Deep Paresh Dodhiwala

213-561-7036 | <u>Email</u> | <u>LinkedIn</u> | <u>Github</u> | <u>Portfolio</u>

EDUCATION

University of Southern California (USC)
Los Angeles, CA

Master of Science in Computer Science, CGPA: 3.75/4 — *Aug. 2022 – May 2024*

Coursework: Analysis of Algorithms, Machine Learning, Deep Learning and its Applications, Database Systems.

Sardar Patel Institute of Technology
Mumbai, India

Bachelor of Technology in Information Technology, CGPA: 9.5 / 10 — *Aug. 2018 – May 2022*

TECHNICAL SKILLS

Languages: Python, Java, C, C++, JavaScript, C#, Bash, HTML, CSS, PHP, TypeScript.
Frameworks: Git, Spring Boot, Kubernetes, Jira, Angular, PyTorch, React, Node.js, AWS, Jenkins, REST API, Docker.
Databases: MySQL, Oracle, PostgreSQL, MongoDB, Firebase, phpMyAdmin, Amazon Dynamo DB.
Libraries: Numpy, Pandas, TensorFlow, Plotly, Matplotlib.

EXPERIENCE

Algorithm Developer
Sept. 2023 – Present

Autonomous Networks Research Group (ANRG) — *Los Angeles, CA*

- Researched and Implemented task graph scheduling algorithms **(Min-Min and Sufferage)** for distributed computing environment within the SAGA (Scheduling Algorithms Gathered) project.
- Improved task execution efficiency and load balancing by incorporating financial constraints using **E-HEFT**.

Data Science Intern
Oct. 2021 – Nov. 2021

Globalshala, in collaboration with Saint Louis University — *Mumbai, India*

- Analyzed statistical information of the firm's ad campaigns based on **Facebook Data** - Cost Per Click, Click-Through Rate, Cost Per Result and Average Unique Clicks.
- Extracted valuable insights from these data measures and proposed to optimize campaign performance by discontinuing **20%** of the least profitable campaigns using **Tableau** and **Scatterpolar**.

Software Engineer Intern
May 2021 – July 2021

JPMorgan Chase — *Mumbai, India*

- Worked on **Asset Record Software**- computes and stores net holdings of a legal entity.
- Enhanced Asset Record's End-Of-Day functionality by analyzing the current design and suggested modifications to include SubLine-Of-Business for a particular legal entity.
- Suggested critical code changes in **Spring Boot** and database structure modifications to **Apache Cassandra** to achieve the desired goals.

PROJECTS

Text-Based Active Reinforcement Learning | *Python, PyTorch, Git*
- Developed a Reinforcement Learning (RL) agent using **Stable Baselines3** to interpret and execute text-based instructions for navigating in a Mario Game within the OpenAI Gym Environment.
- Optimized and fine-tuned **Llama 3 8B** using LoRA to generate various reward functions of the RL model.
- Improved the completion time by **5x** and final score by **1.4x** than the baseline reward function.

Intelligent Call Prioritization Using Speech Emotion Recognition | *Python, Anvil, Git*
- Built a website using **Anvil** that uses a short voice note of waiting callers and prioritizes calls based on emotion predicted through speech and transcribed text.
- Deployed BERT model for **textual analysis** and Random Forest Classifier for the **Speech Emotion Recognition** for the best results.
- The combined model achieved an overall accuracy of **92%** and with an average waiting time of **30 seconds**, a **1.5x** improvement on the standard scheduling algorithms.

Diabetes Prediction System | *Python, Flask, HTML, CSS*
- Developed a web-based application to determine probability of having diabetes using Artificial Neural Network.
- Conducted extensive **exploratory data analysis** and **data mining** to build a robust prediction model.
- Constructed the Neural Network using TensorFlow which obtained a training accuracy of **93.48%** and testing accuracy of **87%** an improvement on the existing prediction algorithms.

Eggshub | *Python, Plotly, MongoDB, Angular, Node.js*
- Led a group of 6 to develop a specialized website focused on egg-based statistics
- Created a website to display the data metrics through interactive chloropleth world maps that could effectively display export and import statistics related to eggs in an effective manner using **Plotly**.