**Mukesh Kumar**

[Your City, India] | [Email] | [Phone] | [LinkedIn] | [GitHub]

**OBJECTIVE**

Detail-oriented and aspiring Data Scientist with a BTech in Computer Science and certification in FullStack Data Science with Generative AI . Experienced in building end-to-end machine learning solutions spanning NLP, computer vision, and recommendation systems. Skilled in deploying models through interactive web applications, eager to contribute data-driven insights and AI innovation.

**EDUCATION**

**Bachelor of Technology (BTech), Computer Science**

[Your University], [City, Year of Graduation]

**Certification**:

* FullStack Data Science with Generative AI — [Issuing Platform/Institution]
* Second certification

**TECHNICAL SKILLS**

**Programming** : Python, TensorFlow, Keras, OpenCV, Docker

**Machine Learning** : Deep Learning, CNNs, Transfer Learning, Transformers, NLP

**Libraries/Frameworks** : Hugging Face Transformers, scikit-learn, NLTK, MediaPipe, AutoPy

**Data Handling**  : Pandas, NumPy

**Tools** : Streamlit, Gradio, Jupyter Notebook, TMDb API, Pickle

**Deployment** : Docker, Web UI development

**CAPSTONE PROJECT**

**Language Translator (English to Hindi):**

* Developed a Seq2Seq LSTM model from scratch in TensorFlow for English-Hindi translation.
* Handled comprehensive data processing including tokenization, padding, and parallel corpus preparation.
* Built encoder-decoder architecture, trained the model end-to-end, and optimized translation accuracy.
* Deployed a Streamlit web app demonstrating real-time translation, highlighting production readiness.
* Utilized Python libraries such as NumPy, Pandas, and Pickle for efficient data and model management.

**Tools**: Python, TensorFlow 2.x, Streamlit, NumPy, Pandas, Pickle, Jupyter Notebook

**PROJECTS**

**Wine Quality Prediction (ML Project)**

* Analyzed the UCI Wine Quality dataset to identify key features affecting wine quality using univariate, bivariate, and multivariate analysis.
* Cleaned and prepared data by handling missing values, removing duplicates, capping outliers (IQR), and scaling features.
* Engineered features by mapping wine quality into 3 categories for multi-class classification.
* Trained and evaluated models (KNN, Naive Bayes, SVM, Decision Tree), using GridSearchCV for hyperparameter tuning.
* Achieved 85.66% accuracy (best model); interpreted results using classification reports and feature importance analysis.

**Tools**: *Python, Pandas, Scikit-learn, Matplotlib, Seaborn*

**Heart Disease Prediction using Machine Learning**

* Performed exploratory data analysis: univariate, bivariate, multivariate analysis, and correlation heatmap
* Cleaned data, handled missing values with IterativeImputer, and treated outliers
* Applied feature scaling and categorical encoding techniques
* Trained and compared 10+ ML models: Logistic Regression, SVM, Random Forest, XGBoost, LightGBM, CatBoost, etc.
* Tuned hyperparameters using GridSearchCV and evaluated models using confusion matrix
* Automated model evaluation to identify the best-performing classifier

***Tools****: Python, scikit-learn, XGBoost, LightGBM*

**Discovering Vehicle Segments using Unsupervised Clustering**

* Performed univariate, bivariate, and multivariate analysis to understand feature relationships
* Cleaned and pre-processed data (missing values, scaling) for unsupervised learning
* Applied PCA for dimensionality reduction and visualized data in reduced dimensions
* Used K-Means, DBSCAN, and Hierarchical Clustering to explore inherent groupings in vehicles
* Determined optimal number of clusters using Elbow Method and Silhouette Score
* Identified distinct vehicle categories based on performance and efficiency features without using labels

**Tools**: Python, scikit-learn, Pandas, NumPy, Seaborn, Matplotlib, Jupyter Notebook

**Content-Based Movie Recommendation System:**

* Created a recommender system using TMDb dataset by combining metadata and computing cosine similarity to recommend movies.
* Integrated TMDb API for fetching movie posters and developed an interactive Streamlit UI.

**Tools**: Python, Pandas, scikit-learn, NLTK, Streamlit, TMDb API

**Cat vs. Dog Image Classifier (Custom CNN):**

* Built binary classifier from scratch with data augmentation and dropout, containerized using Docker.

**Tools**: TensorFlow, Keras, OpenCV, Docker, scikit-learn

**Cat vs. Dog Classification (Transfer Learning):**

* Applied VGG16, ResNet50, and MobileNetV2 pre-trained CNNs for classification with fine-tuning and feature extraction.
* Evaluated with accuracy, confusion matrix, and F1 score metrics.

**Tools**: TensorFlow, Keras, scikit-learn, Matplotlib, Docker

**Gesture-Controlled Virtual Mouse:**

* Developed real-time hand gesture-based virtual mouse using OpenCV and MediaPipe, enhancing hands-free interaction.

**Tools**: Python, OpenCV, MediaPipe, AutoPy

**GENERATIVE AI PROJECTS**

* **CLI and UI Chatbot:** Created conversational agents using DialoGPT, Flan-T5, and GODEL models, deployed with Gradio UI.
* **Text Summarizer Tool:** Built a text summarization app using facebook/bart-large-cnn model with Streamlit interface.
* **Image Classification with Vision Transformers**: Developed transformer-based image classifiers with interactive Gradio UI.
* **Image Captioning System**: Generated image captions using Salesforce blip-image-captioning-base model and Gradio UI.
* **Object Detection System**: Implemented object detection leveraging Facebook detr-resnet-50 with Gradio visualization.
* **Visual Question Answering (VQA) System**: Built VQA app with Salesforce blip-vqa-base model and real-time Gradio interface.

**Tools**: Python, Hugging Face Transformers, Gradio, Streamlit

**ADDITIONAL INFORMATION**

* Strong foundation in NLP, computer vision, deep learning, and model deployment.
* Effective communicator with experience in building interactive web apps for AI demos.
* Enthusiastic about continuous learning and applying AI to solve practical problems.