Comparative Analysis of Optimizers to Improve the Model of Image Classification

Abstract

Image classification is an important task in computer vision applications, and it is crucial to maximizing the model's accuracy and computational efficiency. In this project, we perform comparative analysis on various optimizer (Jia, 2019)[1] to enhance the performance of the image classification model. We examine the performance of four popular optimization algorithms Stochastic Gradient Descent (SGD), Adagrad, Adam, and RMSprop using different dataset to train our model. We evaluate the performance of each optimizer using measures for accuracy and computational efficiency, and we observe how their hyperparameters affect the performance of the model. This analysis provides useful insights into the selection of optimization algorithms for different types of datasets.

Introduction

Image Classification is the process of categorizing an image into one or more classes based on its characteristics. It is one of the predominant techniques used widely in Deep Learning. Image classification has numerous applications such as medical imaging, self-driving cars, and surveillance and for this reason, the accuracy of the image classification models is crucial. The performance of these models greatly depends on the selection of hyperparameters, most importantly Optimizers (Ashia C. Wilson, 2018)[2]. Optimizer algorithms iteratively update the attributes of the neural network such as weights and biases during training to reduce the loss function, thus improving the accuracy.

Several types of Optimizers are used in training neural networks. Stochastic Gradient Descent (SGD) is a widely used optimizer, but it has a significant disadvantage of slow convergence and difficulty in obtaining the local minimum. To overcome this, adaptive algorithms like Adam, Adagrad, and RMSprop were introduced. These algorithms accelerate convergence and improve accuracy by modifying the learning rate for each parameter based on its historical gradients. In addition to these improvements, the adaptive algorithms also showed several disadvantages such as the tendency to oscillate and converge to suboptimal solutions. Adam Optimizer has recently become quite popular due to its superior performance and quick convergence rates. It combines the advantages of momentum-based and adaptive optimization techniques.

In this project, we aim to perform a comparative analysis of different optimization algorithms to improve the Image classification models. We will specifically analyze the efficiency of the SGD, Adam, Adagrad, and RMSprop optimization algorithms on popular image classification architectures like ResNet50, Xception, InceptionResnet50, and VGG16. We will examine the impact of the model performance across various datasets and offer insights into the relationship between optimizers, model performance, and dataset characteristics. The results of this project will assist in choosing the best optimization algorithm for specific model architecture and datasets, thus improving the accuracy and reliability of the image classification models.

Problem Statement

The problem addressed by this project is the challenge of improving the accuracy of image classification models by performing analysis and comparison of different optimizers. Image classification is one of the crucial tasks in computer vision and it is used in wide range of applications, some of which are critical and can have life-saving or life-threatening consequences.

Medical imaging is one of the applications that uses image classification for diagnosing various diseases like neurological disorders, heart diseases and cancer. Accuracy in this case is very crucial since it can facilitate the patients with timely diagnosis, appropriate treatment which in turn can improve patient's condition. Image classification is used in video surveillance and when achieved high accuracy, suspicious activities or individuals can be identified, potential threats such as weapons or dangerous materials can be detected. Autonomous vehicles are currently one of the most cutting-edge areas of technologies and it uses image classification to identify objects and obstacles on the road. High accuracy is essential to prevent accidents and to ensure the safety of passengers and pedestrians. One of its other applications is environment monitoring, it identifies and tracks wildfires, deforestation, and climate change. Models must be accurate to detect environmental risks and protect natural resources.

Performance of image classification models are influenced by various factors and one of the critical factors is the optimizer that is used to train the model. The goal of this project is to perform a comparative analysis of various optimizers, including stochastic gradient descent (SGD), Adam, Adagrad, and RMSProp, to determine which optimizer is the best choice for image classification tasks. The result of this project helps to increase the accuracy of image classification models.

Method

In this project, we compare the performance of four optimization algorithms – Stochastic Gradient Descent (SGD), Adam, Adagrad, and RMSprop over 4 different Architectures – VGG16, ReseNet50, InceptionResNetV2, Xception using 4 datasets. These optimization algorithms differ in the way they update weights and biases during the training phase.

Data Preparation

We considered four datasets to evaluate the performance of different optimizers and architectures in Image classification tasks.

MNIST dataset consists of a collection of 70,000 grayscale images of handwritten digits (0-9) with 60,000 training images and 10,000 testing images, each of which is 28x28 pixels in size.

Flower Recognition is a dataset containing 4242 images divided into 5 categories namely chamomile, tulip, rose, sunflower, and dandelion. The images are not standardized to a single size and have different proportions, with an average size of 320x240 pixels.

Water Level Recognition is a collection of 486 images that has 3 categories full water level, half water level, and Overflow. Out of 486 images, 308 are full water level, 139 are half water level, and 39 are overflowing.

Chest X-Rays dataset contains 5,856 validated chest X-ray images that are in the anterior-posterior view, including both normal and pneumonia cases. Each image is labeled as either 'NORMAL', 'BACTERIA', or 'VIRUS' followed by a random patient ID and image number.

Optimization Algorithms

Adam (Adaptive Moment Estimation) (Kingma D. P., 2015)[3] combines the benefits Adagrad and RMSProp. It computes the individual adaptive learning rate for different parameters by using the estimates of first and second moments of gradients which helps for the faster convergence during optimization process.

Adagrad (Adaptive Gradient) (John Duchi, 2011)[5] Algorithm adapts the learning rate for each parameter based on past gradients during training phase. It does this by gathering the squared gradients of each parameter over time and uses this information to adapt the learning rate. By this the Adagrad gives more importance to parameters that receive smaller gradients and vice-versa.

RMSProp (Root mean square propagation) (Ruder)[6] adjusts the learning rate based on the magnitudes of gradients, and this help to resolve the vanishing and exploding gradient problem.

SGD (Stochastic Gradient Descent) (Yingjie Tian)[7] approximates the gradients of loss function by using a random sample of training data. It updates the model parameters in the negative direction with a fixed learning rate. It allows for more frequent updates of model parameters which is used to escape from local minima.

Architectures

VGG16 (Qing Guan, 2021)[8] consists of 16 layers, including 13 convolutional layers and 3 fully connected layers and all layers are stacked one after the other, each layer has a filter of size 3x3 with a stride of 1. The last layer uses a SoftMax activation function which generates class probabilities. Also, it is one of most used architecture for image classification tasks.

ResNet50 is a 50-layer network, including 49 convolutional layers and 1 fully connected layer, in this the convolutional layers are arranged into groups where each group has different number of filters. It also uses the skip connections to address the vanishing gradient problem.

InceptionResNetV2 (Christian Szegedy)[9] consists of 164 layers, and it is a hybrid architecture that combines the inception module with residual connections. It has a greater number of filters in each layer, and the number of filters increases as the network becomes deeper.

Xception consists of 126 layers; it also consists of depth wise separable convolutions which was introduced in inception to minimize the number of parameters and increase the efficiency. It also uses the skip connections to prevent vanishing gradient problem.

Implementation

For each dataset, we loaded the pre-trained model with ImageNet weights and froze the first few layers, as these layers have already learned the general features from the ImageNet. Then we added a top model on top of these frozen layers, which includes a flattened layer to convert the output of the convolutional layers into a 1-D vector. We also added two dense layers and two dropout layers, which are helpful for preventing overfitting. Finally, an output layer with appropriate number of nodes and a softmax function was added based on the number of classes in each dataset.

While conducting this analysis, we will implement each model such that it has the same values for hyperparameters (James Bergstra, 2012)[4]. The hyperparameters include the learning rate, the number of epochs, and the activation function. The reason for holding the values of other hyperparameters as the constant is to facilitate the most accurate study of the impact of the Optimization algorithms on the efficiency of the Image classification model.

Evaluation metrics that are used to measure the efficiency of the Optimization algorithms for each architecture are Accuracy and Loss. Accuracy is defined as the ability of the model to correctly classify the images, that is, the percentage of correctly classified images out of the total number of images in the dataset. On the other hand, loss refers to the error between the predicted outputs and the actual outputs. Loss is typically calculated using cross-entropy loss, which measures the difference between the predicted probability distribution and the true probability distribution of the classes. The goal is to minimize the loss while training to improve the accuracy of the model.

Executed System Configuration

Processor: Apple M1 Pro 10-Core CPU
Graphics: 14-core Apple M1 Pro GPU
Memory: 16 GB unified memory
Storage: 512 GB SSD
Operating System: macOS Ventura 13.2.1
Programming Language: Python 3
Tools: Google Colab, VSCode

Results

Water Bottles Dataset

Architecture	**Adam**	**Adagrad**	**RmsProp**	**SGD**
VGG16	91.54	95.77	88.73	85.91
ResNet50	71.83	73.23	66.19	64.78

InceptionResnetV2	85.91	90.14	85.91	80.28
Xception	94.36	91.54	83.09	83.09

Table 1. Water Bottles – Accuracy (%) Table

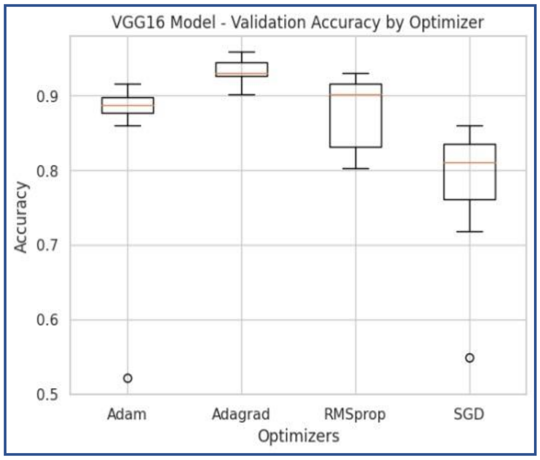

Fig 1.1. Water bottles-VGG16

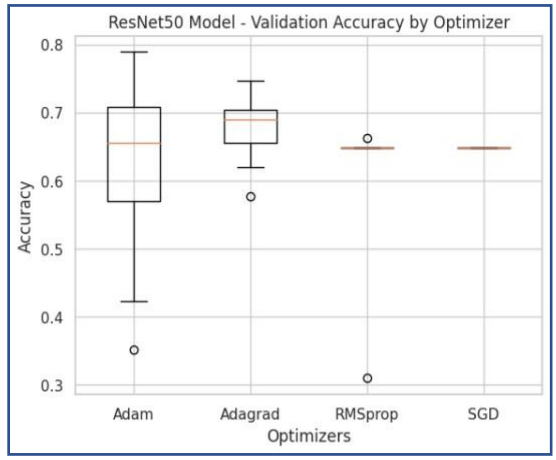

Fig 1.2. Water bottles-ResNet50

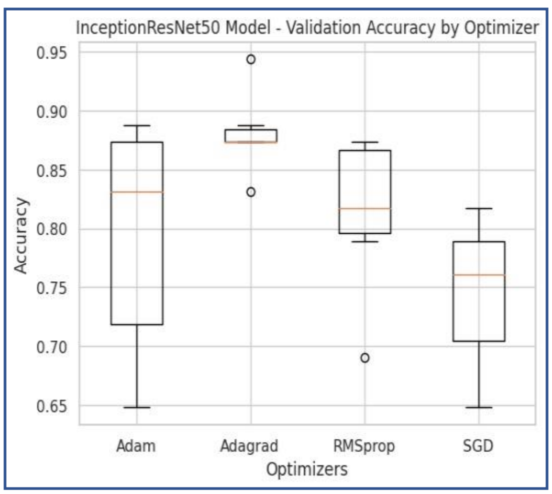

Fig 1.3. Water bottles-InceptionResNet50

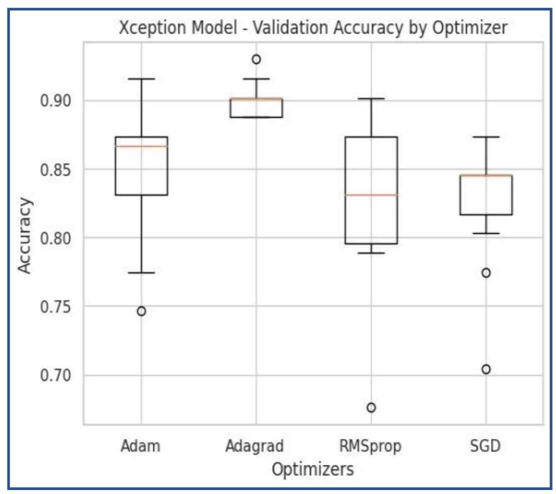

Fig 1.4. Water bottles-Xception

MNIST Dataset

Architecture	**Adam**	**Adagrad**	**RmsProp**	**SGD**
VGG16	78.24	78.05	78.98	25.09
ResNet50	81.01	83.33	83.14	25.18
InceptionResnetV2	25.55	25	25.09	25.09

Xception	27.40	25.46	25.09	25.09

Table 2. MNIST – Accuracy (%) Table

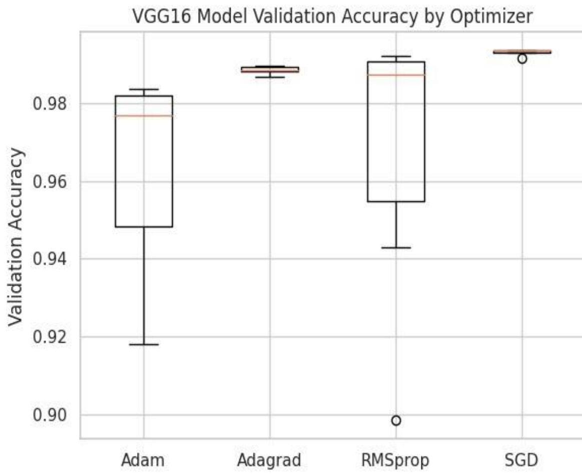

Fig 2.1. MNIST-VGG16

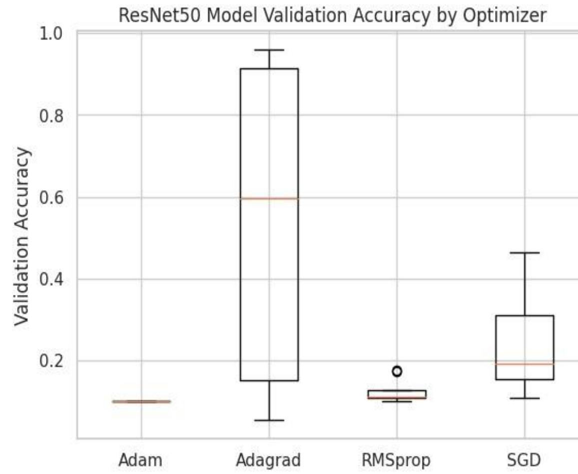

Fig 2.2. MNIST-ResNet50

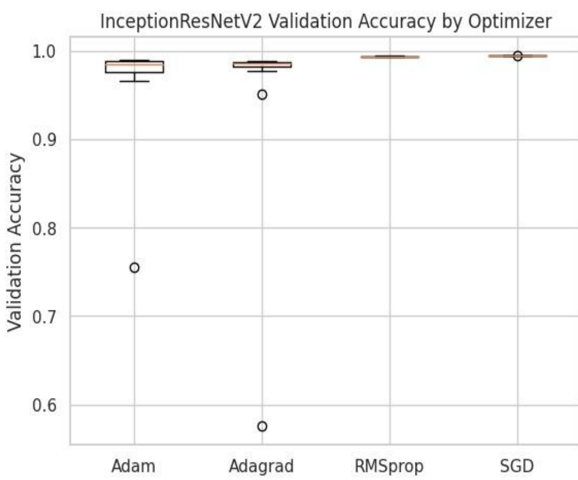

Fig 2.3. MNIST-InceptionResNetV2

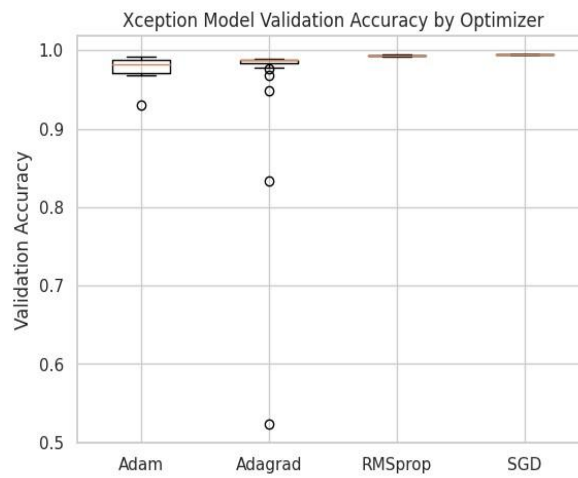

Fig 2.4. MNIST-Xception

Appendix

Flowers Dataset

Architecture	**Adam**	**Adagrad**	**RmsProp**	**SGD**
VGG16	97.73	97.32	97.19	97.06
ResNet50	93.11	93.23	90.94	74.23
InceptionResnetV2	97.95	97.95	98.46	98.34
Xception	96.55	97.83	97.44	98.08

Table 3. Flowers-Accuracy (%) Table

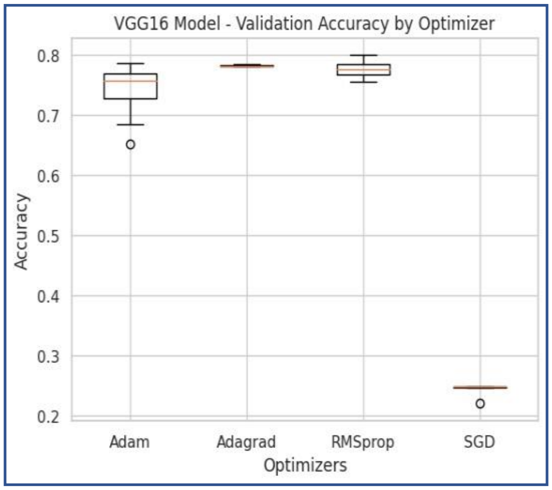

Fig 3.1. Flowers-VGG16

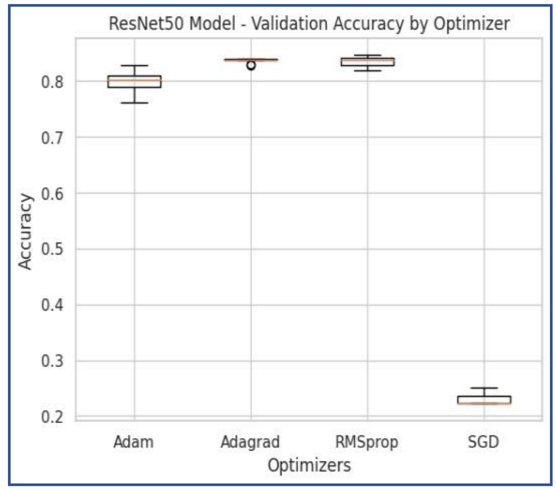

Fig 3.2. Flowers-ResNet50

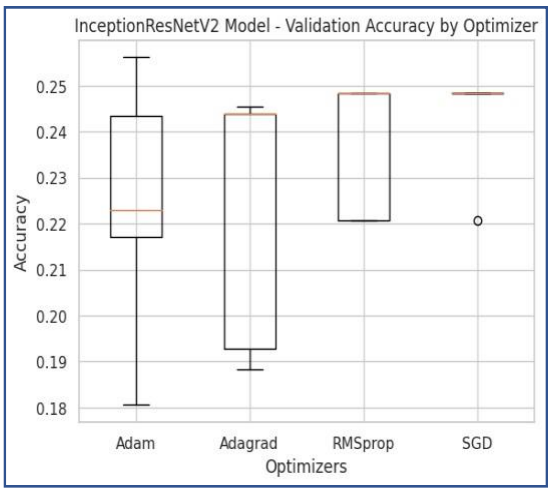

Fig 3.3. Flowers-VGG16

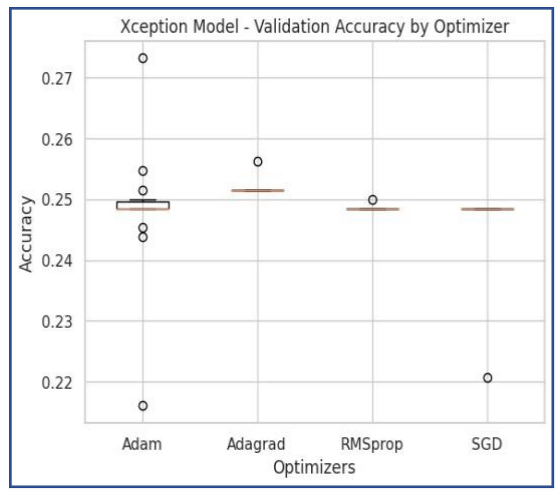

Fig 3.4. Flowers-ResNet50

Pneumonia Dataset

Architecture	Adam	Adagrad	RmsProp	SGD
VGG16	98.95	98.55	98.44	96.08
ResNet50	94.35	15.06	24.05	90.54
InceptionResnetV2	98.35	98.64	99.45	99.45
Xception	94.36	91.54	83.09	83.09

Table 4. Pneumonia-Accuracy (%) Table

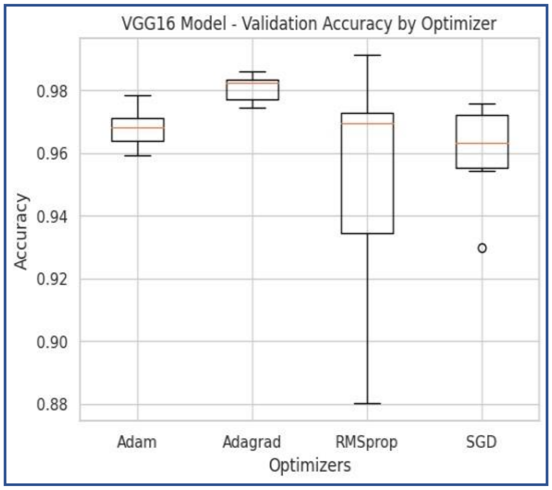

Fig 4.1. Pneumonia-VGG16

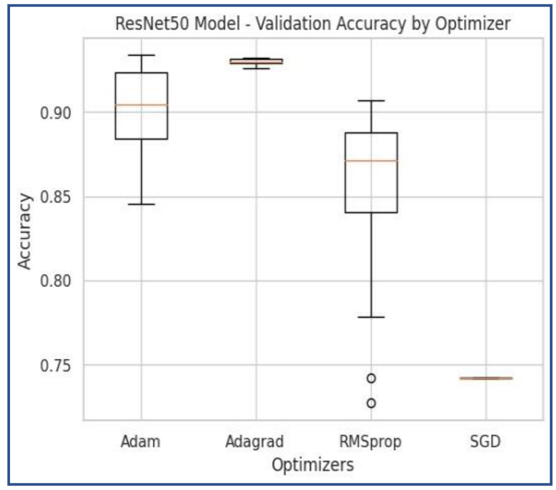

Fig 4.2. Pneumonia-ResNet50

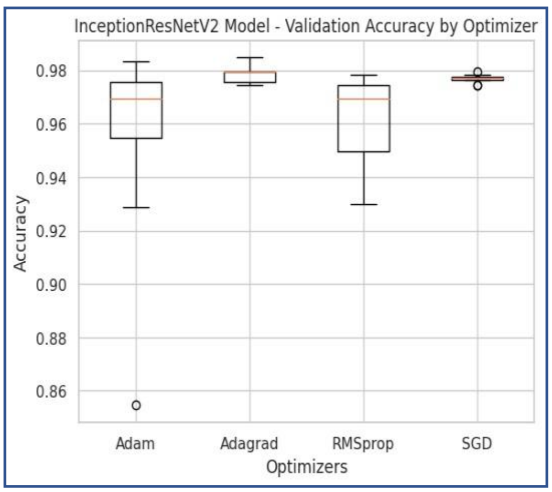

Fig 4.3. Pneumonia-InceptionResNetV2

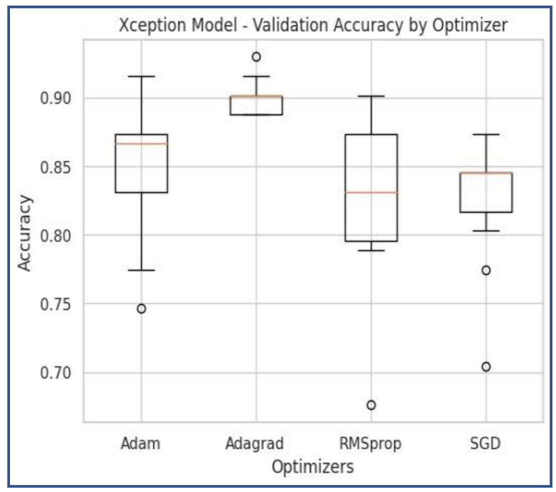

Fig 4.4. Pneumonia-Xception

From the above plots, we can compare the range of validation accuracy values for each optimizer, as well as the distribution of values within the box. We can also look for any outliers, which may indicate that an optimizer is particularly good or bad at handling certain data points. Additionally, we can calculate the standard deviation of the validation accuracy values for each optimizer to get a more precise estimate of the variability of the optimizers

Overall, Adam optimizer outperforms other optimizers across different architectures and datasets. This is because Adam is adaptive to all the architectures due to its adaptive learning rates. It can adjust the learning rate based on the magnitude of the gradients. These speeds up the convergence and help to avoid getting stuck in the local minima. Also, Adam Optimizer is robust to noisy and sparse gradients, which may occur in some applications.

References

1. Jia, D. L. (2019). A Survey of Optimization Methods for Deep Learning.
2. Ashia C. Wilson, R. R. (2018). The Marginal Value of Adaptive Gradient Methods in Machine Learning. *Journal of Machine Learning Research*.

3. Kingma, D. P. (2015). Adam: A Method for Stochastic Optimization. *International Conference on Learning Representations (ICLR).*
4. James Bergstra, Y. B. (2012). Random Search for Hyper-Parameter Optimization. *Journal of Machine Learning Research*, 281-305.
5. Christian Szegedy, S. I. (n.d.). Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. *Thirty-First AAAI Conference on Artificial Intelligence* (pp. 4278-4284). San Francisco, California, USA.: AAAI Press.
6. Ruder, S. (n.d.). An overview of gradient descent optimization algorithms.
7. Yingjie Tian, Y. Z. (n.d.). Recent Advances in Stochastic Gradient Descent in Deep Learning.
8. Qing Guan, Y. W. (2021). Deep convolutional neural network VGG-16 model for differential diagnosing of papillary thyroid carcinomas in cytological images: a pilot study. *Journal of Medical Systems*, 122