Integrating ML/AI workflows in a Streaming Data Management and Processing Platform for Building Energy Research

Jaewoo Shin
Rosen Center for Advanced Computing, Purdue University
shin152@purdue.edu

Lan Zhao
Rosen Center for Advanced Computing, Purdue University
lanzhao@purdue.edu

Carol X. Song
Rosen Center for Advanced Computing, Purdue University
cxsong@purdue.edu

Dikai Xu
Lyles School of Civil Engineering, Purdue University
xu1718@purdue.edu

Ming Qu
Lyles School of Civil Engineering, Purdue University
mqu@purdue.edu

Ananth Grama
Computer Science Department, Purdue University
ayg@purdue.edu

Dongyan Xu
Computer Science Department, Purdue University
dxu@purdue.edu

ABSTRACT

Aimed at reducing energy consumption and improving efficiency and sustainability, the state-of-the-art research on building energy prediction and optimization is increasingly driven by advanced ML/AI technologies using large volumes of Internet of Things (IoT) data continuously streamed from smart building facilities. Despite significant progress in recent years, researchers still encounter major challenges on data storage, access, processing, and integration due to large volumes, heterogeneous formats, continuous generation, and non-standardized metadata of IoT sensor data, as well as significant computational demands of ML/AI models. In this paper we describe the design and implementation of a scalable open-source AnalytiXIN (AXIN) data lake platform for streaming data management, modeling, and analysis. We will also describe how AXIN data lake has been used to enable data driven energy prediction and optimization research using ML/AI approach. Built on open-source software stacks including StreamCI[1], JupyterHub, and HUBzero, and integrated with both composable and GPU high performance computing (HPC) resources, the AXIN data lake infrastructure provides a flexible and scalable solution for streaming data driven research beyond the energy domain.

CCS CONCEPTS

• **Software and its engineering**; • **Computer systems organization** → Architectures; Distributed architectures; Cloud computing; • **Information systems** → Information systems applications; Computing platforms;

PEARC '24, July 21–25, 2024, Providence, RI, USA

ACM ISBN 979-8-4007-0419-2/24/07
https://doi.org/10.1145/3626203.3670599

KEYWORDS

Streaming data, Cyberinfrastructure, Building energy modeling, StreamCI, ML/AI

ACM Reference Format:
Jaewoo Shin, Lan Zhao, Carol X. Song, Dikai Xu, Ming Qu, Ananth Grama, and Dongyan Xu. 2024. Integrating ML/AI workflows in a Streaming Data Management and Processing Platform for Building Energy Research. In *Practice and Experience in Advanced Research Computing (PEARC '24), July 21–25, 2024, Providence, RI, USA.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3626203.3670599

1 INTRODUCTION

The building sector, accounting for 40% of overall energy consumption and contributing to 35% of the nation's carbon emissions in the United States, plays a substantial role in shaping our nation's energy landscape and carbon footprint. Prediction and optimization of building energy is the key to identifying the most efficient ways to operate and control buildings, ultimately leading to reductions in energy consumption and carbon emissions. Traditional physics-based statistical models such as engineering model and statistical model are often complex, inaccurate, and inflexible, and cannot support energy analysis or intelligent control of such complex systems with massive noisy data [2]. On the other side, the increasingly prevalent IoT sensors in smart facilities brought new opportunities for developing data driven AI-based energy analytics and smart controls for advanced facilities to optimize energy consumption and achieve energy sustainability [3].

Despite significant advancement in developing AI based energy prediction and optimization solutions, researchers still encounter many data and computation challenges that need to be addressed to further improve the efficiency and effectiveness of their research. For example, sensor data are continuously generated, and the large volumes of streaming data present a major problem in data collection, storage, management, sharing and access. Furthermore, sensor data come from different sources with different formats and resolutions. It takes researchers significant effort to wrangle the

data, such as cleaning, normalization, feature extraction, aggregation, and synchronization, to convert them into the format that is ready to be used by their models. Training complex AI models also requires advanced computation resources, including high performance CPUs, GPUs, and large amounts of memory. The availability and affordability of such resources is often a limiting factor for researchers from small institutions. Many applications in building energy optimization require real-time predictions and control, thus demanding automating data processing and modeling pipelines supported by computation resources in an on-demand manner, which is hard to achieve without an easy-to-use cyberinfrastructure (CI).

To address the pressing needs of advanced CI in streaming data enabled research, we designed and implemented AXIN data lake, a scalable streaming data management and modeling platform. It builds upon and extends the functionalities of StreamCI [1], a cloud based streaming data management system that enables both streaming data providers to easily collect and store their data and data consumers to easily query and access the data. StreamCI has been successfully used to manage and disseminate sensor data in multiple domains including ecology, agriculture, and manufacturing. In this project, we focused on expanding the computation capabilities of StreamCI to support streaming data driven ML/AI modeling workflows and effectively mitigate the IoT data processing challenges encountered by the researchers, The AXIN data lake includes (1) a message queue based sensor data ingestion pipeline and repository, (2) a set of REST APIs for data collection and access, (3) a data portal for both users and administrators to manage the datasets and infrastructure, (4) a streaming data engine for real time streaming data processing, (5) integration of GPU resources and NVIDIA Triton inference server at the backend to support ML/AI modeling workflows, and (6) a built-in personal workspace for model development and testing. It has been used by researchers to streamline their data processing tasks and develop a web application for real time building energy data monitoring, prediction, and optimization.

2 ANALYTIXIN DATA LAKE

Motivated by the rapidly developing manufacturing and building energy research as well as the accompanying data and computation challenges, the researchers in this project collaborated and developed a comprehensive cloud platform for not only managing streaming sensor data collected from the field but also enabling researchers to easily access, process, and connect the data into ML/AI modeling workflows.

2.1 System Design

The AXIN Data Lake platform consists of the following main components: first, it has a general purpose, scalable, and secure streaming data management system built based on StreamCI for data collection, preprocessing, storage, and access. The data management system connects to a Flink streaming engine to support real time streaming data processing. The output of the streaming engine can be used to support continuous model training as well as model prediction and data analytics. A model repository is deployed in an Anvil [4] object storage and managed by the NVIDIA Triton inference server for standardized AI model deployment and execution. Researchers have access to a personal workspace powered by a JupyterHub server with preinstalled StreamCI and ML/AI libraries which makes it easy to access the streaming data sources, configure the streaming data engine, process the data, develop, test, deploy and execute ML/AI models. Finally, the system comes with data access APIs and data analytics libraries and tools which are applications deployed and run in the JupyterHub environment.

2.2 Streaming Data Management

At the core of this data lake infrastructure is a scalable and flexible streaming data collection and processing platform provided by StreamCI. It is a ready-to-use CI solution for streaming data providers to manage their data and for data consumers to easily access the data through facile APIs and a user portal interface. Data owners can use the portal interface to register new data sources, view a sample of the latest ingested data, the number of data records, plot time series, and modify the data table schema, all by themselves. Once a new data source is registered, all the required data pipelines are automatically set up at the back end. A role-based access control mechanism is implemented in StreamCI for users to manage data sharing and access permissions.

StreamCI is developed using an open-source software stack including RabbitMQ, node.js, MongoDB, Grafana, InflexDB, Slack, and HUBzero. The backend of StreamCI is deployed on Purdue's Geddes composable system providing scalable operations using Singularity containers and Kubernetes autoscaling services. There are several groups of container pods responsible for API endpoints, message queues, data processing, query, authentication, database, management, and real-time resource monitoring/alert support. Sensor data in JSON and GeoJSON formats (i.e., Point, LineString, Polygon, and other GeoJSON types standardized in IETF RFC 7946 (https://datatracker.ietf.org/doc/html/rfc7946)) can be ingested into StreamCI via its REST API end points. An Apache Flink streaming engine is integrated with StreamCI to create programmable real-time data processing pipelines on incoming data. The streaming processing capabilities include stream joining, data mapping, aggregation, windowing, filtering, and custom user-defined operations. It enables StreamCI to handle continuous data flows and perform computations on the fly, unlocking new possibilities for real-time analytics, event processing, and data-driven decision-making.

2.3 Computation Environment

A flexible computation environment is seamlessly integrated with StreamCI to support data-driven scientific workflows. This environment consists of two components: a front-end JupyterHub server deployed on the Anvil composable system, and a back-end CPU/GPU resource pool. At the front-end, the JupyterHub server offers a personal workspace for running Jupyter Notebook-based data analytics tools and applications. It is integrated with commonly used data analytics, ML/AI libraries, as well as the StreamCI data API and Triton server API. This integration enables users to leverage a wide range of analytical capabilities and access data from various sources. The flexible back-end CPU/GPU resources can be selectively mounted to Jupyter Notebook containers for authorized users, providing powerful computing capabilities for resource-intensive tasks such as model training and inference. This personal workspace empowers model developers with a seamless

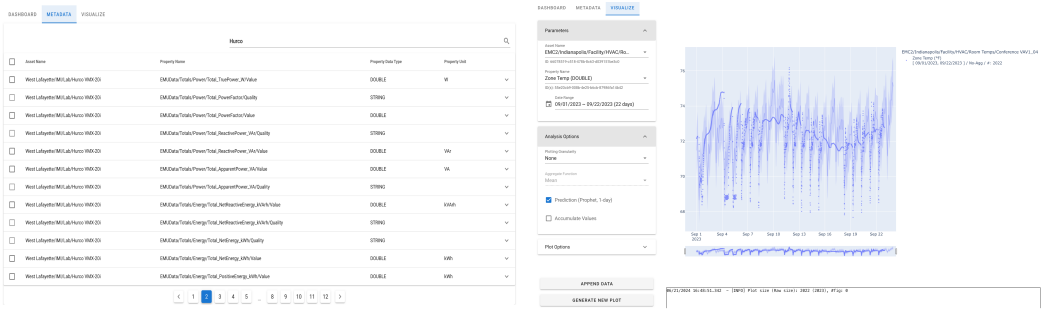

Figure 1: The AXIN dashboard application. Left: A metadata explorer tool that helps user browse and filter the information for hundreds of sensors; Right: A data viewer that allows users to visualize sensor data in different resolutions as well as perform univariate forecast.

computation environment, granting them access to both the IoT data processing pipelines and Anvil's CPU/GPU resources for model development and execution. Moreover, research groups can publish their Jupyter Notebook-based data analytics tools to run in the app mode within this JupyterHub environment for collaboration and sharing of data and analytical results.

2.4 Computation Workflows

The AXIN data lake infrastructure supports four types of sensor data-driven modeling workflows: 1) **Model Development and Testing**: Users can leverage the GPU-enabled Jupyter Notebook environment to develop their own models and validate them using the built-in StreamCI data API. This setup allows for iterative model development and testing against real-time sensor data. 2) **Model Deployment**: Once a model is thoroughly developed and tested, it can be deployed to the Triton Inference Server for inference and production use. The Triton Server provides a scalable and optimized environment for serving models efficiently. 3) **Model Training**: Users can train their models using the StreamCI data API and the powerful CPU/GPU resources of the Anvil system. Additionally, they can leverage the distributed computing capabilities of the streaming engine for automated input data processing at scale and training large-scale models. 4) **Model Execution**: The trained models can be executed on the Triton Inference Server for real-time inference and prediction. The Triton Server exposes a RESTful API, enabling users to integrate their models seamlessly with other applications and services, without the need for GPU resources on the client side.

2.5 API, Analytics libraries, and tools

Several data analytics tools have been developed and published to the JupyterHub hosting environment. They perform a variety of operations such as to help stakeholders monitor their sensor data pipelines, manage data and metadata, and perform data exploration and analysis. As an example, the AXIN dashboard application provides a quick overview of the data managed in the data lake, including information about the facilities, assets and properties, the amount of data received, and the information about the last data received. As shown in figure 1, the dashboard also includes a metadata explorer that enables users to easily query and filter hundreds of metadata attributes of the sensors in the facilities managed by the data lake. Another useful feature of the dashboard is it allows users to dynamically plot and visualize the assets and properties of interest for a specific time range at different temporal resolutions. It also leverages the open-source Prophet library to enable users to generate quick forecasts for univariate time series.

In addition to ready-to-use applications, users can programmatically access the sensor data managed in StreamCI in their models using the StreamCI REST API. They can also interact with the Triton inference server via its REST API for inference and model management requests. These libraries are pre-installed in the AXIN personal workspace in addition to several commonly used data science and ML/AI libraries such as PyTorch and TensorFlow, making it very convenient for the users to develop and execute sensor data driven applications and workflows.

3 BUILDING ENERGY MODEL INTEGRATION AND WORKFLOWS

The science team in this project has been working on the development of two primary models: a recurrent neural network (RNN) based framework for building energy prediction and a reinforcement learning (RL) based approach for building control optimization [5]. With the two models, they can assess various energy conservation measures and refine building control strategies. The development of the AI models encompasses various components, including data collection, data processing, feature extraction and feature selection, cloud computing, AI learning, and others. The accuracy and generality of the AI models highly depend on the availability, quantity, and range of the training data. A typical workflow for model development comprises stages such as data acquisition, storage, retrieval, preprocessing, feature selection, and model training. The researchers used to first upload the sensor data from the smart facilities to some cloud storage service such as AWS, and then download selected data to their local machines for subsequent processing. One notable challenge arises from the continuous generation of building operational data at relatively frequent intervals with some sensors generating data every minute, requiring repeated manual downloads and metadata inspection, which is resource intensive, time consuming, and error prone. Moreover, the

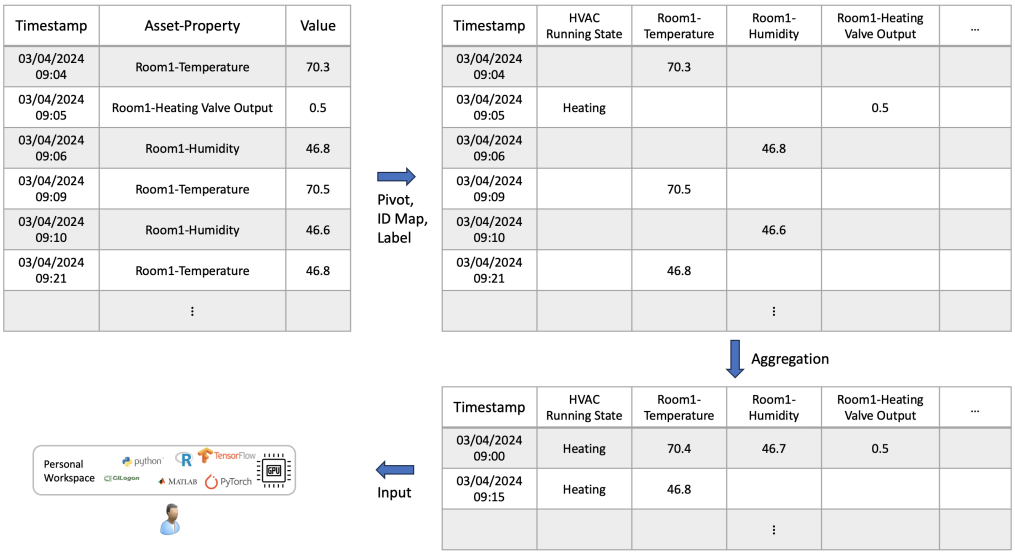

Figure 2: Real time data preprocessing at the streaming engine significantly reduced the workload to prepare the data for ML models.

raw data obtained from the site are heterogeneous in format, resolution, and quality. Extensive data processing is needed to make the data ready for model development. For example, the data needs to be normalized and synchronized since different equipment and sensors update their status at different intervals and use different units. Processing such a large amount of data and training the developed models also demand significant computational resources.

The AXIN data lake provides the researchers an effective solution for automating and streamlining the data processing and modeling tasks. It has been successfully used by the science team in developing and integrating their models to simulate and predict energy consumption profiles for buildings based on sensor data. The IoT data from the sensors (i.e., over 1,000 asset-property pairs) installed in the Emerging Manufacturing Collaboration Center (EMC2), a smart cross-sector manufacture research facility at Indianapolis, were first collected and streamed to a S3 storage on AWS cloud and then automatically ingested into the AXIN data lake using the StreamCI data ingestion API. The input data for the energy prediction and optimization ML models consists of 185 attributes including a combination of HVAC operation data, temperature and humidity readings, weather data, and historical energy consumption data to forecast future energy consumption levels. These data need to be aggregated into 15-minute windows. The Flink streaming engine is configured to automatically process incoming sensor data in real-time, pivoting raw data streams, mapping attribute names, labeling values, and aggregating the data into 15-minute windows, as exemplified in Figure 2. The aggregated data is then fed into the ML models for training and inference. The automated data access and preprocessing operations greatly reduced the data wrangling workload for researchers, allowing them to focus on developing and improving the ML models. Furthermore, the team utilizes the JupyterHub environment to develop, test, and deploy their models and uses the Triton Inference Server to serve the models for real-time predictions. This setup allows the team to streamline the entire model development and deployment lifecycle, from data preprocessing to model serving.

The team successfully deployed their first set of models in the data lake environment and is currently working on optimizing these models to improve the accuracy of energy consumption predictions further. Powered by these models and the data lake infrastructure, a building energy prediction and optimization tool (Figure 3) is currently under development which provides real-time monitoring support for facility energy consumption, carbon emissions, and energy operating costs, as well as the capability to predict future energy usage and identify conservation measures to improve energy performance and reduce negative environmental impacts. The dashboard offers a comprehensive up-to-the-minute view of a site's energy usage and performance, aiding users in understanding environmental factors influencing energy consumption. The prediction tab runs the ML models to forecast future energy consumption and compare it with historical data. The feature to suggest energy conservation measures based on the site's energy usage and performance is still under development. This project demonstrates the potential of combining real-time sensor data processing with machine learning techniques to address real-world challenges in optimizing building energy efficiency.

4 CONCLUSION

AXIN data lake empowers users to streamline the entire lifecycle of sensor data-driven modeling workflows, from development and training to deployment and execution. By combining the capabilities of Jupyter Notebooks, StreamCI, Anvil's computing resources, and the Triton Inference Server, researchers and developers can

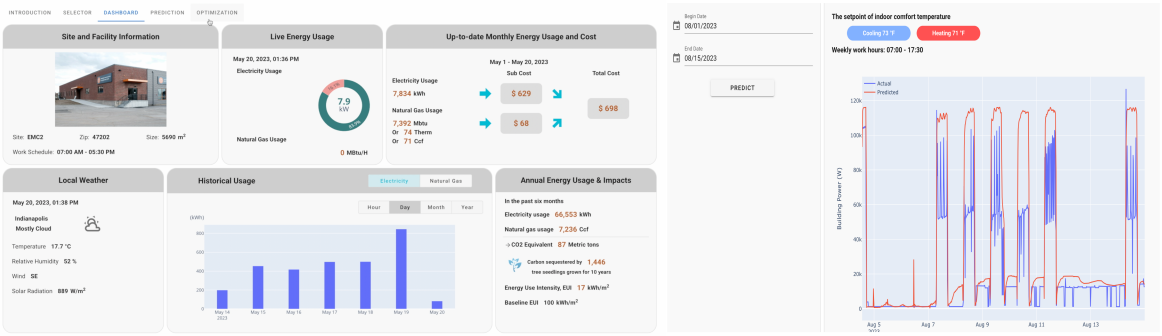

Figure 3: A building energy monitoring, prediction and optimization tool supported by the AXIN data lake.

efficiently build, train, and operationalize their models, leveraging the power of real-time sensor data and high-performance computing resources. The infrastructure provides a general-purpose platform for data collection, storage, streaming processing, and ML/AI modeling for disciplines beyond building energy research. It is deployed within a scalable cloud environment with the API layer ensuring secure access control. Our future plan includes expanding the streaming engine and ML/AI capabilities to allow for easy configuration through a user-friendly web portal, enabling users to manipulate and configurate data processing pipelines without requiring extensive implementation knowledge, enhancing accessibility and usability of the system.

ACKNOWLEDGMENTS

This work was supported in part by a contract from the Central Indiana Corporate Partnership (CICP) under the AnalytiXIN Program and by the National Science Foundation HDR grant #1835822.

REFERENCES

[1] Jaewoo Shin, Lan Zhao, Carol X. Song, Rajesh Kalyanam, Jian Jin, Jacob D. Hosen, Ananth Grama and Dongyan Xu. Enabling Scalable and Reliable Real Time Data Services for Sensors and Devices in StreamCI. Gateways 2022, October 18-20, 2022, San Diego, CA

[2] Sun, Ying & Haghighat, Fariborz & Fung, Benjamin. (2020). A Review of the-State-of-the-Art in Data-driven Approaches for Building Energy Prediction. Energy and Buildings. 221. 110022. 10.1016/j.enbuild.2020.110022

[3] Lu, Chujie & Li, Sihui & Lu, Zhengjun. (2021). Building Energy Prediction Using Artificial Neural Networks: A Literature Survey. Energy and Buildings. 10.1016/j.enbuild.2021.111718.

[4] Carol Song, Preston Smith, Rajesh Kalyanam, Xiao Zhu, Eric Adams, Kevin Colby, Patrick Finnegan, Erik Gough, Elizabett Hillery, Rick Irvine, Amiya Maji and Jason St. John. Anvil - System Architecture and Experiences from Deployment and Early User Operations. Practice & Experience in Advanced Research Computing Conference, July 10-14, 2022, Boston, MA. https://doi.org/10.1145/3491418.3530766

[5] Dikai Xu, Jaewoo Shin, Lan Zhao, Ming Qu, Optimizing Controls of IoT-based Manufacturing Buildings through Deep Reinforcement Learning. 8th International High Performance Buildings Conference at Purdue, July 15-18, 2024, West Lafayette, IN