IMAGE CLASSIFICATION

FASHION MNIST

THE UNIVERSITY OF CHICAGO

Helen Lykos

AGENDA

- PROBLEM STATEMENT
- ASSUMPTIONS/HYPOTHESIS
- DATA OVERVIEW
- EXPLORATORY DATA ANALYSIS
- FEATURE ENGINNERING & TRANSFORMATIONS
- PROPOSED APPROACH(MODEL) WITH CHECKS FOR OVERFITTING / UNDERFITTING
- MODEL WITH REGULARIZATION
- RESULTS (ACCURACY) AND LEARNINGS FROM THE METHODOLOGY
- FUTURE WORK/ RECOMMENDATIONS

PROBLEM STATEMENT

Fashion-MNIST is a dataset of Zalando's article images—consisting of a training set of 60,000 examples and a test set of 10,000 examples. Each example is a 28x28 grayscale image, associated with a label from 10 classes. Zalando intends Fashion-MNIST to serve as a direct drop-in replacement for the original MNIST dataset for benchmarking machine learning algorithms. It shares the same image size and structure of training and testing splits.

E- commerce companies have an extensive inventory of products available for online sale. It is very crucial for attractive customers to display numerous product images on websites, social media platforms and applications. We will classify images of different pieces of clothing into their respective classes which will assist businesses in automating the image categorization process.

ASSUPTIONS/ HYPOTHESIS

Assumptions

- The dataset is well-balanced, with an approximately equal number of samples for each class.
- The images in the dataset possess sufficient quality and resolution, enabling accurate classification.
- Dataset does not contain any substantial biases or artifacts that would negatively impact the model's performance

Hypothesis

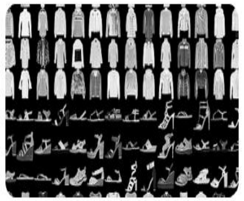
- The model can effectively learn and extract meaningful features from the fashion images.
- Training the model with a sufficient number of epochs will allow it to converge to an optimal solution.
- Regularization techniques, such as dropout or weight decay, can help prevent overfitting.
- The model's performance can be improved by fine-tuning hyperparameters, such as learning rate and batch size.

DATA OVERVIEW

Fashion MNIST

An MNIST-like dataset of 70,000 28×28 labeled fashion images

kaggle



https://www.kaggle.com/datasets/zalando-research/fashionmnist

fashion-mnist_train

60,000 images

fashion-mnist_test

10,000 images

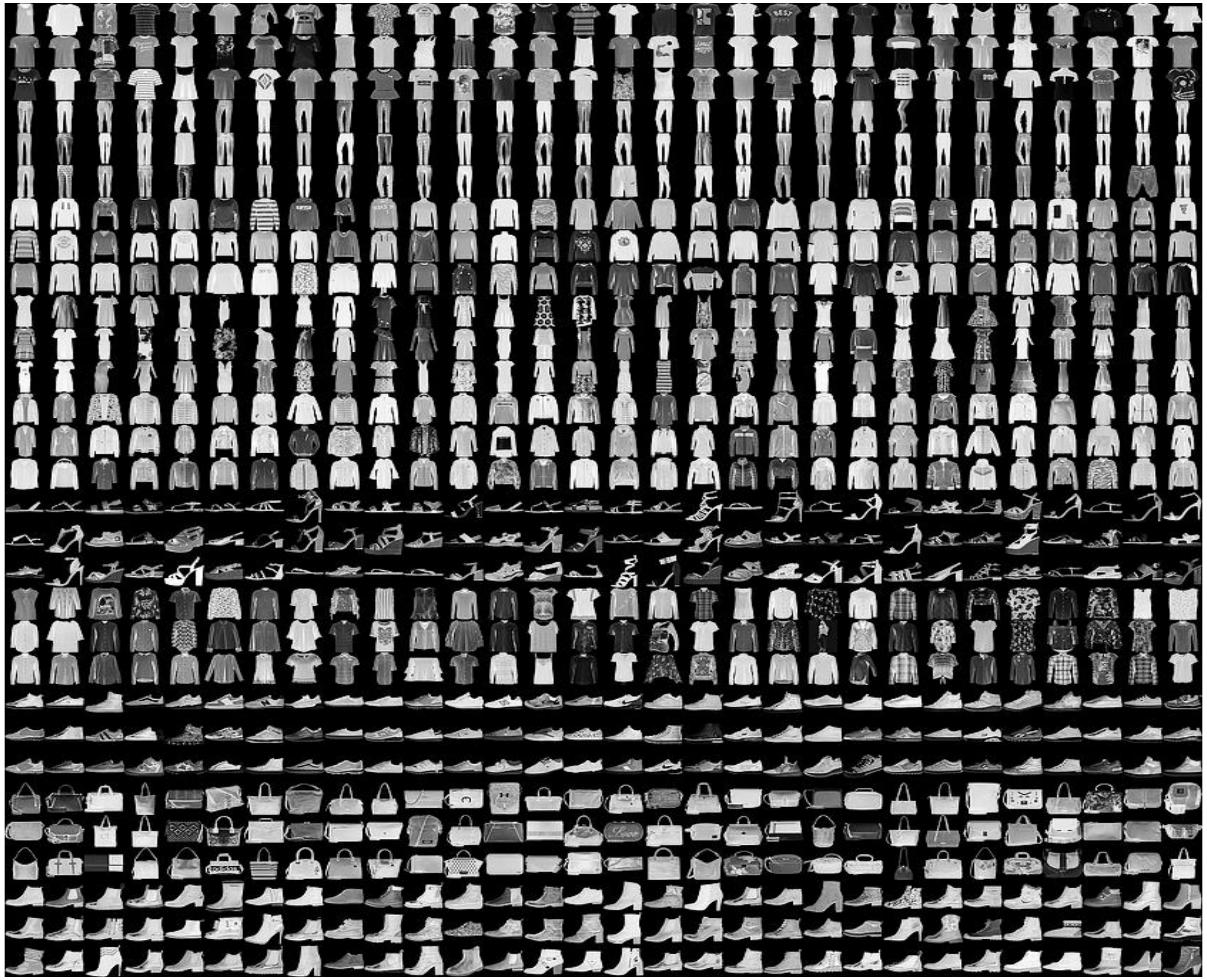
LABELS

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

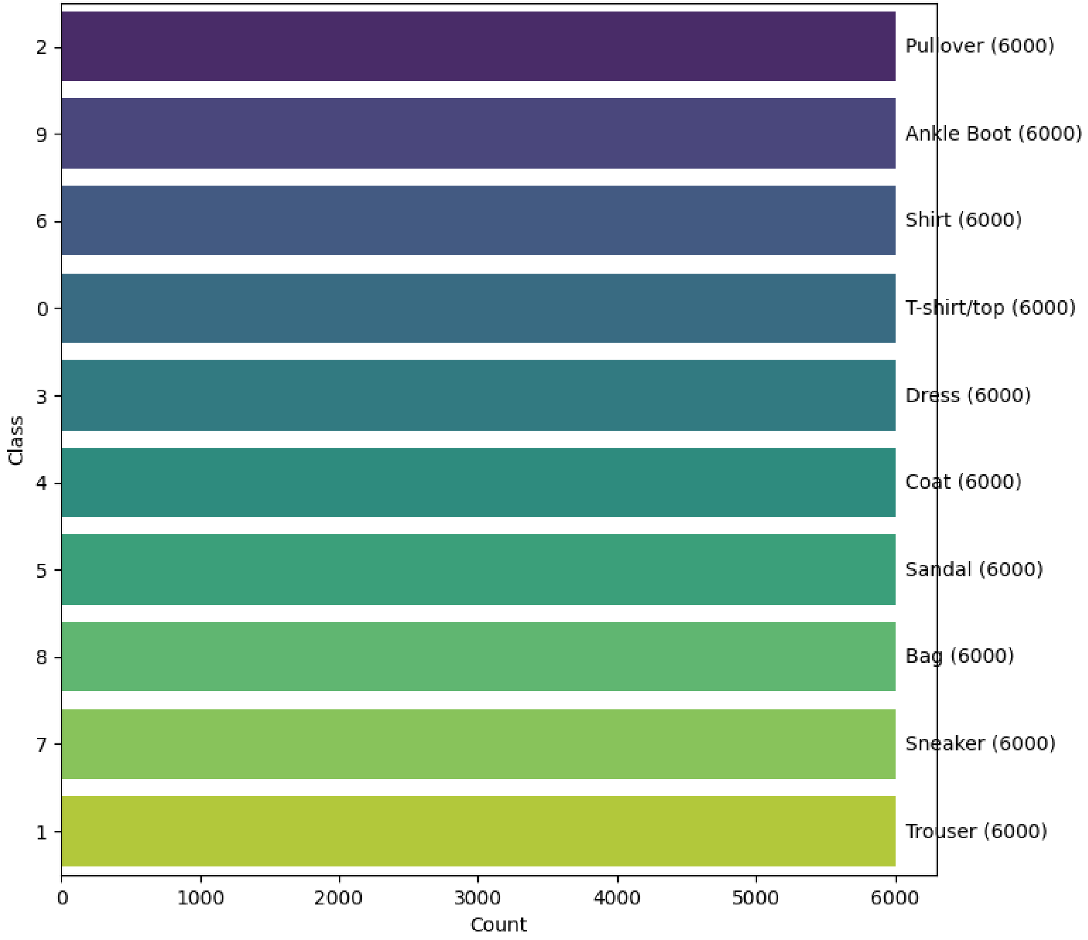
- Each row is a separate image
- Column 1 is the class label.
- Remaining columns are pixel numbers (784 total).
- Each value is the darkness of the pixel (1 to 255)

EXPLORATORY DATA ANALYSIS

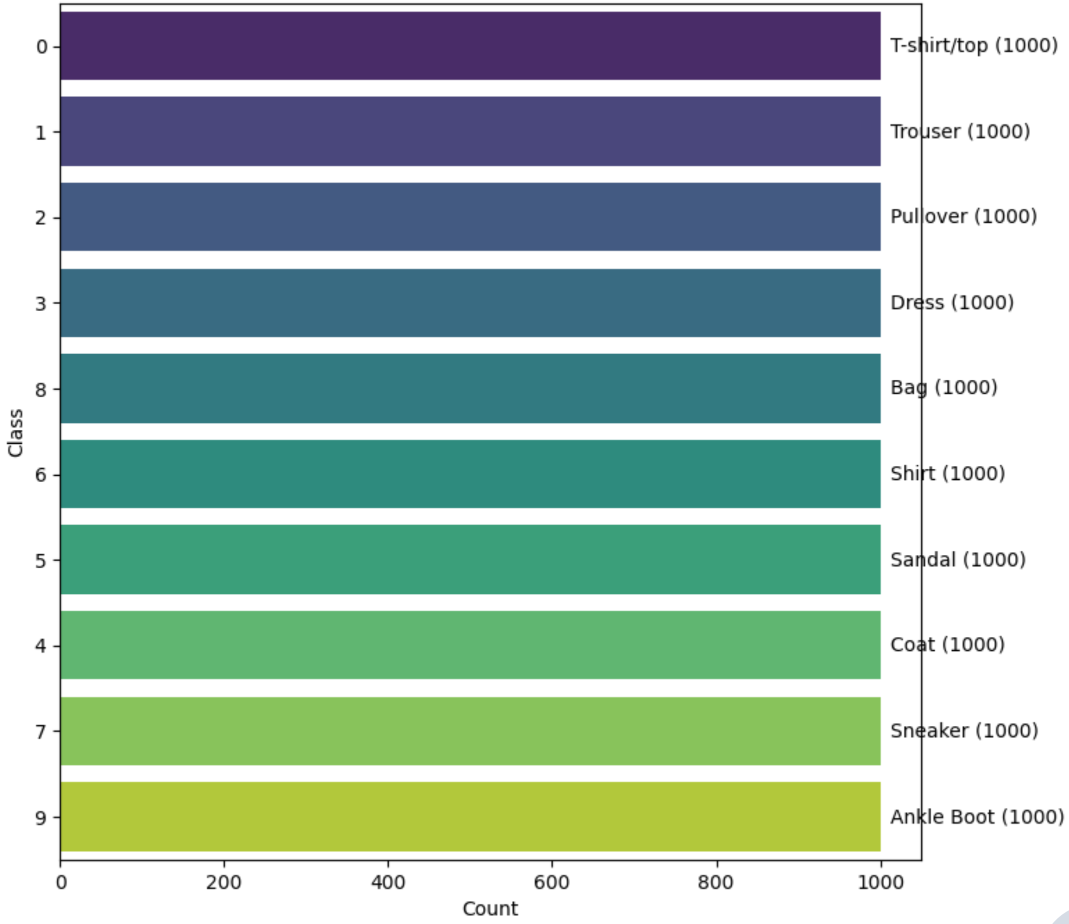
EVERY CLASS TAKES THREE ROWS



NUMBER OF LABELS FOR EACH CLASS IN TRAIN DATA



NUMBER OF LABELS FOR EACH CLASS IN TEST DATA



FEAUTURE ENGINEERING/ TRANSFORMATIONS

01

One-Hot Encoding:

raw.label

(categorical labels)

02

Reshaping:

The input image data is reshaped from a 2-dimensional array (60000,785) to a 4-dimensional array (60000,28,28,1).

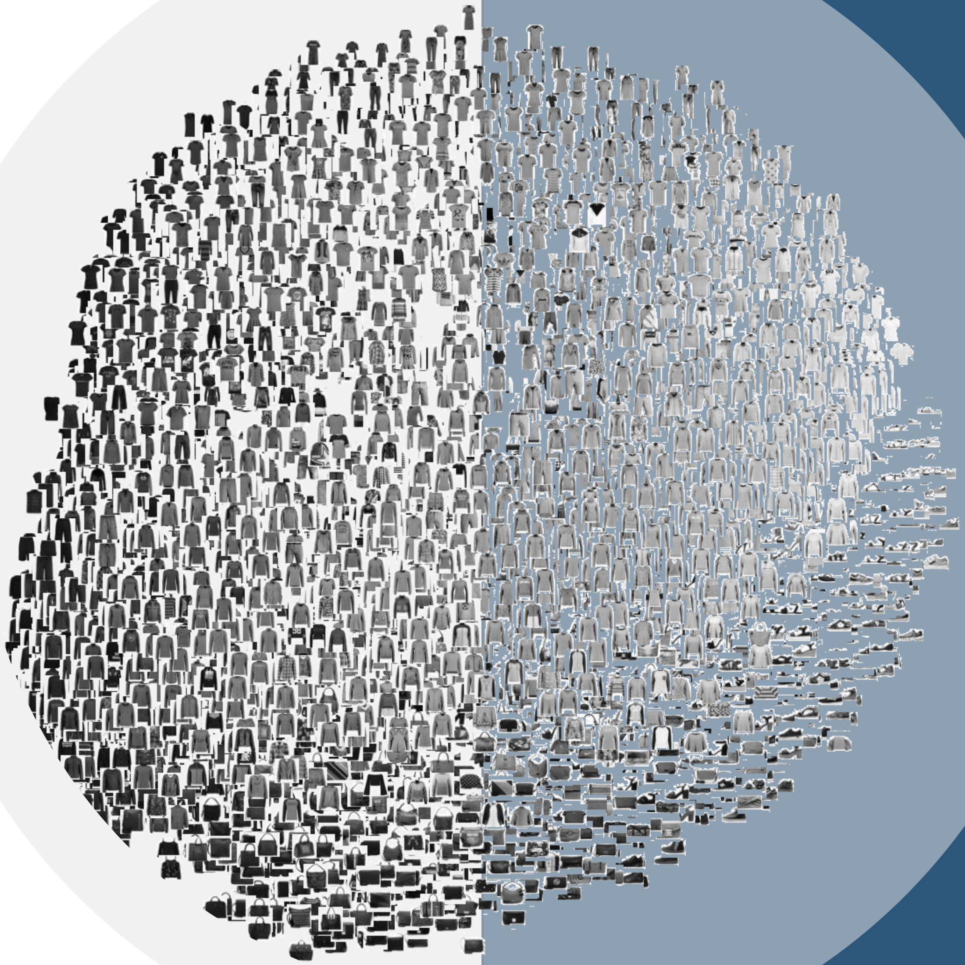
03

Normalize:

Normalize the pixel values between 0 and 1.

MODELLING


Ankle boot
T-shirt/top
T-shirt/top
Dress
T-shirt/top
Pullover
Sneaker
Pullover
Sandal
Sandal
T-shirt/top
Ankle boot
Sandal
Sandal
Sneaker
Ankle boot
Trouser
T-shirt/top
Shirt
Coat
Dress
Trouser
Coat
Bag
Coat

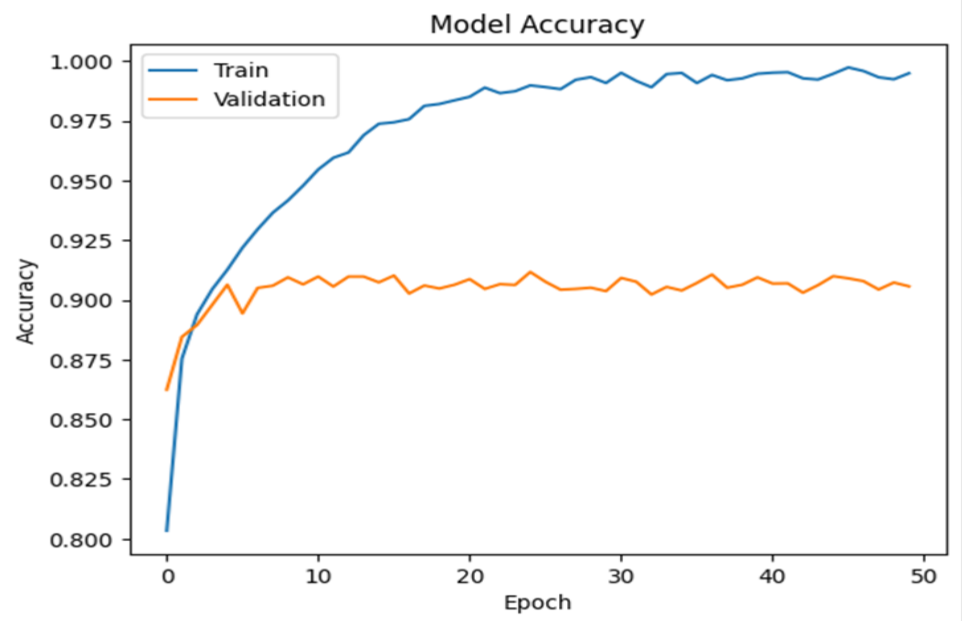


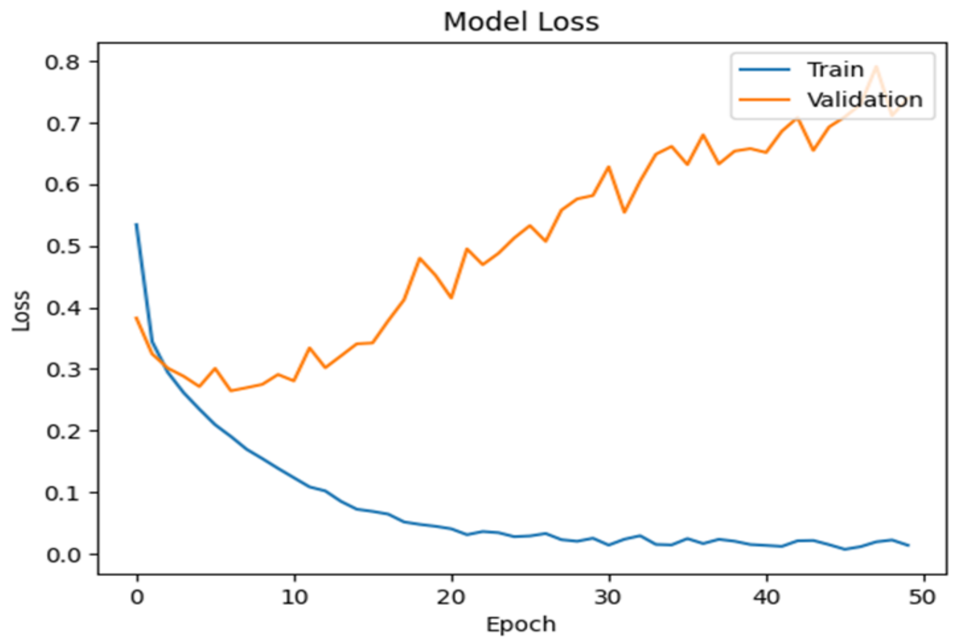
CONVOLUTIONAL NEURAL NETWORKS (CNN's)

The overall model architecture used is explained as follows:

- **Convolution Layers**: Extract features from the input image.
- **Pooling Layers**: Added after each convolutional layer to reduce spatial dimensions.
- **Fully Connected Layers**: Learn a function between the high-level features
  - Flatten: Transform multi-dimensional feature maps into a 1D vector.
  - Dense Layer (128 neurons, ReLU activation): Learn complex patterns.
  - Dense Layer (10 neurons, softmax activation): Produce class probabilities.

To ensure the model perform well on unseen data, we split dataset into Train, Validation and Test sets.





OVERFITTING / UNDERFITTING ?

- Training Accuracy :0.9948124885559082
- Validation Accuracy :0.9105833172798157
- Training Loss : 0.014748914167284966
- Validation Loss : 0.7869294285774231

- **Validation Accuracy :**

  No improvement after a few epochs

- **Validation Loss:**

  Increasing after few epochs

- Verify from the previous slide plots

MODEL OVERFITTING

REGULARIZATION

Added a total of four Dropout layers to the model.

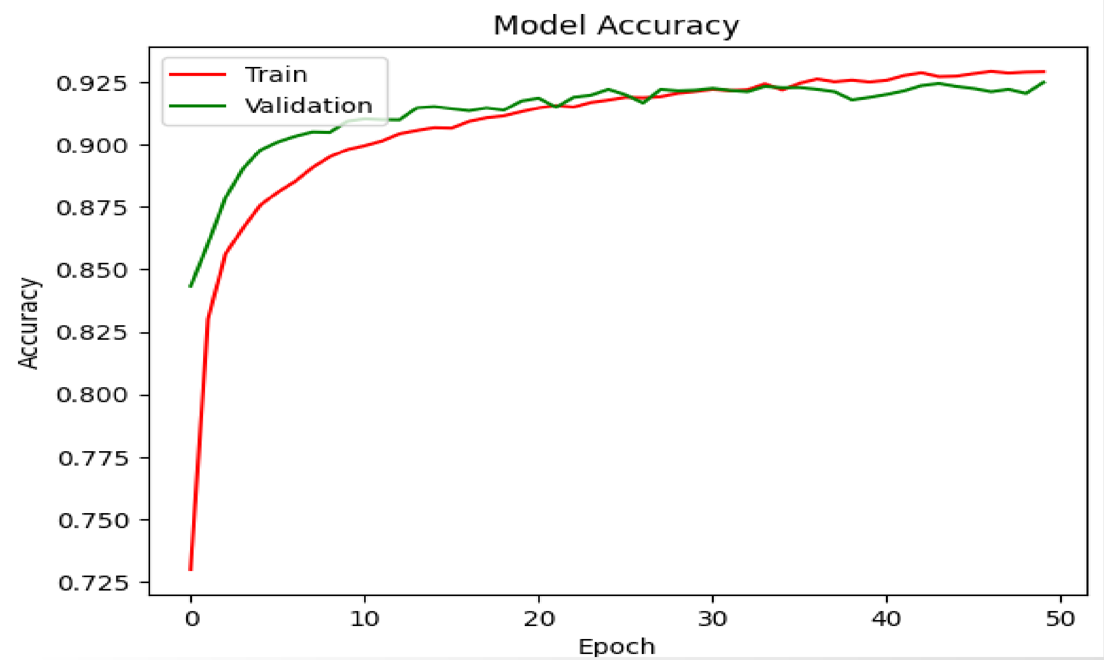
FINAL MODEL OVERVIEW:

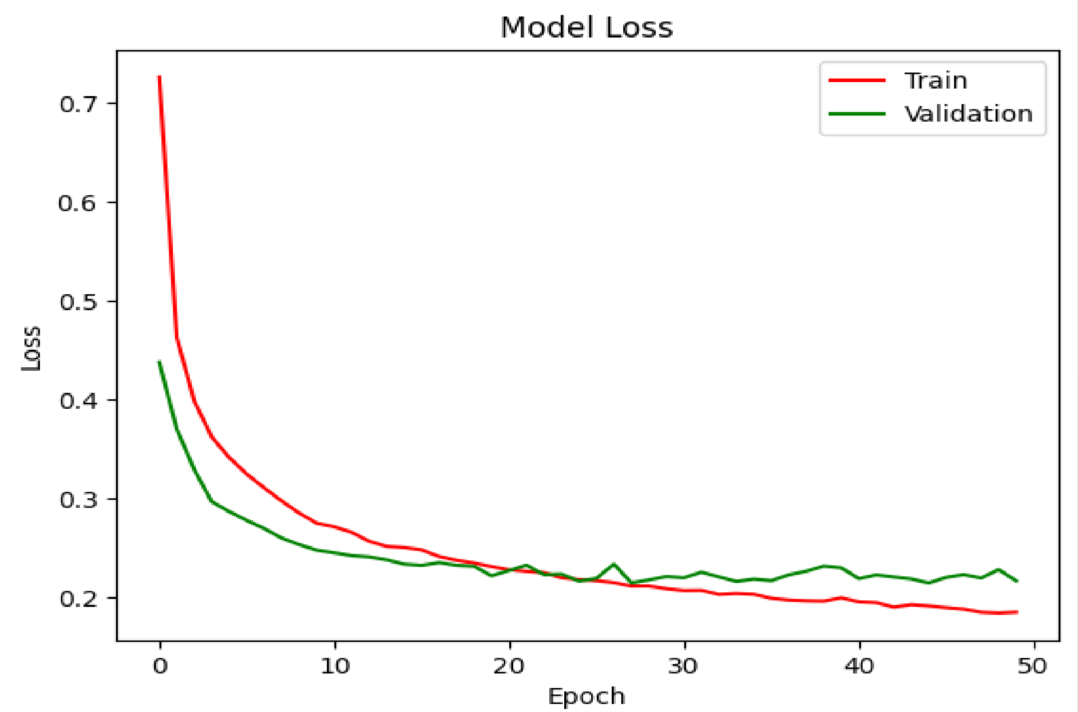
- ❑ Three Convolutional Layers COV2D with 32, 64, 128, uses 3x3 filters & applies ReLU activation function.
- ❑ Two Max Pooling Layers with 2x2 pooling
- ❑ Two Dropout layers with a dropout rate of 0.25.
- ❑ Flatten Layer to convert the previous output to 1D vector.
- ❑ Two Fully Connected Layers with 128 & 10 neurons, using the ReLU activation function
- ❑ Two Dropout layers with a dropout rate of 0.3 and 0.4.

Results:

Test Loss : 0.20

Accuracy: 93 %

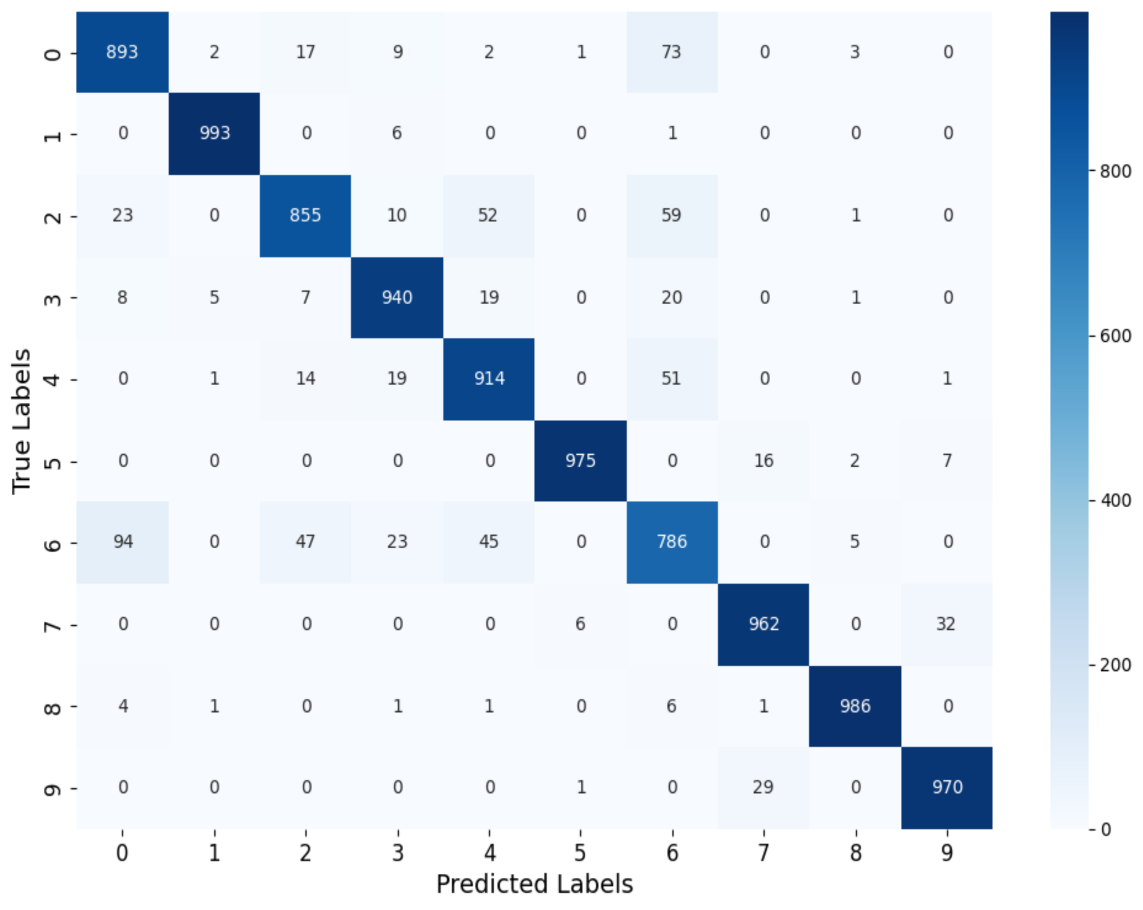




CLASSIFICATION REPORT

	precision	recall	f1-score	support
Class 0 (T-shirt/top) :	0.87	0.89	0.88	1000
Class 1 (Trouser) :	0.99	0.99	0.99	1000
Class 2 (Pullover) :	0.91	0.85	0.88	1000
Class 3 (Dress) :	0.93	0.94	0.94	1000
Class 4 (Coat) :	0.88	0.91	0.90	1000
Class 5 (Sandal) :	0.99	0.97	0.98	1000
Class 6 (Shirt) :	0.79	0.79	0.79	1000
Class 7 (Sneaker) :	0.95	0.96	0.96	1000
Class 8 (Bag) :	0.99	0.99	0.99	1000
Class 9 (Ankle Boot) :	0.96	0.97	0.97	1000
accuracy			0.93	10000
macro avg	0.93	0.93	0.93	10000
weighted avg	0.93	0.93	0.93	10000

CONFUSION MATRIX



	HIGHER	LOWER
ACCURACY	Classes: 1, 5 ,7, 8, 9	Class : 6
RECALL	Classes: 1,5,8	Classes: 2 , 6
F1- SCORE	Classes: 1,5,8	Classes: 0 ,2, 6

Most of the misclassifications are happening between the classes **Shirt, T-shirt/top, Pullover and Coat** which are majorly impacting the performance of the classifier.

EVALUATING THE FINAL MODEL:
Performance on Unseen Test

CORRECT CLASSIFIED IMAGES

INCORRECT CLASSIFIED IMAGES

RESULTS (ACCURACY) AND LEARNINGS FROM THE METHODOLOGY

RESULTS (ACCURACY)

- Our model reached a test accuracy of 92.7%
- Throughout the training process, the validation accuracy and loss were closely monitored to assess the performance of the model on unseen data.

LEARNINGS

- The provided dataset exhibited a well-balanced distribution with an equal number of samples for each class.
- The image quality and resolution within the dataset proved sufficient for effective classification.
- The model was designed with convolutional layers, pooling layers, and fully connected layers. To address the issue of overfitting, the model implemented regularization techniques, such as incorporating dropout layers. With the inclusion of these layers, the updated model achieved a test prediction accuracy of around 92.4%.
- Finally, the confusion matrix reveals that there are four classes (Shirt, T-shirt/top, Pullover, and Coat) that are not consistently classified correctly all the time.

Future Work - Recommendations

Collect more images

Improve the classifier's performance by incorporating more images from the categories of Shirt, T-shirt/top, Pullover, and Coat into the training data. By including a larger and more diverse set of samples, it will enable the classifier to learn additional features and improve its overall performance

Exploring advanced deep learning architectures

Consider exploring advanced deep learning architectures like ResNet, DenseNet, to enhance the classification performance. These architectures, characterized required more layers and complex structures to capture intricate features present in fashion images.

Data Augmentation

By implementing a range of data augmentation techniques, such as rotation, scaling, and cropping, we can enhance our training dataset. This approach helps improve the model's ability to generalize and reduces overfitting issues




THANK YOU FOR YOUR ATTENTION

GitHub Link:
https://github.com/elykos94/MSCA-31009