**Term Project Proposal**

**Team Members:** *Aisha Dantuluri, Mayank Gupta*

**Parallel Convolution Neural Networks (CNN)**

**Distributed learning with MPI**

**Dataset:**

Fashion-MNIST is a dataset of [Zalando](https://jobs.zalando.com/tech/)'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. Fashion-MNIST serve as a direct **drop-in replacement** for the original [MNIST dataset](http://yann.lecun.com/exdb/mnist/) for benchmarking machine learning algorithms. It shares the same image size and structure of training and testing splits.

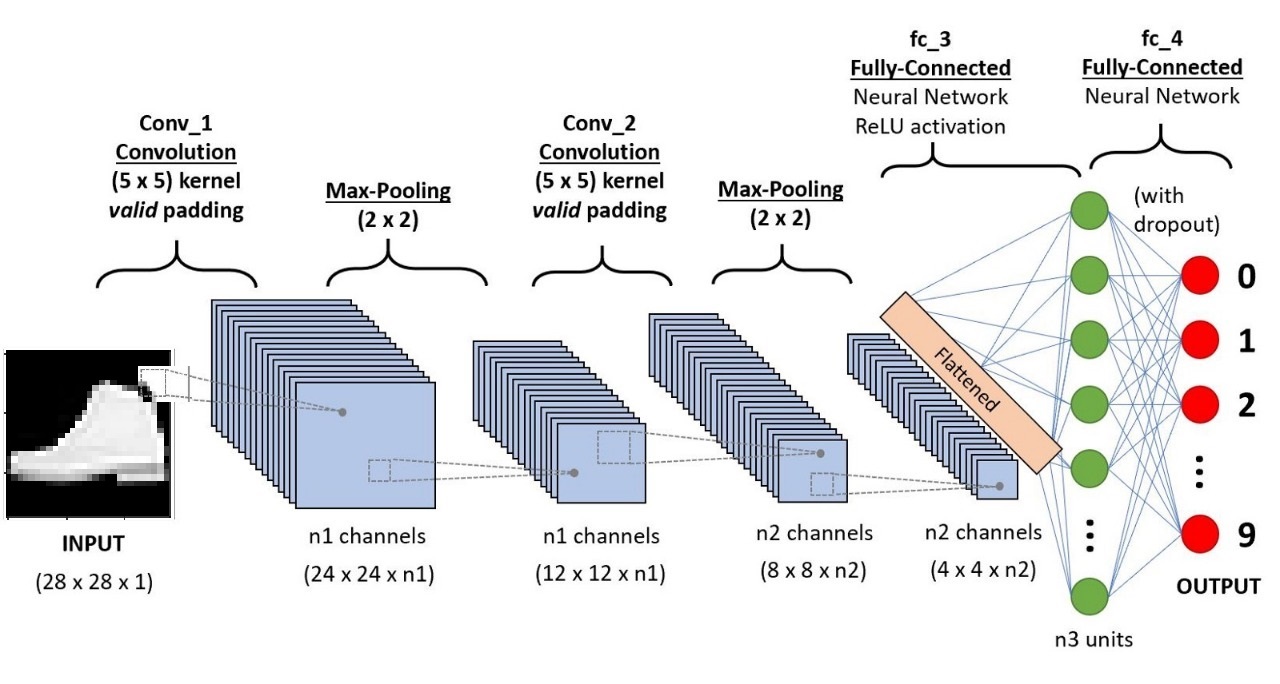
*Here is some images of how the dataset looks like:*



We are using Fashion-MNIST instead of MNIST because MNIST is too easy, overused, and cannot represent modern CV tasks. Below is the link of the data-set: <https://github.com/zalandoresearch/fashion-mnist>

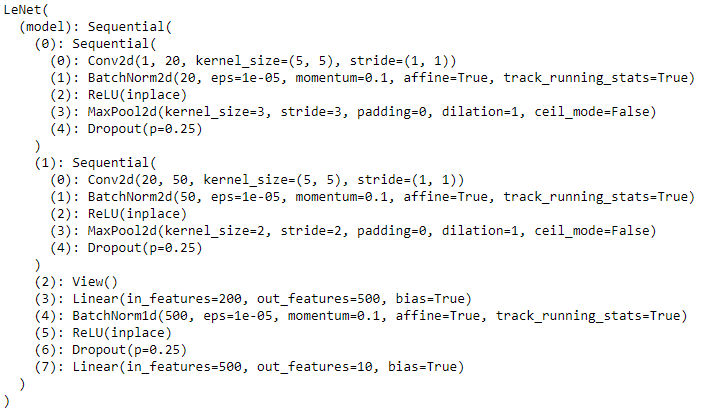
**Convolution Neural Networks:**

Convolutional Neural Networks (CNN) are a class of neural networks which involve using k\*k matrix as a filter and perform a sliding window based multiplication on the input feature map. It can have one or more convolution layers (often with a subsampling step) and then followed by one or more fully connected layers. The advantage of CNN over classical Neural Networks is that they are easier to train and with exponentially reduced parameters. The CNN has become one of the most important tool that present day Artificial Intelligence problems involving images employ. Below figure summarizes all the computations that are done in the convolutional neural network including the pooling as well as linear layers.



**Architecture:**

Our end goal is to increase the test accuracy on a dataset using Message Passing Interface between CNNs running on a dataset. So the initial step is to define the good architecture of a CNN which gives descent results without using MPI and then we will try to use MPI to increase the accuracy of the combined architecture. We tried and tested three different architecture and the best results were given by the below architecture when run sequentially. Below is the architecture of our model

****

Where Conv2d means 2 dimensional Convolution layer, BatchNorm2d means Batch Normalization which prevents gradient vanishing problem. Here ReLU is an activation function which we chose because it makes the algorithm to work faster. We do MaxPool2d to increases the receptive field which makes the model more robust and invariant to small movements in input. Dropouts with some probability makes the model computationally cheaper and prevents dependency on other nodes. The architecture consists of two convolution layers followed by fully connected network with one hidden layer with an output layer of 10 units followed by a softmax function. Each of the 10 units represents each class of Fashion MNIST dataset. We will chose the class corresponding to the highest response after the softmax function.

**After running the above architecture for 5 epochs we get an accuracy of 85.53% on Fashion MNIST test dataset with a final loss of 0.35.**