**Intro**

When the same dataset, model architecture, and hyperparameters are used, but with varying initial parameters, the roles and contributions of specific neurons can differ following distinct training processes. This variability also extends to the different kernels of a neural network; the function of each kernel can change from one training session to another. However, the extent of this variation in their functions remains largely unexplored.

This paper specifically focuses on investigating the variations in layer feature attribution across the distinct kernels of a Naive Inception neural network (Szegedy et al., 2014). Feature attribution in machine learning is the process of identifying and quantifying the contribution of each individual input feature to the output of a model. Layer feature attribution extends this concept to the layers of a neural network, determining the impact and importance of each layer, or specific components within these layers, in influencing the model's predictions. In the context of an Inception module, layer feature attribution assesses the importance of each kernel, such as 1x1, 3x3, and 5x5. The Naive Inception model was chosen for its unique architecture that incorporates various kernels with distinct characteristics and functions within a single layer, allowing for a detailed analysis of their individual contributions. Our goal is to examine the variability in how these kernels, through their feature attribution, contribute to the overall decision-making process of the model, particularly focusing on how this contribution of different kernels might change with different hyperparameters and datasets.

Specifically, multiple datasets have been used to train the Naive Inception model architecture numerous times, during which accuracy, loss, and feature attribution were recorded. These training sessions employed a variety of hyperparameter combinations.

Our findings suggest that smaller batch sizes, higher learning rates, and more complex tasks contribute to reduced variance in feature attribution across the datasets tested. This conclusion was drawn from paired T-tests that evaluated the effects of the studied variables on feature attribution variance while holding other parameters constant.

The goals of this research are threefold:

1. To demonstrate that lower variance in feature attribution leads to more consistent models. This consistency suggests a more uniform learning pattern across the network's layers, enhancing the model's interpretability. Thus, our results provide valuable insights into hyperparameter selection for creating models that are both consistent and interpretable.

2. To establish a baseline for understanding the variance in feature attribution in relatively small models trained on simple datasets.

3. To investigate the potential of using variance in feature attribution as a tool to study the inherent learning tendencies of different layers, leading to potential branch specialization. For instance, our study shows high variance in the 3x3 layer of the Inception model; this observation leads to a hypothesis that in an Inception model, the 1x1 layers might be more adept at processing lower-level features, while the 5x5 layers could be inclined towards handling higher-level features. Meanwhile, the 3x3 layers may serve as intermediaries, adapting to various needs and, as a result, potentially exhibiting greater variance in feature attribution. This aspect of the research underscores the rationale behind selecting the Inception model for our analysis.

**Related work:**

下面可以放到related work里Inception model is a model that uses different rbanches in each stack and each branch is meant to fit on some function, fore example for a kernel with size of 1\*1, it is more efficient when processing more details, and 5\*5 is good at processing the broader picture rathe than the details.

Related work 也可以写nas和hyperprameter turning 的

Inception Model intro,

Kernel function studies

Kernel function Variability Studies,

Feature Attribution Methods intro,

Impact of Hyperparameters on Neural Networks confirm

Connection with NAS and Hyperparameter Tuning

**Experiment**

**3.1Setup**

The central concept of this experiment involves selecting specific hyperparameter combinations for each dataset through a grid search process targeting the hyperparameters we aim to investigate. Subsequently, for each hyperparameter combination, we execute the model training a total of 15 times. The number of training epochs remains fixed, contingent upon the dataset's complexity, predetermined through preliminary testing. During this training process, we record the training and test accuracy achieved after every epoch of training, as well as the feature attribution of the kernels or branches at the conclusion of the last training epoch, under the assumption that the model performs adequately on the test set. DeepLIFT and Integrated Gradient are utilized to determine feature attribution, with both scores being recorded and analyzed separately (described in section 4).

hyperparams\_choice\_list = {

'initial\_lr': [0.01, 0.001, 0.007, 0.003],

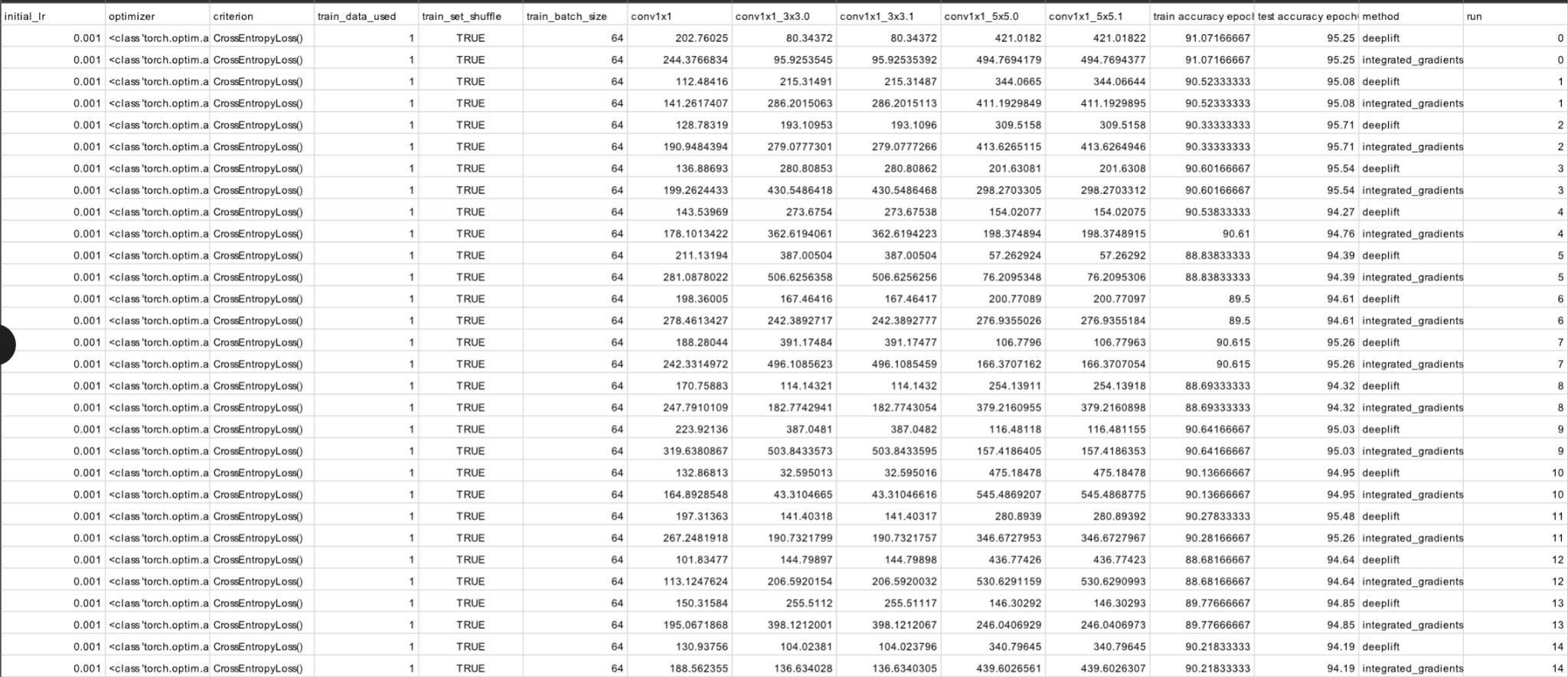
'optimizer': [torch.optim.Adam, torch.optim.SGD],

'train\_data\_used': [0.5, 1],

'train\_set\_shuffle': [True, False],

'train\_batch\_size': [64, 32, 16]

}



**3.2 preliminary conclusion and stage 2 setup**

After conducting two sets of tests on the MNIST and Fashion MNIST datasets, the results indicate that the percentage of train data used does not have a significant influence on feature attribution. Consequently, the second stage of the experiment is now solely focused on variations in layer feature attribution caused by training set shuffling, initial learning rate, and batch size. While SGD has a similar effect when using a smaller batch size, we have also omitted it from our set of focusing variables.

3.3 baseline

For the Fashion MNIST models, we constructed three distinct inception-like architectures to serve as a baseline for assessing variations in feature attribution for comparison purposes. Each of these architectures featured kernels of the same size in each branch, with the option to use all 1\*1 kernels, all 3\*3 kernels, or all 5\*5 kernels.

3.4 model in experiment

Naive inception model architecture

3.5 dataset used

For the actual experiment, we did experiment describe in setup on dataset mnist fashion mnist, svhn, grtsb, from the most simple to the most difficult.

3.6 feature attribution method

Feature attribution method deep lift and inter grated gradient is chosen.

3.7 data analysis steps

After record the variance of feature attribution in 3\*3\*2\*15 = 270 train process, for each method, we first ignore all the combinations that have an average of the 15 test accuracy after their final epoch lower than 70.

Then we normalize the feature attribution scores, such as the branches within this layer or stack sums to 1. And for each combination, within these 15 training process, we can get 15 feature attribution scores on each branch. We can calculate the average of these scores and a variance on these scores and normalize the variance using the average, so the resulted one variance for each branch is normalized to be compared fairly.

And then the we do a variance feature attribution paired T test when fixing all the hyperparamtser and leave one to vary. For example, when fixing all other rhyperparamters , we can compare the variance difference between the result we got when batch size is 16 and when batch size is 64. We both compared the average variance of all branches’ feature attribution and also the variance of each branches’ feature attribution.

The T test fixing all hyperparamter leaving one out to see the impact of that hyper amateur, and a cross dataset analysis is also done for comparing the impact of dataset on these differences.

4result

4.1 preliminary experiments result

The preliminary experiments is only on fashion mnist and mnist, the result shows that optimizer of sgd , and its influence is the same as decreasing the batch size, as it should be.

4.2 baseline result

4.3 experiment result

graph shown to give a big picture then only talk about paired t test result

5 discussion

More model architecture can be test, more dataset, this is an experiment only fixing on naive inception . As model architecture and dataset complexity both increase , it is interesting to see what will happen then, Willy he variance normalized decrease? Or will they increase.

Other than methods of feature attribution, to study the specialization of branches itself, methods like T sne probing, visualization such as a normalized cumulative grad cam can be applied, Will provide better and more direct insights.

Further deeper understanding of the variance in mode’s inner layer functions will further improve the field of nas by understanding the innate specialization potential in certain layer design on some data. To build more controlled and interpretable models.

6 conclusion

Reference

Szegedy, Christian, et al. "Going deeper with convolutions." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015.