Comparative Analysis of Optimizers and Neural Network Architectures:

SGD vs. Adam and MLP vs. CNN

Kim Min-Seo

June 2, 2024

Abstract

The purpose of this study is to compare the performance of optimization algorithms and neural network architectures. First, we compared the performance of Stochastic Gradient Descent (SGD) and Adam optimization algorithms on a Multilayer Perceptron (MLP) model using the MNIST dataset. The comparison results showed that the Adam algorithm converged faster and more stably than SGD, achieving higher performance. Subsequently, using the Adam algorithm, we compared the performance of MLP and Convolutional Neural Networks (CNN). The results indicated that CNN outperformed MLP on the validation dataset, achieving over 99% accuracy and lower loss. These findings suggest that the choice of optimization algorithm and neural network architecture significantly impacts model performance, highlighting the importance of selecting appropriate algorithms and architectures for various problems.

**Contents**

1. Introduction

1.1 Objectives of the Study

2. Literature Review

2.1 Overview of Stochastic Gradient Descent (SGD) and Adam Optimizer

2.2 Overview of Multilayer Perceptron (MLP) and Convolutional Neural Network (CNN)

3. Methodology

3.1 Description of the MNIST Dataset

4. Experience

4.1. SGD vs. Adam Optimizer on MLP

4.1.1 Model Architecture of MLP

4.1.2 Training Procedure

4.1.3 Performance Metrics (Loss and Accuracy)

4.2. MLP vs. CNN using Adam Optimizer

4.2.1. Model Architectures of MLP and CNN

4.2.2. Training Procedure

4.2.3. Performance Metrics (Loss and Accuracy)

5. Result and Discussion

5.1 Comparative Analysis of SGD and Adam on MLP

5.2 Comparative Analysis of MLP and CNN using Adam

6. Conclusion

6.1 Summary of Findings

1. Introduction

1.1 Objectives of the Study

The purpose of this study is to compare optimization algorithms and the performance of CNN and MLP models.

First, we compare the optimization algorithms by analyzing the performance of Stochastic Gradient Descent (SGD) and Adam optimization algorithms. Through this comparison, we aim to clearly understand the differences in the learning processes of a Multilayer Perceptron (MLP) model and the respective strengths and weaknesses of each algorithm.

Next, we compare the neural network architectures. Using the better-performing algorithm from the initial comparison, we evaluate the performance of MLP and Convolutional Neural Networks (CNN). This comparison aims to analyze the performance differences between the two neural network architectures and assess the characteristics and applicability of each model.

2. Literature Review

2.1 Overview of Stochastic Gradient Descent (SGD) and Adam Optimizer

Stochastic Gradient Descent (SGD) is a widely used algorithm for optimizing machine learning and deep learning models. Instead of using the entire dataset, SGD updates the model's weights by selecting a small, random subset of the data (batch) in each iteration. This approach reduces computational costs and allows efficient operation on large datasets. However, SGD can experience instability and oscillation during the training process, leading to slower convergence.

Adam (Adaptive Moment Estimation) is an optimization algorithm that combines the advantages of Momentum and RMSProp. Adam adjusts the learning rate for each weight individually and utilizes past gradient information to facilitate fast and stable learning. By reducing oscillations during training, Adam converges faster than SGD. Due to these characteristics, Adam is widely used across various deep learning models.

2.2 Overview of Multilayer Perceptron (MLP) and Convolutional Neural Network (CNN)

A Multilayer Perceptron (MLP) is one of the most basic forms of neural networks, consisting of an input layer, multiple hidden layers, and an output layer. Each neuron in a layer is connected to all neurons in the previous layer, and activation functions add non-linearity to the model. While MLP can be applied to various data types like images, text, and audio, it has limitations in handling high-dimensional data such as images.

Convolutional Neural Networks (CNN) are neural network architectures primarily used for image recognition and processing. A CNN consists of convolutional layers, pooling layers, and fully connected layers. Convolutional layers extract local patterns from images, and pooling layers reduce spatial dimensions to create feature maps. Due to this structure, CNNs effectively learn the spatial hierarchical structure of images and perform exceptionally well in image classification, object detection, and other computer vision tasks.

3. Methodology

3.1 Description of the MNIST Dataset

The MNIST dataset is an image classification dataset consisting of 60,000 images of handwritten digits from 0 to 9. Each image is a grayscale image with a size of 28x28 pixels and pixel values ranging from 0 to 255. This MNIST dataset is used to compare the performance of the experiments described below.

4. Experience

4.1. SGD vs. Adam Optimizer on MLP

4.1.1 Model Architecture of MLP

In this experiment, we used an MLP model with two hidden layers for MNIST image classification. The input layer flattens the 28x28 images into a 1x784 vector. The first hidden layer is a fully connected layer with 784 input neurons and 256 output neurons, followed by a ReLU activation function and a 30% dropout. The second hidden layer is a fully connected layer with 256 input neurons and 10 output neurons. The 10 output neurons correspond to the 10 classes in the MNIST dataset. Finally, a softmax function is used to calculate the probabilities for each class from the 10 output neurons.

4.1.2 Training Procedure

The entire training dataset is randomly shuffled and then split into training and validation sets in a ratio of 0.85:0.15. The data is then loaded into a DataLoader with a batch size of 32, preparing the dataset in mini-batch form. CrossEntropyLoss is used as the loss function during training, and the model's performance is evaluated on the validation set at each epoch.

4.1.3 Performance Metrics (Loss and Accuracy)

During the training process, loss and accuracy are recorded at each epoch. These metrics are used to evaluate the performance of the optimizers, assessing their effectiveness from various perspectives, such as learning trends and absolute values. This allows for a comparison of the performance of the MLP model trained with SGD and Adam optimization algorithms, analyzing the impact of each optimizer on the learning process and outcomes.

4.2. MLP vs. CNN using Adam Optimizer

4.2.1. Model Architectures of MLP and CNN

The MLP model used in this section is the same as the one described in the previous section.

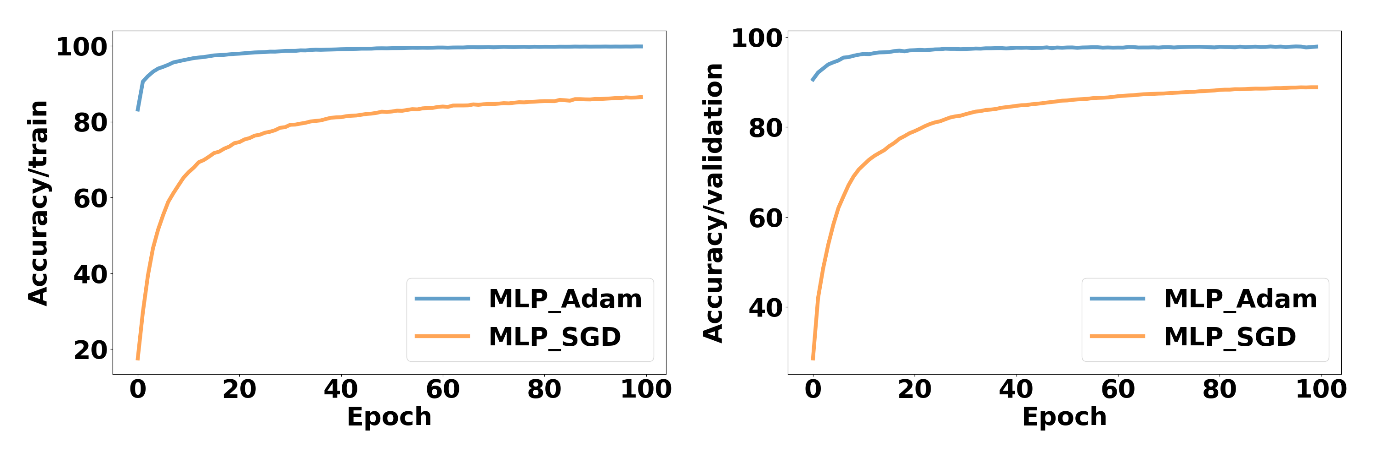
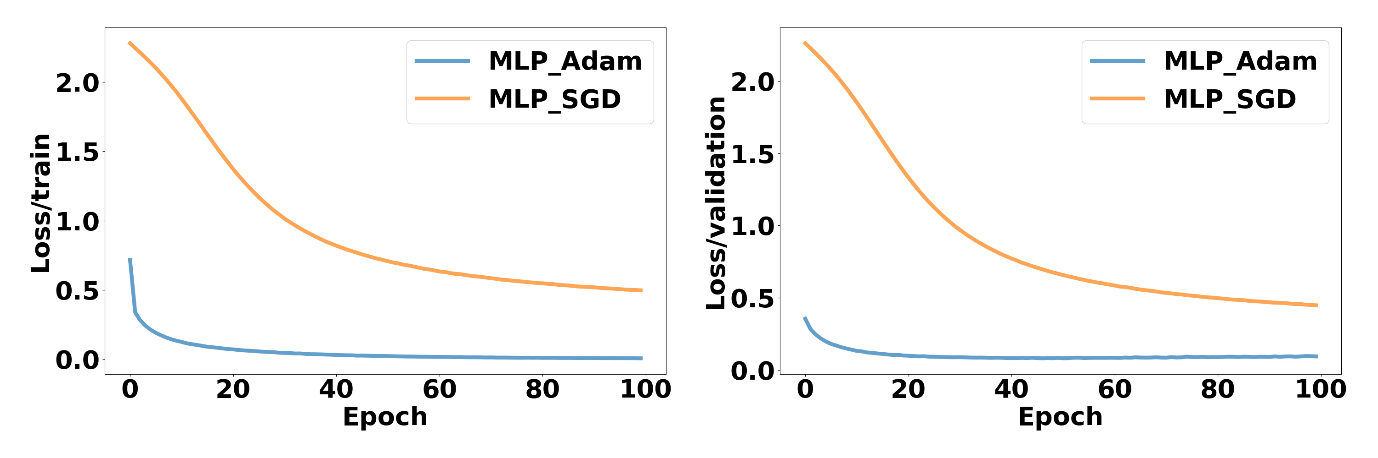
The CNN model consists of two convolutional layers. The first convolutional layer has 1 input channel, 32 output channels, a kernel size of 3, and padding of 1. The second convolutional layer has 32 input channels, 64 output channels, a kernel size of 3, and padding of 1. Each convolutional layer is followed by a ReLU activation function, a max-pooling layer with a kernel size of 2 and padding of 2, and a 25% dropout. Following these layers, the input is flattened to a size of 7x7x64 and passed through a fully connected layer with 256 nodes, followed by a ReLU activation function and a 50% dropout. Finally, the output is passed through another fully connected layer with 10 nodes, corresponding to the 10 classes in the MNIST dataset.

4.2.2. Training Procedure

The same data loading and preprocessing, loss function, and training procedures as in the previous experiments are applied here.

4.2.3. Performance Metrics (Loss and Accuracy)

Similar to the previous experiment, loss and accuracy are recorded at each epoch during the training process. These metrics are used to compare the performance of MLP and CNN, analyzing learning trends and absolute values from various perspectives.

5. Result and Discussion

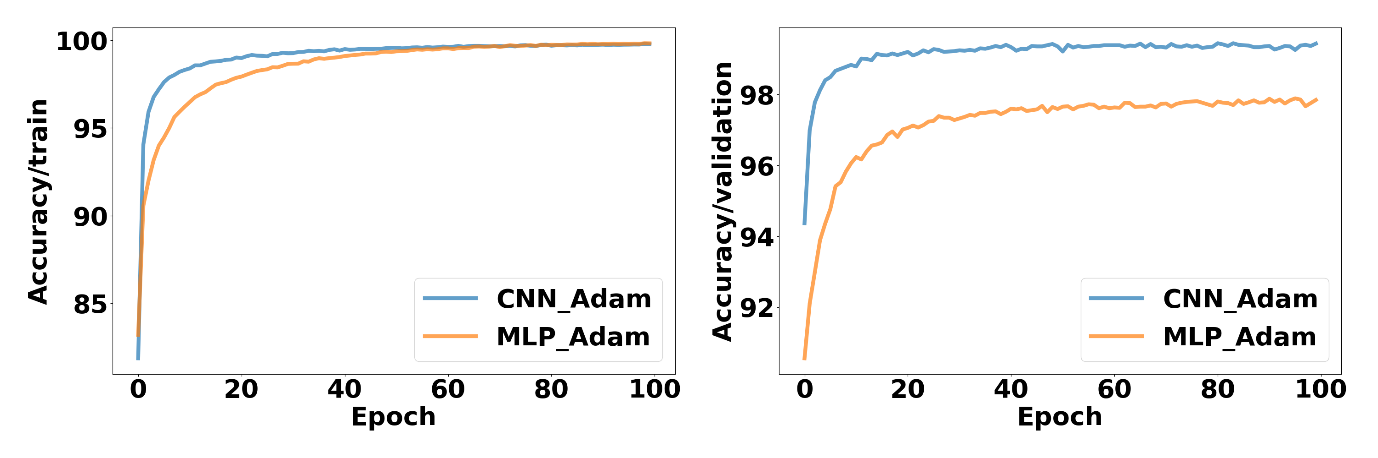
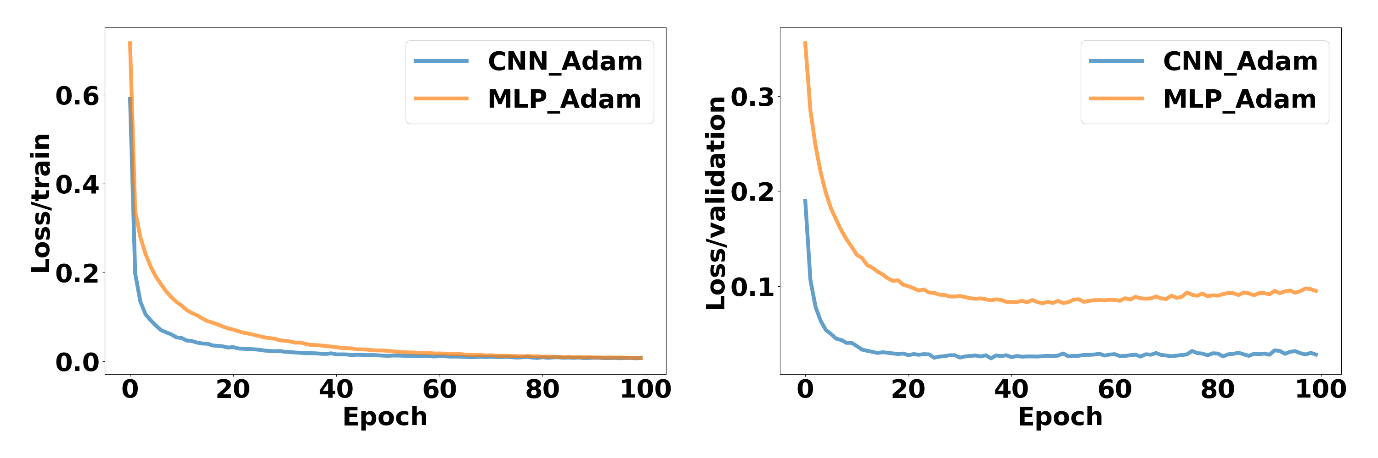
5.1 Comparative Analysis of SGD and Adam on MLP

The MLP model using the Adam optimization algorithm showed a tendency for accuracy to increase rapidly from the initial stages of training. In contrast, the model using SGD displayed a relatively slower increase in accuracy over the same number of epochs. This can be interpreted as Adam achieving faster and more stable optimization because it adjusts the learning rate for each parameter individually.

In terms of the loss metric, the model using the Adam optimizer demonstrated a more rapid decrease in loss. While the loss for the SGD model gradually decreased, the Adam model's loss dropped sharply during the initial epochs and consistently maintained a lower loss value thereafter. This is because Adam combines the advantages of momentum and RMSProp, reducing oscillations during training and converging more stably.

The difference in learning speed between SGD and Adam was also very noticeable. SGD showed slower learning, with gradual improvements in loss reduction and accuracy increase. This can be attributed to SGD using a single learning rate, which can lead to oscillations during training. On the other hand, Adam rapidly decreased the loss and increased accuracy during the early stages of training, consistently maintaining high performance. Adam's dynamic adjustment of the learning rate and utilization of past gradient information allowed for faster convergence.

In conclusion, the Adam optimization algorithm proved to be more efficient and faster in enhancing the performance of the MLP model using the MNIST dataset compared to SGD. This suggests that Adam could also exhibit superior performance when dealing with more complex neural network structures and larger datasets. The changes in loss and accuracy during the training process clearly indicate that Adam is a more effective optimization algorithm than SGD.

5.2 Comparative Analysis of MLP and CNN using Adam

This section provides a comparative analysis of the performance of MLP and CNN models using the Adam optimization algorithm. The performance metrics used are accuracy and loss, and the performance changes were observed over a total of 100 epochs.

In the training dataset, the CNN model showed a rapid increase in accuracy from the early stages of training. In contrast, the MLP model's accuracy increased more slowly compared to CNN, but eventually, both models achieved over 99% accuracy within 100 epochs. Regarding the loss metric, the CNN model initially experienced a faster reduction in loss; however, the MLP model also steadily decreased its loss, with both models reaching approximately 0.005 in loss by the end of training.

In the validation dataset, the performance differences between the two models were more pronounced. The MLP model achieved validation accuracy exceeding 97%, while the CNN model demonstrated superior performance with accuracy over 99%. For the loss metric, the CNN model's validation loss decreased to 0.002, whereas the MLP model's validation loss only reduced to 0.008. This indicates that CNN has better generalization performance compared to MLP.

Overall, the CNN model recorded higher accuracy and lower loss in both training and validation datasets compared to the MLP model. The superior performance of CNN, particularly in the validation dataset, is attributed to its ability to effectively learn spatial features of images through convolutional layers

6. Conclusion

6.1 Summary of Findings

The comparison between the SGD and Adam optimization algorithms revealed that the Adam algorithm provided faster and more stable performance in training the MLP model. The MLP model using Adam showed a rapid increase in accuracy and a sharp decrease in loss from the initial stages of training, maintaining high performance throughout the training process. In contrast, the MLP model using SGD exhibited gradual improvements in accuracy and loss, but the learning speed was slower compared to Adam.

Next, comparing the performance of MLP and CNN models using the Adam optimization algorithm showed that the CNN model outperformed the MLP model. Both models recorded high accuracy and low loss in the training dataset; however, the CNN model demonstrated higher accuracy and lower loss in the validation dataset. This superior performance is attributed to the CNN's ability to effectively learn the spatial features of images through its convolutional layers. Specifically, the CNN model achieved over 99% accuracy and a loss below 0.002 in the validation dataset, confirming its superior generalization performance compared to the MLP model.

In conclusion, this study confirmed that the Adam optimization algorithm is more effective for training the MLP model, and the CNN model performs better than the MLP model in image classification tasks. These findings highlight the significant impact of choosing the right optimization algorithm and neural network architecture on model performance, emphasizing the importance of selecting appropriate algorithms and architectures for various datasets and problem types.