Analyzing the Relations of Misclassified Inputs Between Models

Hadar Shavit
Independent Researcher
shavit.hadar@gmail.com

Abstract

A common belief in the machine learning community is that many of the misclassified images are "difficult" images (*e.g.*, the differentiation between classes is based on small details). We compare the misclassified images of various deep learning models and check which model misclassifies which image. We find that the misclassified images of each model are different. Moreover, despite having similar accuracy on ImageNet, one model can classify correctly more than 15% of the misclassified images of another model. This can encourage further research to use two or more architectures when performing a prediction, such as ensemble methods. The code for our analysis can be found at https://github.com/hadarshavit/analysing-relations.

1 Introduction

Today, in the field of computer vision, neural networks are the most used method for most pattern recognition tasks, such as image classification or semantic segmentation. The popular ImageNet benchmark is typically used to compare different models. In the last decade, the best accuracy on this benchmark increased by around 20%, as AlexNet [9] reached an accuracy of 62.5% while the recent ConvNeXt reached 20%. Many new architectures emerged with various key features, such as residual networks [5], patchify stem [17], and more. It is known that every architecture produces a different features map [4], even two models from the same family (e.g., transformers) can have a substantially different features map.

In recent years, a few papers have been published trying to understand the differences between classified and misclassified images of deep learning models. Specifically, Shankar et al. [12] showed that there are difficult images in the ImageNet benchmark that both human and AI models cannot classify correctly, for example, due to classes with similar attributes like different dog breeds. In addition, they showed that the computer models had more difficulties in classifying some objects correctly, while humans did not have this problem. Wen et. al [18] showed the inter and intra relations between images in a superclass of a few similar classes (such as types of dogs).

In addition, combinations between machine learning models have been studied for a long time in the form of ensemble methods [3, 2]. In this paper, we explore the relations between classified and misclassified images between models. We do it because the different features map of each model can mean that each model can learn different features, which can allow it to classify different images correctly. In addition, we do this to understand better the strengths and limitations of ensemble methods, as looking at the theoretical boundaries can give insights into what can work and what cannot. We find that one model can classify between 15% to 25% of the misclassified images of another model. In the next sections, we check how two (or more) models can benefit from combining them and check whether some models are stronger in certain classes.

I Can't Believe It's Not Better Workshop at NeurIPS 2022.

2 Relation Between Misclassified Images

First, we check how many images that one model classifies incorrectly, another model classifies correctly. Therefore, we define the *Potential Correction Rate* of model B over model A as the ratio of misclassified images of A that B classifies correctly:

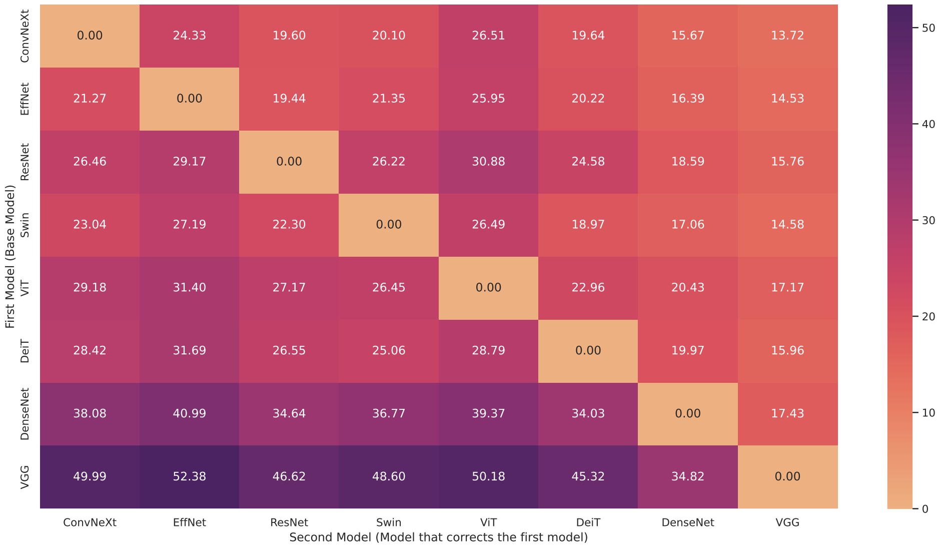
$$PotentialCorrectionRate = \frac{|F_A \cap T_B|}{|F_A|} \tag{1}$$

Where we denote F_A as the set of images that are classified correctly by the first model. T_B denotes the images that are classified correctly by the second model. In addition, we look into the maximum achievable accuracy of combining two models (e.g., the accuracy that can be achieved if for every image, we use the model that classifies it correctly, if such model exists). We define the *Virtual Accuracy* of models A and B as the percent of the validation set that either model A or B can classify correctly:

$$VirtualAccuracy = \frac{|T_A \cup T_B|}{|V|} \tag{2}$$

Where we denote T_A as the set of images that are classified correctly by the first model, T_B denotes the images that are classified correctly by the second model, and V denotes the full validation set. This definition is similar to the *virtual best solver* definition from the automated algorithm selection area [8].

We use the timm library [19] for trained weights of various models, as well as the inference script in order to check which images each model classifies correctly from the ImageNet validation dataset. In our experiment, we use the following models: ConvNeXt-T [11], EfficientNet-B4 [15], ResNet-50 [5, 20], Swin-T [10], Vit-S [1, 14], DeiT-S [16], DenseNet-121 [7], VGG11 [13]. We choose those models as they are from various families (transformer-based, CNNs) and years. In Figure 1 we show the potential correction rates of those models.



	ConvNeXt	EffNet	ResNet	Swin	ViT	DeiT	DenseNet	VGG
ConvNeXt	0.00	24.33	19.60	20.10	26.51	19.64	15.67	13.72
EffNet	21.27	0.00	19.44	21.35	25.95	20.22	16.39	14.53
ResNet	26.46	29.17	0.00	26.22	30.88	24.58	18.59	15.76
Swin	23.04	27.19	22.30	0.00	26.49	18.97	17.06	14.58
ViT	29.18	31.40	27.17	26.45	0.00	22.96	20.43	17.17
DeiT	28.42	31.69	26.55	25.06	28.79	0.00	19.97	15.96
DenseNet	38.08	40.99	34.64	36.77	39.37	34.03	0.00	17.43
VGG	49.99	52.38	46.62	48.60	50.18	45.32	34.82	0.00

Figure 1: Correction rate of various models. Every model can correct between 15%-25% of the misclassified images of other models.

We can see that for all model pairs, the second model can correctly classify a substantial amount of the images misclassified by the first model, even when the base accuracy of the second model is lower. For example, VGG can classify 15% of the misclassified images of ViT, even though VGG has lower accuracy by 10%. This is especially important as some ensemble methods, such as weighting use the accuracy of the model for the performing the ensemble [3]. Indeed, models with a higher base accuracy have a generally higher potential correction rate (e.g., they can classify more of the misclassified images of other models). However, this is not always the case. For example, ResNet has lower accuracy than Swin Transformer, but it has a higher correction rates than Swin for many models.

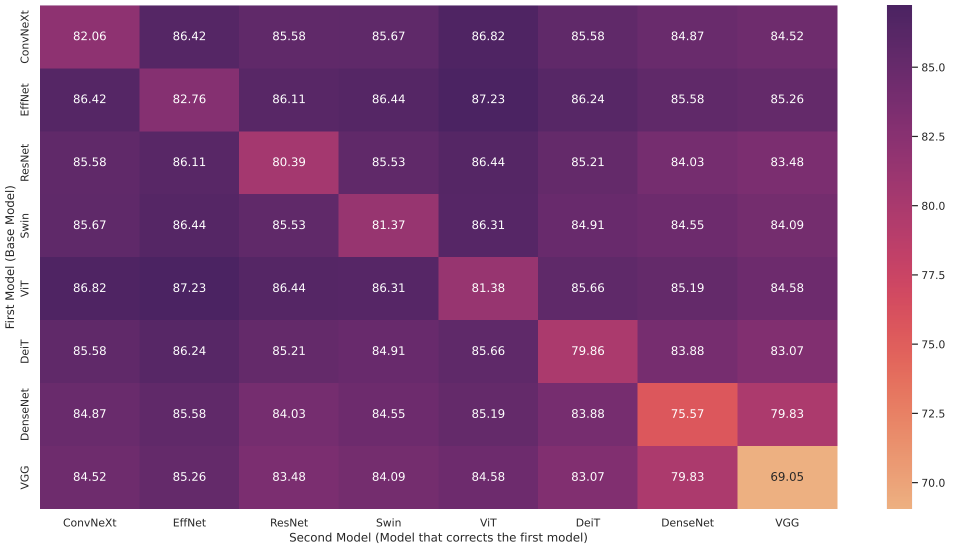


Figure 2: Virtual accuracy obtained by combining the classified images of the first and second models. The accuracies of single models are the diagonal of the matrix. All combinations reach higher virtual accuracy.

In addition, we show the virtual accuracy of model pairs. This number shows how many percents of the validation set one of the models can classify correctly. The values are presented in Figure 2. The diagonal of the matrix is the accuracy of every single model. We can see that those are the lowest values in the matrix. We can see that combining two models reaches substantially higher (virtual) accuracy by 3-5% of the better model.

Combining three models gives even better virtual accuracy. For example, by combining ViT, ConvNeXt, and EfficientNet, we get a virtual accuracy of 88.8%. For detailed accuracies, see appendix A.

3 Analysis of Images

In this section, we investigate for which classes each model is better than the other models. In Figure 3 we can see the total number of examples each model corrected per class for all models. Higher bars mean that the model is stronger in this class relative to other models, as it can classify correctly many instances that other models cannot classify correctly. In Figure 4, we can see the total number of examples each model got corrected by other models, per class. A higher bar in this figure means that the model is weaker in this class relative to other models. We can see that each model has different classes that it can correct, as the patterns in each graph are different from each other. For example, we can see in Figure 3 that ConvNeXt has a denser pattern than ViT for classes 0-500. This can show that each feature map is different, and some feature maps make predictions of specific classes easier than others.

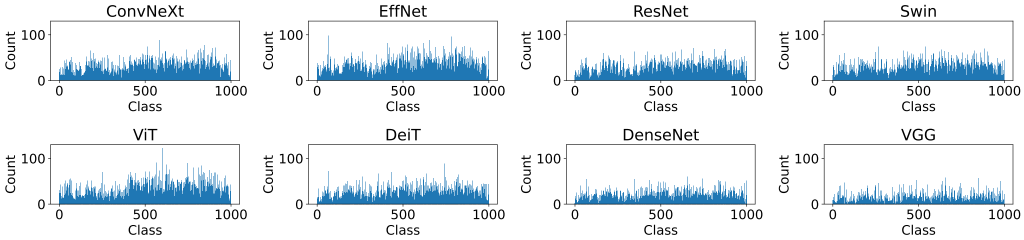


Figure 3: Sum of the corrected images per model per class across all models. Each model has a different pattern, showing that each model is stronger in different classes.

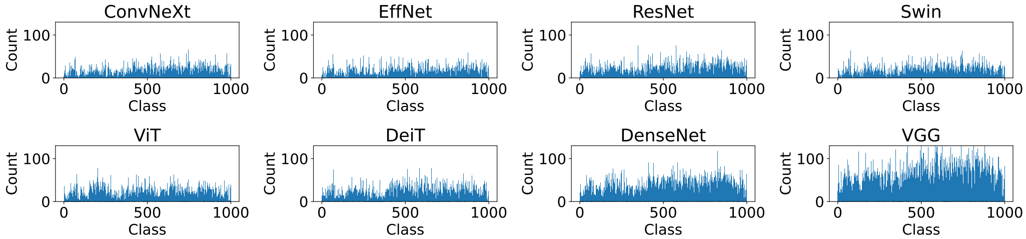


Figure 4: Sum of the images corrected by other models per model per class. Each model has a different pattern, showing that each model is weaker in different classes.

In Table 1 we can see the top 5 classes that each model corrected. Those are the classes that the model is stronger in classifying them related to others. Interestingly, we can see that some classes are repeated a few times, like ’missle’. We check the predictions for this class and find that all models classify correctly less than 50% of the images in this class.

Table 1: Top-5 classes that each model corrected.

Model	Classes
ConvNeXt	’hair spray’, ’tape player’, ’cornet, horn, trumpet, trump’, ’wok’, ’wallet, billfold, notecase, pocketbook’
EffNet	’sidewinder, horned rattlesnake, Crotalus cerastes’, ’screwdriver’, ’missile’, ’ladle’, ’ashcan, trash can, garbage can, ...’
ResNet	’overskirt’, ’spatula’, ’tub, vat’, ’ladle’, ’projectile, missile’
Swin	’toy poodle’, ’drum, membranophone, tympan’, ’velvet’, "loupe, jeweler’s loupe", ’ashcan, trash can, garbage can, ...’
ViT	’hook, claw’, ’frying pan, frypan, skillet’, ’purse’, ’letter opener, paper knife, paperknife’, ’stove’
DeiT	’projectile, missile’, ’horned viper, cerastes, sand viper, horned asp, Cerastes cornutus’, ’bathtub, bathing tub, bath, tub’, ’maillot’, ’polecat, fitch, foulmart, foumart, Mustela putorius’
DenseNet	’missile’, ’projectile, missile’, ’horned viper, cerastes, sand viper, horned asp, Cerastes cornutus’, ’bighorn, bighorn sheep, cimarron, ...’, ’sunglasses, dark glasses, shades’
VGG	’missile’, ’tape player’, ’CD player’, ’maillot’, ’promontory, headland, head, foreland’

4 Conclusions and Future Work

Although the different deep learning models for image classification have similar accuracy on the ImageNet benchmark, each model has its own distinctive set of correctly labeled images. While some misclassified images are mutual to many models, there is a substantial amount of images that one model can classify correctly while other model(s) cannot. This can show that there is a group of images that are the "most difficult" to classify, as two or more models fail to classify them correctly. However, a considerable amount of the misclassified images of a model can be classified correctly by other models. Especially, we can see that by using two models the virtual accuracy increases by a few percent, and by using three models, the accuracy further increases. This shows us that the belief that all the misclassified images are difficult to classify is not entirely correct.

We can also see that combining models can result in higher virtual accuracy. Using per-instance algorithm selection [8] techniques can be a further research direction. It is possible to use this existing work from machine learning with reject option literature [6], as they try to predict when a model cannot classify an input correctly. Another possible research is to use two or more feature maps to perform predictions, as used in ensemble learning [3].

References

[1] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *CoRR*, abs/2010.11929, 2020.

[2] Nick Erickson, Jonas Mueller, Alexander Shirkov, Hang Zhang, Pedro Larroy, Mu Li, and Alexander Smola. Autogluon-tabular: Robust and accurate automl for structured data. *arXiv preprint arXiv:2003.06505*, 2020.

[3] Mudasir A. Ganaie, Minghui Hu, Mohammad Tanveer, and Ponnuthurai N. Suganthan. Ensemble deep learning: A review. *CoRR*, abs/2104.02395, 2021.

[4] Ali Hassani, Steven Walton, Jiachen Li, Shen Li, and Humphrey Shi. Neighborhood attention transformer, 2022.

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[6] Kilian Hendrickx, Lorenzo Perini, Dries Van der Plas, Wannes Meert, and Jesse Davis. Machine learning with a reject option: A survey. *CoRR*, abs/2107.11277, 2021.

[7] Gao Huang, Zhuang Liu, and Kilian Q. Weinberger. Densely connected convolutional networks. *CoRR*, abs/1608.06993, 2016.

[8] Pascal Kerschke, Holger H. Hoos, Frank Neumann, and Heike Trautmann. Automated algorithm selection: Survey and perspectives. *CoRR*, abs/1811.11597, 2018.

[9] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

[10] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *CoRR*, abs/2103.14030, 2021.

[11] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[12] Vaishaal Shankar, Rebecca Roelofs, Horia Mania, Alex Fang, Benjamin Recht, and Ludwig Schmidt. Evaluating machine accuracy on ImageNet. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 8634–8644. PMLR, 13–18 Jul 2020.

[13] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2014.

[14] Andreas Steiner, Alexander Kolesnikov, Xiaohua Zhai, Ross Wightman, Jakob Uszkoreit, and Lucas Beyer. How to train your vit? data, augmentation, and regularization in vision transformers. *CoRR*, abs/2106.10270, 2021.

[15] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. *CoRR*, abs/1905.11946, 2019.

[16] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. *CoRR*, abs/2012.12877, 2020.

[17] Asher Trockman and J. Zico Kolter. Patches are all you need? *CoRR*, abs/2201.09792, 2022.

[18] Shixian Wen, Amanda Sofie Rios, Kiran Lekkala, and Laurent Itti. What can we learn from misclassified imagenet images? *CoRR*, abs/2201.08098, 2022.

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

[20] Ross Wightman, Hugo Touvron, and Herve Jegou. Resnet strikes back: An improved training procedure in timm. In *NeurIPS 2021 Workshop on ImageNet: Past, Present, and Future*, 2021.

A Virtual Accuracies of 3 models combinations

In table 2, we present the virtual accuracies of combinations of 3 models.

Table 2: Virtual Accuracies of 3 models combinations

Model 1	Model 2	Model 3	Virt. Acc.
convnext	effnet	resnet	87.87
convnext	effnet	swin	87.93
convnext	effnet	vit	**88.80**
convnext	effnet	deit	88.02
convnext	effnet	densenet	87.70
convnext	effnet	vgg	87.69
convnext	resnet	swin	87.42
convnext	resnet	vit	88.04
convnext	resnet	deit	87.43
convnext	resnet	densenet	86.99
convnext	resnet	vgg	86.94
convnext	swin	vit	88.24
convnext	swin	deit	87.18
convnext	swin	densenet	87.11
convnext	swin	vgg	87.02
convnext	vit	deit	88.09
convnext	vit	densenet	87.98
convnext	vit	vgg	87.97
convnext	deit	densenet	87.01
convnext	deit	vgg	86.89
convnext	densenet	vgg	86.14
effnet	resnet	swin	87.96
effnet	resnet	vit	88.68
effnet	resnet	deit	87.86
effnet	resnet	densenet	87.44
effnet	resnet	vgg	87.43
effnet	swin	vit	88.67
effnet	swin	deit	87.81
effnet	swin	densenet	87.72
effnet	swin	vgg	87.70
effnet	vit	deit	88.50
effnet	vit	densenet	88.42
effnet	vit	vgg	88.40
effnet	deit	densenet	87.61
effnet	deit	vgg	87.52
effnet	densenet	vgg	86.81
resnet	swin	vit	88.21
resnet	swin	deit	87.18
resnet	swin	densenet	86.97
resnet	swin	vgg	86.91
resnet	vit	deit	87.99
resnet	vit	densenet	87.77
resnet	vit	vgg	87.71
resnet	deit	densenet	86.73
resnet	deit	vgg	86.62
resnet	densenet	vgg	85.46
swin	vit	deit	87.63
swin	vit	densenet	87.75
swin	vit	vgg	87.63
swin	deit	densenet	86.61
swin	deit	vgg	86.45
swin	densenet	vgg	85.91
vit	deit	densenet	87.35
vit	deit	vgg	87.11
vit	densenet	vgg	86.57
deit	densenet	vgg	85.36