Analysing the Relations of Misclassified Inputs Between Models

Anonymous Author(s)
Affiliation
Address
email

Abstract

A common thought in the machine learning community is that many of the misclassified images are "difficult" images (for example images where the details are too small to differentiate between two classes). We evaluate those misclassified images of various deep learning models and check if other models can correctly classify those images. We find that the misclassified images of each model are different. Moreover, despite that models have similar accuracy on ImageNet, one model can classify correctly more than 15% of the misclassified images of another model. This means can encourage further research to use two or more architectures when performing a prediction. The code for our analysis can be found in https://github.com/USER/REPO

1 Introduction

Today, in the field of computer vision, neural networks are the most used method for most tasks. The popular ImageNet benchmark is used to compare different models. In the last decade, the top-1 accuracy on this benchmark increased by more than 10%. Many new architectures emerged, with many key features. It is known that every architecture produces a different features map [3], 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. [10] 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 kinds of dogs. 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 [15] showed the inter and intra relations between images in a superclass of a few similar classes (such as dogs).

In this paper, we explore the relation 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.

2 Relation Between Misclassified Images

First, we define the *Correcttion Rate* of model B over model A as the ratio of misclassified images of A that B classifies correctly:

$$\frac{|\overline{A} \cap B|}{|\overline{A}|} \tag{1}$$

Where we denote A as the set of images that are classified correctly by the first model, B denotes the images that are classified correctly by the second model. In addition, we define the *Virtual Accuracy*

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

of models A and B as the percent of the validation set that either model A or B can classify correctly:

$$\frac{|A \cup B|}{|V|} \tag{2}$$

Where we denote A as the set of images that are classified correctly by the first model, B denotes the images that are classified correctly by the second model, and V denotes the full validation set.

We use timm library [16] 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 [9], EfficientNet-B4 [13], ResNet-50 [4], Swin-T [8], Vit-S [1, 12], DeiT-S [14], DenseNet-121 [6], VGG11 [11]. In Figure 1 we show the correction rates of those models.

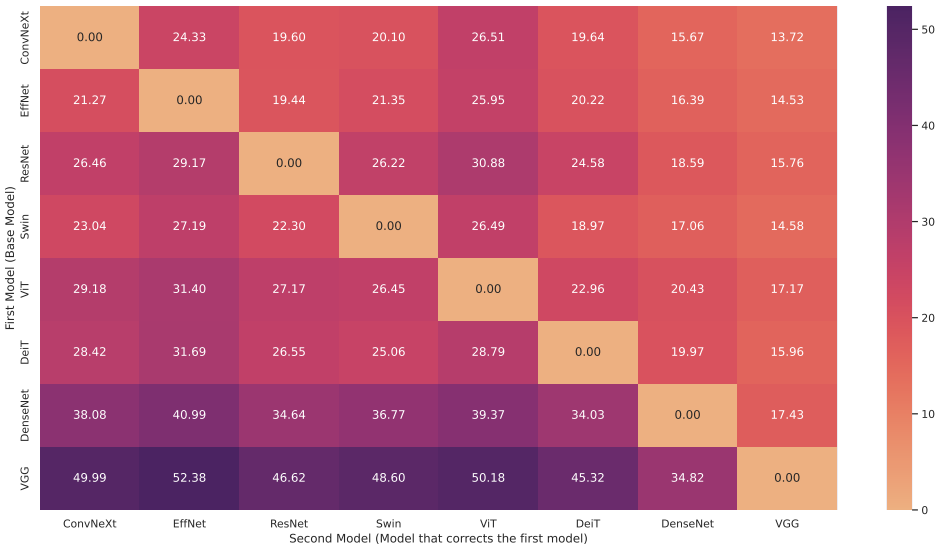

Figure 1: Correction rate of various models.

We can see that for all model pairs, the second model can classify a substantial amount of the misclassified images correctly by the first model, even if the second model has a lower base accuracy. Indeed, models with a base higher accuracy have a generally higher rate. However, this is not consistent as for example, ResNet has lower accuracy than Swin Transformer, but it has higher correction rates than Swin for many models. In addition, even models with relatively low accuracy can correctly classify some of the misclassified images, despite that if the first model has significantly higher accuracy. For example, VGG can classify 15% of the misclassified images of ViT, even though VGG has lower accuracy by 10%. We can also observe that models with higher accuracy have a higher correction rate than the percent of the base models' accuracy improvement. For example, if we take VGG and ConvNeXt, their base accuracies are 69% and 82%. Therefore, ConvNeXt has an improvement of $\frac{82-69}{100-69} = \frac{13}{31} = 41\%$, however the correction rate of ConvNeXt on VGG is 49.99%. This indicates that the set of correctly classified images of VGG is not a subset of ConvNeXt's correctly classified images, but those are rather different groups. Indeed there is a similarity between the groups (for example, images that are relatively easy to classify), but the images that each model classifies correctly differ.

Figure 2: Virtual accuracy obtained by combining the classified images of the first and second models.

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, 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 what 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 3, 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, as it is We can see that each model has different classes that it can correct. This can show that each feature map is different, and some feature maps make predictions of specific classes easier than others.

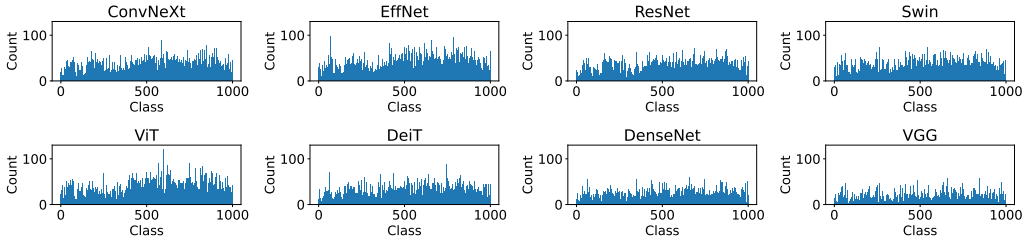

Figure 3: Sum of the corrected images per model per class

In Table 1 we can see the top 5 classes each 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 that repeated a few times, like 'missle'. We check the predictions for this class and found that all models classify correctly less than 50% of the images in this class correctly.

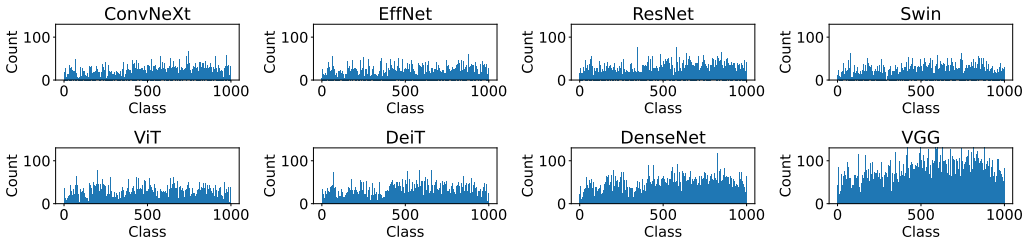

Figure 4: Sum of the images corrected by other models per model per class.

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'

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

4 Conclusions and Future Work

Although the accuracy of different deep learning models for image classification has 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 substantial 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 higher virtual accuracy. Using per-instance algorithm selection [7] techniques can be a further research direction. It is possible to use this existing work from machine learning with reject option literature [5], 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 [2].

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

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

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

[5] 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.

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

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

[8] 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.

[9] 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.

[10] 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.

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

[12] 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.

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

[14] 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.

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

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

A Virtual Accuracies of 3 models combinations

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

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

Table 2: Virtual Accuracies of 3 models combinations