DIVYA KHUNT

8320909334 divyakhunt3325@gmail.com linkedin.com/divyakhunt github.com/divyakhunt

Summary

Machine Learning Engineer with strong foundations in deep learning, computer vision, NLP, and audio processing. Skilled in Python and building scalable, intelligent systems with clean, research-driven implementation. Passionate about ethical AI and solving real-world problems through data-driven innovation.

Technical Skills

Languages: Python, Java, HTML, CSS, and JavaScript
Machine Learning: scikit-learn, LightGBM, XGBoost; Hyperparameter Tuning (GridSearchCV, RandomizedSearchCV), Cross-Validation; Model Evaluation Metrics (Precision, Recall, F1-score, ROC-AUC)
Data Analysis & Visualization: NumPy, Pandas, Matplotlib, Seaborn; Data Cleaning, Feature Engineering, Data Augmentation
Deep Learning: TensorFlow, Keras, and PyTorch; CNNs, RNNs (LSTM, BiLSTM, GRU), Encoder–Decoder, Transformers, Transfer Learning, and Text Generation
Computer Vision: OpenCV, MediaPipe; Image Captioning, Hand Gesture Recognition, Facial Emotion Recognition, and Sign Language Translation
Natural Language Processing: NLTK, spaCy, and Word2Vec; Tokenization and Embeddings
Speech & Audio Processing: Librosa, SoundFile, pyAudioAnalysis, and pyttsx3; MFCC, Chroma, ZCR, Spectrograms
Tools & Frameworks: Jupyter Notebook, Google Colab, VS Code, and Git/GitHub; TensorBoard, Streamlit, and Flask
Databases & APIs: MySQL; Google Drive API, and Kaggle Datasets

Education

Sarvajanik College of Engineering and Technology (SCET) Aug 2022 – May 2026
B.Tech in Computer Engineering — ***CGPA: 9.35 / 10*** *Surat, Gujarat*

P.P. Savani Vidhyabhavan Mar 2022
Higher Secondary Certificate (HSC), GSEB — ***Percentage: 77.54%*** *Surat, Gujarat*

Projects

Speech Emotion Recognition (CNN-BiLSTM) | *TensorFlow, Keras, Librosa, SoundFile*
- Designed a CNN-BiLSTM model for speech emotion recognition using MFCC, Chroma, ZCR, and RMS features.
- Balanced distribution with data augmentation and trained on RAVDESS, TESS, CREMA-D, and SAVEE datasets.
- Optimized with early stopping, ReduceLROnPlateau, and batch normalization to prevent overfitting.

Sign Language to Speech Translator | *TensorFlow, MediaPipe, OpenCV, pyttsx3*
- Built a DNN system to recognize both-hand gestures using MediaPipe and convert them to speech via pyttsx3.
- Wrote modular scripts for data capture, model training, and OpenCV-based inference with sub-2s response.
- Enabled gesture updates via CSV and retraining; visualized training metrics for performance tracking.

Next Word Prediction Web App | *TensorFlow, Flask, Bi-LSTM, NLP*
- Built a Flask app that predicts the next word in real-time using a Bi-LSTM model trained on Sherlock Holmes text.
- Processed and tokenized text data; trained the model for next-word prediction with smooth frontend integration.
- Enabled keyboard interaction for fast word insertion by pressing `Tab`; deployed the app live using Render.
- Optimized model and code for real-time inference with low latency.

Image Captioning (DenseNet201 + LSTM) | *TensorFlow, Keras, NLP, Flickr8k*
- Generated image descriptions using DenseNet201 as encoder and LSTM as decoder.
- Preprocessed captions, extracted features, and trained on Flickr8k with 5 captions per image.
- Built a custom data generator and used greedy search to generate coherent captions.
- Serialized features and tokenizer for efficient inference and deployment.

Facial Emotion Recognition (ResNet50) | *TensorFlow, Keras, OpenCV, FER Dataset*
- Built a facial emotion classifier using ResNet50 trained on the FER dataset.
- Converted grayscale to RGB, normalized inputs, and applied real-time augmentation.
- Added GAP, Dense, and Softmax layers for multi-class emotion prediction.
- Used early stopping; visualized confusion matrix and accuracy/loss plots.