Supplementary Material: Ranking Neural Checkpoints

Yandong Li [1,2*] Xuhui Jia[2] Ruoxin Sang[2]
Yukun Zhu[2] Bradley Green[2] Liqiang Wang[1] Boqing Gong[2]
[1]University of Central Florida [2]Google
lyndon.leeseu@outlook.com lwang@cs.ucf.edu {xhjia,rxsang,yukun,brg,bgong}@google.com

A. Appendix

In this appendix, we provide the following details to support the main text:

Section A.1: Descriptions of the 4 downstream tasks.

Section A.2: Training details of pre-training and fine-tuning.

Section A.3: Comparison results on the combined group of checkpoints in Groups I, II and III.

Section A.4: Another group of checkpoints with ResNet101s at different pre-training stages.

Section A.5: More experiment results on Groups I-IV.

Section A.6: Neural checkpoints ranking on object detection and instance segmentation.

A.1. Downstream tasks

In this section, we describe the datasets used for the downstream tasks as shown in Table 1. More specifically, **Caltech101** [2] contains 101 classes, including animals, airplanes, chairs and etc, the image size varies from 200 to 300 pixels per edge. **Flowers102** [6] have 102 classes, with 40 to 248 training images per class, each image has at least 500 pixels. **Patch Camelyon** [10] contains 327,680 images of histopathologic scans of lymph node sections with image size of 96x96, which is collected to predict the presence of metastatic tissue. **Sun397** [11] is a scenery benchmark with 397 classes, including cathedral, staircase, shelter, river, or archipelago. There are at least 100 images per class. The images are in 200x200 or higher resolutions. We believe the dataset portfolio well represents a broad set of vision tasks.

A.2. Hyper-parameter Sweep

We adopt the similar experiment setting as in [12] to fine-tune the neural networks on the downstream tasks. Specifically, we set the batch size to 512 and use SGD with momentum of 0.9. We do not use weight decay for fine-tuning, and we set it to be 0.01 times the learning rate [4] when training from scratch. We perform per-task hyper-parameter search. For each task, we sweep the learning rate in $\{0.0001, 0.001, 0.005, 0.01, 0.05, 0.1, 0.2, 0.5\}$ and the training steps in $\{2500, 5000, 10000, 15000, 20000, 400000\}$. We incorporate inception data augmentation [8] for pre-training checkpoints and we do not use data-augmentation when we fine-tune the neural networks on the downstream tasks to emphasize the effect of transfer learning.

A.3. Comparison results on all checkpoints in Groups I, II, III

To obtain a comprehensive analysis, we also consolidate the checkpoints from Group I, II and III into one group (including 41 checkpoints in total) and then apply the ranking methods on it. Table 2 shows the comparison results. The results further evaluate our observations in Section 4 of the main text. $\mathcal{N}$LEEP performs consistently well on the big group of checkpoints with lowest computation cost. Linear separability of the feature representation is also a good indicator for ranking a large group of neural checkpoints. Fine-tuning with early stopping and mutual information estimator produce poor correlations. The ranking qualities of different ranking methods on the large group of checkpoints are in sharper contrast than on small groups. For instance, the Pearson's r of $\mathcal{N}$LEEP *vs.* Finetune (5 epochs) on the large group is 83.71 *vs.* 27.84 but they perform 72.84 *vs.* 68.47 on Group II (Table 2 in the main text). It indicates that $\mathcal{N}$LEEP is a low-variance and low-bias checkpoint ranking estimator, while early stopping may produce high-variance ranking results.

A.4. Group IV: Supervised ResNet101s

We incorporate another group of checkpoints, including 12 ResNet101 [3] models pre-trained by fully supervised learning on ImageNet [1], iNaturalist [9], and Places-365 [13]. We obtain the checkpoints in the same way as we have done for Group II, but with ResNet101 architecture. We want to study how different model architecture and model size affect the ranking quality.

Figure 1 and Table 7 show the fine-tuning accuracy on 4 downstream tasks. The relative fine-tuning accuracies are similar to the accuracies on Group II. We also observe that a converged checkpoint does not necessarily demonstrate the best performance on the downstream tasks (cf. Img-270k is better than Img-300k on Flowers102 [6]). Table 3 shows the comparison results of ranking methods on those checkpoints. The relative performance among the ranking methods is similar to what they do in Group II (Table 2 in the main text). Except that they perform better on ResNet101s, e.g., Linear (converged) can achieve 68.60 in terms of Kendall's τ on ResNet50s versus 73.48 on ResNet101s, $\mathcal{N}$LEEP can get 72.84 in terms of Pearson's r on ResNet50s versus 83.22 on ResNet101s. The observation reveals that the ranking of deeper checkpoints may be more predictable than shallow ones.

A.5. More experimental results on Groups I-IV

We show more comparison results on NeuCRaB in this section. Figures 2 and 3 show the best fine-tuning accuracies offset by their mean (for better visualization) on Groups II and III, respectively. Table 4, 5, 6, 7 demonstrate the absolute best fine-tuning accuracies on Groups I-IV, respectively.

A.6. Neural checkpoint ranking for object detection and instance segmentation

We also evaluate on object detection and instance/semantic segmentation and show the results in Tables 8. Specifically, we incorporate the recent self-supervised MoCo models (MoCov1, MoCov2, MoCov2-800epoch) and a ResNet50 model (supervised pretrained on ImageNet) into a new group of checkpoints. We evaluate checkpoint ranking on Pascal VOC (object detection) and Cityscapes (instance segmentation). In order to adapt $\mathcal{N}$LEEP to detection and segmentation tasks, we assign multiple ground truth labels for one image if it includes multiple object categories and extract the image-level features to perform GMM. We adapt $\mathcal{N}$LEEP to detection and segmentation tasks by assigning multi-labels to images with multiple object categories. The experiment results demonstrate that $\mathcal{N}$LEEP consistently outperforms the fine-tune and linear evaluation based approachs. We plan to include more diverse downstream tasks in NeuCRaB to facilitate future research.

References

[1] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[2] Li Fei-Fei, Rob Fergus, and Pietro Perona. One-shot learning of object categories. *IEEE transactions on pattern analysis and machine intelligence*, 28(4):594–611, 2006.

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[4] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[5] Cuong V Nguyen, Tal Hassner, Cedric Archambeau, and Matthias Seeger. Leep: A new measure to evaluate transferability of learned representations. *arXiv preprint arXiv:2002.12462*, 2020.

[6] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pages 722–729. IEEE, 2008.

[7] Ben Poole, Sherjil Ozair, Aaron van den Oord, Alexander A Alemi, and George Tucker. On variational bounds of mutual information. *arXiv preprint arXiv:1905.06922*, 2019.

[8] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alex Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. *arXiv preprint arXiv:1602.07261*, 2016.

[9] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8769–8778, 2018.

[10] Bastiaan S Veeling, Jasper Linmans, Jim Winkens, Taco Cohen, and Max Welling. Rotation equivariant cnns for digital pathology. In *International Conference on Medical image computing and computer-assisted intervention*, pages 210–218. Springer, 2018.

[11] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3485–3492. IEEE, 2010.

[12] Xiaohua Zhai, Joan Puigcerver, Alexander Kolesnikov, Pierre Ruyssen, Carlos Riquelme, Mario Lucic, Josip Djolonga, Andre Susano Pinto, Maxim Neumann, Alexey Dosovitskiy, et al. The visual task adaptation benchmark. *arXiv preprint arXiv:1910.04867*, 2019.

[13] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

Dataset	Training	Evaluation	Number of Classes
Caltech101 [2]	3060	6084	101
Flower102 [6]	2040	6149	102
Patch-Camelyon [10]	262144	32768	2
Sun397 [11]	76128	10875	397

Table 1. Statistics of the datasets associated with the downstream tasks

Method	Recall@1	Rel@1	Recall@3	Rel@3	Pearson	Kendall	GFLOPS
Linear (1 epoch)	0.00	99.13	25.00	99.46	22.30	13.42	4.45E4
Linear (5 epochs)	0.00	99.13	25.00	99.21	42.99	31.64	4.47E4
Linear (converged)	25.00	99.42	**50.00**	99.73	76.22	61.22	4.79E4
Fine-tune (1 epoch)	0.00	96.69	0.00	98.16	3.84	6.50	5.85E5
Fine-tune (5 epochs)	0.00	**99.49**	0.00	99.49	27.20	27.16	3.84E6
MI (α=0.01) [7]	0.00	77.50	0.00	81.44	1.12	7.16	1.52E5
MI (α=0.50)	0.00	66.51	0.00	90.07	-4.05	-14.22	1.52E5
MI w/ PCA (α=0.01)	0.00	89.18	**50.00**	**99.84**	12.14	20.99	5.57E4
MI w/ PCA (α=0.50)	0.00	97.07	0.00	98.70	-14.03	-2.39	5.57E4
LEEP [5]	–	–	–	–	–	–	–
$\mathcal{N}$LEEP	**50.00**	99.47	**50.00**	99.78	**83.71**	**68.18**	**12.86**

Table 2. Comparison results on all checkpoints in Group I, II, III (GFLOPS excludes a forward pass on training data, which takes 2.73E5 GFLOPS shared by all).

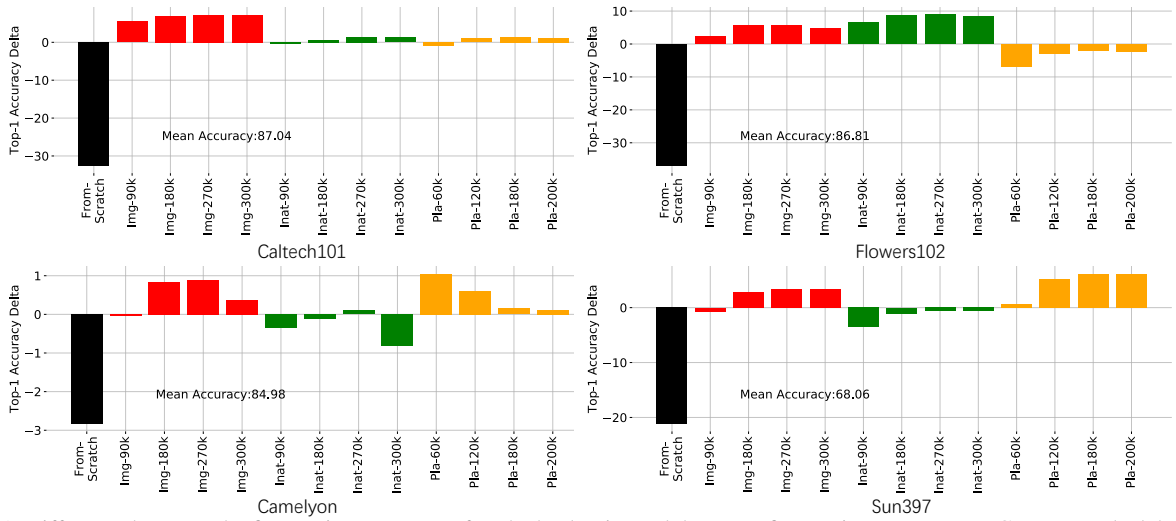

Figure 1. Difference between the fine-tuning accuracy of each checkpoint and the mean fine-tuning accuracy on Group IV. Black bar means From-Scratch. Red, green and orange bars represent ImageNet models, iNaturalist models and Places365 models, respectively. Img-90k means the checkpoint obtained by early stopping at the 90k-th iteration on ImageNet, and so on.

Method	Recall@1	Rel@1	Recall@3	Rel@3	Pearson	Kendall	GFLOPS
Linear (1 epoch)	0.00	98.58	25.00	98.99	46.75	27.27	1.021E5
Linear (5 epochs)	0.00	98.72	75.00	99.95	59.27	41.32	1.023E5
Linear (converged)	25.00	99.81	75.00	99.95	82.17	73.48	1.06E5
Fine-tune (1 epoch)	0.00	96.19	25.00	99.34	29.64	21.21	1.34E6
Fine-tune (5 epochs)	**75.00**	**99.98**	75.00	99.94	69.19	50.00	8.81E6
MI (α=0.01) [7]	0.00	97.25	75.00	98.46	12.96	13.21	1.62E5
MI (α=0.50)	25.00	98.60	50.00	99.54	30.16	18.21	1.62E5
MI w/ PCA (α=0.01)	0.00	99.85	75.00	99.95	51.85	48.91	5.58E4
MI w/ PCA (α=0.50)	0.00	95.99	50.00	98.41	48.64	44.31	5.58E4
LEEP [5]	25.00	99.52	75.00	99.72	54.54	46.43	378.31
$\mathcal{N}$LEEP	**75.00**	**99.98**	**100.00**	**100.00**	**83.22**	**73.80**	**12.95**

Table 3. Comparison results on Group IV (GFLOPS excludes a forward pass on training data, which takes 6.27E5 GFLOPS shared by all).

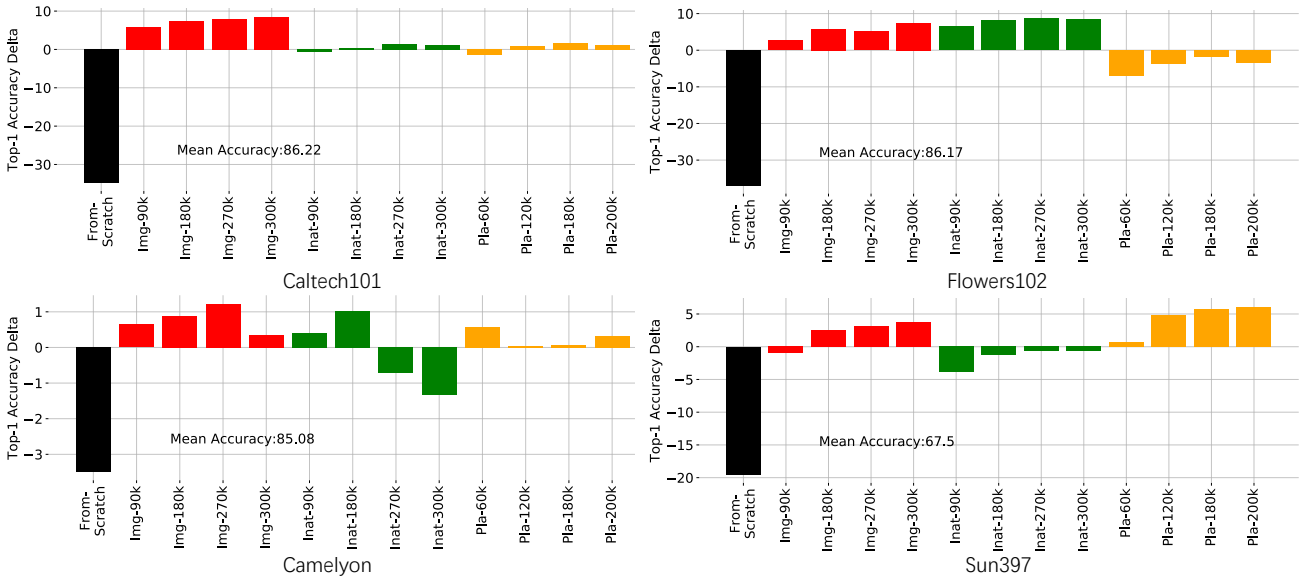

Figure 2. Difference between the fine-tuning accuracy of each checkpoint and the mean fine-tuning accuracy on Group II. Black bar means From-Scratch. Red, green and orange bars represent ImageNet models, iNaturalist models and Places365 models, respectively. Img-90k means the checkpoint obtained by early stopping at the 90k-th iteration on ImageNet, and so on.

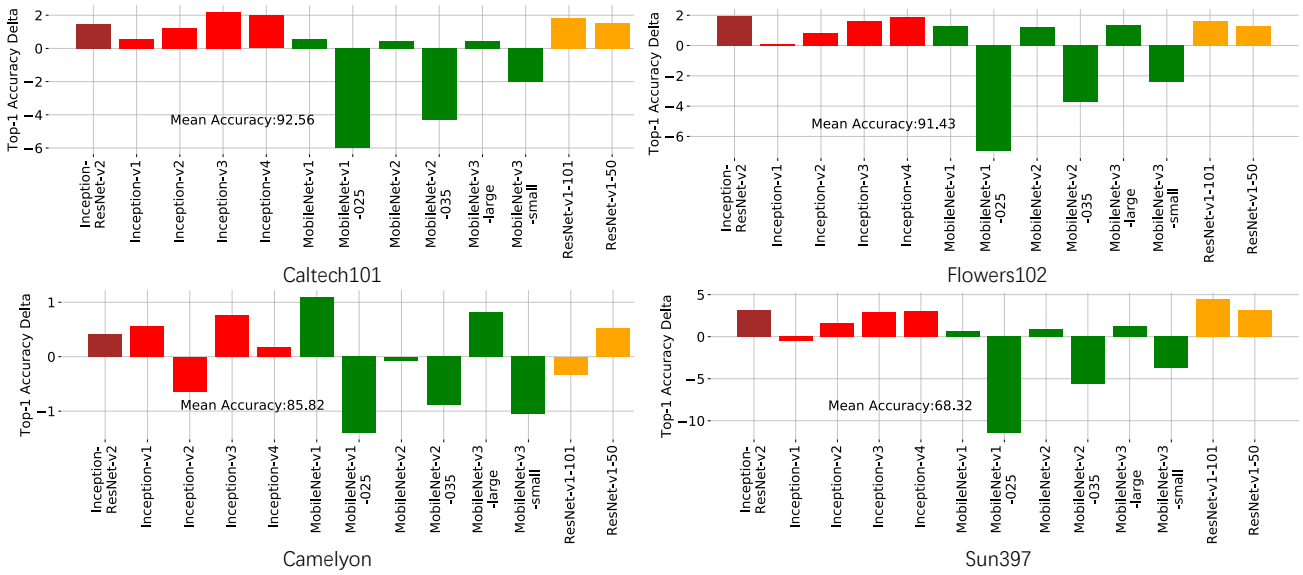

Figure 3. Difference between the fine-tuning accuracy of each checkpoint and the mean fine-tuning accuracy on Group III. The colors of bars represent the models trained with different architectures. Brown: Inception-ResNet-V2. Red: Inception family. Green: MobileNet family and their variants. Orange: ResNet-v1 family.

Dataset	From-Scratch	WAE-UKL	WAE-GAN	Cond-BigGAN	WAE-MMD	VAE	Uncond-BigGAN	Jigsaw	Rel.Pat.Loc	Exemplar	Rotation	Semi-Rotation-10%	Semi-Exemplar-10%	Sup-100%-Img	Sup-Exemplar-100%	Sup-100%-Inat	Sup-100%-Pla
Caltech101	51.44	41.99	42.37	73.88	51.83	54.91	73.15	78.85	79.09	80.02	87.91	92.73	92.77	94.42	94.5	87.00	87.27
Flowers102	49.28	17.46	17.96	69.67	26.05	26.98	61.72	77.69	78.08	78.87	83.29	91.46	91.28	93.05	93.91	94.47	82.79
Camelyon	81.59	80.55	79.71	80.73	80.73	80.87	82.14	85.43	86.58	85.27	85.81	85.09	85.83	85.37	84.98	83.33	84.77
Sun397	47.97	30.5	31.52	44.99	37.56	39.85	47.36	59.71	58.06	58.05	60.00	66.74	67.27	70.98	70.06	66.65	73.45

Table 4. Absolute fine-tuning accuracy on Group I.

Dataset	From-Scratch	Img-90k	Img-180k	Img-270k	Img-300k	Inat-90k	Inat-180k	Inat-270k	Inat-300k	Pla-60k	Pla-120k	Pla-180k	Pla-200k
Caltech101	51.44	92.08	93.55	94.18	94.73	85.69	86.54	87.66	87.43	84.91	87.11	88.01	87.48
Flowers102	49.28	88.8	91.96	91.36	93.6	92.71	94.32	94.78	94.6	79.15	82.47	84.36	82.88
Camelyon	81.59	85.73	85.97	86.3	85.43	85.48	86.12	84.37	83.75	85.67	85.13	85.16	85.4
Sun397	47.97	66.56	70.07	70.69	71.24	63.63	66.24	66.84	66.87	68.14	72.34	73.25	73.6

Table 5. Absolute fine-tuning accuracy on Group II.

Dataset	Inception-ResNet-v2	Inception-v1	Inception-v2	Inception-v3	Inception-v4	MobileNet-v1	MobileNet-v1-025	MobileNet-v2	MobileNet-v2-035	MobileNet-v3-large	MobileNet-v1-small	ResNet-v1-101	ResNet-v1-50
Caltech101	94.02	93.13	93.77	94.76	94.55	93.15	86.59	93.02	88.28	93.02	90.52	94.42	94.09
Flowers102	93.39	91.5	92.25	93.07	93.31	92.69	84.44	92.68	87.71	92.81	89.01	93.03	92.72
Camelyon	86.23	86.38	85.18	86.58	86.0	86.91	84.42	85.75	84.93	86.64	84.77	85.49	86.35
Sun397	71.52	67.82	69.95	71.23	71.41	69.03	56.88	69.22	62.7	69.61	64.63	72.74	71.44

Table 6. Absolute fine-tuning accuracy on Group III.

Dataset	From-Scratch	Img-90k	Img-180k	Img-270k	Img-300k	Inat-90k	Inat-180k	Inat-270k	Inat-300k	Pla-60k	Pla-120k	Pla-180k	Pla-200k
Caltech101	54.47	92.56	93.91	94.23	94.32	86.68	87.64	88.51	88.28	86.22	88.09	88.44	88.14
Flowers102	49.85	89.13	92.45	92.37	91.47	93.46	95.44	95.7	95.26	79.96	84.05	84.72	84.62
Camelyon	82.14	84.95	85.81	85.87	85.35	84.64	84.87	85.08	84.16	86.03	85.59	85.13	85.09
Sun397	46.87	67.36	70.83	71.41	71.44	64.6	66.97	67.44	67.42	68.78	73.21	74.22	74.24

Table 7. Absolute fine-tuning accuracy on Group IV.

Method	Recall@1	Pearson	Kendall
Linear (1 epoch)	✗	18.45	-33.33
Linear (5 epoch)	✗	40.77	0.00
Linear (converged)	✓	61.57	54.77
Fine-tune (1 epoch)	✗	20.55	0.00
Fine-tune (5 epoch)	✓	50.46	33.33
$\mathcal{N}$LEEP	✓	**66.59**	**66.67**

Method	Recall@1	Pearson	Kendall
Linear (1 epoch)	✗	23.55	-33.33
Linear (5 epoch)	✗	55.43	0.00
Linear (converged)	✓	68.32	33.33
Fine-tune (1 epoch)	✗	38.85	-33.33
Fine-tune (5 epoch)	✗	51.22	33.33
$\mathcal{N}$LEEP	✓	**82.66**	**66.67**

Table 8. Left: Checkpoint ranking results on the Pascal VOC Object Detection Benchmark (trained on VOC 2007 train+val + VOC 2012 train+val, tested on VOC 2007 using AP). Right: Checkpoint ranking for Cityscapes instance segmentation.