Grounding Representation Similarity with Statistical Testing

Anonymous Author(s)
Affiliation
Address
email

Abstract

To understand neural network behavior, recent works quantitatively compare different networks' learned representations using canonical correlation analysis (CCA), centered kernel alignment (CKA), and other dissimilarity measures. Unfortunately, these widely used measures often disagree on fundamental observations, such as whether deep networks differing only in random initialization learn similar representations. These disagreements raise the question: which, if any, of these dissimilarity measures should we believe? We provide a framework to ground this question through a concrete test: measures should have *sensitivity* to changes that affect functional behavior, and *specificity* against changes that do not. We quantify this through a variety of functional behaviors including probing accuracy and robustness to distribution shift, and examine changes such as varying random initialization and deleting principal components. We find that current metrics exhibit different weaknesses, note that a classical baseline performs surprisingly well, and highlight settings where all metrics appear to fail, thus providing a challenge set for further improvement.

1 Introduction

Improving our understanding of neural network representations is not only scientifically interesting, but critical for applying deep networks in high-stakes situations. Recent work has highlighted the value of analyzing not just the final outputs of a network, but also its intermediate representations [14, 24]. This has motivated the development of representation similarity measures, which can provide insight into how different training schemes, architectures, and datasets affect networks' learned representations.

A number of similarity measures have been proposed, including centered kernel alignment (CKA) [10], ones based on canonical correlation analysis (CCA) [19, 25], single neuron alignment [14], vector space alignment [3, 5, 27], and others [2, 8, 11, 13, 16, 33]. Unfortunately, these different measures tell different stories. For instance, CKA and projection weighted CCA disagree on which layers of different networks are most similar [10]. This lack of consensus is worrying, as measures are often designed according to different and incompatible intuitive desiderata, such as whether finding a one-to-one assignment, or finding few-to-one mappings, between network neurons is more appropriate [14]. As a community, we need well-chosen formal criteria for evaluating metrics to avoid over-reliance on intuition and the pitfalls of too many researcher degrees of freedom [12].

The question *are two representations A and B the same?* calls to mind the classical statistics question *are two datasets A and B generated from the same distribution?* For example, when assessing whether a medical treatment is effective, we ask whether the outcomes of the treatment and control groups are overall the same. It is standard to answer this with hypothesis testing: is there enough evidence to reject the null hypothesis that the two samples arose from the same distribution?

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

Methods for hypothesis testing are evaluated by their *sensitivity* and *specificity*. Sensitivity, or power, ensures a method tracks signal, while specificity ensures it is robust against noise. We propose similar criteria for similarity measures. A good measure should be *sensitive* to "important" changes while having *specificity* against changes that are less important.

To instantiate this idea, we initially consider two intuitive criteria: first, that metrics should have specificity against random initialization; and second, that they should be sensitive to deleting important principal components (those that affect probing accuracy). Unfortunately, popular metrics fail at least one of these two tests. CCA is not specific – random initialization noise overwhelms differences between even far-apart layers in a network (Section 3.1). CKA on the other hand is not sensitive, failing to detect changes in all but the top 10 principal components of a representation (Section 3.2).

While these failed tests are worrying, the discussion so far rests only on intuitive, qualitative desiderata. To ground ourselves, we turn to the functional behavior of the representations we are comparing, measured through several means such as probing accuracy [4, 22, 30] and out-of-distribution performance [6, 18, 20]. An important change is then one that affects these measures. We formalize this through a quantitative score based on the rank correlation between the distance measure and these functional behaviors (Section 4).

Using this rank correlation score, we construct a series of miniature "benchmarks" by varying models along one or more axes and measuring functional behavior. Each benchmark consists of 100 or more Transformer models, and we study which distance measures consistently achieve high rank correlation with functional behavior such as probing accuracy or out-of-distribution accuracy.

We first build two benchmarks that confirm our two intuitive observations: when varying random seed and layer depth, CCA has poor rank correlation with probing accuracy; and CKA similarly has poor rank correlation when deleting principal components. Meanwhile, the Orthogonal Procrustes distance, a classical but often overlooked[1] dissimilarity measure, balances gracefully between CKA and CCA and consistently performs well. This underscores need for systematic evaluation, otherwise we may fall to recency bias that undervalues classical baselines.

We then turn to out-of-distribution accuracy, motivated by the observation that random initialization sometimes has large effects on OOD performance [18]. While CCA and Orthogonal Procrustes distance can sometimes detect these differences, achieving high rank correlation with OOD accuracy, other OOD behaviors presented a challenge for all the measures we studied. Our protocol thus reveals a challenge set for developing new and better dissimilarity measures, which we hope will help measure and spur progress in the future.

2 Problem Setup

Our goal is to quantify the similarity between two different groups of neurons (usually layers). We do this by comparing how their activations behave on the same dataset. Thus for a layer with p_1 neurons, we define $A \in \mathbb{R}^{p_1 \times n}$, the matrix of activations of the p_1 neurons on n data points, to be that layer's representation of the data. Similarly, let $B \in \mathbb{R}^{p_2 \times n}$ be a matrix of the activations of p_2 neurons on the same n data points. In this work we study dissimilarity measures $d(A, B)$ that allow for quantitative comparisons of representations both within and across different networks. We colloquially refer to values of $d(A, B)$ as distances, although they do not necessarily satisfy the triangle inequality required of a proper metric.

In this paper we study five dissimilarity measures: centered kernel alignment (CKA), three measures derived from canonical correlation analysis (CCA), and a measure derived from the orthogonal Procrustes problem. As argued in Kornblith et al. [10], similarity measures should be invariant to left orthogonal transformations to accommodate the symmetries of neural networks, and all studied measures satisfy this requirement.

Centered kernel alignment (CKA) uses an inner product to quantify similarity between two representations. It is motivated by the idea that one can first use a kernel to measure the similarity between every pair of datapoints in each representation separately, and then compare these similarity structures. The measure of similarity is determined by one's choice of kernel; in this work we focus

[1] For instance, Raghu et al. [25] and Morcos et al. [19] do not mention it, and Kornblith et al. [10] relegates it to the appendix; although Smith et al. [27] does use it to analyze word embeddings and prefers it to CCA.

on **Linear CKA**:

$$d_{\text{Linear CKA}}(A, B) = 1 - \frac{\|AB^\top\|_F^2}{\|AA^\top\|_F \|BB^\top\|_F} \tag{1}$$

as proposed in Kornblith et al. [10]. Other choices of kernel are also valid; we focus on Linear CKA here since Kornblith et al. [10] report similar results from using either a linear or RBF kernel.

Canonical correlation analysis (CCA) finds orthogonal bases (w_A^i, w_B^i) for two matrices such that after projection onto w_A^i, w_B^i, the projected matrices have maximally correlated rows. For $1 \leq i \leq p_1$, the ith canonical correlation coefficient ρ_i is computed as follows:

$$\rho_i = \max_{w_A^i, w_B^i} \frac{\langle {w_A^i}^\top A, {w_B^i}^\top B \rangle}{\|{w_A^i}^\top A\| \cdot \|{w_B^i}^\top B\|} \tag{2}$$

$$s.t. \quad \langle {w_A^i}^\top A, {w_A^j}^\top A \rangle = 0, \ \forall j < i, \qquad \langle {w_B^i}^\top B, {w_B^j}^\top B \rangle = 0, \ \forall j < i \tag{3}$$

To transform the vector of correlation coefficients into a scalar measure, two options considered previously are the **mean correlation coefficient,** $\bar{\rho}_{\textbf{CCA}}$, and the **mean squared correlation coefficient,** $R^2_{\textbf{CCA}}$, defined as follows:

$$d_{\bar{\rho}_{\text{CCA}}}(A, B) = 1 - \frac{1}{p_1} \sum_i \rho_i, \qquad d_{R^2_{\text{CCA}}}(A, B) = 1 - \frac{1}{p_1} \sum_i \rho_i^2 \tag{4}$$

To improve the robustness of CCA, Morcos et al. [19] propose **projection-weighted CCA (PWCCA)** as another scalar summary of CCA:

$$d_{\text{PWCCA}}(A, B) = 1 - \frac{\sum_i \alpha_i \rho_i}{\sum_i \alpha_i}, \qquad \alpha_i = \sum_j |\langle h_i, a_j \rangle| \tag{5}$$

where a_j is the jth row of A, and $h_i = {w_A^i}^\top A$ is the projection of A onto the ith canonical direction. We find that PWCCA performs far better than $\bar{\rho}_{\text{CCA}}$ and R^2_{CCA}, so we focus on PWCCA in the main text, but include results on the other two measures in the appendix.

The **orthogonal Procrustes** problem consists of finding the left-rotation of A that is closest to B in Frobenius norm, i.e. solving the optimization problem:

$$\min_R \|B - RA\|_{\text{F}}^2, \quad \text{subject to } R^\top R = I. \tag{6}$$

The minimum is the squared **orthogonal Procrustes distance** between A and B, and is equal to:

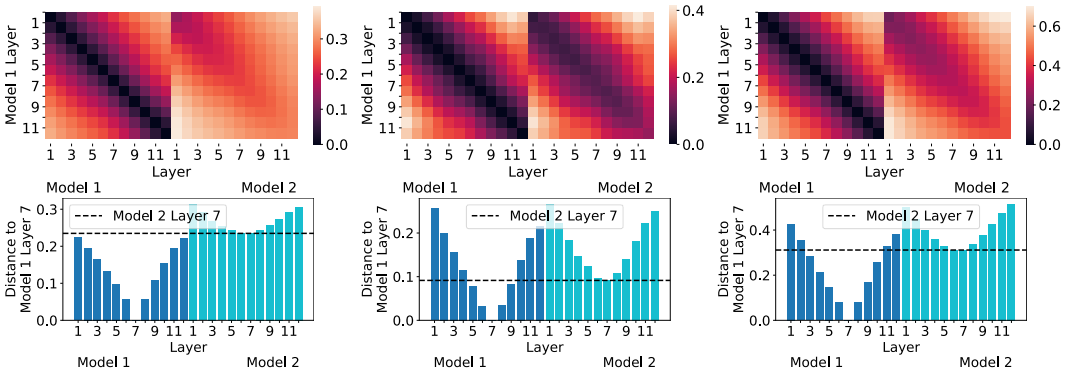
$$d_{\text{Proc}}(A, B) = \|A\|_F^2 + \|B\|_F^2 - 2\|A^\top B\|_*, \tag{7}$$

where $\|\cdot\|_*$ is the nuclear norm [26]. Unlike the other metrics, the orthogonal Procrustes distance is not normalized between 0 and 1.

Models studied. We investigate the representations computed by the BERT model family [7] on sentences from the Multigenre Natural Language Inference (MNLI) dataset [34]. We study BERT models of two sizes: BERT base, with 12 hidden layers of 768 neurons each, and BERT medium, with 8 hidden layers of 512 neurons. We use the same architectures as in the open source BERT release[2], but we study 3 variations of models in order to provide a variety of testbeds for dissimilarity measures: 1) BERT base models we pretrained with different random seeds but did not fine-tune for particular tasks, 2) BERT medium models we pretrained with different random seeds and fine-tuned on MNLI with different fine-tuning seeds, and 3) BERT base models [3] released by McCoy et al. [18] that were initialized from the pretrained BERT model in [7] and fine-tuned on MNLI with different seeds. We pretrain models with 10 different random seeds for variations 1 and 2, using code from [7] and the pretraining corpus from Li et al. [15]. To collect models under variation 2, we further finetune the pretrained models with 10 different fine-tuning seeds each, resulting in 100 total finetuned models to analyze. Further training details, as well as checks that our training protocol results in models with comparable performance to the original BERT release, can be found in Appendix A.

[2] available at https://github.com/google-research/bert

[3] available at https://github.com/tommccoy1/hans/tree/master/berts_of_a_feather



(a) PWCCA distances (b) CKA distances (c) Orthogonal Procrustes distances

Figure 1: **PWCCA fails the intuitive specificity test.** Top: PWCCA, CKA, and Orthogonal Procrustes pairwise distances between each layer of two differently initialized networks, averaged over 5 independent pairs of networks. Bottom: We zoom in to analyze the 7th layer, plotting this layer's distance to every other layer. The dotted line is this layer's distance to the corresponding 7th layer in the second model. For PWCCA, the dotted line fails to cross any of the bars on the left half of the plot, indicating that it is not specific against changes in random initialization.

3 Intuitive tests for sensitivity and specificity

Sensitivity and specificity are common statistical measures of the performance of a binary classifier. Sensitivity, or true positive rate, is the proportion of positives that are correctly identified as positive. Specificity, or true negative rate, is the proportion of negatives correctly identified as negative.

Assessing sensitivity and specificity requires a ground truth – which representations are truly different? To answer this, we begin with the following two intuitions: 1) neural network representations trained on the same data but from different random initializations are similar, and 2) representations lose crucial information as principal components are deleted. These motivate the following intuitive tests of specificity and sensitivity: we expect a similarity measure to: 1) assign a small distance between architecturally corresponding layers of two neural networks only differing in initialization seed, and 2) assign a large distance between a representation A and the representation $\hat{A}$ after significant dimensionality reduction (enough to affect accuracy). In this section we show that PWCCA fails the intuitive specificity test, while CKA fails the intuitive sensitivity test.

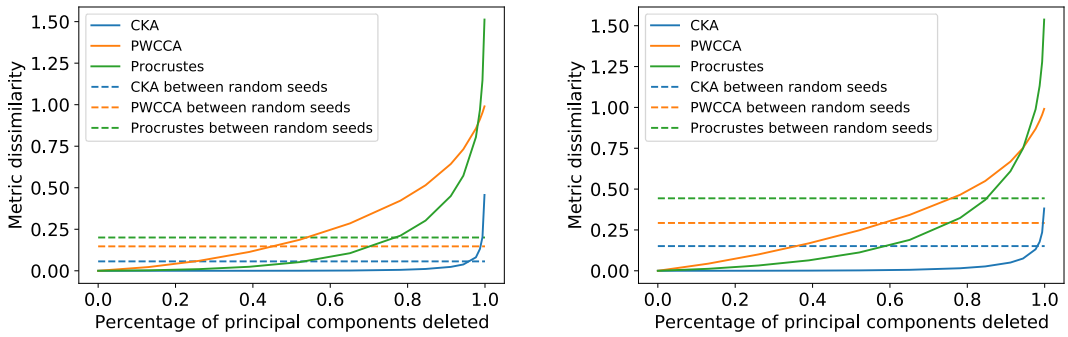
3.1 Specificity against changes to random seed

Neural networks with the same architecture trained from different random initializations show many similarities, such as highly correlated predictions on in-distribution data points [18]. Thus it seems natural to expect a good similarity measure to assign relatively small distances between architecturally corresponding layers of networks only differing in initialization seed.

To check this property we take two BERT base models pre-trained with different random seeds and, for every layer in the first model, compute its dissimilarity to every layer in both the first and second model. We do this for 5 separate pairs of models and average the results. To pass the intuitive specificity test, a dissimilarity measure should assign relatively small distances between a layer in the first network and its architecturally corresponding layer in the second network.

Figure 1 shows heatmaps for the average pair-wise PWCCA, CKA, and Orthogonal Procrustes distances between layers of two networks, in addition to a zoomed-in view to analyze distances from the 7th layer in particular. According to PWCCA, two different networks' representations at layer 7 are quite dissimilar, with a higher PWCCA distance than that between layer 7 and any other layer within the same network. Clearly, PWCCA is not specific against random initialization.

In contrast, CKA can separate layer 7 in a different network from layers 4 or 10 in the same network, showing better specificity to random initialization. Orthogonal Procrustes exhibits smaller but non-trivial specificity, distinguishing layers once they are 4-5 layers apart.



(a) First layer of BERT

(b) Last layer of BERT

Figure 2: **Metrics disagree on the representation dissimilarity induced by deleting principal components.** We fix a layer depth (here layer 1 (a) and layer 12 (b) of BERT) and compute dissimilarities between that layer's representation and a low-rank approximation to that representation found by deleting principal components (starting from the smallest). We plot these dissimilarities as a function of the fraction of PCs deleted (solid lines). We also plot the baseline average distance between networks trained from random seeds, at that layer depth, with the dotted line. The dotted line and solid line intersect at different points for each dissimilarity measure, indicating disagreement on the relative dissimilarity induced by random seed and by deleting principal components.

3.2 Sensitivity to removing principal components

Dissimilarity measures should also be sensitive to deleting important principal components of a representation. To quantify which components are important, we fix a layer of a pre-trained BERT base model and analyze how probing accuracy degrades as principal components are deleted. We can compare this to how dissimilarity increases, and assess whether a dissimilarity measure is sensitive to the loss of important information.[4]

Figure 2 plots the dissimilarity induced by deleting principal components, compared to a baseline of dissimilarity between networks with different random initialization. We see that measures disagree on the fraction of principal components that need to be deleted to be equivalent to random seed dissimilarity. For the last layer of BERT, CKA considers random initialization variation to be on the same order as deleting over 97% of a representation's principal components; for Orthogonal Procrustes, $\sim$85%, and for PWCCA, $\sim$55%. We may interpret these fractions as each measure's threshold for "detectable" principal component deletion.

To contextualize this, we also compute probing accuracy, a common measure of the information captured in a representation [4], as principal components are deleted from the last layer of BERT. We probe linear classifation performance on the Stanford Sentiment Tree Bank task (SST-2) [28], following the experimental protocol in Tamkin et al. [29]. Figure 3b shows how probing accuracy degrades with component deletion.

We find that at CKA's threshold of detectable deletion (97% of the principal components), there is an accuracy drop from 80% to 63% (chance is 50%), which seems unreasonably large to be minimally "detectable". At Orthogonal Procrustes' threshold (85%) the drop in accuracy is from 80% to 70%, while at PWCCA's threshold (55%) there is only a small drop, from 80% to 75% accuracy.

PWCCA's failure of specificity and CKA's failure of sensitivity on these intuitive tests are worrying. However, before declaring definitive failure, in the next section, we turn to making our assessments more rigorous.

[4]For a representation A, we define $\hat{A}_{-k}$, the result of deleting the k smallest principal components from A, as follows: we compute the singular value decomposition $U\Sigma V^T = A$, construct $U_{-k} \in \mathbb{R}^{p \times p-k}$ by dropping the lowest k singular vectors of U, and finally take $\hat{A}_{-k} = U_{-k}^T A$.

4 Rigorously evaluating dissimilarity metrics

In the previous section, we saw that CKA and PWCCA each failed intuitive tests, based on sensitivity to principal components and specificity to random initialization. However, these were based primarily on intuitive, qualitative desiderata. Is there some way for us to make these tests more rigorous and quantitative?

To start with, consider the intuitive layer specificity test (Section 3.1), which revealed that random initialization matters more to PWCCA than large changes in layer depth. To justify why this is undesirable, we can turn to probing accuracy, which is strongly affected by layer depth, and only weakly affected by random seed (Figure 3a). This suggests a path forward: grounding the layer test in the concrete differences in functionality captured by the probe.

More generally, we want metrics to be sensitive to changes that affect functionality, while ignoring those that don't. This motivates the following general procedure, given a distance metric d and a functionality f (which assigns a real number to a given representation):

1. Collect a set S of representations that differ along one or more axes of interest (e.g. layer depth, random seed).
2. Choose a reference representation $A \in S$. When f is an accuracy metric, it is reasonable to choose $A = \arg\max_{A \in S} f(A)$.[5]
3. For every representation $B \in S$:
   - Compute $|f(A) - f(B)|$
   - Compute $d(A, B)$
4. Report the rank correlation between $|f(A) - f(B)|$ and $d(A, B)$ (measured by Kendall's τ or Spearman ρ).

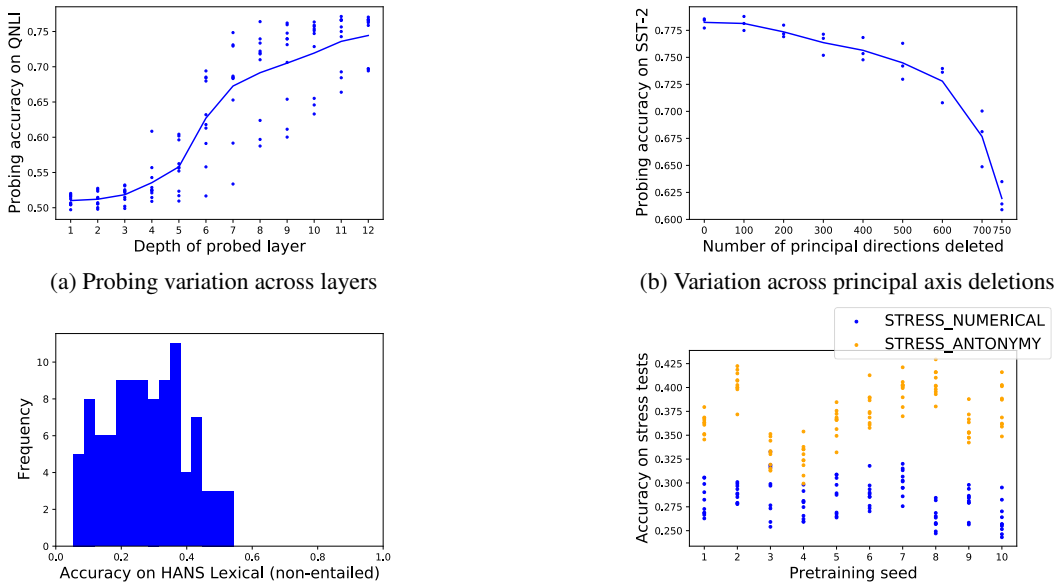
The above procedure provides a *quantitative* measure of how well the distance metric d responds to the functionality f. Intuitively, the most informative sets S contain multiple axes of variation, some of which affect functionality and some of which don't. For instance, in the layer specificity test, since depth affects probing accuracy strongly while random seed affects it only weakly, a dissimilarity measure needs to be strongly responsive to layer depth and weakly responsive to seed, to achieve a high rank correlation, quantitatively formalizing the test from Section 3.1.

Note, however, that while this intuition is helpful for generating S, our procedure does not rely on it. For instance, Figure 3a shows that some variation in random seed actually does affect accuracy, and the procedure rewards metrics that pick up on this, even though it would increase sensitivity to random seed. The crucial feature of S is that it contains models with varying functionality. Different sets S can be thought of as miniature "benchmarks" that surface complementary perspectives on dissimilarity measures' sensitivity and specificity to functional differences.

In the rest of this section, we instantiate this quantitative benchmark for several functional differences f and collections of models S, starting with the layer and principal component tests from Section 3 and continuing on to several tests of OOD performance. The overall results are summarized in Table 1. Note that for any single benchmark, we expect the correlation coefficients to be significantly lower than 1, since the metric D must capture all important axes of variation while f measures only one type of functionality. A good metric is one that has consistently high correlation across many different functional measures.

Layer depth. To turn the layer test into a benchmark, we construct a set S of 120 representations by pretraining 10 BERT base models with different initialization seeds and including each of the 12 BERT layers as a representation. We separately consider two functionalities f: probing accuracy on QNLI [32] or SST-2 [28]. To compute the rank correlation, we take the reference representation A to be the depth-1 (initial layer) representation with highest probing accuracy. We compute the Kendall's τ and Spearman's ρ rank correlations between the dissimilarities and the probing accuracy

[5] Choosing the highest accuracy model as the reference makes it more likely that as accuracy changes, models are on average becoming more dissimilar. A low accuracy model may be on the "periphery" of model space, where it is dissimilar to models with high accuracy, but potentially even more dissimilar to other low accuracy models that make different mistakes.



(a) Probing variation across layers

(b) Variation across principal axis deletions

(c) Variation across finetuning seeds

(d) Variation across different seeds

Figure 3: **Our perturbations induce substantial variation on probing tasks and stress tests:** (3a): On average, probing accuracy on QNLI increases significantly with layer depth, across layers of pretrained BERT base. Variation due to different random initializations is displayed as the scatter plot surrounding the mean trend. (3b): Truncating principal components from pretrained BERT base (specifically, layer 12 shown here) significantly degrades probing accuracy on SST-2. Variation due to different random initializations is displayed through the scatter plot surrounding the mean trend. (3c): Finetuning BERT base with different seeds leads to variation in accuracies on the Lexical (non-entailed) subset of HANS [18]. (3d): Pretraining and finetuning BERT medium with 10 different pretraining seeds and 10 different finetuning seeds per pretrained model leads to variation in accuracies on the Antonymy and Numerical stress tests [20].

differences. We then repeat this for depths $2, 3, \ldots, 12$. Results are consistent across the 12 depths (see Appendix C), so we average these values and report summary statistics in Table 1.

We find that PWCCA and Procrustes have lower rank correlations compared to CKA for both probing tasks. This corroborates the intuitive specificity test (Section 3.1), suggesting that PWCCA registers too large dissimilarities across random initializations.

Principal component (PC) deletion. In Section 3.2, CKA appeared to lack sensitivity to principal component deletion, requiring 97% of a representation's principal components to be removed in order to match the dissimilarity between two networks trained from different initializations. We quantify this as a benchmark by constructing a set of representations S that varies in both random initialization and in fraction of principal components deleted: we take 10 BERT-base models pretrained with different initializations, and from each pretrained model we obtain 14 different representations by deleting that layer representation's k smallest principal components, with $k \in \{0, 100, 200, 300, 400, 500, 600, 650, 700, 725, 750, 758, 763, 767\}$. Thus S has $10 \times 14 = 140$ elements. The representations themselves are the layer-ℓ activations, for $\ell \in \{7, 8, \ldots, 12\}$,[6] so there are 6 different choices of S. We use SST-2 probing accuracy as the functionality of interest f, and select the reference representation A as the element in S with highest accuracy. We average rank correlation results over the 6 choices of S and report results in Table 1.

We find that PWCCA has the highest rank correlation between dissimilarity and probing accuracy functionality, followed by Procrustes, and distantly followed by CKA. This corroborates the intuitive observations from Section 3.2.

[6]Earlier layers have near-chance accuracy on probing tasks, so we ignore them.

Table 1: Summary of rank correlation results

Perturbation	Functionality	Procrustes		CKA		PWCCA	
		ρ (p)	τ (p)	ρ (p)	τ (p)	ρ (p)	τ (p)
Pretraining seed, layer depth	QNLI probe	0.574	0.435	**0.628**	**0.473**	0.491	0.368
	SST-2 probe	0.624	0.474	**0.679**	**0.512**	0.556	0.42
Pretraining seed, PC deletion	SST-2 probe	0.895	0.73	0.795	0.643	**0.918**	**0.76**
Finetuning seed	HANS: Lexical (non-entailed)	**0.551 (4.8e-07)**	**0.398 (6.1e-07)**	0.462 (2.3e-3)	0.329 (2.6e-3)	**0.568 (1.2e-6)**	**0.412 (1.8e-6)**
Pretraining, finetuning seeds	Antonymy stress test	**0.243**	**0.178**	0.227	0.16	0.204	0.152
	Numerical stress test	0.071	0.049	**0.122**	**0.084**	0.031	0.023

4.1 Variation across random seeds

So far our benchmarks are based on probing accuracy, which only measures in-distribution behavior (the train and set set of the probe are typically IID). In addition, representations were always pretrained but not finetuned. To add diversity to our benchmarks, we next consider the out-of-distribution performance of several collections of fine-tuned models.

Changing fine-tuning seeds. McCoy et al. [18] showed that a single pretrained BERT-base model fine-tuned with different random initializations will produce models with highly correlated predictions on in-distribution data, but widely variable performance on out-of-distribution data (see Figure 3c). We thus create a benchmark S out of McCoy et al.'s 100 released fine-tuned models on MNLI, using OOD accuracy on the Lexical heuristic (non-entailment) subset of the HANS dataset [17] as our functionality f. We construct 12 different S (one for each of the 12 layers of BERT-base), taking A in each case to be the representation of the model with maximum accuracy according to f. As before, we report each dissimilarity measure's rank correlation with f in Table 1, averaged over the 12 runs.

All three dissimilarity measures show substantive correlation with OOD accuracy, with Orthogonal Procrustes and PWCCA being more correlated than CKA. The representations were computed on in-distribution MNLI data, meaning that the dissimilarity measures can detect OOD differences without access to OOD data. This suggests that some of PWCCA and Procrustes' sensitivity to random initialization noise actually tracks functionality.

To check that differences in rank correlations are statistically significant, for every pair of distances we subsampled 20 out of the 100 perturbations and computed the rank correlations for this subsample. We repeated this for 2000 subsamples, and reported the fraction of subsamples for which the rank correlations are greater for each metric. We found statistically significant differences between PWCCA and CKA, and Orthogonal Procrustes and CKA (the full results are in Appendix D). We did not apply this procedure for the previous two benchmarks, because the different models have correlated randomness and so any p-value based on independence assumptions would be invalid.

Changing both pretraining and fine-tuning seeds with BERT-medium. We also construct benchmarks from a collection of 100 BERT-medium models, trained with all combinations of 10 pretraining and 10 fine-tuning seeds. The models are fine-tuned on MNLI, and we consider two different functionalities of interest f: accuracy on the Antonymy stress test and the Numerical stress test [20], which both show significant variation in accuracy across the different models (see Figure 3d). We obtain 8 different S sets (one each of the 8 layer depths in BERT-medium), again taking A to be the representation of the highest-accuracy model according to f. Rank correlations for each dissimilarity measure are averaged over the 8 runs and reported in Table 1.

None of the studied dissimilarity measures show a large rank correlation for either task, and the associated p-values are non-significant at the 0.025 level (even if they were below this threshold, the

p-values are invalid and underestimates of the true p-value because they assume independence; here models with the same pretraining seed are correlated). Thus we conclude that all measures fail to be sensitive to OOD accuracy in these settings. One reason for this could be that there is less variance in the OOD accuracies for these networks and on these stress tests compared to the previous experiment (compare 3c to 3d). It is also possible that a newly proposed dissimilarity measure could improve upon these results, and we present this setting as a challenge task to help benchmark progress.

5 Discussion

In this work we proposed a quantitative performance measure for evaluating similarity metrics, based on the rank correlation with functional behavior. Using this, we generated tasks motivated by sensitivity to deleting important directions, specificity to random initialization, and sensitivity to out-of-distribution performance. Popular existing metrics such as CKA and CCA often performed poorly on these tasks, sometimes in striking ways. Meanwhile, the classical Orthogonal Procrustes transform attained consistently good performance.

Given the success of Orthogonal Procrustes, it is worth reflecting on how it differs from the other metrics and why it might perform well. To do so, we consider a simplified case where A and B have the same singular vectors but different singular values. Thus without loss of generality $A = \Lambda_1$ and $B = \Lambda_2$, where the Λ_i are both diagonal. In this case, the Orthogonal Procrustes distance reduces to $\|\Lambda_1 - \Lambda_2\|_F^2$, or the sum of the squared distances between the singular values. We will see that both CCA and CKA reduce to less reasonable formulas in this case.

Orthogonal Procrustes vs. CCA. All three metrics derived from CCA assign *zero* distance even when the (non-zero) singular values are arbitrarily different. This is because CCA correlation coefficients are invariant to all invertible linear transformations. This invariance to invertible linear transformations may help explain why CCA metrics generally find networks trained with different randomness to be very different–random initialization introduces noise, particularly in unimportant principal components, and CCA may overfit on this noise when optimizing for the optimal linear alignment of the two representations.

Orthogonal Procrustes vs. CKA. In contrast to the squared distance of Orthogonal Procrustes, CKA actually reduces to a quartic function based on the dot products between the *squared* entries of Λ_1 and Λ_2. As a consequence, CKA is dominated by representations' largest singular values, leaving it insensitive to meaningful differences in smaller singular values as illustrated in Figure 2. This lack of sensitivity to moderate-sized differences may also help explain why CKA fails to track out-of-distribution error effectively.

While rank correlation provides a unified framework for understanding similarity measures, it also suggests how to improve them. While Orthogonal Procrustes performed well on most of our tasks, no method was sensitive to accuracy on the Antonymy and Numerical stress tests [20], possibly due to a lower signal-to-noise ratio. Recent work shows how to regularize Orthogonal Procrustes to handle high noise [23], and perhaps similar techniques could be adapted here.

A limitation of this work is that we only consider a handful of functional behaviors, and overreliance on the behaviors measured here could result in neglecting other important differences in representations, such as modularity. Thus we envision a paradigm in which a rich tapestry of benchmarks are used to ground and validate neural network interpretations. Other axes of variation to study could include training on more or fewer examples, training on shuffled labels vs. real labels, and training from specifically chosen initializations [9]. These benchmarks could also be applied to other interpretability tools beyond dissimilarity. For example, sensitivity to deleting principal components could provide an additional sanity check for saliency maps and other visualization tools [1].

Another risk of this style of work is that it provides an illusion of interpretability, while in reality neural networks are too complex for any simple quantitative measures to track their variation. We believe this can be mitigated by considering many axes of variation early on. For example, one avenue of interesting future work is to study how choice of architecture and training parameters affects learned representations. We can also check whether dissimilarity metrics correlate well with other interpetability tools such as neuron visualisations [21]. While it is possible that such checks will ultimately show that attempts at interpreting deep networks are misguided, we are optimistic that they will instead provide deeper understanding and more reliable models.

References

[1] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, and B. Kim. Sanity checks for saliency maps. *Advances in Neural Information Processing Systems*, 31:9505–9515, 2018.

[2] G. Alain and Y. Bengio. Understanding intermediate layers using linear classifier probes, 2018.

[3] S. Arora, Y. Liang, and T. Ma. A simple but tough-to-beat baseline for sentence embeddings. 2017.

[4] Y. Belinkov, N. Durrani, F. Dalvi, H. Sajjad, and J. Glass. What do neural machine translation models learn about morphology? In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 861–872, 2017.

[5] A. Conneau, G. Lample, M. Ranzato, L. Denoyer, and H. Jégou. Word translation without parallel data, 2018.

[6] A. D'Amour, K. Heller, D. Moldovan, B. Adlam, B. Alipanahi, A. Beutel, C. Chen, J. Deaton, J. Eisenstein, M. D. Hoffman, et al. Underspecification presents challenges for credibility in modern machine learning. *arXiv preprint arXiv:2011.03395*, 2020.

[7] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[8] Y. Feng, R. Zhai, D. He, L. Wang, and B. Dong. Transferred discrepancy: Quantifying the difference between representations. *arXiv preprint arXiv:2007.12446*, 2020.

[9] J. Frankle and M. Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *International Conference on Learning Representations*, 2018.

[10] S. Kornblith, M. Norouzi, H. Lee, and G. Hinton. Similarity of neural network representations revisited. In *International Conference on Machine Learning*, pages 3519–3529, 2019.

[11] A. Laakso and G. Cottrell. Content and cluster analysis: assessing representational similarity in neural systems. *Philosophical psychology*, 13(1):47–76, 2000.

[12] M. L. Leavitt and A. Morcos. Towards falsifiable interpretability research. *arXiv preprint arXiv:2010.12016*, 2020.

[13] K. Lenc and A. Vedaldi. Understanding image representations by measuring their equivariance and equivalence, 2015.

[14] Y. Li, J. Yosinski, J. Clune, H. Lipson, and J. Hopcroft. Convergent learning: Do different neural networks learn the same representations? In *Feature Extraction: Modern Questions and Challenges*, pages 196–212. PMLR, 2015.

[15] Z. Li, E. Wallace, S. Shen, K. Lin, K. Keutzer, D. Klein, and J. Gonzalez. Train big, then compress: Rethinking model size for efficient training and inference of transformers. In *International Conference on Machine Learning*, pages 5958–5968. PMLR, 2020.

[16] R. Liang, T. Li, L. Li, J. Wang, and Q. Zhang. Knowledge consistency between neural networks and beyond. In *International Conference on Learning Representations*, 2019.

[17] R. T. McCoy, E. Pavlick, and T. Linzen. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. *arXiv preprint arXiv:1902.01007*, 2019.

[18] R. T. McCoy, J. Min, and T. Linzen. Berts of a feather do not generalize together: Large variability in generalization across models with similar test set performance. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 217–227, 2020.

[19] A. Morcos, M. Raghu, and S. Bengio. Insights on representational similarity in neural networks with canonical correlation. In *Advances in Neural Information Processing Systems*, pages 5727–5736, 2018.

[20] A. Naik, A. Ravichander, N. Sadeh, C. Rose, and G. Neubig. Stress test evaluation for natural language inference. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2340–2353, 2018.

[21] C. Olah, A. Mordvintsev, and L. Schubert. Feature visualization. *Distill*, 2017. doi: 10.23915/distill.00007. https://distill.pub/2017/feature-visualization.

[22] M. E. Peters, M. Neumann, L. Zettlemoyer, and W.-t. Yih. Dissecting contextual word embeddings: Architecture and representation. *arXiv preprint arXiv:1808.08949*, 2018.

[23] T. Pumir, A. Singer, and N. Boumal. The generalized orthogonal procrustes problem in the high noise regime. *Information and Inference: A Journal of the IMA*, Jan 2021. ISSN 2049-8772. doi: 10.1093/imaiai/iaaa035. URL http://dx.doi.org/10.1093/imaiai/iaaa035.

[24] A. Raghu, M. Raghu, S. Bengio, and O. Vinyals. Rapid learning or feature reuse? towards understanding the effectiveness of maml. In *International Conference on Learning Representations*, 2019.

[25] M. Raghu, J. Gilmer, J. Yosinski, and J. Sohl-Dickstein. Svcca: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. In *Advances in Neural Information Processing Systems*, pages 6076–6085, 2017.

[26] P. H. Schönemann. A generalized solution of the orthogonal procrustes problem. *Psychometrika*, 31(1):1–10, 1966.

[27] S. L. Smith, D. H. P. Turban, S. Hamblin, and N. Y. Hammerla. Offline bilingual word vectors, orthogonal transformations and the inverted softmax, 2017.

[28] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642, 2013.

[29] A. Tamkin, T. Singh, D. Giovanardi, and N. Goodman. Investigating transferability in pretrained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 1393–1401, 2020.

[30] I. Tenney, P. Xia, B. Chen, A. Wang, A. Poliak, R. T. McCoy, N. Kim, B. V. Durme, S. Bowman, D. Das, and E. Pavlick. What do you learn from context? probing for sentence structure in contextualized word representations. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=SJzSgnRcKX.

[31] I. Turc, M.-W. Chang, K. Lee, and K. Toutanova. Well-read students learn better: On the importance of pre-training compact models. *arXiv preprint arXiv:1908.08962*, 2019.

[32] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. Bowman. 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, 2018.

[33] L. Wang, L. Hu, J. Gu, Z. Hu, Y. Wu, K. He, and J. Hopcroft. Towards understanding learning representations: To what extent do different neural networks learn the same representation. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31, pages 9584–9593. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper/2018/file/5fc34ed307aac159a30d81181c99847e-Paper.pdf.

[34] A. Williams, N. Nangia, and S. Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122. Association for Computational Linguistics, 2018. URL http://aclweb.org/anthology/N18-1101.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Section 5.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See Section 5.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See Section 2, Section 4, and Appendix A for details to reproduce the experimental results.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Appendix A and B .
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See Figure 3, Appendix C and Appendix D.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Appendix A and B.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See Section 2 and Section 4
   (b) Did you mention the license of the assets? [Yes] See Appendix E
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]