VISION SPACE

Tanagra Space

esa

→ DISCOVERY

ON-BOARD SOFTWARE AS A SERVICE FOR MACHINE LEARNING

Executive Summary Report

Study

Campaign: The Open Space Innovation Platform (OSIP) for OPS-SAT Experiments

Affiliation: VisionSpace Technologies GmbH (Prime), TanagraSpace (Sub)

Activity summary:

The Software as a Service for Machine Learning (SaaSyML) app aims to provide on-board machine learning (ML) capabilities to the experimenters of the ESA OPS-SAT platform. It focuses on abstracting complex ML operations to spare the users the difficulties of provisioning data sets, training models, and performing inference on new data. The SaaSyML architecture takes advantage of the OPS-SAT NanoSat Mission Operations Framework Java ecosystem and dual-core computer for multi-threaded execution. Experimenters can interact with SaaSyML via API endpoints. EM sessions with OPS-SAT demonstrated that SaaSyML can be used on-board by multiple experimenters in parallel. A ML use case for fault detection (on-board camera exposure to sunlight) was successfully replicated locally with SaaSyML.

Publishing Date: 28/11/2022
Contract Number: 4000137384/22/NL/GLC/ov
Implemented as ESA Express Procurement - EXPRO

→ THE EUROPEAN SPACE AGENCY
ESA Discovery & Preparation
From breakthrough ideas to mission feasibility. Discovery & Preparation is laying the groundwork for the future of space in Europe
Learn more on www.esa.int/discovery
Deliverables published on https://nebula.esa.int

EXECUTIVE SUMMARY

The Software as a Service for Machine Learning app, SaaSyML, aims to provide on-board machine learning capabilities to the experimenters of the ESA OPS-SAT platform. It focuses on abstracting complex machine learning operations to spare the users the difficulties of provisioning data sets, training models, and performing inference on new data.

The machine learning work loop is a complex set of operations that mainly consists of fetching data, training models with algorithms, and using those models to make predictions with new data observations. In general, machine learning models on-board of spacecraft are trained on the ground and either integrated with the spacecraft during launch, or uplinked from the ground after launch. Both approaches impose limitations on access to on-board data. For example, every time an experimenter wishes to retrain the model with new data acquired by the spacecraft, they will need to download it, retrain the model on-ground, and re-uplink the updated model to the spacecraft.

As a technology demonstrator, SaaSyML proposes to introduce the "Software as a Service" approach for on-board data feed subscription and machine learning operations services. With SaaS apps, instead of developing, or installing and maintaining their own software, users can remotely access the functionalities they need from existing third-party hosted software.

The OPS-SAT Space Lab dual-core computer and Linux OS enables experimenters to develop and test their own C, C++, Java, and Python apps on-board. In addition, the lab's set of sensors and actuators can provide a steady stream of data for the apps to use.

The design choices for SaaSyML architecture take advantage of the OPS-SAT NanoSat Mission Operations Framework (NMF) Java ecosystem and dual-core computer for multi-threaded execution. This allows for execution of parallel requests by multiple app users, supported in SaaSyML by the Eclipse Vert.x toolkit.

SaaSyML makes available numerous machine learning algorithms via integration with an open-source Java library, JSAT. This integration enables users to train models for classification, regression, clustering, and outlier detection tasks. The app also allows experimenters to introduce their own label computation customized code, via an extension mechanism for third-party plugins with PF4J. Experimenters can interact with SaaSyML via API endpoints to fetch data, train models, and perform inference.

EM sessions with OPS-SAT, consisting of exhaustive tests of multiple scenarios for the entire machine learning loop, showed that SaaSyML can be used on-board by multiple experimenters for non-blocking training and inferencing requests for different models. In addition to the EM sessions results, a machine learning use case for fault detection

(FDIR), where a model learns to detect when the on-board camera is exposed to sunlight, was successfully replicated locally with SaaSyML. This is a follow-up to the previous OPS-SAT experiment, OrbitAI.

SaaSyML source code is available at https://github.com/visionspacetec/opssat-saasy-ml. A paper submission has been peer-reviewed and accepted for IEEE Aerospace Conference 2023, as part of sharing knowledge and awareness regarding SaaSyML.

As SaaSyML has been validated during OPS-SAT EM sessions, the next step is in-orbit deployment of the app and execution of a use case such as the above mentioned fault detection of camera exposure. Ongoing and future enhancements such as subscription to an inference feed, and algorithm customization via plugins and parameters in requests will introduce even more flexibility for app users to run ML experiments on-board OPS-SAT.

Traditional SaaS is a way of delivering an application's capabilities over the Internet — as a service. Instead of installing and maintaining their own software, users become clients who can remotely access the functionalities they need from existing third-party hosted software thus avoiding the complexities and costs of developing, hosting, and maintaining their own systems. The Web's "as-a-Service" ecosystem has rapidly matured in recent years with the developments of options such as Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Everything as a service (XaaS). As a technology demonstrator, the SaaSyML experiment introduces this approach to software in the space segment thus contributing the concept of Satellite Platform as a Service (SPaaS) to the "as-a-Service" ecosystem.