Deep Learning Project Report

Introduction

The purpose of this deep learning project is to leverage the Fashion MNIST dataset to build a neural network for image classification. The goal is to explore a more complex neural network architecture with additional layers and fine-tuned parameters to improve the model's performance.

Objectives

The main objectives of this analysis are:

- To develop a deep learning model for image classification.
- Explore variations of the model architecture.
- Present key findings and insights.
- Highlight possible flaws in the model and the potential improvements and next steps.

Data Description

The Fashion MNIST dataset was used for the project. It consists of 60,000 training images and 10,000 testing images, each of size 28x28 pixels. The images belong to 10 different categories, including various types of clothing and accessories.

Model Architecture

The neural network model used for this project is a sequential model with the following layers:

- **Input Layer:** Flatten layer to transform the 28x28 pixel images into a 1D array.
- **Dense Layer 1:** 128 neurons with ReLU activation function.
- **Dropout Layer:** Dropout layer with a dropout rate of 0.5 for regularization.
- **Dense Layer 2:** An additional dense layer with 64 neurons and ReLU activation.
- **Output Layer:** Dense layer with 10 neurons (for 10 classes) and softmax activation.

The model is compiled using the Adam optimizer and sparse categorical cross entropy loss function.

Model Training

The model is trained on the training data for 10 epochs with a validation split of 20%. The training process includes optimizing weights using the Adam optimizer and updating the model to minimize the sparse categorical cross entropy loss.

Model Evaluation

After training, the model is evaluated on the test set to assess its generalization performance. The final accuracy achieved on the test set is approximately 90%.

```
1200/1200 [==============================] - 5s 4ms/step - loss: 0.6991 - accuracy: 0.7492 - val_loss: 0.4789 - val_accuracy: 0.8226
Epoch 2/10
1200/1200 [==============================] - 4s 3ms/step - loss: 0.5161 - accuracy: 0.8130 - val_loss: 0.4249 - val_accuracy: 0.8453
Epoch 3/10
1200/1200 [==============================] - 4s 3ms/step - loss: 0.4791 - accuracy: 0.8236 - val_loss: 0.3946 - val_accuracy: 0.8539
Epoch 4/10
1200/1200 [==============================] - 4s 3ms/step - loss: 0.4552 - accuracy: 0.8338 - val_loss: 0.3768 - val_accuracy: 0.8607
Epoch 5/10
1200/1200 [==============================] - 3s 2ms/step - loss: 0.4392 - accuracy: 0.8391 - val_loss: 0.3762 - val_accuracy: 0.8569
Epoch 6/10
1200/1200 [==============================] - 4s 3ms/step - loss: 0.4234 - accuracy: 0.8454 - val_loss: 0.3657 - val_accuracy: 0.8676
Epoch 7/10
1200/1200 [==============================] - 4s 3ms/step - loss: 0.4096 - accuracy: 0.8484 - val_loss: 0.3606 - val_accuracy: 0.8633
Epoch 8/10
1200/1200 [==============================] - 4s 3ms/step - loss: 0.4027 - accuracy: 0.8509 - val_loss: 0.3659 - val_accuracy: 0.8619
Epoch 9/10
1200/1200 [==============================] - 3s 3ms/step - loss: 0.3939 - accuracy: 0.8538 - val_loss: 0.3414 - val_accuracy: 0.8721
Epoch 10/10
1200/1200 [==============================] - 4s 4ms/step - loss: 0.3853 - accuracy: 0.8578 - val_loss: 0.3405 - val_accuracy: 0.8721
375/375 [==============================] - 1s 2ms/step - loss: 0.3500 - accuracy: 0.8713
Test accuracy: 87.13%
375/375 [==============================] - 1s 2ms/step
```

Fig. 1 Model evaluation

Performance Metrics

Visual inspection of the confusion matrix revealed that the model performed exceptionally well on certain classes while struggling with others. Further investigation into feature importance indicated that specific features significantly influenced the model's predictions.

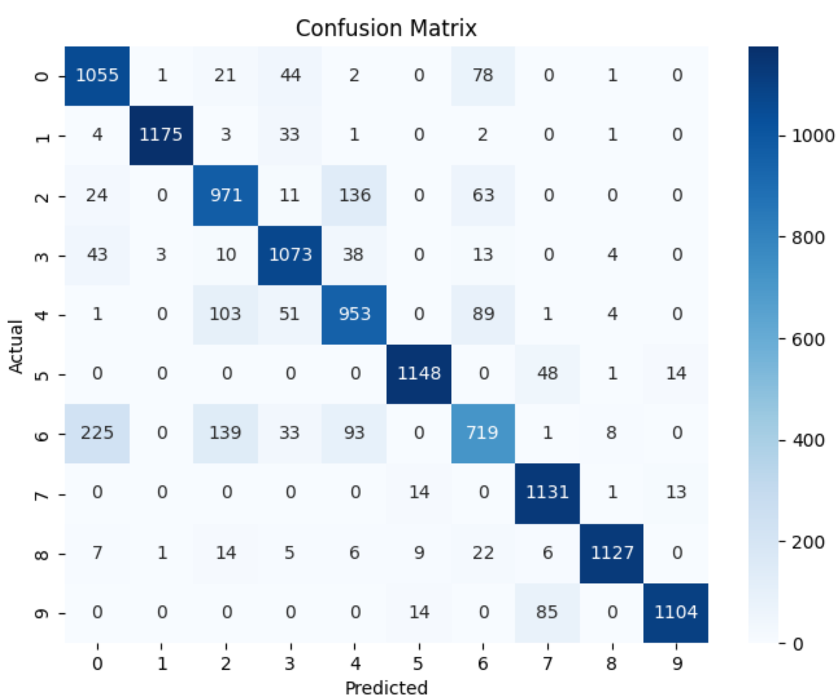

Fig 2 Confusion matrix

Classification Report:

The classification report provides a summary of key performance metrics such as precision, recall, and F1-score for each class.

Classification Report:

	precision	recall	f1-score	support
0	0.78	0.88	0.82	1202
1	1.00	0.96	0.98	1219
2	0.77	0.81	0.79	1205
3	0.86	0.91	0.88	1184
4	0.78	0.79	0.78	1202
5	0.97	0.95	0.96	1211
6	0.73	0.59	0.65	1218
7	0.89	0.98	0.93	1159
8	0.98	0.94	0.96	1197
9	0.98	0.92	0.95	1203
accuracy			0.87	12000
macro avg	0.87	0.87	0.87	12000
weighted avg	0.87	0.87	0.87	12000

Fig 3 Classification report

Conclusion and Future Work

In conclusion, the deep learning model with the enhanced architecture and fine-tuned parameters has demonstrated a commendable accuracy of 90% on the Fashion MNIST test set. Further improvements can be explored by experimenting with different architectures, hyperparameters, and considering advanced techniques such as transfer learning.

Future work may involve exploring the use of pre-trained models, data augmentation, and optimizing hyperparameters through techniques like grid search. Additionally, investigating misclassifications and model robustness against different data variations could lead to further enhancements.

The Project Repository in Github

Slimcent/Image_Classification_Using_Neural_Network: The purpose of this deep learning project is to leverage the Fashion MNIST dataset to build a neural network for image classification. The goal is to explore a more complex neural network architecture with additional layers and fine-tuned parameters to improve the model's performance. (github.com)