

Sai siva reddy Maddula

✉ maddulasaisivareddy@gmail.com ☎ 3614881962 📍 Tampa, FL 🔗 <https://o1mssr.online>

🌐 [linkedin.com/in/sai-siva-reddy-m-964504224](https://www.linkedin.com/in/sai-siva-reddy-m-964504224) 🏠 github.com/Mpr9640

Skills

Machine Learning, Python, C++, HTML, CSS, JavaScript, React.js, Next.js, Numpy, Linear algebra, 3D Modeling, DSA, Computer Networks, Pandas, DFA, NFA,Regex, Colab, Blender, Docker, PostgreSQL, TensorFlow, Teamwork, Github pages, UI, Fast API, Chrome extension and Google forms.

Projects

o1 Application Autofill Extension,Phase 1(Part of Ongoing Project)

Tech Stack: HTML, CSS, JavaScript, JSX, React, FastAPI, Python, Alembic, PostgreSQL, Chrome Extension

- Developed a full-stack application with user authentication (account creation, email confirmation, login) and data storage using FastAPI and PostgreSQL.
- Built a Chrome Extension that detects job application forms, highlights a popup icon, and auto-fills fields by intelligently mapping backend user data to input elements.
- Implemented secure data fetching using tokens, with robust form parsing across varied layouts and input types.
- Managed database schemas and migrations using Alembic.

Personal Portfolio Website, My Description

Tech Stack: HTML, CSS, JavaScript

- Designed and developed a dynamic, responsive personal portfolio to showcase skills, projects, and accomplishments.
- Implemented a modern UI with HTML for structure, CSS for styling, and JavaScript for interactivity, including animations, form validation, and dynamic content rendering.
- Ensured cross-browser compatibility and optimized mobile responsiveness for consistent user experience across devices.
- Demonstrated strong front-end development capabilities and provided an engaging platform for potential employers and clients to explore professional work.

Fairness Evaluation using TensorFlow Fairness Indicators – ACS Dataset

Tech Stack: TensorFlow, TensorFlow Model Analysis (TFMA), Fairness Indicators, Python, Pandas

- Conducted a fairness audit on a machine learning model trained on the **ACS (American Community Survey)** dataset to detect and mitigate **bias across demographic groups** (e.g., gender, race, income level).
- Used **TensorFlow Fairness Indicators** to evaluate model performance metrics like **accuracy**, **false positive rate**, and **equal opportunity** across sensitive attributes.
- Integrated **TFMA (TensorFlow Model Analysis)** to compute and visualize **per-slice metrics** for better interpretability and transparency.
- Identified disparities in model performance among subgroups and adjusted data preprocessing, threshold tuning, and sample reweighting to reduce bias.
- Presented results through **interactive fairness dashboards**, highlighting areas where the model failed to maintain equitable performance.
- Demonstrated the use of **responsible AI practices** to ensure fairness and accountability in predictive models built on real-world census data.

Transfer Learning & Multiclass Classification – Fashion MNIST Dataset

Tech Stack: TensorFlow, Keras, NumPy, Matplotlib, Scikit-learn

- Implemented **transfer learning** by training a base convolutional neural network (CNN) on 5 classes from the Fashion MNIST dataset and reusing learned features for a new target task involving the remaining 5 classes.
- Applied **data augmentation** techniques (e.g., horizontal flip, width/height shift) using ImageDataGenerator to increase training data diversity and reduce overfitting.
- Selected a **limited dataset** for the target task (10 samples per class) to simulate few-shot learning and validated model generalization on unseen classes.
- **Froze initial CNN layers** and fine-tuned the final dense layers for the target classification task, showcasing effective feature transfer.
- Measured and compared **training/validation accuracy and loss** before and after augmentation to highlight its impact on performance.
- Used early stopping and visualization techniques (matplotlib) to monitor training progress and prevent overfitting.
- Achieved improved classification accuracy on both base and target tasks, demonstrating strong feature reuse and model generalization capability.

Data Augmentation & Performance Evaluation – MNIST Dataset

Tech Stack: TensorFlow, NumPy, Matplotlib, Scikit-learn

- Applied **data augmentation** techniques to the MNIST handwritten digit dataset to increase dataset diversity and improve model robustness.
- Augmentation strategies included **random rotation**, **shifts**, **zoom**, and **flipping**, implemented using ImageDataGenerator from TensorFlow.
- Trained a baseline model without augmentation and recorded validation loss and accuracy metrics.
- Retrained the model on the augmented dataset and **compared performance**, observing a **decrease in validation loss** and improvement in generalization.
- Visualized and analyzed the impact of data augmentation on model training and validation using matplotlib.

- Demonstrated the effectiveness of augmentation in **reducing overfitting** and enhancing model performance on unseen data.

Multiclass Classification – Fashion MNIST Dataset

Tech Stack: TensorFlow, Scikit-learn, NumPy, Matplotlib

- Developed a multiclass image classification model using **TensorFlow** to classify clothing items from the Fashion MNIST dataset into 10 categories.
- Preprocessed and normalized image data using NumPy, and split the dataset into training and test sets using train_test_split from Scikit-learn.
- Designed and trained a neural network with multiple layers, optimizing performance using appropriate activation functions and loss metrics for multiclass classification.
- Visualized training progress, accuracy, and sample predictions using matplotlib.pyplot to evaluate model effectiveness.
- Achieved strong classification performance and improved accuracy by fine-tuning model architecture and hyperparameters.

ML Linear Regression – Chicago Taxi Dataset

Tech Stack: Colab,Python, NumPy, Pandas, Keras, Plotly, Seaborn

- Built a supervised learning model using **linear regression** to predict outcomes from the Chicago Taxi dataset.
- Used pandas and numpy for data preprocessing, feature selection, and identifying high-impact columns influencing target values.
- Refined the dataset by removing irrelevant features and trained the model using **Keras** with adjusted hyperparameters such as epochs, learning rate, and optimizers to improve accuracy.
- Visualized data trends, model training, and predictions using plotly.express , plotly.graph_objects, and seaborn to provide interactive and static graphical insights.
- Performed model evaluation by predicting outcomes on new data and calculating overall accuracy to assess performance.

Fire Flame Detector, Detects the fire and makes noise.

Tech Stack: Arduino Uno, IR Flame Sensor, MQ-2/MQ-135 Smoke Sensor, Buzzer, LED, Resistors, Breadboard, Jumper Wires

- Built a fire detection system using Arduino Uno that identifies flame and smoke using IR Flame Sensor and MQ-2/MQ-135 Smoke Sensor.
- Triggered immediate audio-visual alerts (buzzer sound and LED indication) when fire or smoke was detected.
- Designed and assembled the circuit on a breadboard with appropriate resistors and jumper wires, powered via USB/adaptor.
- Demonstrated real-time detection capabilities for fire safety applications using embedded hardware and sensor integration.

Professional Experience

Datics Inc, Software Engineer Internship 01/2024 – 04/2024

- Execute full software development life cycle (SDLC).
- Develop flowcharts, layouts and documentation to identify requirements and solutions.
- Write well-designed, testable code.
- Produce specifications and determine operational feasibility.
- Integrate software components into a fully functional software system.
- Develop software verification plans and quality assurance procedures.
- Document and maintain software functionality.
- Troubleshoot, debug and upgrade existing systems.
- Deploy programs and evaluate user feedback.

Education

Auburn University at Montgomery, Master of Science in Computer Science 08/2022 – 05/2024
Montgomery, United States of America

- Graduated with a CGPA of 3.83.
- Developed and designed websites using HTML, CSS, JavaScript, and React focusing on responsive and user-friendly interfaces.
- Worked on neural network projects within machine learning, applying advanced algorithms to solve complex problems.
- Ran robust and efficient C++, and Python (OOPS) and Attended lab sessions to gain practical knowledge for emphasizing best practices and maintainability.
- Gained Knowledge in data structures and algorithms to efficiently organize and manage a large amount of data.
- Implemented machine learning code in Google Co-lab, enhancing data analysis and model training efficiency.
- Possess strong SQL skills for querying and managing databases.
- Completed projects on 3D Modeling Using the Blender Application.
- Experienced with NFA, DFA, and Regular Expressions.
- Developed codes in Python code for supervised Learning and Unsupervised Learning Projects.

Vasireddy Venkatadri Institute of Technology, 06/2018 – 06/2022 | Guntur, India
Bachelor of Technology in Electrical and Electronics Engineering

- Graduated with a CGPA of 3.41
- Mathematical Skills: Acquired proficiency in derivatives, integrations, calculus, linear algebra, statistics, and discrete mathematics.
- Physics and Mechanics: Worked extensively on thermodynamics, electromagnetism, quantum mechanics, optics, and wave mechanics, gaining practical knowledge through laboratory experiments.
- Chemistry Fundamentals: Studied atomic structure, the periodic table, chemical bonding, stoichiometry, chemical reactions, states of matter, gas laws, and molecular geometry, complemented by hands-on experience in laboratory work.
- Mathematical models used in designing digital networks.
- Advanced Topics: Familiar with neural networks, fuzzy logic, and programming for microprocessors and microcontrollers.
- Programming Skills: Acquired strong computer programming and C programming knowledge, with applications in various engineering and technical projects.