CS-233: Introduction to Machine Learning - Milestone 2 Report

Yshai Dinée-Baumgarten (356356) Ethan Boren (361582) Gabriel Taieb (360560)

June 2, 2024

1 Introduction

For the second milestone of our project, we implemented several machine learning methods to classify images from the Fashion-MNIST dataset. The dataset contains 60,000 training images and 10,000 test images of fashion items categorized into 10 classes. We implemented three neural network architectures using PyTorch: a Multi-Layer Perceptron (MLP), a Convolutional Neural Network (CNN), and a Vision Transformer (ViT). Additionally, we utilized Principal Component Analysis (PCA) for dimensionality reduction, specifically for the MLP. This report outlines our implementation process, experimentation results, and insights gained from our analyses.

2 Methodology

2.1 Data Preparation

We began by loading the Fashion-MNIST dataset using the provided `load_data()` function from `data.py`. We flattened the images obtained into vectors and then split the data into training and validation sets (80-20 split) to facilitate model evaluation during development. Data normalization was performed so that each feature has a mean of 0 and a standard deviation of 1. This results in faster convergence. Finally, we reshaped the data to fit each model individually.

2.2 Models and Implementation

The three models were implemented in `deep_network.py`.

2.2.1 Multi-Layer Perceptron (MLP)

The MLP was implemented with solely fully connected layers. The architecture consisted of an input layer, three hidden layers with ReLU activation, and an output layer with softmax activation (which was already implemented into the `nn.CrossEntropyLoss` function) . We used PCA for dimensionality reduction to improve training efficiency and performance. The PCA was implemented in `pca.py`, where we computed the principal components and reduced the data dimensionality.

2.2.2 Convolutional Neural Network (CNN)

Our Convolutional Neural Network (CNN) architecture is designed to effectively capture the spatial hierarchies present in image data. The network consists of three convolutional layers, each followed by ReLU activation and max-pooling, and three fully connected layers.

This architecture leverages the spatial structure of images, capturing local patterns and reducing computational complexity through pooling layers. The combination of convolutional and fully connected layers allows the network to learn both local and global features, leading to robust performance in classification tasks.

2.2.3 Vision Transformer (ViT)

Our Vision Transformer (ViT) model, leveraging attention mechanisms, is designed to capture long-range dependencies within image data. Unlike traditional convolutional neural networks (CNNs) that focus on local features through convolutional layers, the ViT treats an image as a sequence of patches, enabling it to model global relationships between distant pixels effectively.

This approach includes multi-head self-attention layers, which allow the model to weigh the importance of different patches when making predictions, followed by feed-forward networks that process these relationships further.

As a relatively novel approach to image processing, the ViT represents a significant shift from traditional convolution-based methods, opening new possibilities for future research and application.

Despite these advantages, the ViT model's performance can be dependent on the amount and quality of training data, often requiring large datasets to achieve its full potential. This makes it a compelling choice for applications where data availability and computational resources are abundant.

2.3 Training and Evaluation

Each model was trained using the cross-entropy loss function and the Adam optimizer. We experimented with different learning rates and batch sizes to optimize performance. The models were evaluated based on accuracy and macro F1-score, computed using validation data.

3 Experiments and Results

3.1 Model Parameters

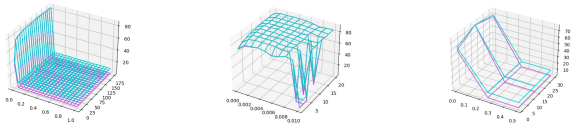
The table below gives the parameters used to achieve the performances detailed in table 2.

Parameter	**MLP**	**CNN**	**ViT**
Learning rate	2e-3	2e-3	1e-3
Batch size	256	64	256
Number of Epochs	10	6	10
PCA	Yes	No	No
PCA components	100	-	-

Table 1: Model Parameters

3.2 Hyperparameter Tuning

We conducted extensive hyperparameter tuning to determine the optimal configurations for each model. For the MLP, we tested different numbers of principal components in PCA, finding that reducing the data to 100 components provided a good balance between training time and performance. We varied learning rates and number of epochs to achieve the highest accuracy.



(a) MLP (b) CNN (c) ViT

Figure 1: Accuracy and F1-score for different learning rates and number of epochs.

3.3 Performance Comparison

The table below summarizes the performance of each model on the validation set:

Model	Accuracy (%)	Macro F1-Score
MLP	86.175	0.8610
CNN	90.392	0.9037
ViT	79.450	0.7830

Table 2: Performance of each model on the validation set.

3.4 PCA Impact on MLP

Incorporating PCA significantly reduced the training time for the MLP without a substantial loss in accuracy. The reduced dimension MLP trained faster (approximately 60% less time) compared to the non-reduced version, with only a slight drop in accuracy.

3.5 Runtime Analysis

Training times varied across models. The CNN required the longest training time due to its complex layers, while the MLP with PCA was the fastest. Prediction times also varied depending on the model, with the MLP being again the fastest.

Model	Training Time (s)	Prediction Time (s)
MLP	3.82	0.03
CNN	202.16	2.63
ViT	1477.68	15.01

Table 3: Training and prediction times for each model (10 epochs on a MacBook Pro M1).

4 Discussion and Conclusion

The Convolutional Neural Network (CNN) demonstrated superior performance in accuracy and F1-score, underscoring its capability to effectively capture and utilize spatial hierarchies within image data. This was achieved through its layered structure of convolutions and pooling, which progressively abstracted and combined features from the input images.

The Vision Transformer (ViT), while not outperforming the CNN, exhibited a novel approach by leveraging self-attention mechanisms to model global relationships between image patches. This innovative architecture holds promise, especially in scenarios with ample data and computational resources, and could benefit from further hyperparameter tuning and potentially larger datasets to fully realize its potential.

The Multi-Layer Perceptron (MLP) benefited significantly from the application of Principal Component Analysis (PCA), which reduced the dimensionality of the input data and, in turn, decreased training time without a notable loss in performance. This highlights the practical advantage of incorporating PCA for efficiency, particularly when dealing with high-dimensional data.

In conclusion, each model brought unique strengths to the table: the CNN for its robust spatial feature extraction, the ViT for its advanced handling of global dependencies, and the MLP for its efficiency aided by dimensionality reduction. Future work could explore hybrid models that combine these strengths, as well as further optimization and experimentation with additional datasets and architectures.