**Brief Report for LAB-1**

For Part 1, We had to develop a Wide and Deep Recommender System for the Book review dataset using PyTorch. The process involved data preparation, feature engineering with wide and deep components, and the creation of a model architecture combining logistic regression and neural networks. The model was trained and evaluated on an 80/20 train-test split, with hyperparameter tuning for optimization. Results demonstrated the model's accuracy and performance metrics.

For Part2 , we had to train the neural network by exploring different parameters. We undertook a meticulous exploration of architectural and hyperparameter choices like different number of layers and neurons. Activation functions played a pivotal role in shaping the network's non-linearity. Through experimentation, we evaluated ReLU, Sigmoid, and elu. ReLU, the widely used activation function, introduces non-linearity by replacing negative values with zero. Sigmoid is apt for binary classification tasks, producing outputs between 0 and 1, while elu activation function produces output values in the entire real number range, ranging from negative infinity to positive infinity.

We also explored the impact of dropout layers. Dropout, acting as a regularization technique, randomly deactivates neurons during training, enhancing generalization. We trained the network with and without dropout layer=0.5. Also, we used batch normalization and L2 as well. These techniques help in solving overfitting. Different optimizers were applied like Adam, SGD.

These models’ performance was illustrated through a graph, plotting F1 score versus epochs for both training and validation sets. This graph provided insights into the model's learning trajectory, guiding further adjustments to hyperparameters.

For Part3, Object detection ( for ships) using the 'train\_df.csv' dataset, with pixel information indicating ship presence in images. For this we have done the Dataset exploration involved visual analysis, plotting random training set images based on ship counts. Ship locations were highlighted using pixel information from 'train\_df.csv,' offering a spatial understanding of ship distribution. Random test set images were also plotted for feature exploration. Then the object detection using custom model, where we have used U-net architecture ( where we have made the slight changes to the U-net layers) also added the average layers with the architecture to predict the object.