Jerome Siljan

JeromeSiljan@gmail.com • (469) 733-8585 • GitHub • LinkedIn
Data Scientist | Software Engineer

Core Competencies:

- HDFS with Apache Hive, Impala
- Random forest/decision trees
- Time series analysis
- Pyspark for processing and training models on large datasets

Technical Skills:

Languages: C, C++, Python, Java, SQL, ARM Assembly, Verilog (HDL), HTML, CSS
Tools/Libraries: pyspark, tensorflow, Apache Hive, Apache Impala, opencv, pandas, matplotlib, numpy, flask, CockroachDB, SQLite3, AutoSys git, BitBucket, GitHub, Confluence

Education:

Bachelor's of Science in Computer Engineering at The University of Texas at Arlington
Spring 2024

Work Experience:

Citigroup – PBWMT Summer Analyst

- Developed a high-impact program that integrated Impala querying functionality into an HDFS database, processing approximately 20 thousand records daily using AutoSys.
- Collaborated with a team to realize a document scanning and OCR application that streamlines the credit card application process for users at Citi.
- Led an overhaul of legacy code, replacing hardcoded implementations with a configuration file based solution to significantly improve readability and maintainability.

Projects:

Forest for Flood

- Trained and tuned random forest to assign flood insurance policies based on features such as latitude, longitude, elevation, etc.
- Processed over 50 million data points from FEMA's flood insurance database with pyspark's RDDs to ease fast, multithreaded processing.
- Optimized random forest by adjusting hyperparameters with a grid search.
- Analyzed and culled features in the preprocessing stage to reduce training time and improve accuracy.

Monocle

- This project was built to solve CBRE's challenge from the 2022 Texas A&M Datathon.
- Used machine learning to identify, read, and cluster text in order to sort them into sequences using **CRAFT** and **tesseract**.
- Designed and implemented a custom clustering algorithm that groups text based on their coordinates.
- Increased OCR accuracy from 2% to 80% by including text detection (using **CRAFT**) in the preprocessing stage.
- Developed **flask** web UI that interfaces with the OCR component to display results.
- Successfully organized and managed the work load between a team of engineers in order to complete the project in a tight timeline.

Pistachio Detective

- Classified images of pistachios as being either Kirmizi or Siirt pistachios using a convolutional neural network in **tensorflow.**

- Accurately classified pistachios with an 85% accuracy with only 2148 images and a 9/1 training/validation split.
- The Pistachio Detective uses a convolutional neural network to both speed up compile times and minimize the impact of “noise” in the images. Dropouts were used to combat overfitting.
- Methodically used machine learning fundamentals, such as producing an appropriate training/testing set balance, applying different techniques to avoid overfitting, and methods more specific to image classification, such as CNNs and image augmentation to achieve a high accuracy.