Knowledge Distillation through Representational Alignment

Anonymous ACL submission

Abstract

Knowledge distillation is a common paradigm for transferring capabilities from a larger model to smaller models. Assuming white box access to the larger model, traditional knowledge distillation methods often draw a probabilistic measure over the activations and minimize a divergence measure between the larger and smaller model. These methods are often limited to last-layer activations, and do not leverage any meaningful information from representations included in the hidden layers. In this work, we propose a distillation method that explicitly utilizes popular measures of representational alignment: CKA and Shape. We show that our method yields statistically significant improvement (up to 2 percentage point and $p < 0.05$) over both fine-tuning and standard logits-based distillation on three tasks (CoLA, RTE and MRCP) of the GLUE benchmark.

1 Introduction

While large models are achieving state-of-the-art results across almost all vision and language tasks, the "emergent" abilities that are encapsulated in them (Wei et al., 2022; Liang et al., 2023b) are often inaccessible to the public as a result of their inherent size and operating costs. Knowledge Distillation (KD) is one of the many paradigms that aim to bridge the gap between size and performance by inducing ways of transferring knowledge and abilities from a larger, complex model (teacher) to a smaller and accessible model (student).

Assuming white-box access (weights and intermediate representations) to the teacher model during the training process, we can leverage alignment of the teacher-student model through not just their outputs, but also their intermediary representations. Prior works have minimized probabilistic divergences on the distributions of last-layer activations (Hinton et al., 2015; Wen et al., 2023) or used variants of Euclidean norms between student and teacher intermediary activations. (Sanh et al., 2020; Liang et al., 2023a; Tung and Mori, 2019; Sun et al., 2019; Mukherjee and Hassan Awadallah, 2020). Our work provides a framework that allows for intermediary representation in any arbitrary hidden layer of a neural network to be aligned between teacher and student models, taking the geometry of the representational space into account. We anticipate that this alignment in the representational geometry will bias the student model towards better downstream performance.

In picking the similarity function for aligning the representation, we draw from a wide literature in representational alignment (Sucholutsky et al., 2023), particularly with a focus on measuring and bridging the representational space between models (Klabunde et al., 2023). While a broad range of similarity functions have been proposed and used in the literature, we focus on using Centered Kernel Alignment (CKA) (Kornblith et al., 2019) and liner Shape (Williams et al., 2021) since they are both differentiable and invariant to orthogonal transformations. A differentiable metric can be backpropagated to align representations, while invariance to orthogonality is a commonly proposed symmetry of neural networks trained through gradient descent. (Chen et al., 1993; Orhan and Pitkow, 2018). We focus on cases where the student model is minimized using a combination of cross-entropy loss using labels and KL divergence between last layer logits, alongside the alignment of hidden representations. Our core contributions are summarized below:

1. We show that adding representational alignment in the distillation objective leads to a statistically significant improvement in accuracy (upto 2 percentage points) of the student model.

2. Adding more layers while calculating representational similarity leads to better perfor-

mance. CKA, in particular, scales much better when multiple layers are aligned.

2 Background

2.1 Distillation and divergences

The distillation process is usually done by gradient descent on a loss that minimizes the student target loss, as well as a secondary loss that incorporates the difference in the "knowledge" being transferred from the teacher to student model. Specifically, it takes the form of

$$\mathcal{L} = \mathcal{L}_{\text{CE}}(f_S(x), y) + \mathcal{L}_{KD}\left(f_T(x), f_S(x)\right) \quad (1)$$

where $f_S(X)$ and $f_T(x)$ are last-layer logits of the student and teacher model respectively, y is the true output labels. $\mathcal{L}_{KD}$ is the KL divergence between teacher and student logits and $\mathcal{L}_{CE}$ is the cross entropy loss of the student output.

Traditional knowledge-distillation methods have used either the forward (Sanh et al., 2020; Hinton et al., 2015) or reverse (Agarwal et al., 2024; Gu et al., 2024) KL divergence as the measure of difference between last-layer logits. It has been shown that even when student generalization improves, teacher-student fidelity is still low when knowledge distillation is performed on last-layer features. (Stanton et al., 2021)

Beyond alignment of the last-layer logits, hidden-layer representations can also be aligned. It is natural to assume that $\mathcal{L}_{KD}$ can take the form of any vector p-norm. Variants of Euclidean norms, including cosine-similarity (Sanh et al., 2020), normalized mean-squared, (Liang et al., 2023a; Sun et al., 2019) and ℓ^2 norms (Tung and Mori, 2019; Mukherjee and Hassan Awadallah, 2020) have been used in a distillation setting. An obvious advantage of this method is that, using a variety of higher order projection/dimensionality reduction methods on Euclidean spaces, (PCA, zero-padding, multi-dimensional scaling), cases where the number of activations in a student model is less than the teacher model are supported. However, the curse of dimensionality is a consistent problem when working with high-dimensional vectors. Similarly, Euclidean distances do not reflect the geometry of neural representational spaces, which are often invariant to permutations and orthogonality in the space of activation vectors. (Rombach et al., 2020). We are motivated to use a metric that, by its construction, is invariant to transformation of activations under certain groups.

2.2 Representational Similarity Metrics

Establishing a framework for comparing intermediate representations of neural networks is of significant implications to deeper analysis of neural network based models. Prior works in neuroscience have approached a similar problem in comparing representations of various stimuli to signals generated by the brain based on second order isometries of raw signals (Barrett et al., 2019; Kriegeskorte et al., 2008), while approaches in machine learning have traditionally focused on measures based on correlation analysis (Raghu et al., 2017).

Centered Kernel Alignment (CKA) (Kornblith et al., 2019) is a widely used measure of representational alignment that constructs a kernel similarity matrix and uses Hilbert-Schmidt Independence Criterion (HSIC) (Gretton et al., 2005a) to compute a metric between the similarity matrices. In the context of neural networks, Batched CKA (Nguyen et al., 2021), a slight reformulation of CKA with an unbiased estimator of HSIC (Song et al., 2012) is primarily used to construct a similarity index that is independent of batch size.

Shape metric (Williams et al., 2021; Duong et al., 2023) are a recently proposed extension of alignment based similarity measures, that enforce invariance in the measure with respect to orthogonal transformation group. They can be conceptualized as a similarity measure that works on second-order isometric equivalence, and their construction using ℓ_2 norms means that they are an appropriate choice of similarity metric to back propagate through for knowledge distillation.

By construction, CKA is invariant to both orthogonal transform and isometric scaling. Shape metric can be constructed to be invariant to all invertible linear transformation by preprocessing representations through a whitening transform. (Williams et al., 2021) In this work, due to computational constraints, we do not preprocess our representations. As a result, our implementation of Shape is only invariant to orthogonal transformations. A formal mathematical description of the similarity measures, their construction and invariance properties are included in Appendix A.

3 Methods

3.1 Dataset & Tasks

Our results are reported on the GLUE benchmark (Wang et al., 2018). Specifically, we use three tasks within GLUE: The Corpus of Linguistic Ac-

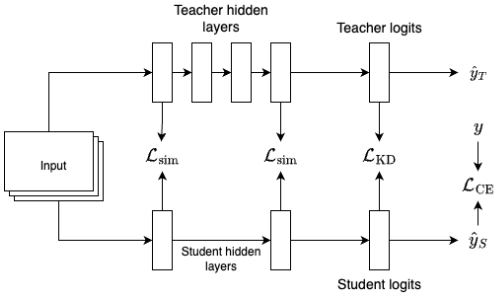

Figure 1: Diagram showing our distillation method. $\hat{y}_T$ is the output of the larger teacher model, $\hat{y}_S$ is the output of the smaller student model, and y are true output labels. $\mathcal{L}_{\text{sim}}$ is the alignment loss between hidden layers, $\mathcal{L}_{KD}$ is the KL divergence between teacher and student logits and $\mathcal{L}_{CE}$ is the cross entropy loss of the student output with respect to the true labels.

ceptability (CoLA) (Warstadt et al., 2019), The Microsoft Research Paraphrase Corpus (MRPC) (Dolan and Brockett, 2005) and The Recognizing Textual Entailment (RTE) (Dagan et al., 2005; Bar-Haim et al., 2006; Giampiccolo et al., 2007; Bentivogli et al., 2009). CoLA involves predicting whether a sequence of words is a grammatical English sentence, and is evaluated using Matthews correlation coefficient (MCC) (Matthews, 1975). MRPC contains two sentences and the task involves predicting if they are semantically equivalent. Since the dataset is imbalanced, we report both accuracy and F1 score. RTE involves an entailment challenge; given a premise sentence and a hypothesis sentence, the task is to predict whether the premise entails the hypothesis. We evaluate RTE using classification accuracy. These tasks were chosen from the 9 GLUE benchmark tasks because they had the greatest discrepancy in performance between teacher and student model after five epochs of fine-tuning.

3.2 Loss functions

Our loss function takes the form of

$$\mathcal{L} = \gamma \mathcal{L}_{CE}\left(f_S, \hat{y}\right) + \alpha \mathcal{L}_{\text{sim}}\left(\phi_T(f_T), \phi_S(f_S)\right) + (1-\alpha)\mathcal{L}_{\text{KD}}\left(f_S, f_T\right) \quad (2)$$

$\mathcal{L}_{CE}$ represents the cross entropy loss of the student logits with respect to output labels, $\mathcal{L}_{\text{sim}}$ represents the loss with respect to the representational similarity measuring function and $\mathcal{L}_{KD}$ is the KL divergence between student and teacher logits.

$\gamma \in \{0, 1\}$ indicates whether we are including supervised cross entropy loss, and $\alpha \in [0, 1]$ controls the interplay between hidden layer and last layer similarities. f_S and f_T are outputs, including hidden representations, of student and teacher models. ϕ is a function that extracts hidden layers from the model. For ease of notation, if $\phi_T = (a, b)$, it is extracting hidden representations from the a^{th} and b^{th} layers of the model.

3.3 Model and training details

We perform all our distillation tasks on the BERT model. (Devlin et al., 2019). As in common in most distillation studies, we use pre-trained BERT-large model, which has 24 encoder layers, as the teacher model and pre-trained BERT-base model with 12 layers as the student model. We fine-tune the pre-trained BERT-large model for 5 epochs on each task, and use this fine-tuned model as the teacher for distillation. The student is not fine-tuned on any tasks; the distillation begins with a pre-trained student model. For calculation of $\mathcal{L}_{\text{sim}}$, we zero pad the student hidden representations to match the dimension of the teacher representations.

To make experiments computationally viable, we use a token size of 128. We optimize using ADAM (Kingma and Ba, 2015) with a learning rate of 2×10^{-5} and a batch size per GPU of 64. We use Hugging Face libraries (Wolf et al., 2020) to perform all our training and evaluation. We run distillation across the three tasks for 6 epochs. Each training run required optimizing over 108,311,810 parameters. Furthermore, to ensure statistical significance in the performance of our distilled model, we use McNemar's test (McNemar, 1947; Dietterich, 1998) to compare all distilled models against the fine-tuned baseline. Unless otherwise noted, all results reported are statistically significant ($p < 0.05$)

α	γ	$\mathcal{L}_{\text{sim}}$	Acc/F1	Remarks
N/A	N/A	N/A	0.68/0.809	RD baseline
N/A	1	N/A	0.816/0.877	FT baseline
0	0	N/A	0.813/0.866 †	KD baseline
0.6	0	Shape	0.791/0.859	Shape+KD
1	0	Shape	0.683/0.812	Shape only
0.6	0	CKA	0.811/0.873	CKA+KD
1	0	CKA	0.683/0.812	CKA only
0.6	1	Shape	**0.835/0.887**	Shape+KD+FT
0.6	1	CKA	0.813/0.846	CKA+KD+FT

Table 1: Performance on MRPC. **RD**: Random baseline, **FT**: Fine-tuning on labels, **KD**: Distillation on KL divergence of the last layer logits. † indicates cases when statistical significance is broken ($p \geq 0.05$)

4 Results & Discussion

α	γ	$\mathcal{L}_{\text{sim}}$	MCC	Remarks
N/A	N/A	N/A	0.0	RD baseline
N/A	1	N/A	0.5702	FT baseline
0	0	N/A	0.5752	KD baseline
0.6	0	Shape	0.5103	Shape+KD
1	0	Shape	0.1194	Shape only
0.6	0	CKA	0.5803	CKA+KD
1	0	CKA	0.1066	CKA only
0.6	1	Shape	0.5497	Shape+KD+FT
0.6	1	CKA	**0.5804**	CKA+KD+FT

Table 2: Performance on CoLA. **RD**: Random baseline, **FT**: Fine-tuning on labels, **KD**: Distillation on KL divergence of the last layer logits.

4.1 Distillation performance

For all tasks in this section, we assume $\phi_T = (12)$ and $\phi_S = (6)$, i.e we are aligning the middle layer of the teacher model with the middle layer of the student model. All results are noted after minimizing the loss function from Equation 2 with values varying for α, γ and $\mathcal{L}_{sim}$.

Alignment can help improve distillation:

As shown in Table 1, 2, 3 and , including $\mathcal{L}_{\text{sim}}$ alongside $\mathcal{L}_{KD}$ and $\mathcal{L}_{CE}$ increases the performance of the student model across all three tasks. Shape does better in RTE and MRPC, while CKA produces the best student model in CoLA. It is interesting to note that adding similarity measures alongside logits distillation, without even including cross entropy of the labels ($\alpha = 0.6, \gamma = 0$), seems to do better than boths logits distillation and fine-tuning.

Alignment, by itself, is disastrous

When we remove $\mathcal{L}_{KD}$ and $\mathcal{L}_{CE}$ entirely ($\alpha = 1, \gamma = 0$) we see that the performance is significantly worse across all tasks and similarity functions. While leveraging the geometry of hidden representations can steer the student model towards producing the correct output, it cannot by itself bias the model to produce the correct output. Some output information, either through teacher logits or supervised labels, are essential to ensure the model performs well.

α	γ	$\mathcal{L}_{\text{sim}}$	Accuracy	Remarks
N/A	N/A	N/A	0	RD baseline
N/A	1	N/A	0.6173	FT baseline
0	0	N/A	0.6389 †	KD baseline
0.6	0	Shape	0.6337	Shape+KD
1	0	Shape	0.5631	Shape only
0.6	0	CKA	0.6462 †	CKA+KD
1	0	CKA	0.4729	CKA only
0.6	1	Shape	**0.6570**	Shape+KD+FT
0.6	1	CKA	0.6462 †	CKA+KD+FT

Table 3: Performance on RTE. **RD**: Random baseline, **FT**: Fine-tuning on labels, **KD**: Distillation on KL divergence of the last layer logits. † indicates cases when statistical significance is broken ($p \geq 0.05$)

Task	$\mathcal{L}_{sim}$	ϕ_T	ϕ_S	Score
CoLA	CKA	(12)	(6)	0.5803
		(6, 12, 18)	(3,6,9)	**0.5804**
	Shape	(12)	(6)	0.5103
		(6, 12, 18)	(3,6,9)	0.5179
RTE	CKA	(12)	(6)	0.6462 †
		(6, 12, 18)	(3,6,9)	**0.6823**
	Shape	(12)	(6)	0.6337
		(6, 12, 18)	(3,6,9)	0.6606
MRPC	CKA	(12)	(6)	0.8112/0.8739
		(6, 12, 18)	(3,6,9)	**0.8406/0.8896**
	Shape	(12)	(6)	0.7916/0.8595
		(6, 12, 18)	(3,6,9)	0.8357/0.8885

Table 4: Changes in distillation performance while adding layers.† indicates cases when statistical significance is broken ($p \geq 0.05$)

4.2 Layer by layer performance

In this section, we use the previous results and set $\alpha = 0.6$ and $\gamma = 0$. We change ϕ_T and ϕ_S to observe the impact of adding more layers during the calculation of $\mathcal{L}_{\text{sim}}$. To ensure appropriate layers are matched, we match layer n of the student model with layer $2n$ of the teacher model. The first third, middle and second third model are matched. As seen from the results in Table 4, for both shape and CKA, going from aligning a single layer to three layers increases the performance of the distilled student model. In fact, CKA tends to scale much better with a greater number of layers, resulting in the best performance across all three tasks.

5 Conclusion

We propose a novel distillation method that incorporates representations of hidden layers and aligns them using two measures of representational similarity: CKA and shape. We showed that adding these measures besides divergence of teacher-student last layer logits or standard cross entropy with labels can yield better performance, however alignment by itself cannot steer distillation towards the correct output. We also showed that adding the number of layers in the calculation of the similarity leads to performance improvements, particularly in the context of CKA.

6 Limitations

- **Generalization to other models and tasks** Our analysis have been carried out using BERT on three tasks of the GLUE dataset. Analysis on further datasets with models of varying capability would lead to a stronger argument about the efficacy of representational alignment for distillation.

- **Limitations with CKA**: Linear CKA has been previously shown to be sensitive to outlier data points (Nguyen et al., 2022), and high variance principal components in the representations (Ding et al., 2021), while theoretical analysis shows that CKA is sensitive to subset translation (Davari et al., 2023). These studies point out that using just linear CKA as a proxy for model similarity can be flawed. Since we're not using CKA to infer the representational capabilities of models, but instead using it as an intermediary measure to that can be optimized to improve end-to-end distillation performance, we believe some of these issues raised in these works do not apply to our method. However, it is important to be aware of the limitations in using CKA.

- **Runtime considerations**: Computing and optimizing over the representational metrics is extremely time-consuming. Shape, for instance, requires computing the SVD of the covariance matrices of the representations, which is in $\mathcal{O}(n^3)$ on the size of representations. This means that without further work on more efficient calculation of these measures, our method cannot be scaled up to larger models and datasets.

7 Ethical Considerations

Our work proposes a framework for better distillation of larger inaccessible models into smaller, more accessible ones. We intend this work to contribute to a larger process of democratizing access to the impressive abilities of larger models, allowing for the deployment of these models in a resource-constrained settings. However, if the teacher model has inherent biases or has been trained with malicious intent, these biases can be propagated to the student model. Special care must be taken, prior to distillation, to ensure that the teacher model is fair and unbiased.

References

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. 2024. On-policy distillation of language models: Learning from self-generated mistakes. In *The Twelfth International Conference on Learning Representations*.

Roy Bar-Haim, Ido Dagan, Bill Dolan, Lisa Ferro, Danilo Giampiccolo, Bernardo Magnini, and Idan Szpektor. 2006. The second pascal recognising textual entailment challenge. In *Proceedings of the second PASCAL challenges workshop on recognising textual entailment*, volume 1. Citeseer.

David GT Barrett, Ari S Morcos, and Jakob H Macke. 2019. Analyzing biological and artificial neural networks: challenges with opportunities for synergy? *Current Opinion in Neurobiology*, 55:55–64. Machine Learning, Big Data, and Neuroscience.

Luisa Bentivogli, Peter Clark, Ido Dagan, and Danilo Giampiccolo. 2009. The fifth pascal recognizing textual entailment challenge. *TAC*, 7(8):1.

An Mei Chen, Haw-minn Lu, and Robert Hecht-Nielsen. 1993. On the geometry of feedforward neural network error surfaces. *Neural computation*, 5(6):910–927.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The pascal recognising textual entailment challenge. In *Machine learning challenges workshop*, pages 177–190. Springer.

MohammadReza Davari, Stefan Horoi, Amine Natik, Guillaume Lajoie, Guy Wolf, and Eugene Belilovsky. 2023. Reliability of CKA as a similarity measure in deep learning. In *The Eleventh International Conference on Learning Representations*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Thomas G Dietterich. 1998. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural computation*, 10(7):1895–1923.

Frances Ding, Jean-Stanislas Denain, and Jacob Steinhardt. 2021. Grounding representation similarity through statistical testing. In *Advances in Neural Information Processing Systems*.

Bill Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Third international workshop on paraphrasing (IWP2005)*.

Lyndon Duong, Jingyang Zhou, Josue Nassar, Jules Berman, Jeroen Olieslagers, and Alex H Williams. 2023. Representational dissimilarity metric spaces for stochastic neural networks. In *The Eleventh International Conference on Learning Representations*.

Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and William B Dolan. 2007. The third pascal recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL workshop on textual entailment and paraphrasing*, pages 1–9.

Arthur Gretton, Olivier Bousquet, Alex Smola, and Bernhard Schölkopf. 2005a. Measuring statistical dependence with hilbert-schmidt norms. In *Algorithmic Learning Theory*, pages 63–77, Berlin, Heidelberg. Springer Berlin Heidelberg.

Arthur Gretton, Olivier Bousquet, Alex Smola, and Bernhard Schölkopf. 2005b. Measuring statistical dependence with hilbert-schmidt norms. In *International Conference on Algorithmic Learning Theory*.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. 2024. MiniLLM: Knowledge distillation of large language models. In *The Twelfth International Conference on Learning Representations*.

Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the knowledge in a neural network. In *NIPS Deep Learning and Representation Learning Workshop*.

David Kendall. 1989. A survey of the statistical theory of shape. *Statistical Science*, 4:87–99.

Agnan Kessy, Alex Lewin, and Korbinian Strimmer. 2018. Optimal whitening and decorrelation. *The American Statistician*, 72(4):309–314.

Diederik P Kingma and Jimmy Lei Ba. 2015. Adam: A method for stochastic gradient descent. In *ICLR: international conference on learning representations*, pages 1–15. ICLR US.

Max Klabunde, Tobias Schumacher, Markus Strohmaier, and Florian Lemmerich. 2023. Similarity of neural network models: A survey of functional and representational measures. *arXiv preprint arXiv:2305.06329*.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. 2019. Similarity of neural network representations revisited. In *International conference on machine learning*, pages 3519–3529. PMLR.

Nikolaus Kriegeskorte, Marieke Mur, and Peter A. Bandettini. 2008. Representational similarity analysis – connecting the branches of systems neuroscience. *Frontiers in Systems Neuroscience*, 2.

Chen Liang, Simiao Zuo, Qingru Zhang, Pengcheng He, Weizhu Chen, and Tuo Zhao. 2023a. Less is more: Task-aware layer-wise distillation for language model compression. In *International Conference on Machine Learning*, pages 20852–20867. PMLR.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, Benjamin Newman, Binhang Yuan, Bobby Yan, Ce Zhang, Christian Alexander Cosgrove, Christopher D Manning, Christopher Re, Diana Acosta-Navas, Drew Arad Hudson, Eric Zelikman, Esin Durmus, Faisal Ladhak, Frieda Rong, Hongyu Ren, Huaxiu Yao, Jue WANG, Keshav Santhanam, Laurel Orr, Lucia Zheng, Mert Yuksekgonul, Mirac Suzgun, Nathan Kim, Neel Guha, Niladri S. Chatterji, Omar Khattab, Peter Henderson, Qian Huang, Ryan Andrew Chi, Sang Michael Xie, Shibani Santurkar, Surya Ganguli, Tatsunori Hashimoto, Thomas Icard, Tianyi Zhang, Vishrav Chaudhary, William Wang, Xuechen Li, Yifan Mai, Yuhui Zhang, and Yuta Koreeda. 2023b. Holistic evaluation of language models. *Transactions on Machine Learning Research*. Featured Certification, Expert Certification.

Brian W Matthews. 1975. Comparison of the predicted and observed secondary structure of t4 phage lysozyme. *Biochimica et Biophysica Acta (BBA)-Protein Structure*, 405(2):442–451.

Quinn McNemar. 1947. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157.

Subhabrata Mukherjee and Ahmed Hassan Awadallah. 2020. XtremeDistil: Multi-stage distillation for massive multilingual models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2221–2234, Online. Association for Computational Linguistics.

Thao Nguyen, Maithra Raghu, and Simon Kornblith. 2021. Do wide and deep networks learn the same things? uncovering how neural network representations vary with width and depth. In *International Conference on Learning Representations*.

Thao Nguyen, Maithra Raghu, and Simon Kornblith. 2022. On the origins of the block structure phenomenon in neural network representations. *Preprint*, arXiv:2202.07184.

Emin Orhan and Xaq Pitkow. 2018. Skip connections eliminate singularities. In *International Conference on Learning Representations*.

Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Narain Sohl-Dickstein. 2017. Svcca: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. In *Neural Information Processing Systems*.

Robin Rombach, Patrick Esser, and Björn Ommer. 2020. Making sense of cnns: Interpreting deep representations & their invariances with inns. In *Proceedings of the European Conference on Computer Vision*.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2020. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *Preprint*, arXiv:1910.01108.

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

Le Song, Alex Smola, Arthur Gretton, Justin Bedo, and Karsten Borgwardt. 2012. Feature selection via dependence maximization. *Journal of Machine Learning Research*, 13(5).

Samuel Stanton, Pavel Izmailov, Polina Kirichenko, Alexander A Alemi, and Andrew G Wilson. 2021. Does knowledge distillation really work? *Advances in Neural Information Processing Systems*, 34:6906–6919.

Ilia Sucholutsky, Lukas Muttenthaler, Adrian Weller, Andi Peng, Andreea Bobu, Been Kim, Bradley C Love, Erin Grant, Jascha Achterberg, Joshua B Tenenbaum, et al. 2023. Getting aligned on representational alignment. *arXiv preprint arXiv:2310.13018*.

S. Sun, Yu Cheng, Zhe Gan, and Jingjing Liu. 2019. Patient knowledge distillation for bert model compression. In *Conference on Empirical Methods in Natural Language Processing*.

Frederick Tung and Greg Mori. 2019. Similarity-preserving knowledge distillation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1365–1374.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*.

Alex Warstadt, Amanpreet Singh, and Samuel R Bowman. 2019. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022. Emergent abilities of large language models. *Transactions on Machine Learning Research*. Survey Certification.

Yuqiao Wen, Zichao Li, Wenyu Du, and Lili Mou. 2023. f-divergence minimization for sequence-level knowledge distillation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10817–10834, Toronto, Canada. Association for Computational Linguistics.

Alex H. Williams, Erin M. Kunz, Simon Kornblith, and Scott W. Linderman. 2021. Generalized shape metrics on neural representations. *Advances in neural information processing systems*, 34:4738–4750.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

A Supplemental Material

Notation

Consider $\mathbb{R}^d$ to a d-dimensional activation from a particular hidden layer. A representational matrix, $R \in \mathbb{R}^{n \times d}$ is a collection of activations from n different inputs. The similarity measure can be formulated as a function $m : \mathbb{R}^{n \times d} \times \mathbb{R}^{n \times d} \to \mathbb{R}$. We will use $\|F\|$ to represent the Frobenius norm is F is a matrix and the Euclidean 2-norm if F is a vector. $\phi : \mathbb{R}^{n \times d} \to \mathbb{R}^{n \times p}$ is a pre-processing function applied to the rows of the representation matrix. ϕ can be common pre-processing functions that preserve the same dimension (mean-centering, standard scaling, etc), in which case $p = d$. $p << d$, i.e pre-processing by applying a dimensionality reduction function like PCA. The group for orthogonal transformations for a d dimensional vector is given as $\mathcal{O}(d) = \{Q \in \mathbb{R}^{d \times d} : Q^T Q = I\}$

A.1 $\mathcal{L}_{\text{sim}}$: Shape

Generalized shape metrics, as presented in (Williams et al., 2021), uses the theory of statistical shape (Kendall, 1989) to create a similarity function that is a metric in the representational shape space. The similarity between representations is defined with respect to a linear isometry group, $\mathcal{G}$. We define the equivalence relation $\phi(R_x) \sim \phi(R_y) \iff \exists T \in \mathcal{G} : \phi(R_x) = \phi(R_y)T$, where ϕ is a preprocessing function. The similarity score is then calculated as

$$d(R_x, R_y) = \min_{T \in \mathcal{G}} \|\phi(R_x) - \phi(R_y)T\| \quad (3)$$

The orthogonal transformation group in the feature space dimension $\mathcal{O}(d)$ is a commonly used isometry group, however simpler permutation groups can be also be used. Shape metrics can thus be conceptually thought of as a second order isometry on neural network representations, accounting for any first-order differences in the raw activations.

These metrics follow the standard properties of the distance function, including the triangle inequality.

Orthogonal Procrustes Problem

The problem of computing the T that optimizes the $\|R_x - R_y T\|$ when $G = \mathcal{O}(d)$ is solved by (Schönemann, 1966). In fact, we can show that $T = VU^T$, where $R_x^T R_y = U\Sigma V^T$ is the Singular Value Decomposition. Furthermore, $\langle R_x, R_y \rangle = \sum_i \sigma_i$, where $\sigma_1 \geq \sigma_2 \cdots \geq \sigma_n \geq 0$ are the singular values of $R_x^T R_y$.

Invariances in Shape

It is clear that when the $\phi(x) = x$, the shape metric is only invariant to orthogonal transformation. By using the linear whitening transform as the preprocessing function, we can control the functional group our metric is invariant to. The whitening transform takes the form of

$$\phi(R) = CR\left(\beta I_d + (1-\beta)(R^T CR)^{-\frac{1}{2}}\right) \quad (4)$$

where $C = I_n - \frac{1}{n}\mathbf{1}\mathbf{1}^T$ is the $n \times n$ centering matrix, that mean-centers the columns of the representational matrix. When $\beta = 1$, Eq 4, reduces to invariance to orthogonal groups only, since $\phi(R) = CR$ is simply mean-centering the columns. On the other hand, with $\beta = 0$, the $\phi(R) = CR(R^T CR)^{-\frac{1}{2}}$, which is equivalent to ZCA whitening. (Kessy et al., 2018). In this case, all invertible linear transformations are equivalent in the representation; thus the shape metric is invariant to all linear transformations. β is thus an important hyperparameter that we can tune to adjust the strength of our isometry group.

In our implementation of shape, to ease the computational complexity of backpropagating through the metric, we preprocess our representations by setting $\beta = 1$ in Equation 4. As a result, we are only invoking orthogonal invariance in the intermediary representations.

Computational constraints

Computing the SVD of $R_x^T R_y$ takes $\mathcal{O}(n^3)$ time. Classical divergence based approaches and Euclidean distances are often $\mathcal{O}(n)$, so the overhead while gradient descending through a metric calculated by solving the orthogonal Procrustes can be quite expensive.

A.2 $\mathcal{L}_{\text{sim}}$: Centered Kernel Alignment (CKA)

Centered Kernel Alignment, proposed in (Kornblith et al., 2019), draws from older literature studying Representational Similarity Analysis (RSA) in neuroscience (Kriegeskorte et al., 2008). The core idea in both lies in computing a similarly matrix of pairwise activations of each sample, $K_x, K_y \in \mathbb{R}^{n \times n}$. While these matrices can take the form of positive semi definite matrices through a kernel function, and have a rich mathematical structure based on the theory of Reproducing Kernel Hilbert Spaces (RKHS), we limit ourselves to linear kernels. So, we will define $K_x = CR_x R_x^T C$ and $K_y = CR_y R_y^T C$, as centered similarity matrices, where $C = I_n - \frac{1}{n}\mathbf{1}\mathbf{1}^T$ is the $n \times n$ centering matrix.

HSIC and computation of the metric

Hilbert-Schmidt Independence Criterion (HSIC) (Gretton et al., 2005b) is used as a way to compare the two similarity matrices. HSIC can be conceptualized as a generalization of the covariance operation in the context RKHS. For the linear kernel that we are using, the empirical estimator for HSIC takes the form

$$\text{HSIC}(K_x, K_y) = \frac{1}{(n-1)^2}\text{tr}(K_x K_y) \quad (5)$$

However, this estimator of HSIC is biased, and it is impossible to calculate the HSIC of the entire dataset at once. To ensure that the calculated CKA is independent of batch size, we instead use an unbiased estimator of HSIC in our implementation. (Song et al., 2012; Nguyen et al., 2021)

$$\widetilde{\text{HSIC}}(K_x, K_y) = \frac{1}{n(n-3)}\Bigg(\text{tr}(\tilde{K_x}\tilde{K_y}) + \frac{\mathbf{1}^T\tilde{K_x}\mathbf{1}\mathbf{1}^T\tilde{K_y}\mathbf{1}}{(n-1)(n-2)} - \frac{2}{n-2}\mathbf{1}^T\tilde{K_x}\tilde{K_y}\mathbf{1}\Bigg) \quad (6)$$

where $\tilde{K_x}$ and $\tilde{K_y}$ are hollow matrices obtained by setting the diagonal of K_x and K_y to 0.

The CKA value is then calculated as

$$\text{CKA}(K_x, K_y) = \frac{\widetilde{\text{HSIC}}(K_x, K_y)}{\sqrt{\widetilde{\text{HSIC}}(K_x, K_x)\widetilde{\text{HSIC}}(K_y, K_y)}} \quad (7)$$

Invariances in CKA

CKA is invariant to both isotropic scaling and orthogonal transformation. HSIC, by itself, is not

invariant to isotropic scaling. However, the normalization with self HSIC in Equation 7 means that CKA will be invariant to isotropic scaling since the trace as well as all matrix multiplications are linear operators.

Orthogonal invariance in CKA can be seen in the construction of K_x and K_y. For instance when a representation, R_y is transformed through $Q \in \mathcal{O}(d)$, the linear kernel similarity matrix takes the form of

$$\begin{aligned} K_{R_yQ} &= CR_yQ(R_yQ)^TC \\ &= CR_yQQ^TR_y^TC \\ &= CR_yR_Y^TC = K_y \end{aligned}$$

Hence, construction of the similarity kernels are invariant to orthogonal transformation of the representation, and thus the CKA score also remains invariant.