Harikesh Kushwaha

Location: New Delhi, Delhi

LinkedIn | Portfolio | GitHub | Kaggle Email: harikeshkumar0926@gmail.com | Mobile: +919838422934

Technical Skills

Languages	:	Python, SQL, JavaScript, MATLAB, C++
Frameworks	:	TensorFlow, Keras, PyTorch, Scikit-learn, Django, Streamlit
Libraries	:	matplotlib, pandas, NumPy, NLTK, Seaborn, BeautifulSoup, Selenium
Databases	:	MySQL, MSSQL, MongoDB
Dev Tools	:	VS Code, Git, GitHub, Airflow, Jupyter Notebook, Anaconda, AWS, Azure, Kaggle

Experience

Junior Data Scientist June 2023 – Present

Nuvoretail Enlytical Technology Private Limited *New Delhi*

- **Automated Amazon Bidding with Python:** Developed Python scripts to automate Amazon Marketing Services bidding, resulting in a **50% reduction** in manual intervention and a **20% increase** in performance.
- **Improved Log Tracking and Issue Identification with Airflow Scheduling:** Leveraged Airflow DAGs for streamlined **log management** and **rapid task issue identification**, enhancing process reliability.
- **Improved Bidding Accuracy with Machine Learning:** Developed **machine learning** models and **statistical algorithms** to predict the optimal bid for a product resulting in cost-effective advertisement.
- **Custom Flask Server Development:** Designed and implemented a Flask server to facilitate **team interaction** and empower seamless data modification within the system.

Projects

House Prices Prediction *Python, pandas, scikit-learn, kaggle, Matplotlib, Seaborn* Source Code

- Analyzed over **80** features to predict house prices using machine learning.
- Performed **data visualization** and **feature engineering** using Matplotlib and Seaborn, respectively.
- Trained **multiple models** using scikit-learn and selected the best one by applying **grid search** and **cross-validation**. Achieved a **top 12%** ranking on the Kaggle leaderboard.

Digit Recognizer *Python, TensorFlow, Keras, Kaggle* Source Code

- Developed a very deep **convolutional neural network** using TensorFlow and Keras with **dropout** and **batch normalization** to improve performance.
- Achieved an accuracy of **99.48%** on the test set, securing a place in the **top 15%** on the Kaggle leaderboard.
- Deployed the app on **HuggingFace** using **Gradio** and made it available to the public. Find the app here.

Food Vision *Python, TensorFlow, Colab* Source Code

- Developed a deep **neural network** using TensorFlow and Keras to classify **101 categories of food**.
- Used a pretrained **EfficientNet** model to extract features from the food images, and then **fine-tuned** the model to improve its accuracy.
- Achieved an accuracy of **80%** on the test set, demonstrating the effectiveness of the approach in addressing complex image recognition problems.

NLP With Disaster Tweets *Python, TensorFlow, NLP, Text Vectorization, LSTM, GRU, CNN* Source Code

- Developed NLP models to classify disaster and non-disaster tweets using **text vectorization**, various **word embeddings**, and deep learning models including **LSTM**, **GRU**, their **bidirectional** variants, and **1D CNNs**
- Utilized the **Universal Sentence Encoder** to create embeddings on both the character and word levels, and implemented a **multivariate** model using the **functional API** of **TensorFlow**.

TensorFlow Speech Recognition Challenge *Python, pandas, TensorFlow, kaggle* Source Code

- Trained a deep neural network to recognize **30** different commands by creating waveforms and transforming them into **2D spectrograms** using STFT.
- Used a convolutional neural network architecture and achieved an **accuracy of about 90%** on the test set.

Titanic - Machine Learning from Disaster *Python, pandas, sklearn, kaggle* Source Code

- Analyzed the Titanic dataset and performed **data cleaning**, **feature engineering**, and **data visualization**.
- Built several machine learning models including **Logistic Regression**, **Random Forest**, and **Gradient Boosting** and selected the best model using **cross-validation**.
- Achieved a test accuracy of **78.5%**, which was in the **top 12%** of the Kaggle leaderboard at the time.

NNet *Python, NumPy, Neural Network* Source Code

- Developed a module for arbitrary neural network architecture using **Python** and **NumPy**, implementing layers such as **Dense**, **Dropout**, **Conv2D**, **Flatten**, **Reshape** etc.
- Implemented both the forward and backward pass of the layers, demonstrating proficiency in **backpropagation** and **gradient descent**.
- Created an API similar to **Keras** for seamless integration and implemented various activation functions including **ReLU**, **tanh**, **sigmoid**, and **softmax**.
- Demonstrated strong skills in **machine learning**, **Python programming**, and **mathematics** while gaining a deeper understanding of the inner workings of neural networks.

ReVision *Python, NumPy, TensorFlow, PyTorch, CLI* Source Code

- Created a personal project called **ReVision** to learn the concepts and implementation details of groundbreaking **computer vision papers**.
- Utilized popular deep learning frameworks such as **TensorFlow** and **PyTorch** to implement the architectures of seminal papers like **LeNet**, **AlexNet**, **VGG**, **ResNet**, **Inception**, **EfficientNet**, etc.
- Developed a deep understanding of the underlying principles of deep learning and computer vision, while improving skills in **Python programming**, **machine learning**, and **deep learning**.
- Demonstrated proficiency in various computer vision tasks, such as **image classification**, **object detection**, and **semantic segmentation**.

IBM Data Analytics Capstone Project *Python, pandas, Matplotlib, Web Scraping, Web API* Source Code

- Gathered and analyzed data from various sources, including **API** and **web scraping**. Conducted **exploratory data analysis** and **wrangling** to prepare the data for further analysis.
- Built a **dynamic dashboard** to extract valuable insights from the collected data, and effectively **communicated** the findings to others through an **engaging presentation**.

Credit Risk Assesment *Python, pandas, scikit-learn, kaggle* Source Code

- Developed a credit textbfrisk assessment model by analyzing various customer features, performing data cleaning, feature engineering, and exploratory data analysis.
- Established a basic model for initial experimentation, and trained advanced models such as **LR**, **SVM**, **XGBoost**, **Catboost**. Top performing model, achieved a test AUC-ROC score of **0.97** and precision of **0.96**.

pystock *Python, portfolio theory, pytest* Source Code

- Developed **pystock**, a comprehensive **Python library** for **portfolio optimization** and management. Utilizing **object-oriented programming**, created a **user-friendly API** capable of optimizing portfolios with any number of securities.
- The library includes various models, such as the **Capital Asset Pricing Model, Single Index Model**, **Fama-French three-** and **five-factor models**, and has a suite of over **100 unit tests** written with **pytest and fixtures**, spanning more than **1500 lines of code**.

optionalyzer *Python, options, futures, plotly, BS model* Source Code

- Developed **optionalyzer**, a powerful **Python library** for **Options Strategy Builder** that makes it easy to create custom options trading strategies.
- Implemented the **Black-Scholes Model** to accurately calculate **Option prices** and utilized **optimization** techniques to find the **implied volatility** of the Option, enabling users to make better trading decisions.
- Leveraged **Plotly** to create an **interactive Options payoff diagram** for any date, allowing users to explore potential outcomes for different combinations of Options.
- Developed a **user-friendly API** that enables users to easily add **short or long**, **Put and Call Options** to the payoff diagram for strategy building.

frontier *Python, portfolio theory, pytest* Source Code

- Developed **frontier**, a Python module for **plotting the efficient frontier** of a portfolio with an arbitrary number of securities.

- Utilizes **Monte Carlo simulations** to create an **interactive efficient frontier**, enabling users to easily explore different portfolios and their expected returns and risks.
- Built on top of **pystock**, the module supports all the models that are supported by pystock, including **CAPM, SIM, FF3FM** and **FF5FM**, making it a comprehensive tool for portfolio optimization and management.

Tableau Dashboards *Tableau, Web Scraping, Web API, BeautifulSoup* Music Books

- Created an interactive **Tableau viz** showcasing my **Spotify streaming history** over several years, using **data blending** and **calculated fields** to present key insights.
- Utilized **web scraping** techniques to extract my book reading history from **Goodreads** and created an interactive **Tableau** dashboard to analyze and visualize the data.

Education

Indian Institute of Technology Delhi New Delhi, India
Master of Science in Physics, (8.6 GPA) *July 2021 – May 2023*

Banaras Hindu University Varanasi, Uttar Pradesh India
Bachelor of Science in Physics, (8.4 GPA) *July 2018 – May 2021*

Certifications

- Machine Learning Specialization (DeepLearning.AI) Certificate
- Simulation Models for Decision Making (University of Minnesota) Certificate
- IBM Data Analyst Capstone Project (IBM) Certificate
- Financial Markets (Yale University) Certificate
- Deep Learning Specialization (DeepLearning.AI) Certificate
- TensorFlow Developer Certificate in 2022: Zero to Mastery (Udemy) Certificate
- TensorFlow: Advanced Techniques Specialization Certificate