MATH201 Final Project: FashionNet

Prepared By: R Mohan Poornachandra

Problem Introduction

Imagine you've just joined a cutting-edge e-commerce company that sells thousands of fashion products every day. The company's current product search and categorization system relies heavily on manual tagging and keyword searches, leading to inefficiencies and missed sales opportunities.

To modernize operations and create a more intelligent user experience, your task is to build a machine learning model that automatically classifies fashion products (e.g., sneakers, dresses, t-shirts) based on image data.

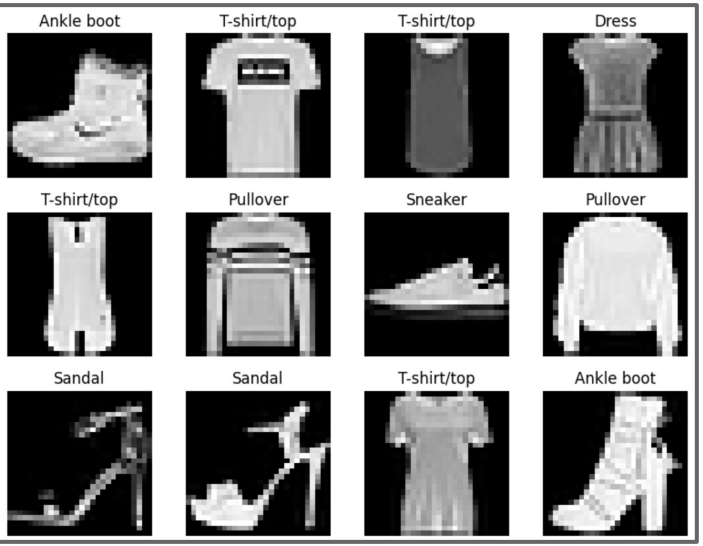
You will train a neural network to recognize these fashion items, optimize the model to improve accuracy, and analyze how feature representations (vectors) help machines "see" and "understand" images.

01 Data collection and EDA

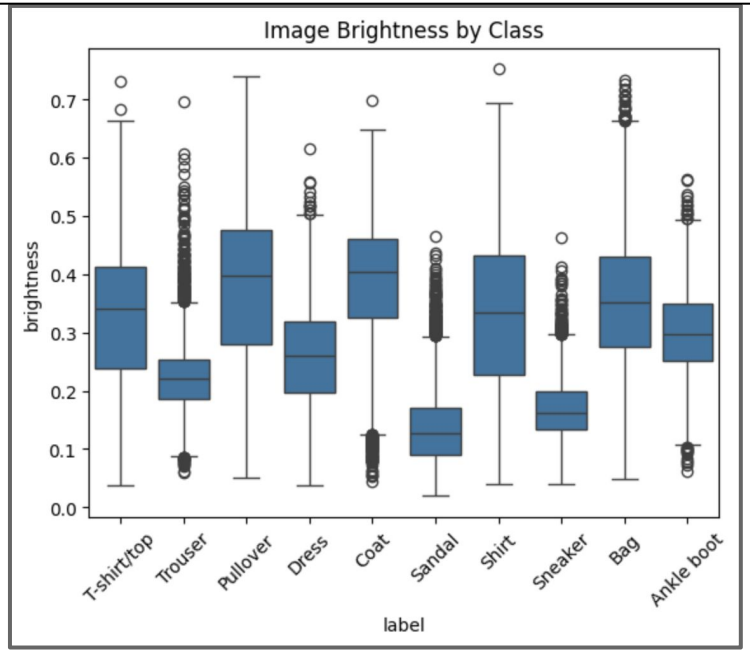
- The dataset is FashionMNIST, taken from tensorflow. It has greyscale images of fashion clothes and shoes.
- Divided into 10 categories with 6000 images for each category, making a total of 60000 images.
- Categories vary from T-shirt/top, shirt, dress, sneaker, sandal, pullover, ankle boot etc.
- There are a few complex patterns and might be confusing like T-shirt/top and shirt.

Google Colab Notebook Link

01 Data collection and EDA



Sample Data



Boxplot of Brightness in each class

label	tag
0	T-shirt/top
1	Trouser
2	Pullover
3	Dress
4	Coat
5	Sandal
6	Shirt
7	Sneaker
8	Bag
9	Ankle boot

Label-tag map

02 Neural network training

Model 1

- 784 inputs passed through a single hidden layer with 2 nodes.
- Both the neurons had sigmoid as the activation function.
- The output layer took the average of the outputs of the 2 nodes in the hidden layer.
- Loss Function: Mean Squared Error
- Optimization : Stochastic Gradient Descent.
- Accuracy: Training - 45.45% , Testing - 44.85%

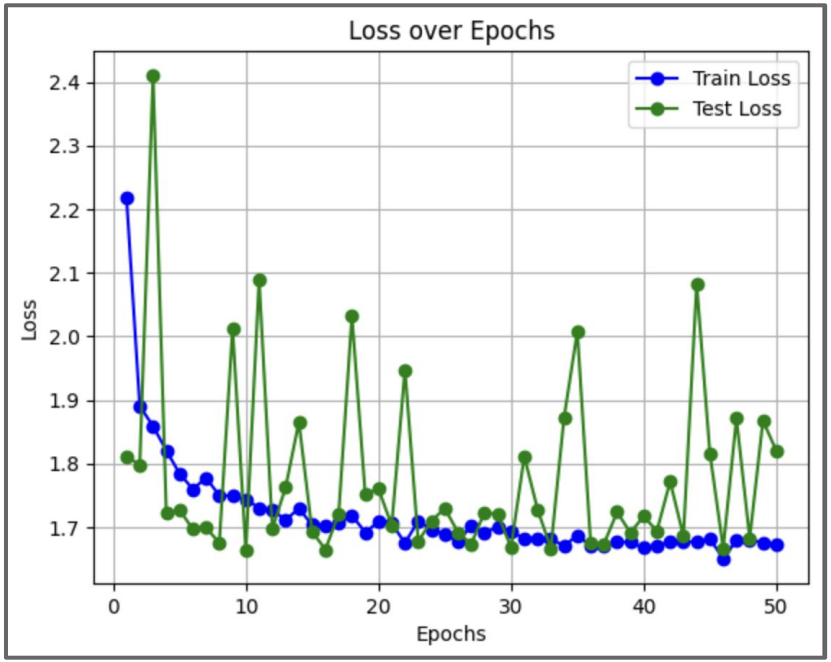
02 Neural network training

Model 2

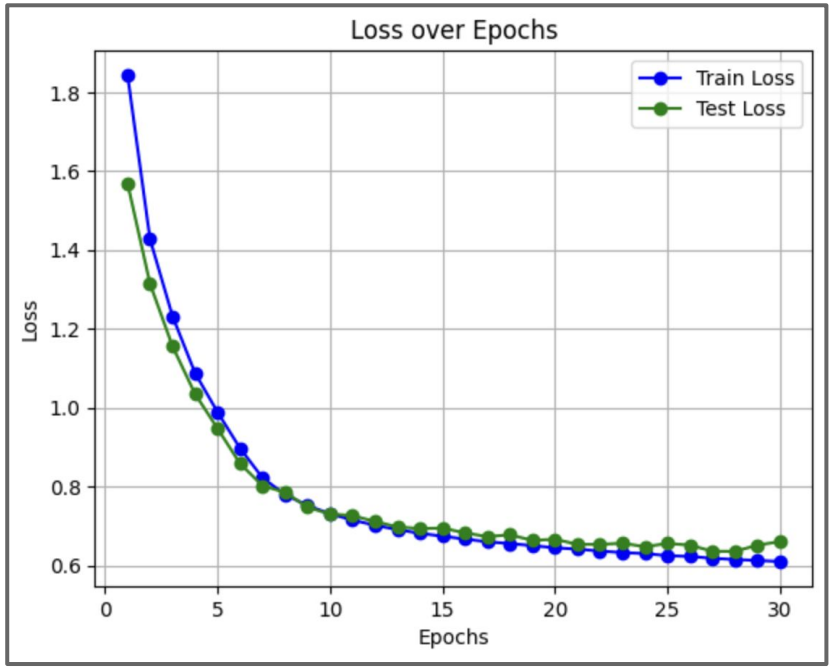
- 784 inputs passed through a sigle hidden layer with 3 nodes.
- All the nodes has sigmoid as the activation function.
- The output layer used softmax as the activation function with a total category count of 10.
- Loss Function: CategoricalCrossEntropy
- Optimization: Adam
- Accuracy: Training - 78.12%, Testing - 76.89%

Google Colab Notebook Link

02 Neural network training



Model 1



Model 2

03 Optimization and Evaluation

- Chose the hyperparameters and a few manual values in the range of initial parameters.

- Applied a Grid Search on the specific chosen hyperparameters with a fixed epoch of 10.

- A few observed points are about the hyper parameters are:

- Most effective batch size: 128

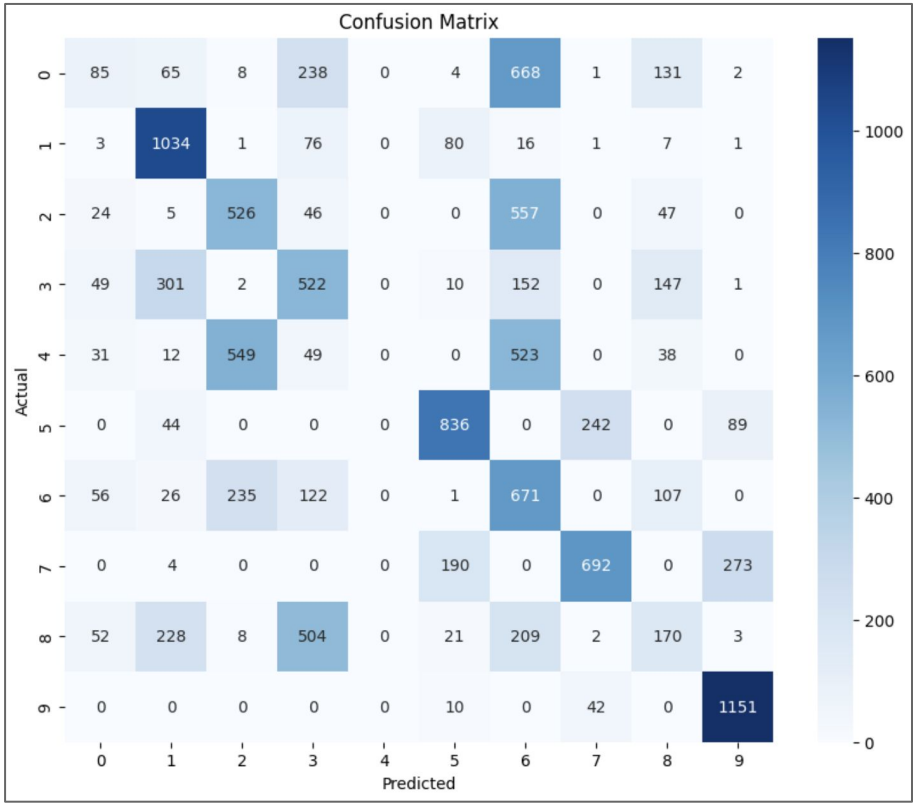
- Suitable optimizer: Adam

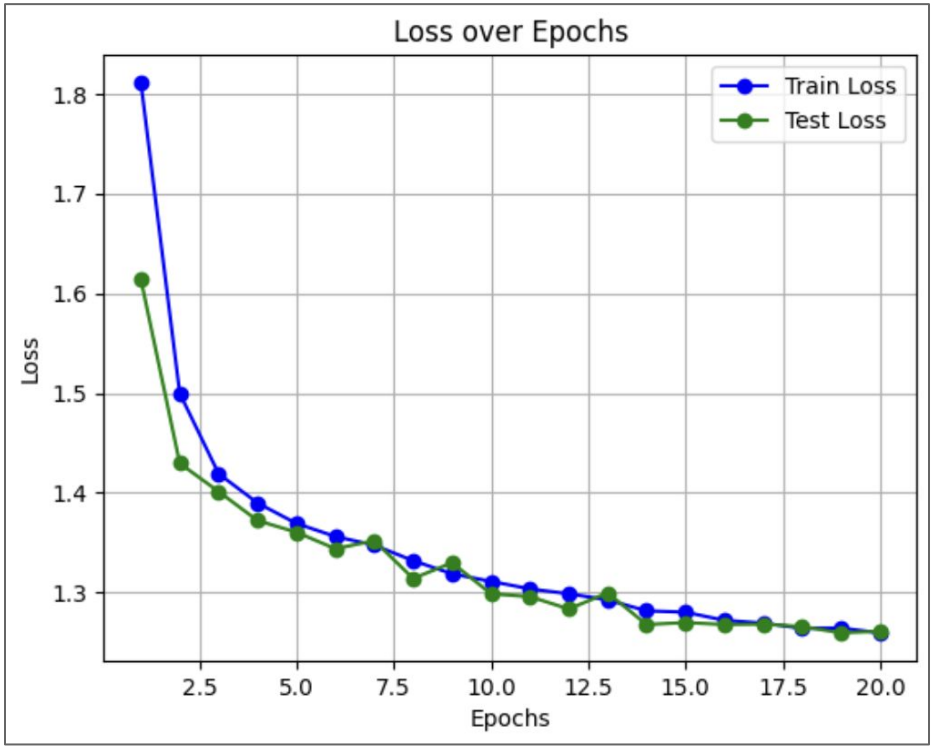
- ReLU performed better than sigmoid.

- Learning rate can be 0.01 or 0.001. As the epochs were fixed, we can't tell much about this.

Google Colab Notebook Link

03 Optimization and Evaluation





04 Final Testing, eval and reflection

Google Colab Notebook Link

Conclusion

Through this project, we built and evaluated neural networks to classify fashion items using the FashionMNIST dataset. Starting with a simple model and progressing to a more optimized architecture, we significantly improved accuracy by experimenting with activation functions, optimizers, and batch sizes. Our best model achieved over 76% test accuracy, highlighting the importance of thoughtful design and tuning. This project demonstrated how neural networks can learn meaningful visual representations and automate image classification effectively, offering a strong foundation for intelligent product tagging in e-commerce platforms.

Thank you!