PRAGYA SOOD

22btrcn209@jainuniversity.ac.in | +919663925902 | linkedin.com/in/pragya-sood2004

EDUCATION

B.Tech in Computer Science and Engineering
JAIN UNIVERSITY, BENGALURU, KARNATAKA
(Aug 2022 - Present)

PUC - Science Stream
DEEKSHA COLLEGE FOR LEARNING, BENGALURU
(May 2020 - May 2022)

CBSE - 10th Grade
KENDRIYA VIDYALAYA D.R.D.O, BENGALURU
(July 2010 - July 2020)

SKILLS

TECHNICAL SKILLS

- **Programming:** Python, Java, JavaScript
- **Frameworks & Libraries:** TensorFlow, Keras, NumPy, Pandas, Scikit-learn, Express.js, React (basics)
- **Databases & Backend**: MySQL, MongoDB, Node.js, REST APIs
- **Domains**: Machine Learning, Speech Recognition, Natural Language Processing, Full Stack Web Development

SOFT SKILLS

Problem Solving | Team Collaboration | Communication (Public Speaking) | Adaptability | Resilience

WORK EXPERIENCE

SOFTWARE TRAINEE

TSECOND, BANGLORE (FEB 2025 - PRESENT)

- Built an offline speech-to-text (STT) system using Vosk, reducing dependency on cloud APIs and enabling real-time transcription on edge devices for field technicians.
- Fine-tuned the model on Indian-accented LibriSpeech data, improving transcription accuracy by **~32%** and reducing average inference latency by **~40%**.
- Achieved seamless integration into the data pipeline, enabling automated logging of audio-to-text records with **98%+** uptime in offline environments

PROJECTS

PRODUCT PRICING SYSTEM

- Engineered a dynamic pricing tool using **MySQL**, factoring in BOM, yield, warranty, and market segment.
- Reduced data redundancy and improved query speed by **40%** with a normalized database schema.
- Cut user navigation time by **25%** through Figma-based UI redesign and usability testing.

SARCASM DETECTION USING LSTM(Tensorflow)

- Built a sarcasm classifier using LSTM-based deep learning model in TensorFlow and Keras.
- Preprocessed text data with tokenization and padded sequences to create fixed-length inputs.
- Achieved 92% accuracy after 30 training epochs on a labeled dataset; performance surpassed baseline models by ~10%.

WEATHER API APP

- Developed a GUI-based desktop weather application using Python and Tkinter to fetch and display live weather conditions.
- Integrated WeatherStack API and used geopy with timezonefinder to determine local weather and timezone from user input.
- Integrated API data handling logic to dynamically update content based on user location.

CERTIFICATIONS

- **Machine Learning with Python - IBM (Coursera)**
- **Generative AI for Everyone - Fractal Analytics (Coursera)**
- **Cybersecurity for Everyone - University of Maryland (Coursera)**
- **Linux Fundamentals - LearnQuest (Coursera)**
- **Cisco CCNAv7 Certification Series - Cisco Networking Academy**