Vishwas Singh

vishwassingh2272@gmail.com | LinkedIn | GitHub | Portfolio

Education

Indian Institute of Information Technology, Guwahati Guwahati, India
Bachelor of Technology in Computer Science and Engineering; ***CGPA: 8.1*** *Aug 2023 – Present*

Coursework: Data Structures & Algorithms, Operating Systems, DBMS, Computer Networks, Artificial Intelligence, Digital Image Processing.

Technical Skills

Languages: C++, Python, Java, JavaScript, SQL, HTML/CSS
Machine Learning: TensorFlow, Keras, PyTorch, OpenCV, YOLOv8, Scikit-learn, Pandas
Development: Flask, Streamlit, FastAPI, Docker, Git/GitHub, Postman
Databases: MySQL, MongoDB

Projects

Advanced Automatic Number Plate Recognition (ANPR) System Python, YOLO, OpenCV, PyTorch

- Architected a real-time license plate detection pipeline using **YOLOv8**, achieving high-speed inference on live video feeds.
- Engineered a robust preprocessing module utilizing **Perspective Transformation** and **Adaptive Thresholding** to correct skewed angles and variable lighting conditions.
- Integrated **Optical Character Recognition (OCR)** with a custom character segmentation algorithm to extract alphanumeric data from localized plates.
- Researched and implemented a **Super-Resolution (SR-GAN)** prototype to enhance low-quality CCTV frames, improving recognition accuracy on distant vehicles by reconstructing high-frequency details.

Shakespearean Text Generator (LSTM & NLP) Python, TensorFlow, Streamlit

- Built a generative language model using **Stacked Long Short-Term Memory (LSTM)** networks (150/100 units) to simulate Shakespearean prose.
- Trained on a large corpus utilizing **100-dimensional Word Embeddings**, optimizing for sequence coherence and grammatical structure.
- Deployed the model via an interactive **Streamlit** web application and containerized the environment using **Docker** for seamless scalability.

CineSentiment: Movie Review Analyzer Python, Keras, Flask

- Designed a sentiment classification system for IMDB reviews using **Recurrent Neural Networks (RNNs)** and custom embedding layers.
- Engineered a production-ready **REST API using Flask** to expose the model as a microservice, handling JSON payloads for external integration.
- Optimized training performance using the **Adam optimizer** and implemented Early Stopping callbacks to prevent overfitting on a 25,000-sample dataset.

Achievements

- Solved **330+** Data Structures & Algorithms problems on LeetCode; Achieved a max rating of **1414+**.
- Maintained active GitHub repositories demonstrating proficiency in Deep Learning architectures and Full Stack Web Development.