

Jerome Siljan

Jerome.Siljan@mavs.uta.edu • (469)733-8585 • [GitHub](#)

Core Competencies:

- Cleaning data
- Natural language processing with machine learning
- Image classification with RNN, CNNs

Technical Skills:

Languages: C, C++, Python, Java, ARM Assembly, Verilog (HDL), HTML, CSS

Applications: Arduino IDE, Quartus Prime, OnShape, Fusion360, Ultimaker Cura

Tools/Libraries: tensorflow, pandas, matplotlib, numpy, flask, SQLite3

Certifications: IBM's Analyzing Data with Python (DA0101EN)

Education:

The University of Texas at Arlington

Bachelors of Science in Computer Engineering

December 2023

Relevant Projects:

Finance "Skimmer"

- A Python application that cleans, organizes, and stores financial data from three different banks. This was created to give a unified summary of transactions. It is started from the command line, but stores data in an SQLite3 database.
- This program used **pandas** to parse through the different .csv statements from the banks. It would then accordingly drop/fill columns and perform basic mathematical operations on others.
- The **sqlite3** library was used to take the pandas dataframes and store them locally. From there, different queries were run to add tags to transactions. Using an SQLite3 database was critical as the tagging makes use of relational databases.
- Finally, **flask** read from the SQLite3 database and provided the WebUI in a more accessible manner.
- This project taught me how to clean, read, and analyze data as well as present it in a clear format. I learned a lot about how pandas works and how to use it in conjunction with SQLite3 (and SQL-like databases in general).

Pistachio Detective

- This application was able to classify images of pistachios as being either Kirmizi or Siirt pistachios. I used an online dataset with only 2148 images and was able to achieve 85% accuracy.
- **Tensorflow** was the framework used in this project. `image_dataset_from_directory` was used to split the testing and training models.
- 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.
- Through the development of this application, I learned a lot about machine learning fundamentals, such as the importance of coming up with 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.