# Final Project: Deep learning

The goal of this project is to :

(1):Compare the performance of different CNN Architectures specifically Alexnet,Resnet, VGGnet, Lenet-5 and Senet on MNIST, FMNIST and CIFAR-10 datasets. The comparsion will be based on metrics such as Loss curves, Accuracy, Recall, Precision and F1 score.

(2): Implementing sequence-to-sequence models with attention mechanism. Models will be trained on a synthetic dataset where the target sequence is the reverse of source sequence.

(3):Create a multifunctional tool that allows users to select and utilize different pre-trained models from Hugging Face for various taks. The tool will support Text summarization, Next word prediction, Story completion, Question & Answering, Chatbot, Image generation and Sentiment analysis. The front end will provide a user-friendly interface to select the task and input the required text or image for processing.

Key Takeaways from this project:

\* Utilizing popular Deep learning frameworks like pytorch.

\* Understanding the CNN Architectures, their Implementaion on different datasets, analyse their performance on various metrices and do comparsion.

\* Utilizing Hugging face pretrained models like GPT2,BART and others.

\* Integrating multiple machine learning models into a single application and user friendly multifunctional tool using Streamlit.

\* Knowledge on GPU and understanding the difference between CPU and GPU usage.

\* Understanding and Utilizing the attention mechanism in Neural networks.

Detailed Information of the project:

1: Comparsion of Different CNN Architetures on Datasets and evaluating their performance based on various metrics. Implemented the necessary modules and libraries as and when required.

Process:

\*Load and preprocess the datasets(Mnist,Fmnist,Cifar-10)

\*Implement the CNN Architetures, train each model on each Dataset and record the loss and accuracy metircs.

\*Plot the Loss curve and other performance metrics for comparsion.

\*Analyse the results to understand the architectures and datasets on the performance of a model.

Note:

Implementation of CNN architectures is done on Google colab platform as they need GPU.

Keeping in view of GPU Limitation, the loss and other performance metrices have been pickled and same files have been added as a part of this project.

2.Implementation of Sequence-to-sequence model with attention mechanism.

Process:

\*Generate a synthetic dataset where each source sequence is random sequence of integers, and each target sequence is the reverse of source sequence.

\*Implement seq-seq model with attention mechanism.

\*Train the model on synthetic dataset,and evaluaute its performance based on loss and accuracy.

\*Plot the loss curve and do the comparititve analysis.

3.Create a Multifunctional tool to select different tasks of Natural Language Processing and evaluate their performance for a given input.

Process:

\*Set up the Environment and install necessary libraries, including Hugging Face Transformers.

\*Implement a user-friendly front end for task selection and input.

\*Load and integrate pretrained models from Hugging face for tasks such as Text Summarization, Chatbot, Nextwordprediction, Story completion, Question&Answering and Sentiment Analysis.

\*Implement the back end logic to process user inputs and generate the output using selected models.

\* Test the application with various inputs and refine the back end logic and user interface.

Note: The streamlit app is run on the colab platform using local tunnel and tool is generated on a browser to choose various NLP tasks and evaluate the performance of the task.