Simulating Future NASA Observatory Data Streams Using James Webb Space Telescope Archives

Dominick Perini and Jeri Brentlinger

Purpose and Background

NASA's Habitable Worlds Observatory (HWO) represents a bold step forward in the search for Earth-like planets and the potential for life beyond our solar system. As the next-generation space observatory, HWO will build on the success of missions like the James Webb Space Telescope (JWST) to directly image exoplanets and analyze their atmospheres for biosignatures. However, to maximize the mission's scientific potential, leveraging the vast data already collected by existing observatories to train and refine AI systems is crucial. There is a significant need for prototype data-engineering pipelines that leverage and transform existing observatory data to simulate a stream of mission data that must be processed live by an AI system. These AI systems, trained on historical datasets, will be critical for real-time analysis, anomaly detection, and decision-making when processing live data streams from HWO, ensuring we can interpret the most subtle signals of habitability and life as they emerge from the next generation of space observatories.

To address the need for an environment that enables AI developers to simulate model deployment, mission data processing, and online learning, this project will include the development of a data engineering pipeline that consumes archived NASA observatory data, applies data "de-conditioning" techniques that will introduce imperfections, null values, corruption, and noise to the data, and finally stream the simulated live-feed data to an AI analytics endpoint for consumption and analysis.

This project will answer the following questions regarding AI development, data engineering, and operationalization of academic/scientific innovations:

- How accessible is archived NASA observatory data, and what conditioning steps have been taken to prepare the data for AI algorithm training and development?
- What data de-conditioning techniques can be used to simulate realistic deployment scenarios for AI-enabled systems, and what are their effects on online learning approaches?
- Can incorporating archived NASA observatory data accelerate AI algorithm development for future large-scale missions? (e.g. transfer learning, domain adaptation, and meta-learning)

The resulting data engineering pipeline will incorporate the following components:

1. Archived Data Consumption
2. Data De-Conditioning
3. Data Stream Simulation
4. Live Data Quality Visualizations

Development Plan

The Archived Data Consumption component will be achieved by accessing the MAST NASA observatory data archive (www.stsci.edu) and downloading a sufficiently diverse dataset from the JWST collection. This dataset will be stored in an S3 bucket instance and accessed by the simulated data pipeline through the AWS API. This dataset will be explored, and the final report will include a short summary of dataset analytics.

The Data De-Conditioning component will manipulate the incoming data such that the downstream simulated live feed will pose a sufficiently challenging deployment simulation for any low-TRL (Technology Readiness Level) AI system. This could include null-value injection, variable data rates, applied noise, stream corruption, and adversarial methods such as data poisoning and concept drift. This component will leverage Python libraries, including Pandas, NumPy, and SciPy.

The Data Stream Simulation will provide the consumption endpoint with a simulated live data feed that can be adjusted and configured in real-time. The pipeline will experience actual data flow through the system using Apache Kafka.

Live Data Quality Visualizations are critically important to understanding the performance of a deployed AI system. In this case, metrics that describe the data on the receiving end of the AI system/endpoint will be updated to reflect important system metrics such as data velocity, null-value count, and time between received samples.

Expected Results

By providing the scientific community with a simple, concise, and reproducible data pipeline for streaming NASA mission data, this project informs and encourages the pursuit of operational AI systems across all domains. The codebase will be entirely containerized and configurable using Docker, and version control will be maintained with GitHub. A walkthrough of the De-Conditioning steps will be exemplified in a Jupyter Notebook, and a final report will be prepared which discusses the findings of this research, the answers to the research questions, a tutorial for the use of the developed pipeline, and potential future applications of the pipeline itself.