

# PLACEMENT PROJECTS

Let's see my projects. All projects available [here](#) .

We have two sections:

- Artificial Intelligence Projects
- Computer Science and Engineering Projects

## *Artificial Intelligence*

### 1. [Object Detection with Voice feedback](#)

- implemented yolov3 algorithm and used pretrained model weights to detect multiple objects in one image.
- used the google text to speech API to generate audio file describing the objects in the image.
- The model identified objects with a confidence of about 99%.

### 2. [Chatbot](#)

- Implemented a retrieval-based chat-bot (named Haritha).
- This Chatbot is specifically designed to mimic the chatbots of Food delivery platforms. It uses Natural Language Processing and Deep learning.
- The chatbot achieved an accuracy of 98% in generating appropriate responses.

### 3. [MNIST Image Classifier](#)

- Implemented each of the components of a convolutional neural network (CNN) from scratch, including backpropagation.
- Using this network built an image classifier trained on MNIST dataset for digit classification and attained accuracy of 88% .
- The results were compared with the CNN built using Pytorch Library.

### 4. [Steganalysis](#)

- Implemented deep learning model for Alaska2 dataset's JPEG Image Steganalysis to detect JUNIWARD-encoded messages in JPEG images.

- Using less than 20% of the data for training got an accuracy of 0.6 for trained model.

## **5. Hackathons**

- Achieved Score: 0.87337 of Kaggle challenge named "Is the driver at fault?" . The machine learning model used is a Stacking Ensemble Model, combining three base models: Random Forest, LightGBM, and XGBoost. These base models' predictions are integrated using a meta-model (LightGBM) to produce the final prediction, often enhancing predictive performance. The ML model is used to assess whether the driver involved in the accident is at fault or not based on 42 features.
- Achieved Score: 0.99828 of Kaggle challenge named "Digit Recognizer" . Built a Convolutional Neural Network (CNN) model for digit recognition, to take an image of a handwritten single digit, and determine what that digit is.

## **6. Analyzing Laptop Specifications**

- A data-driven analysis of laptop specifications was conducted, uncovering valuable insights and market trends that facilitate informed decision-making for laptop purchases and market analysis, applying statistical methods and data visualization, showcasing advanced data analytics skills.

## **7. Binary Classifier**

- Utilized a Stacking Ensemble Model, to combine three base models: Random Forest, LightGBM, and XGBoost.
- Integrated predictions from base models using a meta-model (LightGBM) to develop a binary classifier model to determine whether the driver was at fault in the accident. The model accuracy was 87 percent.

# Computer Science and Engineering

## 1. [Question paper generator](#)

- Implemented automated system for generating unique exam papers, optimizing efficiency and customization through question bank and parameter substitution techniques.
- Streamlined processes of question selection, paper creation, and customization, resulting in improved productivity

## 2. [Development and Management of Database](#)

- Designed a flexible and efficient relational database from scratch for a scientific research paper inquiry website with the referential integrity maintained.
- Developed informative ER diagrams, efficient relational table schema and retrieved information with complex queries.

## 3. [DSA Projects](#)

used Data structures and Algorithms to implement many real-world applications:

- **RISC-V Disassembler** - Implemented a disassembler converting a given RISC-V machine code to equivalent assembly syntax.
- **Sorting Visualizer** - The Program demonstrates various sorting algorithms on a randomly generated array. It visualizes the sorting process step by step.
- **File Zipper** - Implements Huffman coding for file compression and decompression. It utilizes priority queues, binary trees, and variable-length prefix codes for efficient data encoding and decoding.
- **Sudoku solver** - Implemented backtracking algorithm to solve Sudoku puzzles.

## 4. [OS Multithreading](#)

- Utilized multithreading to parallelize the task of finding perfect numbers within a specified range, improving efficiency, reducing processing time and leveraging concurrent execution, demonstrating efficient CPU utilization.