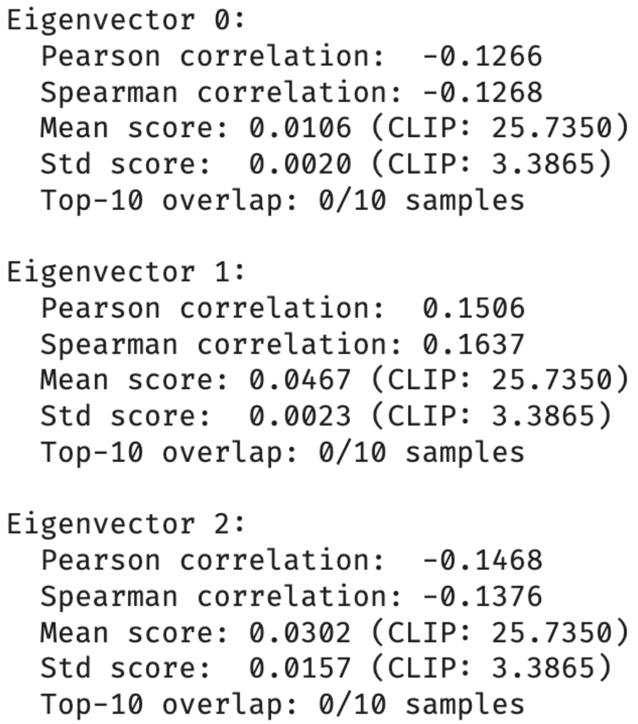


Fig. 2. Top-k agreement between CLIP affinity eigenvectors .

This shows us that as we are using different eigenvectors, they are capturing meaningfully different directions of understanding. This is further visible in the Pearson correlations between them to the original CLIP score - they are pointing in different and meaningfully new directions therefore to make it clear that we are capturing new directions of understanding. We ran this with both the Pearson and Spearman correlations to ensure that this wasn't an artifact of using the implied linear correlations in the parametric method, and Spearman gives us a nonparametric measure of correlation. We see this in Fig. 3.



```
Eigenvector 0:
  Pearson correlation:  -0.1266
  Spearman correlation: -0.1268
  Mean score: 0.0106 (CLIP: 25.7350)
  Std score:  0.0020 (CLIP: 3.3865)
  Top-10 overlap: 0/10 samples

Eigenvector 1:
  Pearson correlation:  0.1506
  Spearman correlation: 0.1637
  Mean score: 0.0467 (CLIP: 25.7350)
  Std score:  0.0023 (CLIP: 3.3865)
  Top-10 overlap: 0/10 samples

Eigenvector 2:
  Pearson correlation:  -0.1468
  Spearman correlation: -0.1376
  Mean score: 0.0302 (CLIP: 25.7350)
  Std score:  0.0157 (CLIP: 3.3865)
  Top-10 overlap: 0/10 samples
```

Fig. 3. Pearson and Spearman correlations between CLIP affinity eigenvectors and the original CLIP score.

Finally, the last sign that the spectral clustering approach on latents is applicable here is in the boosted emphasis on the word "to" in the post-clustering version, where clustering encourages sparsity and exemplars.

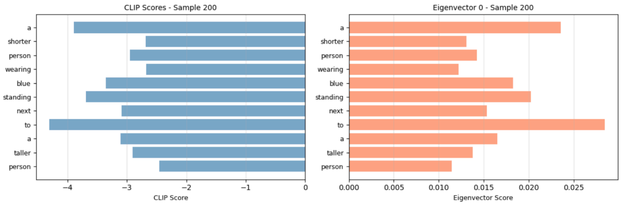


Fig. 4. Boosted emphasis on the word 'to' in the post-clustering version, where clustering encourages sparsity and exemplars.

We can see that since we want to answer relational questions, such as "Is the cat on the table?" vs "Is the table on the cat" in Winoground [6], we want to use this sparsity encouraging clustering approaches to encourage this sparsity at training time, rather than only during inference time.

For example, in Fig. 5, we want to be able to answer whether "the car is on the tree" or "the tree is on the car".

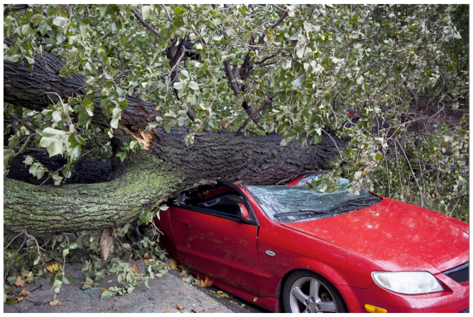


Fig. 5. Example of a question in Winoground .

We then finetuned a "clip-vit-base-patch32" on a standard MSCOCO-captions dataset. We finetuned with all of our above different losses and our best performing loss had a training plot we give below in Fig. 6.

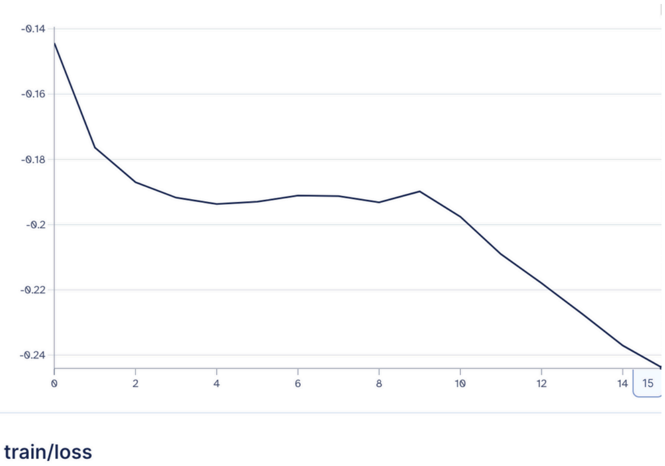


Fig. 6. Training loss curve for the best performing loss.

.

This is a promising training result, and we can further see that the model meaningfully learns. Further, the CLIP score remains roughly constant, suggesting we are retaining the original structure of CLIP over time. Furthermore, we can see this holds in the CLIP score over validation, as seen below in Fig. 7.

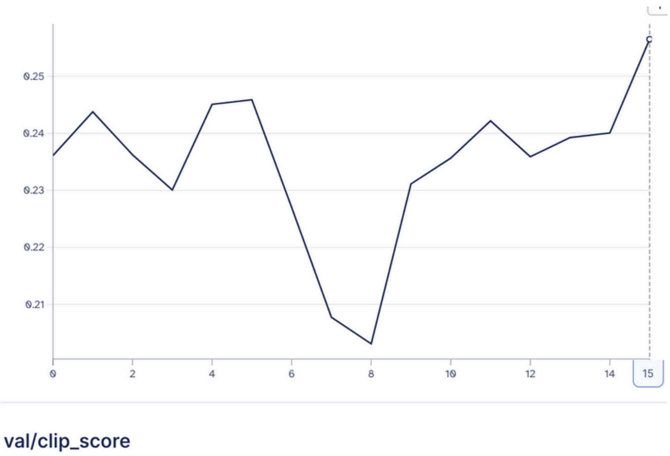


Fig. 7. Validation CLIP score over time.

This suggests that we have not lost meaningful information in the alignment process, and given our sparsity results above, if sparsity is actually beneficial to the model's performance, we should expect to see a boost in performance on both Winoground VQA and MSCOCO-captions.

However, when we run evaluations, we see that our performance drops - given below in Table I.

TABLE I
Evaluation results for the original CLIP and the sparse CLIP.

| **Dataset/Metric** | **Score (Original CLIP)** |
|---|---|
| **Score (Sparse CLIP)** | **Winoground VQA** Text |
| 31.2% | 22.0% |
| **Winoground VQA** Image | 11.2% |
| 7.5% | **Winoground VQA** Group |
| 9.0% | 5.2% |
| **MSCOCO-captions** Image to Text R@1 | 0.1% |
| 0.0% | **MSCOCO-captions** Image to Text R@5 |
| 0.6% | 0.0% |
| **MSCOCO-captions** Text to Image R@1 | 0.1% |
| 0.0% | **MSCOCO-captions** Text to Image R@5 |
| 0.5% | 0.0% |

We can see from the above that encouraging sparsity, despite our result that clustering latents encourages beneficial token emphasis, actually harms compositional reasoning performance. We further show this comparing the latent affinity map below before and after our finetuning.

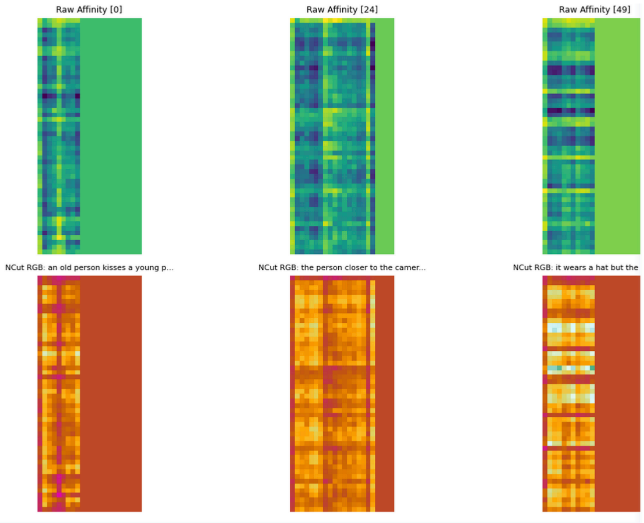


Fig. 8. Latent affinity map before finetuning.

Here, in Fig. 8 we can see that inside each lighter, silver region, we can see that there are intermediate connections between the less significant image tokens and the less significant text tokens. Our results earlier showed that we want to emphasize a small number of directions here as containing new and meaningfully different relationships, and that clustering, which encouraged sparsity improved the emphasis on key directional words, such as “to”, but when we finetune, our decreased performance visually manifests as horizontal banding structure across the affinity map, showing that only the most significant tokens are being emphasized, and the intermediate connections are being lost. While this is exactly the structure we had hoped for from our clustering results, when we attempt to apply it to compositional and spatial reasoning tasks, we see the significant performance drop shown above - from 31.2% to 22.0% on Winoground VQA Text, and from 11.2% to 7.5% on Winoground VQA Image questions. Our drop baseline on COCO is low enough that it is not as clear of a mechanism to measure here. Our affinity plot is in Fig. 9.

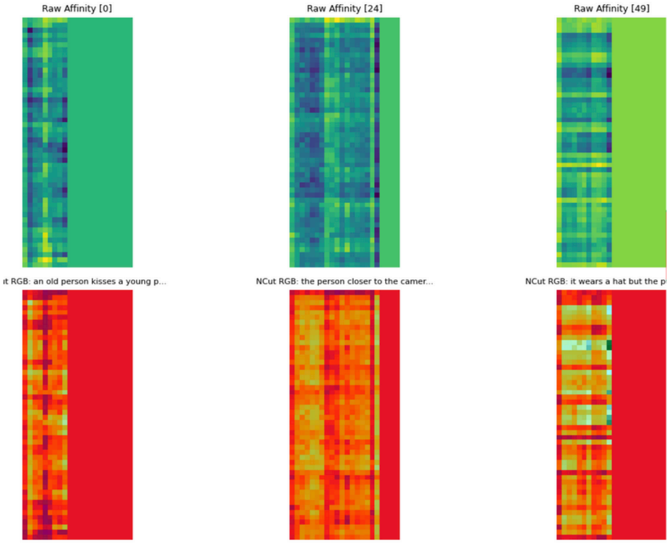


Fig. 9. Latent affinity map after finetuning.

As mentioned above, we can clearly see banding structures here, exactly the pattern we had hoped for with a small number of text tokens, such as "to", the core words for which order to run reasoning tasks in, having significantly more emphasis relative to their neighbors, but given our results, we can see that it is actually the intermediate, sparse connections in the affinity map that enable use to answer these spatial and compositional reasoning tasks, somewhat counterintuitively.

This suggests that the majority of the ability to answer spatial and compositional reasoning questions is in the dense, low magnitude connections, rather than the sparse, high magnitude connections.

## V. Conclusion and Future Work

We find that counter to the patterns that we see in [10], for interpretability, we find that encouraging these sparsity behaviors in latent space during further finetuning destroys compositional reasoning performance, since concentrating the mass of these interactions into a small number of eigenvectors gives us the same challenges outlined in the information bottleneck case of the regular clip score, but in a higher dimensional space. We can see this pattern in our results, where after some fine-tuning, we see clear banding structure across our latents, but lose all the intermediate connections in our latents - ie encouraging massive information sinks, rather than highly connected latents for this compositional reasoning task.

We suggest that for future work, rather than encouraging sparsity in the CLIP latents, exploring densification and graph-based structures to manage the densification would be a germane direction for further work.

Finally, we note that this observation about the structure of the affinity map suggests that the nature of which tokens in the embedding actually are adding the most in terms of performance is not easily motivated by interpretability studies from human intuition, and suggests further work towards visual token based reasoning approaches and more nuanced exploration of how visual question answering tasks can be better aligned with the practical latent space of their embedding models, rather than being hamstrung by the covariate shift of the human inductive biases in their text data and ignoring this density requirement and non-intuitive token significance in their embedding models. We submit this as a direction for future work towards better VLMs and richer CLIP-style latents.

## References

[1] A. Radford *et al.*, "Learning Transferable Visual Models From Natural Language Supervision." Accessed: Nov. 06, 2025. [Online]. Available: http://arxiv.org/abs/2103.00020

[2] H. Liu, C. Li, Y. Li, and Y. J. Lee, "Improved Baselines with Visual Instruction Tuning." Accessed: Dec. 18, 2025. [Online]. Available: http://arxiv.org/abs/2310.03744

[3] H. Liu, C. Li, Q. Wu, and Y. J. Lee, "Visual Instruction Tuning." Accessed: Dec. 18, 2025. [Online]. Available: http://arxiv.org/abs/2304.08485

[4] J. Wu *et al.*, "Reinforcing Spatial Reasoning in Vision-Language Models with Interwoven Thinking and Visual Drawing." Accessed: Dec. 18, 2025. [Online]. Available: http://arxiv.org/abs/2506.09965

[5] J. Li, D. Li, S. Savarese, and S. Hoi, "BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models." Accessed: Dec. 18, 2025. [Online]. Available: http://arxiv.org/abs/2301.12597

[6] T. Thrush *et al.*, "Winoground: Probing Vision and Language Models for Visio-Linguistic Compositionality." Accessed: Dec. 18, 2025. [Online]. Available: http://arxiv.org/abs/2204.03162

[7] N. Tishby and N. Zaslavsky, "Deep Learning and the Information Bottleneck Principle." Accessed: Dec. 18, 2025. [Online]. Available: http://arxiv.org/abs/1503.02406

[8] X. Zhai, B. Mustafa, A. Kolesnikov, and L. Beyer, "Sigmoid Loss for Language Image Pre-Training." Accessed: Dec. 18, 2025. [Online]. Available: http://arxiv.org/abs/2303.15343

[9] R. Kang, Y. Song, G. Gkioxari, and P. Perona, "Is CLIP ideal? No. Can we fix it? Yes!." Accessed: Oct. 14, 2025. [Online]. Available: http://arxiv.org/abs/2503.08723

[10] Y. Gandelsman, A. A. Efros, and J. Steinhardt, "Interpreting the Second-Order Effects of Neurons in CLIP." Accessed: Dec. 18, 2025. [Online]. Available: http://arxiv.org/abs/2406.04341

[11] C. Shinn, C. McCarthy, S. Muralidharan, M. Osama, and J. D. Owens, "The Sparsity Roofline: Understanding the Hardware Limits of Sparse Neural Networks." Accessed: Dec. 18, 2025. [Online]. Available: http://arxiv.org/abs/2310.00496

[12] H. Yang, J. Gee, and J. Shi, "AlignedCut: Visual Concepts Discovery on Brain-Guided Universal Feature Space." Accessed: Oct. 24, 2025. [Online]. Available: http://arxiv.org/abs/2406.18344