A Closer Look at Model Adaptation using Feature Distortion and Simplicity Bias

Anonymous Author(s)
Affiliation
Address
email

Abstract

In order to achieve strong in-distribution (ID) and out-of-distribution (OOD) generalization during transfer learning, it was recently argued that adaptation protocols should better leverage the expressivity of high-quality, pretrained models by controlling feature distortion (FD), i.e., the failure to update features orthogonal to the ID. However, in addition to OOD generalization, practical applications require that adapted models are also safe. To this end, we study the susceptibility of common adaptation protocols to simplicity bias (SB), i.e., the well-known propensity of neural networks to rely upon simple features, as this phenomenon has recently been shown to underlie several problems in safe generalization. Using a controllable, synthetic setting, we demonstrate that solely controlling FD is not sufficient to avoid SB, harming in safe generalization. Given the need to control both SB and FD for improved safety and ID/OOD generalization, we propose modifying a recently proposed protocol with goal of reducing SB. We verify the effectiveness of these modified protocols in decreasing SB on synthetic setting, and in jointly improving OOD generalization and safety on standard adaptation benchmarks.

1 Introduction

Due to rapid improvements in the representation quality of large-scale, pretrained self-supervised models (LSPM) [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13], there has been growing interest in developing transfer learning or adaptation protocols which are expressly designed to exploit these expressive features. However, standard protocols, e.g., fine-tuning (`FT`) all layers or only training a linear-probe (`LP`), do not effectively utilize this expressivity and achieve a sub-optimal in-distribution (ID) vs. out-of-distribution (OOD) generalization trade-off [14, 10, 15, 16]. Kumar et al. argue that *feature distortion* (FD), i.e., the phenomenon of exclusively updating only a subset of features aligned with the ID data, leads to decreased OOD generalization and propose a new family of `LP+FT` protocols to improve this trade-off. By performing `LP` prior to `FT`, `LP+FT` protocols make fewer updates to the LSPM during `FT`, and thus reduce FD without compromising on task performance.

However, in addition to strong ID/OOD generalization, practical deployment requires that models are also safe [17], e.g., well-calibrated, robust to anomalous/corrupted/adversarial examples and avoid shortcuts (see Figure 1). Yet, we find that even the well-studied `LP`, `FT` and `LP+FT` protocols achieve varying levels of success when considering both, these additional safety metrics, and datasets with different degrees of distribution shift from the original, pretraining dataset (ImageNet) (see vanilla protocols in Table. 1). This observation clearly suggests that a complimentary perspective to FD is needed to understand and improve the behavior of adaptation protocols with respect to safety metrics.

Submitted to the Workshop on Distribution Shifts, 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

Proposed Work. To this end, we study the susceptibility of protocols to *simplicity bias* (SB) [18, 19, 20, 21, 22], *i.e.*, the tendency of deep neural networks (DNNs) to prefer simple features over complex features and learn thin, non-robust decision boundaries [23]. Using a configurable, synthetic dataset, we find that FT is particularly prone to SB and that LP+FT does somewhat help mitigate both SB and distortion. However, in settings where both OOD generalization and avoiding SB are required, LP+FT can comprise upon performance by exclusively prioritizing FD mitigation. Using these insights, we aim to systematically mitigate both SB, which is known to influence the robustness of DNNs, *and* FD, which influences ID/OOD generalization during model adaptation. To this end, we propose modifications to LP+FT, where we leverage either optimized perturbations (virtual adversarial training, uncertainty-driven perturbations) or construct *soups* (*i.e.*, average multiple models) of probes to find LP initializations that reduce the SB for the subsequent FT step. On the synthetic dataset, we indeed find that modified protocols decrease SB and, on real datasets, these protocols improve both OOD generalization and safety metrics.

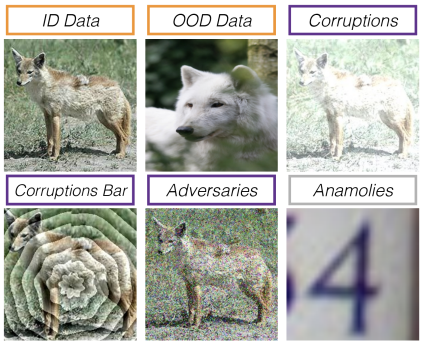

Figure 1: **Strong and Safe Adaptation.** Practical deployment in high risk applications requires that adapted models not only generalize well to in- and out-of distribution data of the downstream task, but also that adapted models are reliable. By considering these widely-adopted safety metrics, we are able to holistically evaluate adaptation protocols.

2 Simplicity Bias, Feature Distortion and Adaptation

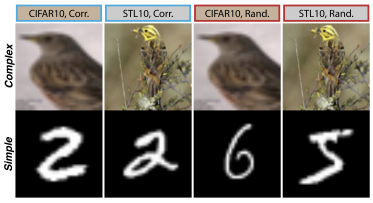

Figure 2: **Understanding Effect of Simplicity Bias.** We introduce a new dataset compromised of "dominoes" of simple (MNIST) and complex (CIFAR10) features to understand the effect of simplicity bias on generalization and safety.

We evaluate the susceptibility to simplicity bias of LP, FT, and LP + FT protocols on a custom variant of the dominoes dataset [23]. As shown in Fig. 2, we create "dominoes" of complex and simple features by pairing each class from CIFAR10 (complex) with the corresponding "digit" class in MNIST (simple), and consider 3 levels of correlation (95%, 99%, 100%) between the simple and complex features. To assess OOD generalization, we create a variant where complex features are sampled from STL10, instead of CIFAR10. (See 2, panels 1/2.)

We consider two metrics: *Randomized Accuracy*, e.g., the accuracy on the variant where samples contain random pairings between simple and complex features, and *Correlated Accuracy*, where pairings between simple and complex features remain correlated. Models that (are)/(are not) susceptible to SB will have high *Correlated Accuracy* and (low/high) *Randomized Accuracy*. We use a MoCo-V2 ResNet-50 [6] pretrained on ImageNet-1K as the base-feature extractor and results are averaged over 3 seeds. See supplementary for additional details.

Results. We make the following observations based on *only on the vanilla protocols, shown in black and gray* in Fig. 3. Here, the *Rand* OOD setting is equivalent to "safety evaluation" as models must avoid shortcuts to perform well under distribution shifts. Across all correlation strengths, FT has lower *Rand.* OOD Acc. and higher *Corr.* OOD Acc. than LP+FT. This clearly indicates that FT is highly susceptible to SB. In contrast, given that LP+FT has higher *Rand.* OOD Acc. and comparable *Corr.* ID Acc., LP+FT more effectively decreases SB in order to do well on the OOD dataset. In the appendix, we include additional results which demonstrate that under high correlation (0.99,1.0), LP is more effective at decreasing SB, as any additional distortion is harmful. However, in moderate correlation (0.95), additional distortion is in fact beneficial to LP+FT.

2.1 Improved Linear Probes for Mitigating Simplicity Bias

While our results on the dominoes dataset indicate that LP+FT and LP are effective protocols for reducing SB, we found that additional distortion can helpful in moderate correlation settings. This indicates that decreasing FD alone is unlikely to achieve to optimal safety performance. To that end, we propose new variants to the LP step of the LP+FT protocol which attempt to enable the subsequent FT step to distort features without compromising generalization or increasing SB. While its possible to modify the FT step as well, modifications to LP are inexpensive as the feature-encoder is not updated and, given that the fine-tuned solution remains in close vicinity of the initial LP initialization, strong starting solution is well-motivated. To this end, we introduce the following modifications.

- LP(VAT): Virtual Adversarial Training (VAT) [24] enforces local distribution smoothness by minimizing the KL-divergence between the predictions of perturbed pairs of examples. Since we are using expressive pretrained models, such perturbations maybe meaningful in the inverted latent space as well, and resulting classifiers become robust in some neighborhood around each latent-space input.
- LP(UDP): Instead of maximizing the loss when training with adversarial perturbations, uncertainty-driven perturbations (UDP) [25] maximize a model's estimated uncertainty and have been shown to be effective in decreasing SB and improving generalization in non-adaptation settings.
- LP(Soup): Inspired by [26], we train multiple, sparse, linear probes jointly and then take the average of their weights (aka soup) as the learned LP for subsequent FT. While soups of large models improve generalization by combining models from the same low-error basin, we consider sparse classifiers soups as an alternative strategy which seeks to average of diverse decision rules, to avoid relying upon a single set of simple features.

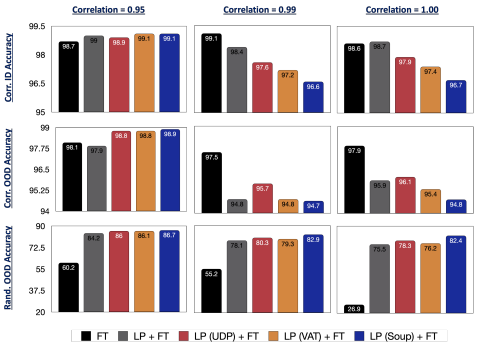

Figure 3: **Hardness Promoting Augmentations help Mitigate Simplicity Bias.** On the dominoes dataset, modified LP+FT protocols improve Rand. OOD Accuracy over vanilla protocols, indicating modified protocols rely less upon simple features.

Empirical Evaluation of Hardness Promoting Augmentations. We evaluate the effectiveness of the above LP variants, which we collectively refer to as "hardness promoting", in reducing SB of LP+FT and summarize the results in Fig. 3. We make the following observations.

Across all correlation strengths, we find that using the modified hardness promoting LPs during LP+FT (aka hp-LP+FT) improves the Rand. OOD Accuracy over vanilla LP+FT($\geq 2\%$) and FT($> 20\%$). This clearly indicates that hp-LP+FT is indeed effective in decreasing reliance on simple features, potentially also leading to improved safety. Furthermore, with the exception of LP(Soup)+FT, hp-LP+FT also performs better than vanilla LP+FT on Corr. ODD accuracy. Lastly, we observe that with respect to Corr. ID Accuracy that hp-LP+FT improves performance at low correlation strength, but slightly loses performance at higher correlation strengths. This is not entirely unexpected as FT's reliance upon simple will be useful in the correlated setting.

3 Evaluating Generalization and Safety of LP+FT Family

Given the effectiveness of incorporating hardness promoting (hp) augmentations with the family of LP+FT protocols (hp-LP+FT) in avoiding shortcuts/SB, we also evaluate on the modified protocols on the three real-world datasets (Living17, DomainNet, and CIFAR10) with respect to ID/OOD generalization and safety metrics. We summarize our results in Table 1 and our observations below. (See supplementary for additional results and details.)

These three datasets represent scenarios where different levels of distortion, measured using CKA scores [27, 28], are necessary when adapting the pretrained model. On Living17, a setting which requires minimal distortion during adaptation, we see that vanilla LP+FT is quite effective with respect to both generalization and safety metrics and is a difficult baseline to surpass. Indeed, while

hp-LP+FT variants do not lead to significant benefits, they generally perform comparably to vanilla LP+FT. On DomainNet, a setting where fairly low distortion is required for LP+FT but FT struggles to find a good solution, we see that hp-LP+FT induces some slight benefits with respect to ID/OOD generalization and robustness, though vanilla LP and hp-LP have better calibration performance. In contrast on CIFAR10, which requires more distortion to obtain an acceptable solution, we see that hp-LP+FT leads to improved generalization and a noticeable boost in corruption robustness. LP(VAT)+ FT is particularly effective in this regard. Lastly, across all datasets, we observe that hp-LP+FT protocols lead to similar distortion to vanilla LP+FT, which suggests that any additional benefits of hp-LP+FT should not be attributed to only reducing feature distortion.

In summary, we find that while vanilla LP+FT is already an effective protocol, especially in settings where low distortion is required, hp-LP+FT can provide some benefits and performs competitively. To this end, we recommend incorporating hardness promoting augmentations during LP as a potential safe-guard to simplicity bias.

Table 1: **Results**: In the low-distortion adaptation setting of Living-17, we see that vanilla LP+FT is an effective baseline and performs comparably to our LP+FT variants. With DomainNet, while relatively low distortion is induced by LP+FT, FT struggles to find a viable solution. Here, hardness-promoting LP+FT variants, particularly LP(VAT)+FT improves ID and OOD generalization as well as robustness. Finally, in CIFAR10, FT is more effective than LP+FT with respect to safety metrics and performs comparably on ID/OOD generalization.

	Generalization		Robustness			Calibration				Anomaly Det.	Rep. Similarity
Protocol	ID Acc.	OOD Acc.	C Acc.	$\overline{\text{C}}$ Acc.	Adv. Acc.	ID 1-RMS	C 1-RMS	$\overline{\text{C}}$ 1-RMS	OOD. 1-RMS	Out-of-Class AUROC	ID CKA
Dataset: *Living-17*											
LP	0.9521	0.8124	0.7010	0.7378	0.2350	0.9313	0.8693	0.8802	0.9117	0.9907	1.0000
FT	0.9518	0.7168	0.7011	0.7164	0.1563	0.8873	0.9019	0.8604	**0.9295**	0.9794	0.7847
LP+FT	**0.9643**	**0.8261**	0.7426	0.7671	0.2135	**0.9782**	0.9472	0.9451	0.8742	0.9924	0.9887
LP(UDP)+ FT	0.9637	**0.8265**	0.7448	**0.7681**	0.2157	0.9768	0.9464	0.9467	0.8757	0.9927	0.98927
LP(VAT)+ FT	**0.9647**	0.8247	0.7425	0.7650	0.2224	0.9727	**0.9521**	0.9463	0.8775	0.9925	0.9893
LP(Soup)+ FT	0.9608	0.8163	**0.7456**	**0.7684**	0.1855	0.9760	0.9498	**0.9492**	0.8678	**0.9936**	0.98540
Dataset: *DomainNet*											
LP	0.8913	0.8013	0.6019	0.6020	0.1768	0.9638	0.9264	**0.9045**	0.9014	0.8679	1.0000
FT	0.7613	0.4522	0.5186	0.2744	0.4164	0.8368	0.7234	0.7234	0.6379	0.8841	0.6092
LP+ FT	0.8985	0.7990	0.6343	0.5979	0.1927	0.9566	0.8445	0.8445	0.8899	0.9022	0.9222
LP(UDP)+ FT	0.9033	0.7965	0.6414	**0.6178**	0.1778	0.9436	0.8533	0.79415	0.752	0.8857	0.9662
LP(VAT)+ FT	**0.9048**	0.8009	**0.6466**	0.6131	0.1942	**0.9686**	0.8911	0.8428	0.7985	**0.9204**	0.9370
LP(Soup)+ FT	**0.9051**	0.8013	0.6393	0.6091	**0.1954**	0.9670	0.9042	0.8692	0.8246	0.9097	0.9281
Dataset: *CIFAR10*											
LP	0.9138	0.8190	0.6912	0.6553	0.0003	0.9595	0.8303	0.8142	0.8696	0.6206	1.0000
FT	0.9539	0.8754	0.7434	**0.7553**	**0.0231**	0.9668	0.8364	0.8453	0.9232	**1.0000**	0.6831
LP+FT	0.9442	0.8678	0.6921	0.6790	0.0018	0.9521	0.7849	0.7721	0.8864	0.6511	0.7853
LP(UDP)+ FT	0.944	0.8848	0.7028	0.6986	0.0004	0.9670	**0.8472**	**0.8476**	0.9237	0.9559	0.7764
LP(VAT)+ FT	**0.9611**	**0.8900**	**0.7442**	0.7321	0.0027	0.9294	0.8355	0.8281	0.9178	0.8276	0.7839
LP(Soup)+ FT	0.9466	0.8892	0.7031	0.6931	0.0001	**0.9678**	0.8390	0.8287	0.9216	0.9265	0.7806

4 Discussion

In this work, we considered factors important to the design of adaptation protocols which can induce both strong generalization and strong safety performance when performing transfer learning using high-quality pretrained models. We find that while the recently proposed LP + FT protocol does achieve impressive OOD accuracy by mitigating feature distortion, simple FT orLP can outperform it with respect safety objectives, such as robustness to corruptions, calibration error, and anomaly detection performance. We argue that feature distortion alone is not sufficient to understand this generalization vs. safety trade-off and that protocols should also consider susceptibility to simplicity bias. To this end, we propose using optimized perturbations (virtual adversarial training, uncertainty driven perturbations) or constructing soups to find better LP initialization, which will better enable subsequent FT to decrease SB and improve safety performance. We verify the benefits of modified protocols empirically on synthetic and real datasets, where, respectively, SB is decreased and safety performance improved.

References

[1] I. Zeki Yalniz, Hervé Jégou, Kan Chen, Manohar Paluri, and Dhruv Mahajan. Billion-scale semi-supervised learning for image classification. *CoRR*, abs/1905.00546, 2019.

[2] Xiaohua Zhai, Xiao Wang, Basil Mustafa, Andreas Steiner, Daniel Keysers, Alexander Kolesnikov, and Lucas Beyer. Lit: Zero-shot transfer with locked-image text tuning. In *Proc. Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[3] Xiangning Chen, Cho-Jui Hsieh, and Boqing Gong. When vision transformers outperform resnets without pretraining or strong data augmentations. In *Proc. Int. Conf. on Learning Representations (ICLR)*, 2022.

[4] Andreas Peter Steiner, Alexander Kolesnikov, Xiaohua Zhai, Ross Wightman, Jakob Uszkoreit, and Lucas Beyer. How to train your vit? data, augmentation, and regularization in vision transformers. *Transactions on Machine Learning Research*, 2022.

[5] Ilya Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, Mario Lucic, and Alexey Dosovitskiy. Mlp-mixer: An all-mlp architecture for vision. In *Proc. Adv. in Neural Information Processing Systems (NeurIPS)*, 2021.

[6] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. In *Proc. Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[7] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. In *Proc. Int. Conf. on Machine Learning (ICML)*, 2020.

[8] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Ávila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent - A new approach to self-supervised learning. In *Proc. Adv. in Neural Information Processing Systems (NeurIPS)*, 2020.

[9] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *Proc. Adv. in Neural Information Processing Systems (NeurIPS)*, 2020.

[10] Dan Hendrycks, Mantas Mazeika, Saurav Kadavath, and Dawn Song. Using self-supervised learning can improve model robustness and uncertainty. In *Proc. Adv. in Neural Information Processing Systems (NeurIPS)*, 2019.

[11] Hong Liu, Jeff Z. HaoChen, Adrien Gaidon, and Tengyu Ma. Self-supervised learning is more robust to dataset imbalance. *CoRR*, abs/2110.05025, 2021.

[12] Linus Ericsson, Henry Gouk, and Timothy M. Hospedales. How well do self-supervised models transfer? In *Proc. Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[13] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proc. Int. Conf. on Computer Vision (ICCV)*, 2021.

[14] Simon Kornblith, Jonathon Shlens, and Quoc V. Le. Do better imagenet models transfer better? In *Proc. Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[15] John Miller, Rohan Taori, Aditi Raghunathan, Shiori Sagawa, Pang Wei Koh, Vaishaal Shankar, Percy Liang, Yair Carmon, and Ludwig Schmidt. Accuracy on the line: on the strong correlation between out-of-distribution and in-distribution generalization. In *Proc. Int. Conf. on Machine Learning (ICML)*, 2021.

[16] Ananya Kumar, Aditi Raghunathan, Robbie Jones, Tengyu Ma, and Percy Liang. Fine-tuning can distort pretrained features and underperform out-of-distribution. In *Proc. Int. Conf. on Learning Representations (ICLR)*, 2022.

[17] Dan Hendrycks, Nicholas Carlini, John Schulman, and Jacob Steinhardt. Unsolved problems in ML safety. *CoRR*, abs/2109.13916, 2021.

[18] Alon Brutzkus, Amir Globerson, Eran Malach, and Shai Shalev-Shwartz. SGD learns over-parameterized networks that provably generalize on linearly separable data. In *Proc. Int. Conf. on Learning Representations (ICLR)*, 2017.

[19] Daniel Soudry, Elad Hoffer, Mor Shpigel Nacson, Suriya Gunasekar, and Nathan Srebro. The implicit bias of gradient descent on separable data. *J. Mach. Learn. Res.*, 19:70:1–70:57, 2018.

[20] Suriya Gunasekar, Jason D. Lee, Daniel Soudry, and Nati Srebro. Implicit bias of gradient descent on linear convolutional networks. In *Proc. Adv. in Neural Information Processing Systems (NeurIPS)*, 2018.

[21] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *Proc. Int. Conf. on Learning Representations (ICLR)*, 2019.

[22] Katherine L. Hermann, Ting Chen, and Simon Kornblith. The origins and prevalence of texture bias in convolutional neural networks. In *Proc. Adv. in Neural Information Processing Systems (NeurIPS)*, 2020.

[23] Harshay Shah, Kaustav Tamuly, Aditi Raghunathan, Prateek Jain, and Praneeth Netrapalli. The pitfalls of simplicity bias in neural networks. In *Proc. Adv. in Neural Information Processing Systems (NeurIPS)*, 2020.

[24] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[25] Matteo Pagliardini, Gilberto Manunza, Martin Jaggi, Michael I. Jordan, and Tatjana Chavdarova. Improving generalization via uncertainty driven perturbations, 2022.

[26] Mitchell Wortsman, Gabriel Ilharco, Samir Yitzhak Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S. Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and Ludwig Schmidt. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *Proc. Int. Conf. on Machine Learning (ICML)*, 2022.

[27] Thao Nguyen, Maithra Raghu, and Simon Kornblith. Do wide and deep networks learn the same things? uncovering how neural network representations vary with width and depth. In *Proc. Int. Conf. on Learning Representations (ICLR)*, 2021.

[28] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey E. Hinton. Similarity of neural network representations revisited. In *Proc. Int. Conf. on Machine Learning (ICML)*, 2019.

[29] Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. *Proc. IEEE*, 109(1):43–76, 2021.

[30] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.*, 22(10):1345–1359, 2010.

[31] Polina Kirichenko, Pavel Izmailov, and Andrew Gordon Wilson. Last layer re-training is sufficient for robustness to spurious correlations. *CoRR*, abs/2204.02937, 2022.

[32] Jeffrey O. Zhang, Alexander Sax, Amir Zamir, Leonidas J. Guibas, and Jitendra Malik. Side-tuning: Network adaptation via additive side networks. In *Proc. Euro. Conf. on Computer Vision (ECCV)*, 2019.

[33] Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Tuo Zhao. SMART: robust and efficient fine-tuning for pre-trained natural language models through principled regularized optimization. In *Proc. Assn. for Computational Linguistics, ACL*, 2020.

[34] Hossein Azizpour, Ali Sharif Razavian, Josephine Sullivan, Atsuto Maki, and Stefan Carlsson. Factors of transferability for a generic convnet representation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 38(9):1790–1802, 2016.

[35] Mi-Young Huh, Pulkit Agrawal, and Alexei A. Efros. What makes imagenet good for transfer learning? *CoRR*, abs/1608.08614, 2016.

[36] Hadi Salman, Andrew Ilyas, Logan Engstrom, Ashish Kapoor, and Aleksander Madry. Do adversarially robust imagenet models transfer better? In *Proc. Adv. in Neural Information Processing Systems (NeurIPS)*, 2020.

[37] Francisco Utrera, Evan Kravitz, N. Benjamin Erichson, Rajiv Khanna, and Michael W. Mahoney. Adversarially-trained deep nets transfer better: Illustration on image classification. In *Proc. Int. Conf. on Learning Representations (ICLR)*, 2021.

[38] Zeyuan Allen-Zhu and Yuanzhi Li. Feature purification: How adversarial training performs robust deep learning. In *Proc. Symposium on Foundations of Computer Science, FOCS*, 2021.

[39] Simran Kaur, Jeremy M. Cohen, and Zachary C. Lipton. Are perceptually-aligned gradients a general property of robust classifiers? *CoRR*, abs/1910.08640, 2019.

[40] Dan Hendrycks, Andy Zou, Mantas Mazeika, Leonard Tang, Bo Li, Dawn Song, and Jacob Steinhardt. Pixmix: Dreamlike pictures comprehensively improve safety measures. In *Proc. Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[41] Sangdoo Yun, Dongyoon Han, Sanghyuk Chun, Seong Joon Oh, Youngjoon Yoo, and Junsuk Choe. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proc. of Int. Conf. on Computer Vision, ICCV*, 2019.

[42] Terrance Devries and Graham W. Taylor. Improved regularization of convolutional neural networks with cutout. *CoRR*, abs/1708.04552, 2017.

[43] Dan Hendrycks, Norman Mu, Ekin Dogus Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty. In *Proc. Int. Conf. on Learning Representations (ICLR)*, 2020.

[44] Ekin Dogus Cubuk, Barret Zoph, Dandelion Mané, Vijay Vasudevan, and Quoc V. Le. Autoaugment: Learning augmentation policies from data. In *Proc. Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[45] Ekin Dogus Cubuk, Barret Zoph, Jon Shlens, and Quoc Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proc. Adv. in Neural Information Processing Systems (NeurIPS)*, 2020.

[46] Damien Teney, Ehsan Abbasnejad, Simon Lucey, and Anton van den Hengel. Evading the simplicity bias: Training a diverse set of models discovers solutions with superior OOD generalization. In *Proc. Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[47] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proc. Int. Conf. on Machine Learning (ICML)*, 2021.

[48] Ross Wightman. Pytorch image models. https://github.com/rwightman/pytorch-image-models, 2019.

[49] Shibani Santurkar, Dimitris Tsipras, and Aleksander Madry. Breeds: Benchmarks for subpopulation shift. In *ArXiv preprint arXiv:2008.04859*, 2020.

[50] Dan Hendrycks and Thomas G. Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *Proc. Int. Conf. on Learning Representations, (ICLR)*, 2019.

[51] Eric Mintun, Alexander Kirillov, and Saining Xie. On interaction between augmentations and corruptions in natural corruption robustness. In *Proc. Adv. in Neural Information Processing Systems (NeurIPS)*, 2021.

[52] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. In *Proc. Adv. in Neural Information Processing Systems (NeurIPS)*, 2019.

A Appendix

In this section, we provide a brief overview of directly relevant work. adaptation protocols used with pre-trained representations, widely adopted augmentation strategies and popular metrics used for evaluating the safety of deep neural networks.

For a comprehensive overview of transfer learning, please see the surveys of Zhuang et al. [29] and Pan et al. [30]. Here, we discuss a few directly works directly relevant to our own.

Adaptation Protocols. Given a pre-trained model, common practices for adapting it to a target task of interest are to either fine-tune (FT) all model parameters or train only a linear probe (LP) while freezing the network parameters. In general, large-scale, pre-trained models have sufficiently expressive features to perform well on both ID and OOD data. However, in practice, LP can result in lower classification accuracies due to the limited expressivity of the linear probe, while end-to-end FT can distort pretrained features toward ID data, thus harming OOD performance. Given these inherent limitations of both LPand FT, there is a need to design adaptation protocols that can lead to models with improved generalization performance and safety characteristics. In this spirit, Kumar et al. [16] recently proposed to perform LP prior to FT (abbrev. LP + FT) and demonstrated that this protocol improves OOD performance without comprising ID generalization of the target task by limiting feature distortion. Namely, that FT only modifies features in the ID representation subspace and not in other directions, which can lead higher OOD error as direction outside the ID subspace are necessary for OOD generalization. Concurrently, Kirichenko et al. [31] also found that retraining the last-layer of a model on a minority group or a simple data re-weighting can significantly improve robustness to spurious correlations and argue that classifiers poorly utilize the expressive features learnt by the model, instead relying upon spurious (instead of core) features. Notably, the model is able to learn *both* spurious and core features, so only simple re-weighting on minority data is needed, if we assume disentangled features. In this paper, we focus on the generic protocols such as LP,FT, and LP + FT, since they are effective, inexpensive and do not to perform any additional re-weighting data. Notably, we find that feature distortion can explain the ID vs. OOD generalization behavior of FTand LP, it cannot be straightforwardly extended to understand the trade-off between OOD and safety performance, necessitating a complementary perspective (simplicity bias).

Other modifications and heuristics have also been proposed to improve fine-tuning, including side-tuning [32], which tunes a small secondary network that is then combined with the original model, using larger/smaller learning rates for the classifier, as well as regularization-based methods [33]. We focus on the LP+FT protocol, as it is principled and achieves strong OOD performance.

Additionally, several works have studied properties of the model that influence the effectiveness of transfer learning [34, 35, 14], including the robustness of pretrained features [36, 37]. While the connection between adversarial training and improved feature representations [38, 39] has been studied, we use virtual adversarial training during LP to learn a better classifier that is less reliant upon simple features, and we do not use an adversarially trained feature extractor. Finally, while we consider a holistic evaluation of safety and generalization in the context of transfer learning with highly expressive pretrained models, Hendrycks et al [40] have considered the trade-offs induced by different data augmentation strategies [41, 42, 43, 44, 45] on safety metrics in supervised learning. We emphasize that while our evaluation is similar, that our work focuses on a different context and contains an additional layer of complexity as we consider the interaction between adaptation protocols, generalization behavior and safety performance.

Simplicity Bias. It is well-known that deep neural networks demonstrate a bias toward simple, potentially less expressive features [18, 19, 20, 21, 22], such as textures and backgrounds, and that this bias can lead to shortcuts that limit the generalization of DNNs. Indeed, recently Shah et al. [23] formalized this intuition by more precisely defining simplicity bias, based on the number of linear components to define a decision boundary, and showed that SB leads to non-robust decision boundaries that effects a model's sensitivity to distribution shifts and adversarial perturbations. In

brief, by learning simple features first, models become invariant to complex features, potentially leading to narrow decision boundaries which can fail to generalize under data shifts. While various methods have recently been proposed to mitigate simplicity bias when training from scratch or in the context of pretraining [46], we focus the susceptibility of existing adaptation protocols to simplicity bias as a tool for gaining insights into their safety behavior.

B Experimental Details

Please see the Anonymous Git repository for training details. In brief, we performed grid-search to find the best parameters, which are as follows. For CIFAR-10 and CIFAR-100, we train only the classifier for 200 epochs with LR=30 during `LP`. For `FT`, the entire model is trained for 20 epochs with LR=1e-5. For `LP+FT`, the model's classifier is initialized with the solution found by `LP`, and then it is fine-tuned for 20 epochs. A grid-search was conducted to determine the LR for `LP` and `FT`. For Domain-Net Experiments, we use 200 epochs with LR=30 during `LP`. For `FT`, the entire model is trained for 20 epochs with LR=3e-4. For `LP+FT`, the model's classifier is initialized with the solution found by `LP`, and then it is fine-tuned for 20 epochs, using LR=3e-7. Furthermore, following Kumar et al., we freeze the batchnorm layers during `LP+FT`. A CLIP [47] pretrained ResNet-50 is used for the DomainNet experiments, while a MoCoV2[6] is used for all CIFAR experiments. We use augmentation functions from timm[48] and compute CKA scores using the packaged provided by torch-cka. When using augmented protocols, the same LRs are used. Note, all results were obtained by averaging over 3 seeds. We consider model soups of sizes 5,10,20, tune ϵ in 0.005, 0.01, 0.02 and 0.1 for UDP, and α in 0.001, 0.01, 0.1 for VAT. For CIFAR-MNIST results, LP is done for 100 epochs, and FT is done for 20 epochs.

B.1 Safety Evaluation

`LP`, `FT`, and `LP + FT` protocols are evaluated for generalization and safety performance on three downstream adaptation tasks: CIFAR-10, Living17, and Domainnet-Sketch, where we report the following additional metrics for each dataset. We select these datasets as they correspond to two different types of distribution shifts (standard domain adaptation and subpopulation) and 3 levels of distortion (low, medium, high). Our safety evaluation protocol is similar to [40].

- *OOD Accuracy:* Models are expected to generalize well under the following distribution shifts: Living17(Source) $\rightarrow$ Living17(Target), CIFAR-10 $\rightarrow$ {STL10, CIFAR10.1} and Domainnet-Sketch $\rightarrow$ {Domainnet-ClipArt, Domainnet-Painting, Domainnet-Real}. CIFAR10 and Domainnet-Sketch shifts are popular domain-adaptation datasets, while Living17 is a recently proposed sub-population shift benchmark [49].
- *Mean corruption accuracy (mCA/m$\bar{C}$A):* We consider two sets of corruptions that a model should be robust to: the 15 naturalistic corruptions (C) proposed by [50], and the 10 perceptually dissimilar corruptions ($\bar{C}$) proposed by [51]. Corruptions are applied to each ID test dataset and the average accuracy over each set is reported.
- *Calibration Error (RMSE)*: It is important that models are well-calibrated so that practitioners may trust the provided predictions in high-risk applications. We measure the root mean square error of calibration as follows: $\sqrt{\mathbb{E}_C\left[(\mathbb{P}(Y = \hat{Y} \mid C = c) - c)^2\right]}$, where C indicates the confidence scores, while $\hat{Y}$ and Y denote the model's predictions and ground-truth labels respectively.
- *Anomaly Detection Performance (AUROC):* Recognizing when samples are anomalous allows models to abstain from making uninformed and inapplicable predictions. We consider samples from Blobs, Gaussian, LSUN, Places69, Rademacher, Textures, and SVHN datasets as anomalies and report the AUROC (area under the ROC curve) of the binary classification problem of detecting such samples as anomalies.
- *Adversarial Accuracy:* DNNs are well-known to be fooled by imperceptible distortions [52]. We use a 2/225, 10-step PGD attack to measure the robustness of models to such perturbations.
- *Representational Similarity (CKA)*: Kumar et al [16] claim that feature distortion harms OOD performance of adapted models by only updating representations in the span of the training data. We measure this distortion by computing the batched centered kernel alignment (CKA) score [27] with respect to the representations of the ID test dataset obtained from the pretrained and adapted models.

C Additional Results

Protocol	Generalization		Robustness			Calibration				Anomaly Detection	Rep. Similarity
	ID Acc.	OOD Acc.	C Acc.	$\overline{\text{C}}$ Acc.	Adv. Acc.	ID 1-RMS	C 1-RMS	$\overline{\text{C}}$ 1-RMS	OOD. 1-RMS	Out-of-Class AUROC	ID CKA
LP	0.9138	0.8190	0.6912	0.6553	0.0003	0.9595	0.8303	0.8142	0.8696	0.6206	1.0000
LP+ soup-5	0.9108	0.8348	0.7007	0.6678	0.0002	0.9748	0.8943	0.8835	0.9108	0.8463	1.0000
LP+ soup-10	0.9129	0.8359	0.6985	0.6652	0.0003	0.9669	0.9104	0.8956	0.9205	0.8713	1.0000
LP+ soup-20	0.9052	0.8353	0.6917	0.6588	0.0003	0.9605	0.9205	0.9037	0.9364	0.8859	1.0000
LP+ udp-0.005	0.9129	0.8332	0.7015	0.6702	0.0003	0.9729	0.8879	0.8817	0.9017	0.8708	1.0000
LP+ udp-0.01	0.9033	0.8356	0.6948	0.6643	0.0003	0.9689	0.9111	0.9023	0.9277	0.9033	1.0000
LP+ udp-0.02	0.8885	0.8281	0.6796	0.6492	0.0004	0.9655	0.9259	0.9142	0.9473	0.9217	1.0000
LP+ udp-0.1	0.8573	0.8005	0.6290	0.6064	0.0007	0.9245	0.9235	0.9143	0.9531	0.8570	1.0000
LP+ vat-0.001	0.9189	0.8276	0.6945	0.6606	0.0006	0.9714	0.8564	0.8442	0.8927	0.7159	1.0000
LP+ vat-0.01	0.8977	0.8251	0.6742	0.6483	0.0002	0.9265	0.9255	0.9139	0.9375	0.7200	1.0000
FT	0.9539	0.8754	0.7434	0.7553	0.0231	0.9668	0.8364	0.8453	0.9232	1.0000	0.6831
LP+FT	0.9442	0.8678	0.6921	0.6790	0.0018	0.9521	0.7849	0.7721	0.8864	0.6511	0.7853
(LP+soup-5) +FT	0.9466	0.8832	0.6997	0.6861	0.0001	0.9639	0.8197	0.8051	0.9155	0.9020	0.7603
(LP+soup-10) +FT	0.9467	0.8857	0.7022	0.6907	0.0001	0.9660	0.8307	0.8182	0.9184	0.9161	0.7671
(LP+soup-20) +FT	0.9466	0.8892	0.7031	0.6931	0.0001	0.9678	0.8390	0.8287	0.9216	0.9265	0.7806
(LP+udp-0.005) +FT	0.9458	0.8864	0.6962	0.6893	0.0005	0.9643	0.8127	0.8110	0.9119	0.9180	0.7742
(LP+udp-0.01) +FT	0.9450	0.8869	0.7048	0.6977	0.0004	0.9642	0.8335	0.8311	0.9209	0.9419	0.7746
(LP+udp-0.02) +FT	0.9440	0.8848	0.7028	0.6986	0.0004	0.9670	0.8472	0.8476	0.9237	0.9559	0.7764
(LP+udp-0.1) + FT	0.9435	0.8836	0.6959	0.6952	0.0000	0.9676	0.8449	0.8525	0.9355	0.9651	0.7382
(LP+vat)+FT	0.9611	0.8900	0.7442	0.7321	0.0027	0.9294	0.8355	0.8281	0.9178	0.8276	0.7839

Table 2: **CIFAR10, Hardness Promoting Augmentations**

Protocol	Generalization		Robustness			Calibration				Anomaly Detection	Rep. Similarity
	ID Acc.	OOD Acc.	C Acc.	$\overline{\text{C}}$ Acc.	Adv. Acc.	ID 1-RMS	C 1-RMS	$\overline{\text{C}}$ 1-RMS	OOD. 1-RMS	Out-of-Class AUROC	ID CKA
LP	0.9521	0.8124	0.7010	0.7378	0.2350	0.9313	0.8693	0.8802	0.9117	0.9907	1.0000
LP+ udp-0.005	0.9524	0.8114	0.7012	0.7379	0.2337	0.9304	0.8699	0.8806	0.9108	0.9907	1.000
LP+ udp-0.01	0.9524	0.8110	0.7017	0.7382	0.2353	0.9308	0.8691	0.8801	0.9118	0.9908	1.000
LP+ udp-0.02	0.9500	0.8126	0.7036	0.7387	0.2373	0.9343	0.8621	0.8763	0.9135	0.9913	1.000
LP+ udp-0.1	0.9459	0.8165	0.6840	0.7220	0.2339	0.9032	0.8243	0.8427	0.8990	0.9882	1.000
LP+ soup-5	0.9439	0.7996	0.6874	0.7290	0.2451	0.8806	0.7868	0.8094	0.9064	0.9897	1.0000
LP+ soup-10	0.9373	0.7904	0.6767	0.7220	0.2547	0.8496	0.7478	0.7709	0.8841	0.9887	1.0000
LP+ soup-20	0.9298	0.7841	0.6601	0.7082	0.2575	0.8056	0.7084	0.7305	0.8274	0.9867	1.0000
LP+ vat-0.001	0.9524	0.8122	0.7010	0.7379	0.2345	0.9299	0.8682	0.8791	0.9103	0.9907	1.0000
FT	0.9518	0.7168	0.7011	0.7164	0.1563	0.8873	0.9019	0.8604	0.9295	0.9794	0.7847
LP+FT	0.9643	0.8261	0.7426	0.7671	0.2135	0.9782	0.9472	0.9451	0.8742	0.9924	0.9887
(LP+udp-0.005) +FT	0.9627	0.8243	0.7434	0.7666	0.2153	0.9811	0.9456	0.9445	0.8736	0.9922	0.98950
(LP+udp-0.01) +FT	0.9627	0.8253	0.7436	0.7669	0.2133	0.9812	0.9454	0.9447	0.8737	0.9923	0.98957
(LP+udp-0.02) +FT	0.9637	0.8265	0.7448	0.7681	0.2157	0.9768	0.9464	0.9467	0.8757	0.9927	0.98927
(LP+udp-0.1) +FT	0.9614	0.8249	0.7499	0.7689	0.2165	0.9808	0.9441	0.9420	0.8711	0.9912	0.9861
(LP+soup-5) + FT	0.9608	0.8163	0.7456	0.7684	0.1855	0.9760	0.9498	0.9492	0.8678	0.9936	0.98540
(LP+soup-10) + FT	0.9580	0.8114	0.7445	0.7678	0.1753	0.9838	0.9503	0.9488	0.8748	0.9938	0.98360
(LP+soup-20) + FT	0.9594	0.8165	0.7450	0.7684	0.1782	0.9893	0.9503	0.9490	0.8609	0.9936	0.98190
(LP+vat-0.001) +FT	0.9647	0.8247	0.7425	0.7650	0.2224	0.9727	0.9521	0.9463	0.8775	0.9925	0.9370

Table 3: **Living17, Hardness Promoting Augmentations**

Table 4: **DomainNet: Hardness Promoting Augmentations and Adaptation.** While relatively low distortion is induced by `LP+FT` , `FT` struggles to find a viable solution. Here, hardness-promoting `LP+FT` variants, particularly `LP(VAT)+FT`do slightly improve ID and OOD generalization as well as robustness to corruptions.

	Generalization		Robustness			Calibration				Anomaly Det.	Rep. Similarity
Protocol	ID	OOD	C	$\overline{C}$	Adv.	ID	C	$\overline{C}$	OOD.	Out-of-Class	ID
	Acc.	Acc.	Acc.	Acc.	Acc.	1-RMS	1-RMS	1-RMS	1-RMS	AUROC	CKA
`LP`	0.8913	0.8013	0.6019	0.6020	0.1768	0.9638	0.9264	**0.9045**	0.9014	0.8679	1.0000
`FT`	0.7613	0.4522	0.5186	0.2744	0.4164	0.8368	0.7234	0.7234	0.6379	0.8841	0.6092
`LP+ FT`	0.8985	0.7990	0.6343	0.5979	0.1927	0.9566	0.8445	0.8445	0.8899	0.9022	0.9222
`LP(UDP)`	0.8919	**0.8021**	0.6022	0.6101	0.1345	0.9635	0.9250	**0.9047**	0.8619	0.8714	1.0000
`LP(VAT)`	0.8836	0.7914	0.5893	0.5963	0.1687	0.8897	**0.9552**	0.8905	**0.9178**	0.8735	1.0000
`LP(Soup)`	0.8787	0.7977	0.5951	0.6048	0.1731	0.8844	0.9479	0.8861	**0.9176**	0.8661	1.0000
`LP(UDP)+ FT`	0.9033	0.7965	0.6414	**0.6178**	0.1778	0.9436	0.8533	0.79415	0.752	0.8857	0.9662
`LP(VAT)+ FT`	**0.9048**	0.8009	**0.6466**	0.6131	0.1942	**0.9686**	0.8911	0.8428	0.7985	**0.9204**	0.9370
`LP(Soup)+ FT`	**0.9051**	0.8013	0.6393	0.6091	**0.1954**	0.9670	0.9042	0.8692	0.8246	0.9097	0.9281