



# APPROACH





## STEP 1: IMPORTING LIBRARIES

- We are using Pytorch to build and train the model.
- Import the necessary libraries required for model creation, training and prediction



## STEP 2: download/preprocess the data

- Import the train and test set(fashion\_MNIST comes along with pytorch).
- Resize the set, Transform the set to totensor() form and normalize the dataset
- Split the train dataset to train and validation dataset in the ratio of (5:1)



## STEP 3: BUILD THE MODEL

- We have to build the models to conquer the MNIST dataset.
- Here we have created a resnet18 model and lenet model



## STEP 4: ADD LOSS AND OPTIMIZER

- Create loss function (cross entropy) to generate the loss while training the model
- Create a optimizer(Adam) to update the model after every successful epoch
- Create a scheduler to prevent overfitting in the validation phase



## STEP 5: ACCURACY FUNTION

- Create an accuracy function to find the accuracy of train set and val set while training the model





## STEP 6: TRAINING AND VALIDATION

- Train the model for 25 epochs and find the train accuracy and validation accuracy for every round
- Save the best model in .pt format



## STEP 7: USING INTEL OPTIMIZATION

- We have used the various optimizations provided by intel for intel devices to further optimize the code and increase its efficiency
- We have used open vino for lenet model and used ipex for resnet18 model
- Plot the time difference before and after the optimizations.





## STEP 8: PREDICT THE TEST SET

- With the optimized model you can now predict the all the labels of the images in the test set and compare the true and predict labels
- Then calculate the accuracy of the model



## STEP 9: VISUALIZE THE RESULTS

- Visualize the data using graphs provided by matplotlib
- Visualize the test train accuracy vs no of epochs and test train loss vs no of epochs to see the pattern
- Make a confusion matrix showing the model's predictions.