Viewmaker Networks: Learning Views for Supervised Representational Learning

Truc Cody Ho
codyho@stanford.edu

ABSTRACT

Data augmentation, or the modification of an image before it is used as training data, is a fundamental requirement for all modern image classifiers. However, the majority of research into image classification has focused on how the improve the structure of the network directly, with comparatively little work going into the methods for data augmentation. This project focuses specifically on supervised image classifiers and evaluates a novel method of data augmentation centering around **Viewmaker Networks**, a lightweight generative model that produces useful perturbations on a given input. The Viewmaker can be trained adversarially to the primary image classifier or to maximize the distance between learned views. Remarkably, when combined with a subset of the default CIFAR-10 augmentations, the learned views achieve greater accuracy than the default augmentations alone. These results indicate that Viewmaker networks may allow more general representational learning algorithms, requiring less domain specific knowledge and working towards the goal of general neural networks being able to be trained on arbitrary data.

INTRODUCTION

Data augmentation has long been understood as necessary for image classifiers in order to improve performance and, more importantly, prevent overfitting and allow the model to generalize[9]. Likewise, it can increase the amount of training data by applying a series of transformations (e. g. Gaussian blur, cropping, shearing, color distortions, etc) to the training dataset, which allows the network to learn more from each image than it would otherwise. Traditionally, data augmentation has depended on domain specific knowledge by human experts. The need for humans to be a part of the learning process is a major limitation, despite the refinement in the process over the years. It is still slow, unable to effectively generalize to different datasets, and potentially introduces human error into the learning process. In this work, I present a general method for learning data augmentations using a generative model that can be trained adversarially to an encoder network or to produce diverse views (or views that are extremely different from each other). Rather than generating an entirely new image, the Viewmaker instead generates a pertubation which can then be applied to the image, reducing the overhead required during training and better allowing the network to generalize to arbitrary data.

RELATED WORK

Viewmaker Networks

First introduced by Alex Tamkin et al., Viewmaker networks have been shown to increase performance on unsupervised models with relatively low overhead [12]. Viewmakers improve on previous approaches, which required a pre-trained model and were only able to execute relatively simple transformations [11] by removing the need for a pretrained network and instead can be tuned for the specific encoder used. Aside from image classification, Viewmakers have demonstrated strong performance on speech recordings and wearable sensor data[12]. However, this Viewmaker was only trained adversarially to the main encoder and only in the context of self-supervised learning.

Data Augmentation

Considering the limitations of current data augmentation techniques, it is unsurprising that an emerging area of research is how this process can be automated. Perhaps most notable has been AutoAugment, a model created by Cubuk et al. which used a recurrent neural network to select the optional transformations and the order of said transformations from a provided vector of transformations[2]. Using this method, AutoAugment demonstrated that the performance of image classifiers can be greatly improved over the previous hand crafted augmentations, and further work has continues to refine the method to achieve faster training and improved accuracy[3][13][6][7]. The approach taken in this paper differs from that of AutoAugment as Viewmakers are generative models and thus do not depend upon a vector of provided transformations (which require human understanding to create).

Adversarial Learning

One observation repeated in all current literature is that the majority of image classifiers respond very poorly to adversarial examples, or examples deliberately designed to confuse the network and play on its weaknesses [14] [5]. Various approaches are being researched in order to find the best solution to this problem, with the most notable likely being the Generative Adversarial Network (GAN) first designed in 2014 and focusing on unsupervised neural nets [4], which has played a major role in machine learning research beyond the field of image classifiers. Viewmakers are heavily inspired by GANs,

attempting to achieve the same goals in a way that can generalize to arbitrary data and without the overhead of having to train a network to produce an entire image. Specifically, Viewmakers are quite small (our implementation contains 1.8 million parameters compared to 9 million for the image classifier), and the perturbations produced should be applicable to any image classifier and dataset. At the same time, Viewmakers should still achieve the improved worst-case performance of GANs as well as the improved accuracy.

METHODS

Dataset

CIFAR-10 was chosen because it is an extremely common dataset in the field of image classification and because Viewmakers have been shown to be effective on this dataset in the context of self supervised learning[12][8]. No other preprocessing will be applied to this dataset as the purpose of this project is to automate the process of data augmentation, and thus any manual data augmentation would contrary to its goal. The CIFAR-10 dataset is composed of 60000 32x32 images of ten different objects, and will be partitioned into a training set of 57000 images (95% of the dataset), with the remainder being divided in half into validation and test datasets.

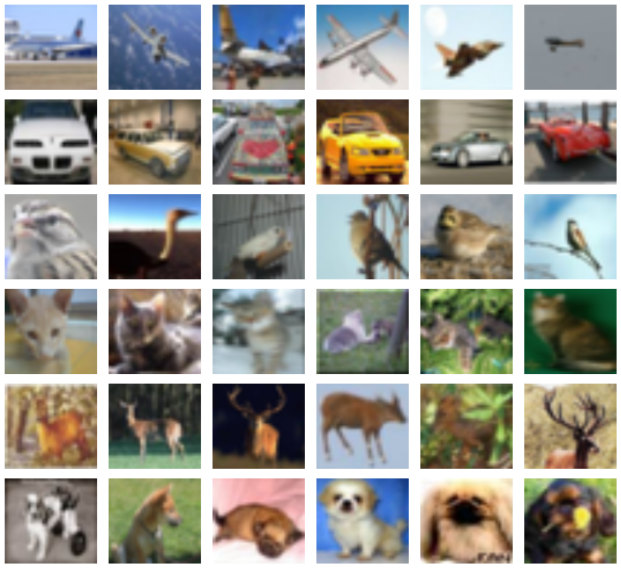

Figure 1: Sample CIFAR-10 images

Network Structure

The Viewmaker itself is implemented as a deep convolutional neural network and is based on one of the example PyTorch generative models. The most significant aspect of the network is the residual layers in the middle, which is where entropy is added and where the actual perturbation is generated. This perturbation is then applied to the image (a unique perturbation is generated for each image in the training dataset and multiple distinct perturbations can be generated per image) before it is passed through the encoder. Important to note is that the Viewmaker is designed to be lightweight compared to any modern image classifier. This implementation only has 1.8 million trainable parameters, compared to 11.2 million for the ResNet-18 model, which is already a very small neural network. As a result, generating views from the Viewmaker adds relatively little overhead when training the encoder.

The image classifier is a supervised Resnet-18 model tuned specifically for the CIFAR-10 dataset (the standard Resnet-18 model does not always perform well on CIFAR as the initial convolutional steps expect larger images and thus can compress the small 32x32 CIFAR images to a point where nothing of value can be learned). This model has already been shown to be an effective, if not particularly noteworthy, image classifier. The choice of classifier is, ultimately, not important, as a pretrained Viewmaker network should function on any image classifier, with a minimal amount of adjusting required. This model was chosen simply because its implementation and performance is already well understood.

Training

There are two ways the Viewmaker can be trained: adversarially and diversity promoting.

Adversarial

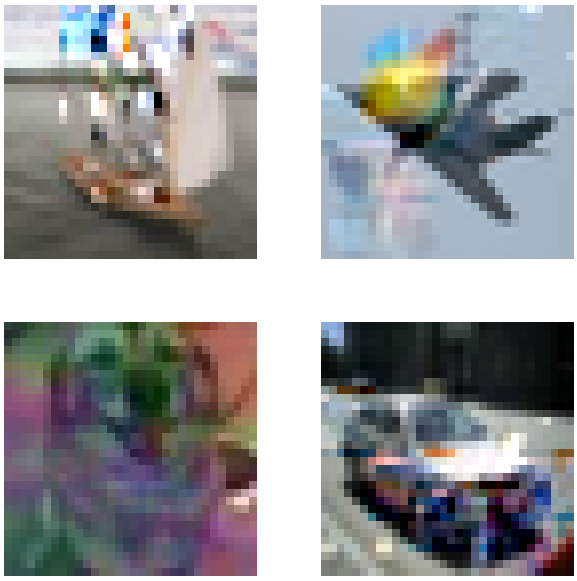

Figure 2: Sample views generated by the Adversarial Viewmaker In the adversarial case, the Viewmaker aims to generate transformations that maximize the loss function of the image classifier. This means that the loss function of the Viewmaker is the same as the loss function of the image classifier. The loss function used in this project is the cross entropy loss function (the pytorch torch.nn.CrossEntropyLoss() is used to calculate the loss of the network) which is essentially the logsumexp() of the guesses of the neural network compared to the labels. This is a standard cost function that is very commonly used in the field of image classification. The Viewmaker is trained by negating the value of the loss function multiplying this value by a constant (to try to increase training speed) then performing optimization (using the Adam optimizer) as usual. The alternative approach to adversarially learning would be to compute the gradients on

the unmodified cost function value, however that would likely be more computationally inefficient and harder to implement while yielding few, if any benefits.

This method of training is primarily intended to remain robust against common corruptions, similar to a GAN. Visual inspection of the generated perturbations reveals that the overwhelming majority consist of some sort of random distortion around the most identifying part of an image (ie, if the image was of a dog its face would be obscured by randomness), which is to be expected given the adversarial training.

Algorithm 1: Training Algorithm for ResNet-18 model with viewmaker

```
def training_step(batch, batch_idx,
                  optimizer_idx):
    x, y = batch
    x = viewmaker(x)
    x = normalize(x)
    preds = forward(x)
    loss = cross_entropy(preds, y)
    optim = optimizers()[optimizer_idx]

    if optimizer_idx % 2 == 1:
        loss = -loss

    return loss
```

Algorithm 1: Psuedocode for the training step of the viewmaker when trained adversarially to the classifier. Note how x is passed through the Viewmaker as the very first step (even before normalization) and how the loss is inverted every other step. In the diverse setting, a different loss function is used that maximizes the difference between generated views.

Diversity

In the diverse case, the Viewmaker aims to generate transformations that are as different as possible from both the original image and other generated transformations. In this case, a novel loss function originally designed to detect anomalies in images is used called NeuTralAD [10]. Similar to the adversarial case, the Viewmaker attempts to maximize the "anomalies" detected by the loss function, however, the diverse case does not interact at all with the encoder loss. The motivation of behind this method of training is to artificially increase the amount of training data and to cover the maximum number of cases possible. Visual inspection of the generated perturbations finds little to no pattern among them except that the area of the image affected tends to differ between images and even within the same image(if multiple perturbations are generated for the same image), which is again to be expected as the Viewmaker is trained to produce as much diversity in generated views as possible (and the Viewmaker samples from a noise distribution for every image, providing a unique source of entropy for every generated transformation).

Figure 3: Sample views generated by the Diverse Viewmaker

Distortion Budget

One crucial aspect of training is the tuning of the most important hyperparameter, the distortion budget. The distortion budget exists to prevent the Viewmaker from simply generating a transformation that makes learning impossible (such as just making the entire image black). Currently this value is hard coded to 5% of the image in the adversarial case and 30% in the diverse case (the diverse case is much higher because there is much less of a danger of the Viewmaker creating impossible augmentations as it is not adversarial). These values were found by trial and error. A better approach would likely be to set some baseline value (ie, 5%) then increment this value by a small amount every time the loss reaches a certain threshold. This should eventually converge to the maximum distortion that still allows the classifier to reach a high degree of accuracy.

RESULTS

Three Viewmaker models were trained (using an NVIDIA RTX2070 on driver version 465.27, which took around three hours to train the adversarial model and six to train the diverse model), one adversarial model, one diverse model, and one diverse model that incorporated a subset (two out of five, cropping and random horizontal flip) of a set of CIFAR-10 default augmentations. Important to note is that the default CIFAR-10 augmentations have been tuned for over a decade and the performance of a ResNet-18 model on the default augmentations can be considered a best-case scenario [1]. As controls, two more ResNet-18 models were trained, one with the default augmentations and one with only the subset mentioned earlier (only cropping and flipping). The results are summarized in the table below:

Remarkably, the diverse Viewmaker, when combined with a subset of the default transformations, was able to surpass the default transformations which have been tuned for over a

Model	Validation Step		Test Step
	Accuracy	Loss	Accuracy
Adversarial Viewmaker	0.81	1.258	0.82
Diverse Viewmaker	0.84	.896	0.85
Diverse Viewmaker+ Subset	0.89	.361	0.90
Default Augmentations	0.89	.359	0.89
Subset	0.85	.514	0.84

Table 1. Key metrics comparing the accuracy of models with the Viewmaker to default augmentations

decade. While more research is needed to determine an exact cause, I believe the relatively poor performance of the adversarial and diverse Viewmakers on their own is their inability to generate feature preserving transformations that require modification of the entire image (such as cropping, rotations, etc). This is likely why the diverse Viewmaker achieved the performance that it did; the subset of the default augmentations were able to compensate for this inability while the Viewmaker was able to replace the Gaussian blur and color distortions of the default augmentations. Alternatives to using this subset of default augmentations are discussed below.

The adversarial Viewmaker yielded remarkably poor performance, well below that of the other models. This is likely to be due to overfitting due to improper hyperparameter tuning rather than a flaw in the model as opposed to a fundamental problem with the method (considering the adversarial Viewmaker has proven effective in self-supervised learning) [12]. This is supported by the training accuracy, which rapidly approaches a value near 100 percent while the validation accuracy fails to improve and, in fact, increases after the first 10 epochs (see the appendix), a clear indication of overfitting.

DISCUSSION

By far the largest limitation of the Viewmaker is the fact that it cannot generate feature preserving views that exceed the distortion budget (ie, cropping, rotations, etc in the context of image classification). Due to the current structure of the Viewmaker, it would be extremely difficult, if not impossible, to add support for these types of transformations as it would require a fundamental shift in how the distortion budget is used. Likewise, it would likely require the Viewmaker to incorporate at least some domain specific knowledge (for instance, inverting an image does not remove any important features while inverting any sort of audio would destroy all features). Thus, implementing alternative data augmentation schemes (such as AutoAugment) but limiting such schemes to the transformations the Viewmaker is unable to generate.

Likewise, layering the augmentations of the diverse and adversarial Viewmakers may provide a path towards improved general and worst-case performance (considering the two models have similar yet distinct goals). This would likely require extensive distortion budget tuning, however, as layering distortions can rapidly destroy an image and make the task impossible (even for a human).

CONCLUSION

This paper extends Viewmakers beyond purely adversarial learning in a self-supervised scenario to both adversarial and diverse learning in a supervised context and demonstrates strong performance in the task of image classification. These results indicate that Viewmakers provide a path towards more general machine learning algorithms capable of pretraining on arbitrary data and domains.

CONTRIBUTIONS

This work is based on my research conducted in CS197 alongside my partners Elijah Vela and Marco Pizarro. As can be verified from the git commit history, the overwhelming majority of the code was written by me, based upon the work of my mentor Alex Tamkin. This entire paper was written solely by me with no help (or even revisions) from either my mentor or my project members.

REFERENCES

[1] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*. PMLR, 1597–1607.

[2] Ekin D. Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V. Le. 2019. AutoAugment: Learning Augmentation Policies from Data. (2019).

[3] E. D. Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V. Le. 2020. Randaugment: Practical automated data augmentation with a reduced search space. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (2020), 3008–3017.

[4] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014a. Generative adversarial networks. *arXiv preprint arXiv:1406.2661* (2014).

[5] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014b. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572* (2014).

[6] Ryuichiro Hataya, J. Zdenek, Kazuki Yoshizoe, and Hideki Nakayama. 2020. Meta Approach to Data Augmentation Optimization. *ArXiv* abs/2006.07965 (2020).

[7] Adil Hamid Khan and Khadija Fraz. 2020. Post-training Iterative Hierarchical Data Augmentation for Deep Networks. In *NeurIPS*.

[8] Alex Krizhevsky, Geoffrey Hinton, and others. 2009. Learning multiple layers of features from tiny images. (2009).

[9] Sharon Y. Li. 2020. Automating Data Augmentation: Practice, Theory and New Direction. (Apr 2020). `https://ai.stanford.edu/blog/data-augmentation/`

[10] Chen Qiu, Timo Pfrommer, Marius Kloft, Stephan Mandt, and Maja Rudolph. 2021. Neural Transformation

Learning for Deep Anomaly Detection Beyond Images. *arXiv preprint arXiv:2103.16440* (2021).

[11] Alexander J. Ratner, Henry R. Ehrenberg, Zeshan Hussain, Jared Dunnmon, and Christopher Ré. 2017. Learning to Compose Domain-Specific Transformations for Data Augmentation. (2017).

[12] Alex Tamkin, Mike Wu, and Noah Goodman. 2021. Viewmaker Networks: Learning Views for Unsupervised Representation Learning. (2021).

[13] Zhiqiang Tang, Yunhe Gao, Leonid Karlinsky, Prasanna Sattigeri, Rogerio Feris, and Dimitris Metaxas. 2020. OnlineAugment: Online Data Augmentation with Less Domain Knowledge. (2020).

[14] Xinyue Zhu, Yifan Liu, Zengchang Qin, and Jiahong Li. 2017. Data augmentation in emotion classification using generative adversarial networks. *arXiv preprint arXiv:1711.00648* (2017).

APPENDIX

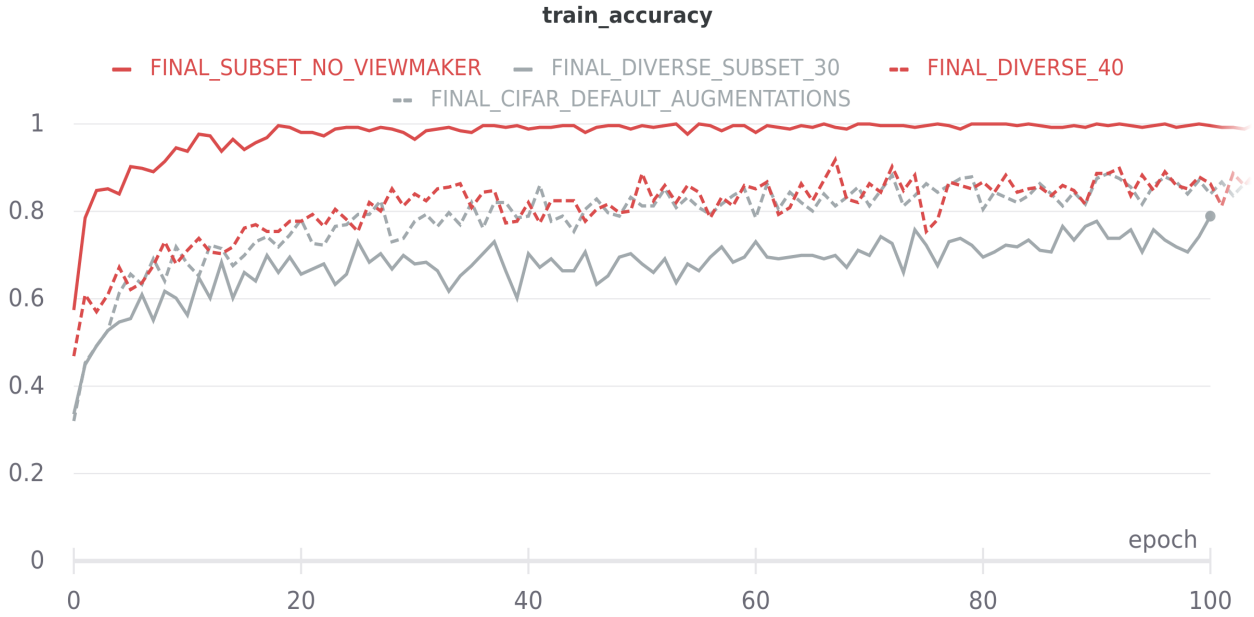

Figure 4: Training Accuracy

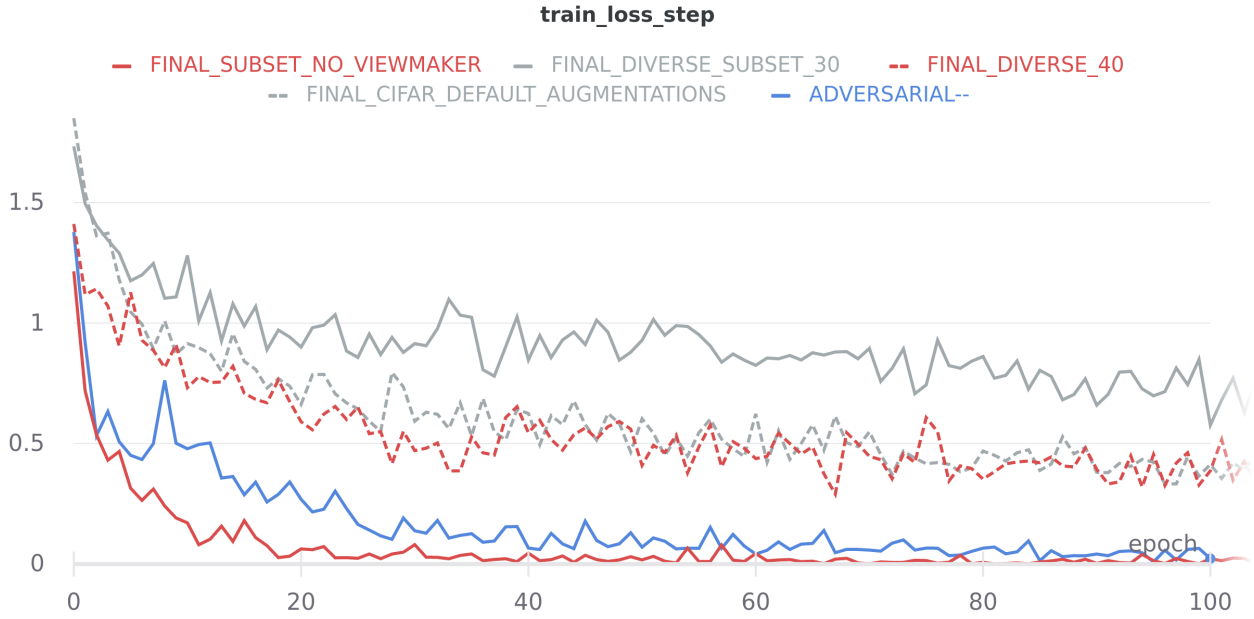

Figure 5: Training Loss

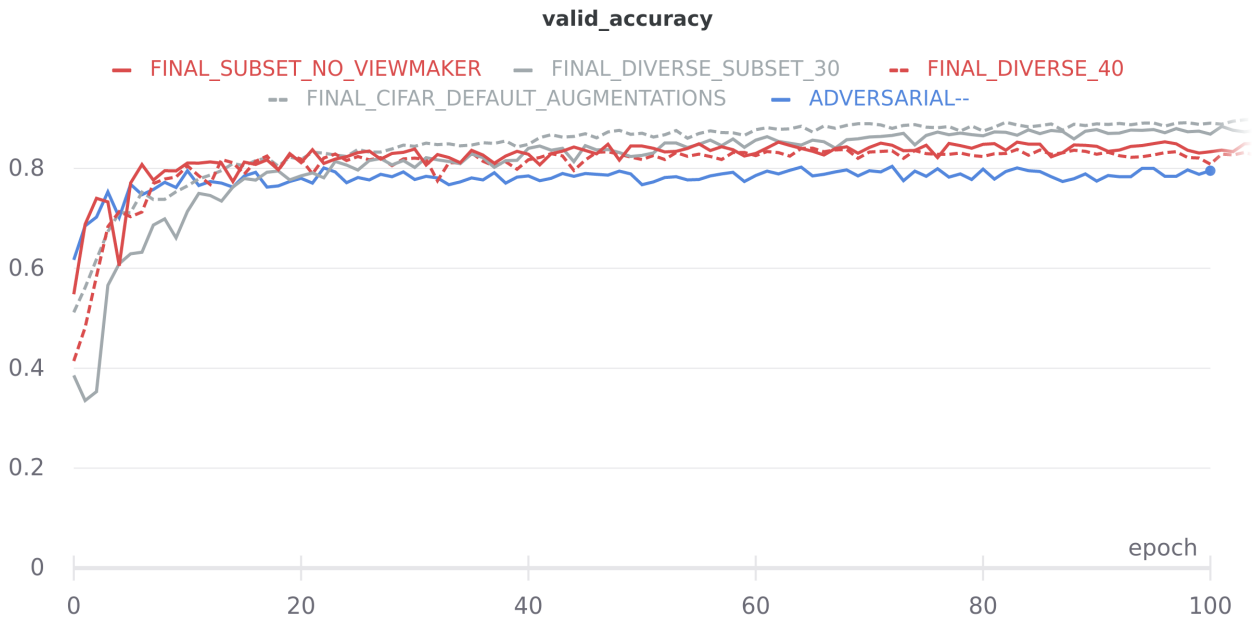

Figure 6: Validation Accuracy

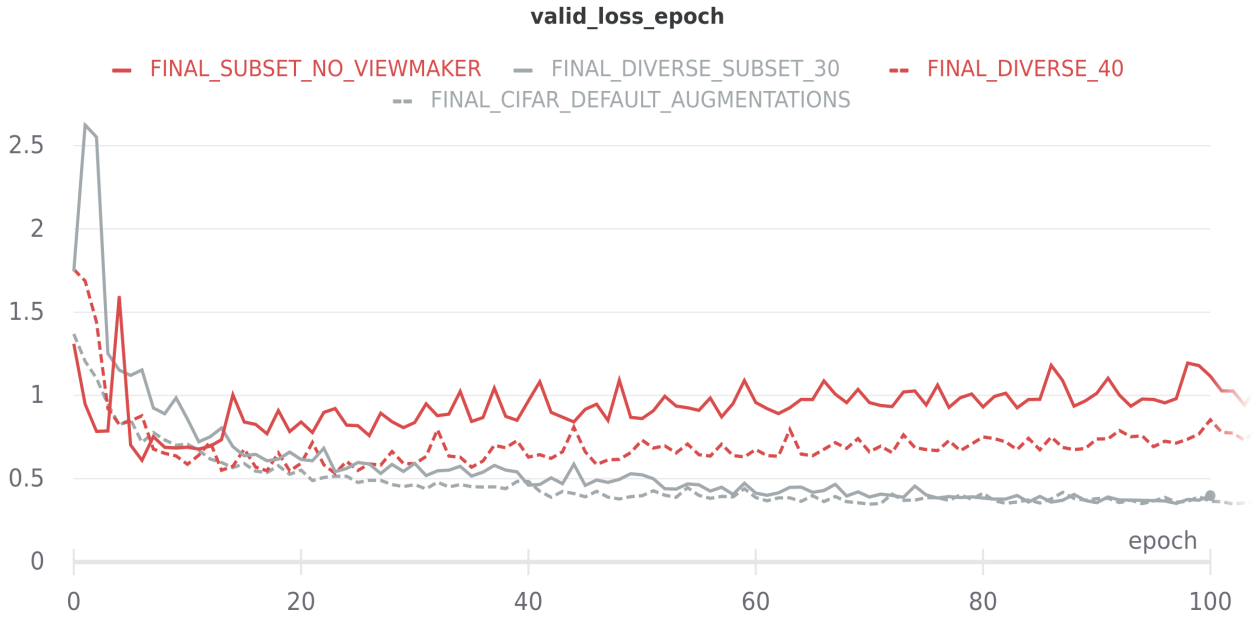

Figure 7: Validation Loss