Adarsh Divase
Data Scientist | Machine Learning Engineer | AI Engineer

Phone: 8451918440 | Email: divaseadarsh608@gmail.com | GitHub: github.com/adarshdivase

---

Summary

Highly analytical AI & Data Science professional leveraging expertise in predictive modeling, deep learning architectures, and scalable MLOps to optimize end-to-end Machine Learning pipelines. Proven track record in leading the development of resilient data engineering and advanced generative AI solutions. Proficient in Python, TensorFlow, PyTorch, SQL, AWS, Hadoop, and Spark, driving innovative solution architecture within dynamic environments.

Education

A. C. Patil College of Engineering, Navi Mumbai   *Expected Graduation: 2025*
- **B.Tech in AI & Data Science** | GPA: 8.7
- Relevant Coursework: Deep Learning, Machine Learning, Data Structures & Algorithms, Distributed Database Systems, Cybersecurity.

Certifications

- **AWS Certified Solutions Architect Associate:** Designed scalable, highly available, and fault-tolerant cloud architectures on AWS.
- **Data Engineering Course (Hadoop & Spark):** Practical skills in large-scale data processing, ETL pipelines, and distributed computing.
- **Mastering Generative AI and ChatGPT:** Specialized training in modern generative models and LLMs.

Experience

Python Backend Developer Intern | NDTR Reflection Engineers (Remote)   *Oct 2024 – Mar 2025*
- Built and optimized scalable backend microservices with Python, FastAPI, and Flask, improving API response times by 25% and handling high daily request volumes with 99.9% uptime.
- Managed PostgreSQL databases via SQLAlchemy ORM, reducing data retrieval time by 15%.
- Implemented JWT-based authentication for over 50 API endpoints, enhancing security significantly.
- Integrated SendGrid (over 1,000 transactional emails/day) and Razorpay.
- Orchestrated Docker-containerized deployments to Render and Heroku, reducing deployment cycles by 40%.
- **Skills:** Python, FastAPI, Flask, RESTful APIs, PostgreSQL, SQLAlchemy, Docker, Render, Heroku, JWT, Razorpay, SendGrid, Microservices.

Technical Skills

- **Programming Languages & Frameworks:** Python (NumPy, Pandas), SQL, FastAPI, Flask, Data Structures, Algorithms.
- **Machine Learning:** Scikit-learn, XGBoost, Random Forests, SVMs, Regression, Classification, Clustering, Feature Engineering, Model Evaluation, SMOTE, Reinforcement Learning.
- **Deep Learning:** TensorFlow, PyTorch, Keras, CNNs, RNNs (LSTMs, GRUs), Transformers, Transfer Learning, GANs.
- **MLOps & Deployment:** Docker, Kubernetes (basics), Render, Heroku, Git, Uvicorn, RESTful API, Microservices, CI/CD, Model Versioning, Monitoring, Hugging Face Spaces.
- **Big Data & Databases:** Hadoop, Spark, Apache Kafka, PostgreSQL, MySQL, SQLAlchemy, MongoDB, SQLite3.
- **Data Visualization & BI:** Plotly, Matplotlib, Seaborn, Dash, Tableau (conceptual), Streamlit.
- **Specialized Tools:** SHAP, LIME, Prophet, Whisper API, YOLOv5, OpenAI API, LLMs (ChatGPT), Financial Modeling, Monte Carlo Simulation, A/B Testing, Alpaca API, Technical Indicators (MACD, RSI, Bollinger Bands, SMAs).

Projects

AI Services Toolkit Pro (Multi-Modal AI Assistant)

- Architected and deployed a comprehensive, integrated Multi-Modal AI Toolkit on Hugging Face Spaces, integrating 9 state-of-the-art Transformer pipelines for diverse AI capabilities.
- Developed a robust FastAPI backend with asynchronous operations and Pydantic models, exposed via `/api` endpoints.
- Designed an intuitive Streamlit frontend for interactive AI service interaction, API call history, and system status monitoring.
- Implemented Text-to-Speech (TTS) with dynamic speaker embeddings and Speech-to-Text (STT) with automatic audio resampling, enhancing accessibility by 15% for 5,000 daily users.
- **Utilized key libraries:** `FastAPI`, `Streamlit`, `Hugging Face Transformers`, `PyTorch`, `soundfile`, `librosa`.

Hybrid Predictive Maintenance System

- Developed and deployed an integrated Hybrid Predictive Maintenance system on Streamlit, integrating supervised learning (LSTM) and reinforcement learning for optimal maintenance recommendations.
- Engineered an LSTM-based deep learning model using TensorFlow/Keras to predict machine health and Remaining Useful Life (RUL) from synthetic time-series data, achieving 30% improvement in prediction accuracy.
- Designed a comprehensive Streamlit multi-page application with a "Live Dashboard" for real-time monitoring and a "Historical Explorer" for data analysis.
- Established a persistent SQLite database to log simulation reports, including health metrics and explainability insights (simulated SHAP/LIME), reducing operational downtime costs by an estimated 20%, resulting in annual savings of $50,000.
- **Utilized key libraries:** `Streamlit`, `TensorFlow`, `Keras`, `NumPy`, `Pandas`, `SQLite3`.

AI-Fueled E-commerce Analytics & Sales Forecasting System

- Developed and deployed an AI-powered E-commerce Analytics and Sales Forecasting platform on Streamlit.
- Constructed a robust sales forecasting model using Facebook Prophet, achieving 92% MAPE for quarterly sales predictions and reducing stockouts by 10%.
- Engineered comprehensive data processing pipelines using Pandas and NumPy, preparing data for time series forecasting and business intelligence.
- Created interactive and dynamic dashboards using Plotly Express and Streamlit, visualizing key e-commerce metrics and forecast performance, leading to 15% increase in revenue strategy impact and 5% uplift in overall quarterly sales revenue.
- **Utilized key libraries:** `Streamlit`, `Prophet`, `Pandas`, `NumPy`, `Plotly Express`.

Customer Churn Prediction and API Deployment

- Architected and deployed an integrated Customer Churn Prediction system on Streamlit with a FastAPI backend for model inference.
- Developed a robust churn prediction model (F1-score of 0.87) using advanced ML techniques, identifying 75% of potential churners within 30 days.
- Engineered data preprocessing pipelines with SMOTE, increasing model recall for the minority class by 25%.
- Launched a high-performance RESTful API using FastAPI and Uvicorn, achieving sub-100ms inference latency and handling up to 500 requests/second.
- **Utilized key libraries:** `FastAPI`, `Streamlit`, `Scikit-learn`, `Pandas`, `NumPy`, `imblearn`.

AI-Powered Trading System with Risk Analytics

- Engineered a real-time AI-driven algorithmic trading system deployed on Streamlit, providing live market data, technical indicators, and automated trade execution, leading to a 5% increase in simulated portfolio returns.
- Developed an enhanced trading strategy combining MACD, RSI, and Bollinger Bands for multi-factor signal generation with volume confirmation.
- Implemented robust risk management protocols, including dynamic position sizing based on portfolio risk (2% per trade), and maximum daily loss limits (2%), ensuring capital preservation.
- Integrated with Alpaca API for fetching historical and live stock data, enabling real-time bar updates and seamless order submission.
- **Utilized key libraries:** `numpy`, `pandas`, `torch`, `scikit-learn`, `plotly`, `alpaca-py`.