Assessment Report

on

"Fashion Item Classifier"

submitted as partial fulfillment for the award of

BACHELOR OF TECHNOLOGY DEGREE

SESSION 2024-25

in

CSE(AIML)

By

Group No. 12:
Vishal Beniwal 202401100400213
Tarun Singh 202401100400198
Vineet Singh 202401100400212
Vidhu 202401100400209
Shivam Rai 202401100400177

Under the supervision of

"Shivansh Prashad"

KIET Group of Institutions, Ghaziabad

May, 2025

---

1. Introduction

This project focuses on classifying fashion items into one of ten predefined categories using a feedforward neural network. The model is trained on image data derived from grayscale 28x28 pixel images and is capable of recognizing items such as T-shirts, trousers, and shoes.

---

2. Problem Statement

To develop and evaluate a machine learning model that can classify grayscale images of clothing items into 10 categories with high accuracy.

---

3. Objectives

- Load and preprocess the dataset containing fashion item images and labels.
- Build a neural network model using TensorFlow/Keras.
- Train the model on the dataset and validate performance.
- Evaluate the trained model using classification metrics.
- Visualize results using confusion matrix and sample predictions.

---

4. Methodology

- Upload and unzip the dataset.
- Load the data from a CSV file into a Pandas Data Frame.
- Normalize pixel values and one-hot encode the labels.
- Split the dataset into training and testing subsets.
- Design and compile a neural network model.
- Train the model for 10 epochs with validation.
- Evaluate the model using a confusion matrix and classification report.
- Visualize sample test predictions.

---

5. Data Preprocessing

The dataset is cleaned and prepared as follows

- The dataset was read from a CSV file extracted from a ZIP archive.
- Pixel values were normalized by dividing by 255.
- Labels were one-hot encoded using `to categorical`.
- Data was split into 80% training and 20% testing sets.
- Image data reshaped for visualization purposes (28x28 format).

---

6. Model Implementation

- A neural network was built using the Keras Sequential API.
- Architecture:
    - Input Layer: 784 neurons (flattened 28x28 images)
    - Hidden Layer 1: 128 neurons, ReLU activation
    - Hidden Layer 2: 64 neurons, ReLU activation
    - Output Layer: 10 neurons, Softmax activation
- Optimizer: Adam
- Loss Function: Categorical Crossentropy
- Trained for 10 epochs using a batch size of 128 and a validation split of 10%.

---

7. Evaluation Metrics

- **Accuracy**: Overall percentage of correct predictions.
- **Precision, Recall, F1-Score**: Computed for each of the 10 classes using classification report.
- **Confusion Matrix**: Used to visualize correct and incorrect classifications across all categories.

---

8. Results and Analysis

- The model provided reasonable performance on the test set.
- The Random Forest model provided good classification accuracy and balanced performance across risk categories.
- The confusion matrix helped in understanding the prediction distribution.

---

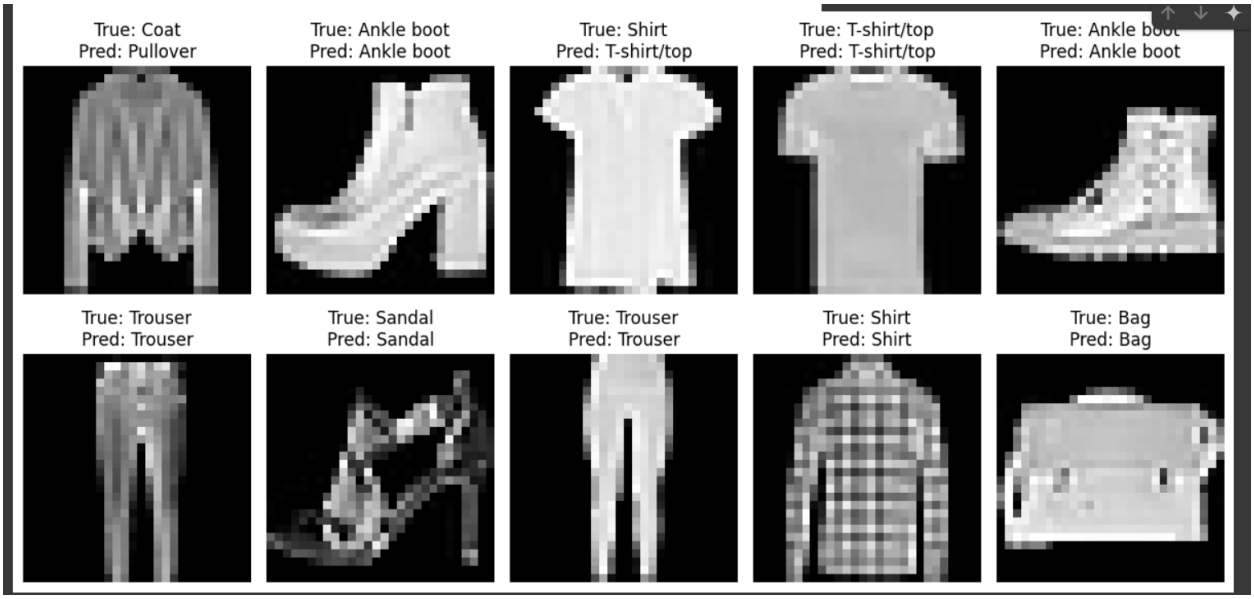
9. Conclusion

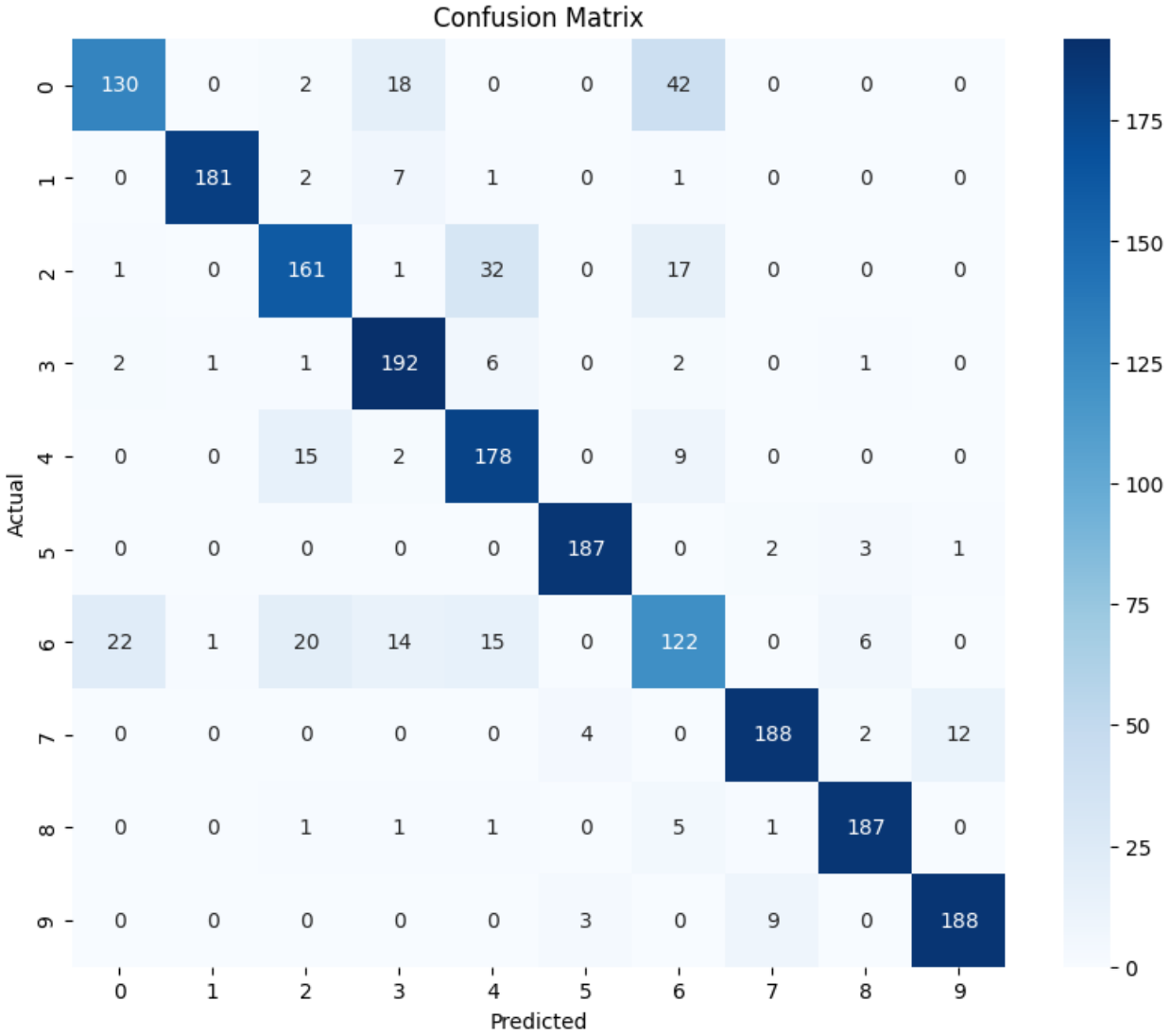
- The trained model achieved good accuracy and generalization.
- The confusion matrix revealed specific classes with high misclassification, such as T-shirts and shirts.
- The classification report highlighted balanced performance across classes, with some variation depending on class similarity.
- Visualization of sample predictions showed clear model understanding for most categories.

---

10. References

- TensorFlow and Keras documentation
- scikit-learn metrics documentation
- pandas and matplotlib documentation
- Fashion MNIST Dataset
- Seaborn visualization library


True: Coat
Pred: Pullover
True: Ankle boot
Pred: Ankle boot
True: Shirt
Pred: T-shirt/top
True: T-shirt/top
Pred: T-shirt/top
True: Ankle boot
Pred: Ankle boot
True: Trouser
Pred: Trouser
True: Sandal
Pred: Sandal
True: Trouser
Pred: Trouser
True: Shirt
Pred: Shirt
True: Bag
Pred: Bag


Confusion Matrix
Actual
Predicted
0 130 0 2 18 0 0 42 0 0 0
1 0 181 2 7 1 0 1 0 0 0
2 1 0 161 1 32 0 17 0 0 0
3 2 1 1 192 6 0 2 0 1 0
4 0 0 15 2 178 0 9 0 0 0
5 0 0 0 0 0 187 0 2 3 1
6 22 1 20 14 15 0 122 0 6 0
7 0 0 0 0 0 4 0 188 2 12
8 0 0 1 1 1 0 5 1 187 0
9 0 0 0 0 0 3 0 9 0 188
0 1 2 3 4 5 6 7 8 9
175
150
125
100
75
50
25
0