APS490 Final Report

Designing a Machine Learning Pipeline for Preventive Maintenance of Building HVAC Systems

March 27, 2022

Group Members:

Elizabeth Chelmecki (1004187131)

Anoja Muthucumaru (995825353)

Chi Zhang (1004051250)

Shirley Zhang (1003906213)

Sherry Zuo (1003881711)

Supervisors:

Markus Bussmann
Professor and Chair of the Department of Mechanical and Industrial Engineering

Seungjae Lee
Assistant Professor in the Department of Civil & Mineral Engineering

Executive Summary

Modern Niagara Group (MNG) is a mechanical and electrical contractor that provides building maintenance services in heating, ventilation, and air conditioning (HVAC) systems. Currently, maintenance is conducted manually through building operator calls and scheduled servicing. MNG would like to automate their building servicing process with AI-driven decision-making based on sensor data they collect from the HVAC systems. Specifically, a proof of concept that uses the sensor information to autonomously identify building system anomalies in the CPPIB building is required.

After performing a literature review and examining the available datasets provided by MNG, the team selected the Air Handling Central Unit (CU) and the Variable Air Volume (VAV) system for demonstration of proof-of-concept. By taking an in-depth look at the system components as well as their relationships through exploratory data analysis (EDA) and domain knowledge, we were able to identify two machine learning models: Valve Position and Air Flow. The chosen solution was a digital twin, a data-driven representation of the VAV and CU. The digital twin is generated through machine learning predictions. These predictions are then fed into an anomaly detector to forecast potential failures ahead of time.

We trained the two models using Vector Autoregression (VAR), Long Short Term Memory (LSTM), and Gated-Recurrent Unit (GRU) models. We compared the results using Mean Squared Error (MSE) and Mean Absolute Error (MAE). During this process, we also tested possible inputs not included in the dataset that could potentially improve the accuracy such as future weather forecasts. Following the development of the prediction model, the team developed an anomaly detector that took in the prediction output as well as temporal and weather data. Two unsupervised learning methods, Isolation Forests and Histogram Based Outlier Selection, were used to identify outlier sensor values that could indicate future building failures. Through evaluation of model results, we identified areas of weakness in the machine learning predictor and anomaly detector that could be improved with additional data and input features. As a proof of concept, the model is successful in meeting the client's requirements of autonomy, efficiency, reliability, generalizability, and modularity.

The design team was able to produce a valuable predictive tool for MNG and a functional UI. The developed ML pipeline will improve the operational performance of the target system with continued development, boosting the comfort of building residents and mitigating costly repair delays. When MNG starts collecting more time-series data and documenting service calls that align with the existing indices, they can tune the proof-of-concept model for further accuracy and reliability.

1.0 Introduction

Modern Niagara Group (MNG) is the largest mechanical contractor in Canada with a strong presence in commercial real estate as well as government, healthcare, and public infrastructure projects [1]. As a contractor for premium projects, MNG aims to maintain a competitive advantage through offering advanced building solutions. Particularly, the firm is expanding its integrated building technology (IBT) services, which aid in automating the control of HVAC systems. MNG hopes to drive additional value and elevate the customer experience through intelligence predictive maintenance (IPM). Thus, MNG has consulted our multidisciplinary capstone team to leverage data-driven decision making and build an in-house IPM pipeline.

2.0 Problem Statement

MNG provides regular maintenance to their customers via a scheduled service approach. For other maintenance issues (i.e. abrupt faults and degradation faults), MNG is reliant on the customer's local building operator to recognize issues and place a call to MNG's 24/7 dispatch for reactive service. This process is shown in Figure 1. This process includes manual fault diagnosis, which requires equipment downtime and can negatively affect building occupants' comfort. This may result in delays in regular building activities inside office towers, hospitals, and manufacturing facilities. As such, maintenance events are costly to MNG and detrimental to the satisfaction of their customers.

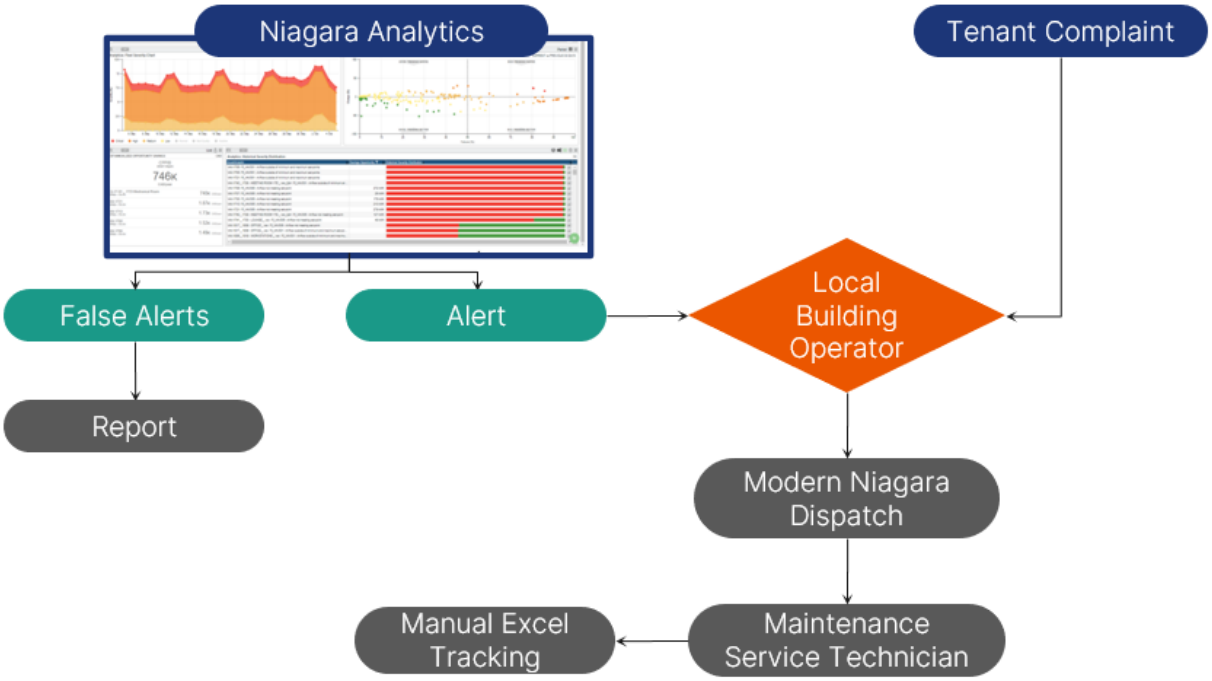


Figure 1: Current MNG Approach to Tenant HVAC Complaints

Currently, MNG employs an analytics software to track metrics from the various systems in their customer's buildings. As such, there are a multitude of system parameters being tracked and stored by MNG that are not being utilized for any function other than monitoring. These system parameters provide valuable information about the status of the building system in real-time. Before a fault occurs, the system parameters often exhibit abnormal relationships relative to each other [2], indicating an impending service need. Predicting these emergency maintenance events ahead of time, while the fault is in the early stage, would allow MNG to preemptively address the issue, save money through optimized repair scheduling, and improve service.

There are currently no available solutions on the market that achieve predictive maintenance. A reason for that is building systems are not homogenous, and HVAC systems are complex (especially in large commercial buildings) requiring extensive engineering effort to develop a digital representation of the interconnected physical systems [3]. Therefore, MNG has proposed a capstone project to establish an in-house, data-driven predictive building-maintenance pipeline for HVAC systems using. Specifically, the client requires the team to develop a proof-of-concept that the current sensor information can be applied to identifying potential failures before they occur, without manual intervention.

3.0 Service Environment

The team has narrowed the scope to two interconnected systems: the Air Handling Central Unit (CU) and Variable Air Volume (VAV) system. Failure of the CU and VAV can cause the indoor temperature to fluctuate and negatively impact the thermal comfort of building occupants. Thus, being able to detect anomalies in the operational state of the mechanical components as well as system outputs such as airflow provides valuable information on system performance. Representing the data behaviour and relationships in two complex and interconnected systems effectively demonstrates the feasibility of data-driven predictive maintenance.

The following section will detail the CU and VAV system, the physical and digital environments in which these systems operate, and how they interact.

3.1 The Physical Environment

The building the team is focusing on is One Queen Street East, pictured in Figure 2, which houses CPPIB offices on the 8-27th floors [4].

Figure 2: One Queen Street East, Toronto, ON M5C 2W5

High occupancy times are likely between the hours of 8 AM and 5 PM. The HVAC system likely is more active during these hours as it responds to the more frequent changes in temperature and humidity from the heat generated by people and electronics.

Additional information about the physical space that the HVAC system services, such as the layout and use cases of different office zones on the seventeenth floor, is unavailable to the team. However, this did not pose any problems to the initial design because any insights obtained from this information can also be obtained from mapping the data provided by MNG (such as how temperature varies throughout the day).

3.2 The Digital Environment

The physical components of the HVAC system are integrated into a digital environment through a building automation system (BAS).

The BAS utilises a control system to:

1. Automate the control of building climate and lighting based on occupancy schedules
2. Monitor the performance of the devices in all the system
3. Provide alarms in case of malfunctions [5]

To automate the control of building climate and monitor the performance of the physical devices in the HVAC system, the BAS collects data from multiple sensors, as well as tracks setpoints and commands which are prescribed by either the building manager or MNG [5].

The team utilized this data in order to create a predictive model to meet the functions and requirements of the project. Specifically, the team used data collected from the CU and 34 VAV

boxes on the seventeenth floor. The control schematics for the seventeenth floor are shown in Figure 3.

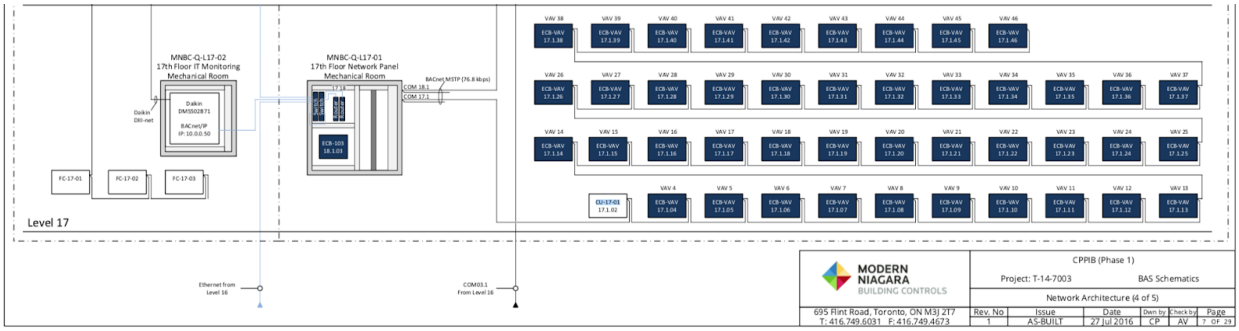


Figure 3: BAS Control Schematics for the seventeenth floor of the CPPIB building

The dataset's variables and meanings are detailed in Table 1. The variables are also classified based on if they are measurements, commands, or setpoints. Measurements are actual values for variables measured by sensors, setpoints are desired or target values for variables, and commands are values that physical components are set to in order to achieve setpoints.

To exemplify the differences between these types, consider the damper command variable. The damper command is a command for the physical position of the damper. It is changed to achieve the zone temperature setpoint, and is determined by the BAS by comparing the actual measured zone temperature with the zone temperature setpoint.

Variable	**Definition**	**Measured/Command/Setpoint**
DateTime	Date and time of the variables recorded	Measured
Occupancy	Boolean indicating if space is occupied	Measured
Unit State	The state of the system	Command
Direct Expansion Cooling. Condenser Water Temperature	Temperature of the condenser water leaving the system (after absorbing heat)	Measured
Direct Expansion Cooling.Cooling Capacity	The measure of how much heat the system can remove	Command
Direct Expansion Cooling.Enable	Enables direct expansion cooling (i.e. air handling unit will operate)	Command

Discharge Air.Static Pressure	The static pressure inside the duct system	Measurement
Discharge Air.Static Pressure Setpoint	The target static pressure inside the duct system	Setpoint
Discharge Air.Temperature	The temperature of air leaving the air handling unit	Measurement
Discharge Air.Temperature Cooling Setpoint	The set temperature of air inside the air handling unit	Setpoint
Discharge Air.Temperature Setpoint	The set temperature of air inside the air handling unit	Setpoint
Mixed Air. Temperature	Temperature of the air in the return loop of the air handling unit after mixing with outdoor air	Measurement
Return Air.Temperature	Temperature of the air in the return loop of the air handling unit before mixing with outdoor air	Measurement
Supply Fan.Speed	Speed of the fan blowing air over the coil	Measurement
Supply Fan.Status	If the supply fan is off or on	Measurement
Fin Tube Radiation. Valve Position	Position of the fin tube valve in each VAV box	Measurement
Primary Air. Air Flow	Airflow out of the damper	Measurement
Primary Air.Air Flow Cooling Maximum Setpoint	The maximum amount of airflow allowed through a damper for cooling	Setpoint
Primary Air.Air Flow Cooling Minimum Setpoint	The minimum amount of airflow allowed through a damper for cooling	Setpoint
Primary Air.Air Flow Heating Maximum	If the heating required the maximum airflow setpoint is 480.	Setpoint

Setpoint [cfm]		
Primary Air.Air Flow Heating Minimum Setpoint [cfm]	If the heating is kept to a minimum the airflow is set to 144.	Setpoint
Primary Air.Air Flow Setpoint [cfm]	Setpoint for airflow dependant on the temperature required in each zone	Command
Primary Air Damper.Damper Command	Position the damper is set to	Command
Primary Air Damper.Damper Position	Position the damper is actually in	Measurement
Zone Air.Temperature [$\circ\ C$]	Actual temperature of zone	Measurement
Zone Air.Temperature Cooling Setpoint [$\circ\ C$]	Temperature setpoint in cooling mode	Setpoint
Zone Air.Temperature Heating Setpoint [$\circ\ C$]	Temperature setpoint in heating mode	Setpoint

Table 1: Description of variables in the dataset provided by MNG

3.3 Interactions

The digital BAS system directly interacts with the physical HVAC system. The BAS maintains the various setpoints and commands set by either the building manager (such as the cooling temperature setpoint) or the designer of the system (such as static air pressure inside the AHU). The BAS uses sensors to measure how different variables are from their setpoints. It then controls various physical components in the system to bring these variables to their setpoints.

4.0 Stakeholders

Table 2 shows the agencies and stakeholders that are influenced by and have an influence upon the design for data-driven prediction of potential service requests.

Stakeholder	Impact

Government (General Impact: Governing bodies must approve that the design satisfies legal regulations)	
City of Toronto	● If the design requires mechanical components to be added to existing buildings, any updates must comply with City of Toronto regulation. [2]
Province of Ontario	● Collection of building sensor data must comply with guidelines set out by the Information and Privacy Commissioner of Ontario [6][7]
Modern Niagara Group	
IntegratedBuilding Technology (IBT) Team (The team in MNG who is directly in communication with the capstone group)	● The design contributes to their goal of achieving more autonomous buildings, as a result, this team will be supplying all the required data for the project. ● The design could generate additional value/revenue for this team ● The design could modify the current database, software pipeline, and workflow of the team ● The design could save valuable time in the IBT team's day-to-day workflow
Building Service Technicians	● The design aims to optimize the scheduling of repairs for building service technicians ● The design will help technicians narrow down on problems and save valuable time and manpower.
Service Dispatchers	● Instead of receiving phone calls, dispatchers will be alerted of service requests and building failure through the designed system ● The design could increase the efficiency of service dispatching ● Dispatchers will need to learn how to interact with the system
People serviced by MNG	
Clients of MNG (Ex. CIBC, Shopify, Government, etc.)	● The design could provide corporate/institutional clients with an elevated building experience, in which the building is more automated ● The design could reduce costly delays caused by emergency repairs ● The design, after deployment, will collect and monitor the sensor data at client buildings
Occupants of buildings	● The design could make the buildings more comfortable

(Ex. Office workers, hospital staff, etc.)	for occupants through preventing the building parameters from entering extreme ranges by predicting them ahead of time
Capstone Team	
Capstone Team	● The team, consisting of five students from different faculties and disciplines, are responsible for designing the project according to the project requirements ● Each student is expected to exercise knowledge from their respective area of study, which is outlined in section 10.3 ● The role of each student in the team is in section 10.3 ● The design can help further the students' industry knowledge and develop their technical skills
Faculty Advisors: Professor Markus Bussmann Professor Seungjae Lee	● Faculty advisors will be consulted for guidance throughout the design process and course deliverables ● Professor Lee specializes in machine learning applications to energy systems, and will be the expert advisor for technical knowledge ● The design can help advisors understand the workflow of companies like MNG and as a result enhance the usability of their research findings

Table 2: Stakeholders affected by IPM design

5.0 Detailed Requirements

5.1 Functions

Representing the physical systems is possible, but since each building is different, solutions are not easily transferred from building to building [8]. Additionally, as a building ages, the systems within that building will degrade and the baseline operating conditions will change [8]. Thus, machine learning methods are appropriate to the predictive maintenance task, as the models are able to dynamically learn component relationships and patterns over time.

Ideally, the output of the predictive maintenance model would be failure classification. However, time-series data on previous system failures that align with sensor data is unavailable during the design, and directly training the model to detect and classify failure is not feasible. Thus, the team has opted to achieve anomaly detection through creating a digital representation of the building VAV and CU system through a digital twin.

A digital twin [9] takes sensor information from core operational components of the building system and learns the system behaviour. It can then be used to generate predictions of how the systems *should* function. Thus, MNG and the building operators can use the predicted system parameters as guidelines to assess whether the physical system may have impending faults. Beyond anomaly detection, digital twins can generate additional future value - the MNG could use the model to run simulations and identify areas of optimization.

5.1.1 Functional Requirements

The designed pipeline processes AHU and VAV sensor data from the 17th floor of the CPPIB building to predict system outputs and identify impending system faults. The predicted system outputs from the model represents what the AHU and VAV system on the 17th floor should look like at a certain time in the future. These predicted outputs will be compared in real-time to the measured system parameters in order to identify any anomalies or inefficiencies.

The model's outputs will create a digital representation of the physical system using previous operating data. If there is a significant discrepancy between the measured and predicted system parameters, the design will send an early warning to the user, which aids in scheduled building maintenance. The design is also able to display ongoing metrics. The functionalities are illustrated in Figure 4.

Functional Basis: Identify impending system faults

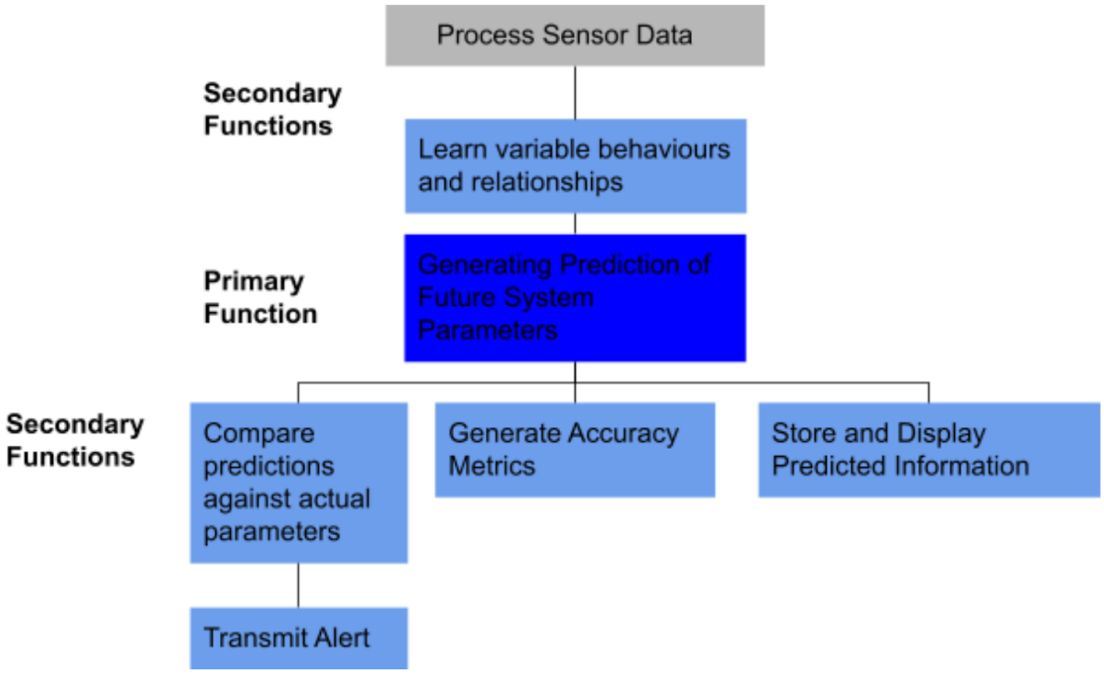


Figure 4: Functional Breakdown of design

5.2 Objectives

The following design objectives were determined based on the MNG's needs. MNG considers autonomy as their top priority in order to reduce manual work performed by building operators and dispatchers. MNG also values reliability which directly impacts the quality of service they will deliver to their customer. In addition, MNG wants an efficient, generalizable, and modular algorithm that can be trained and deployed in a short time frame as well as adapt to different customer sites.

5.2.1 Autonomous

- The design will generate predictions based on input data without supervision.

5.2.2 Reliable

- The design will generate real-time predictions that are reasonably close to the actual operational state of the physical system.
    - The design will achieve 90% training accuracy at the minimum [3].

5.2.3 Computationally efficient

- The design will consume the minimum required computational resources and within available computational resources.
    - The design will require at most 16GB RAM [10] to train and make predictions which can be achieved on standard personal computers.
- The design will generate predictions at the same frequency as data input.
    - The design will make predictions at 5-minute intervals (same as input data).
- The design will be able to be re-calibrated as new historical data becomes available.
    - The design will complete training during company off-hours (overnight or weekend).

5.2.4 Generalizable

- Given the sensor data and specific building characteristics, the design will have the capacity to model different buildings with similar performance accuracy.

5.2.5 Modularity

- The design can be isolated into separate components that are tested individually. The failure of one component will not impact other components.

5.3 Constraints

As the design operates across residential and commercial buildings, it must adhere to all relevant bylaws and building codes enforced federally and city-wide in Toronto. It must also follow specific policies imposed by building management agencies and occupants.

5.3.1 Legal Regulations

The design shall comply with all building maintenance standards enforced by the City of

Toronto. For example, the design should be compatible with the State of Good Repair (Capital) Plan which requires landlords to provide a five (5) year forecast of major building repairs. The design shall also adhere to technical requirements for the energy-efficient design and construction of high-rise towers set out by The National Energy Code of Canada for Buildings (NECB) [11].

5.3.2 Security

The design shall follow all security protocols of the existing systems it will be integrated with. The design shall not introduce any security vulnerabilities in the network systems it is mounted on [7].

5.3.3 Data Collection

MNG should be informed of and consent to any data collected as a part of the design. The data collected from the client shall only be used for the purpose of the design and made available to MNG upon request. Any intended release of data to third parties shall be approved by MNG.

In addition, the design must respect the privacy of unit occupants, under the guidelines of The Personal Information Protection and Electronic Documents Act (PIPEDA) [12]. Any data collected from open-source databases should also comply with the Freedom of Information and Protection of Privacy Act [7].

5.3.4 Affordable

The cost of implementing, deploying, and maintaining the design shall not exceed the cost of the current platform and service request process (i.e. the project will generate a non-negative return on investment).

5.3.5 Compatible

The design shall be compatible with the existing data acquisition system. The format of the given data shall be used as inputs to the design without the need for alterations.

6.0 Data Preparation

Prior to designing any machine learning model, it is essential to first take a close look at the data before making any assumptions. With the help of summary statistics and graphical representations, the team performed initial investigations on the sensor data, weather data, and DateTime data to discover patterns, spot anomalies, test hypotheses, and check assumptions.

6.1 Exploratory Data Analysis (EDA)

6.1.1 HVAC sensor data

EDA is primarily used to see what data can reveal beyond the formal modeling or hypothesis testing task and provides a better understanding of data set variables and the relationships between them. The HVAC systems data provided by MNG contains 17 Excel files with 617 variables each. These variables capture sensor data from 1 CU and 42 VAV boxes on the 17th floor of the CPPIB building over a 10-month period as shown in Figure 5. Because the VAV boxes are placed at different locations with varying characteristics, data from individual VAV boxes will be used to train the model.

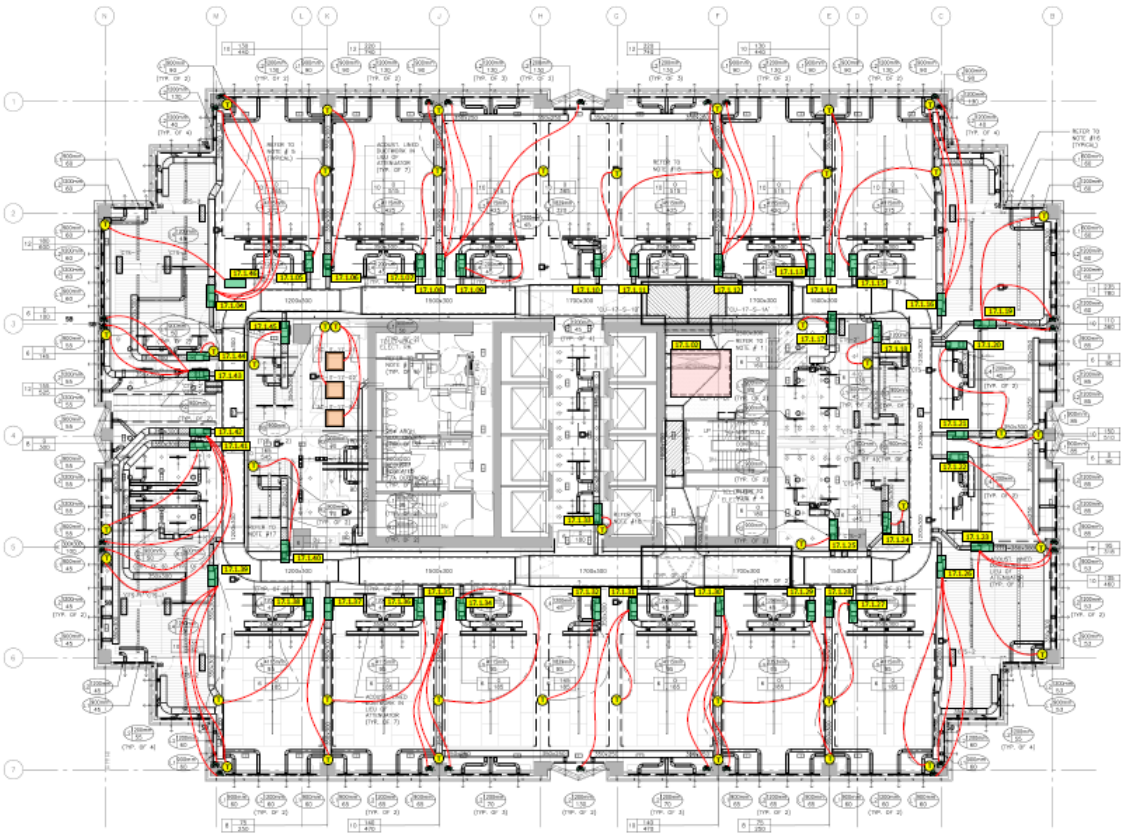


Figure 5: 17th Floor Mechanical HVAC layout for CPPIB Building

The team performed multiple statistical functions and techniques with EDA tools. For example, the team created univariate visualization of different fields in the raw dataset. Particular attention was paid to the output variables (Airflow and Damper position) as shown in Figure 6. It can be observed that the airflow data hovers around 200 but frequently fluctuates between 0 and 600. On the other hand, the damper position is fixed at 100 for the majority of the time with occasional drops. This highly irregular behaviour is likely a result of PID controller tuning and changing temperature conditions which will be further explained in the prediction section.

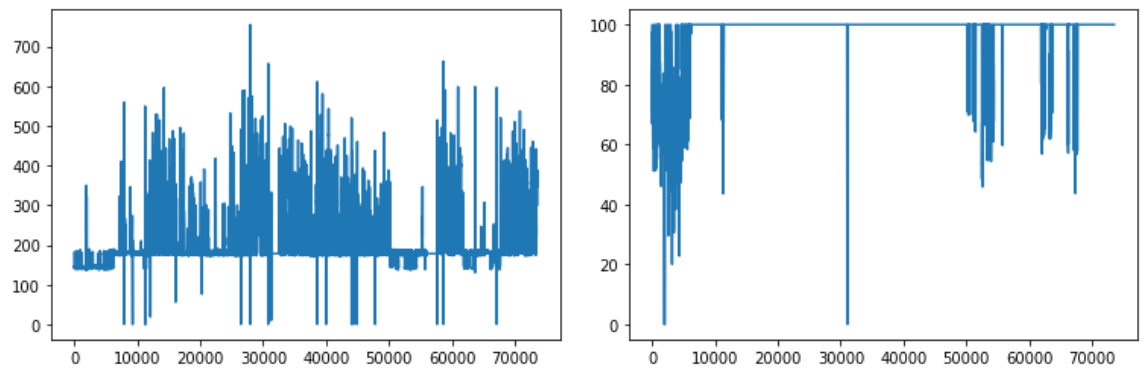


Figure 6: Airflow (left) and Damper Position (right) vs Time

In addition to univariate visualization, the team also used bivariate visualizations to assess the relationship between each variable in the dataset and the target variable. For example, because the air flow setpoint largely determines the actual air flow, they are plotted together in Figure 7. It can be observed that although air flow follows relatively closely with its setpoint, there are some periods of deviation that the team paid attention to in the model training step.

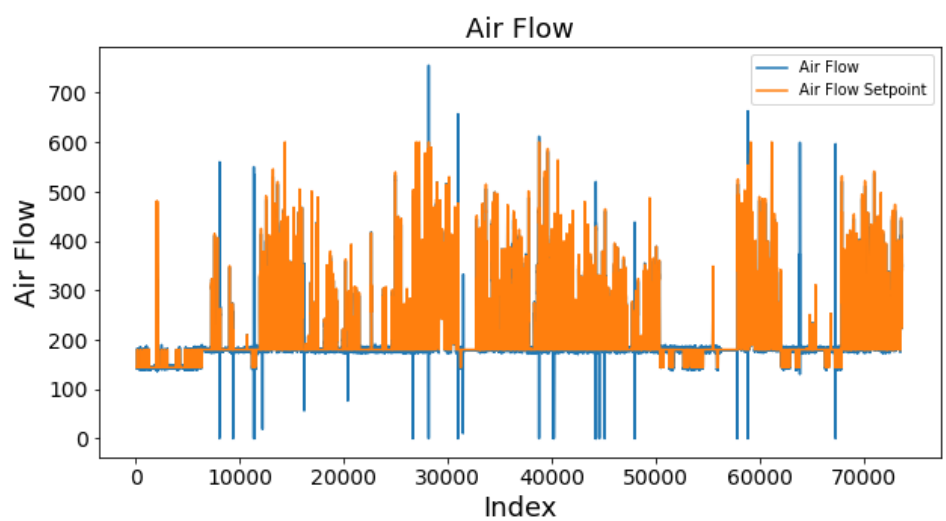


Figure 7: Airflow and airflow setpoint

6.1.2 Weather Data

In addition to sensor data, the team used web scraping techniques to download weather data for Toronto City Centre from the government of Canada website [13]. This dataset is available in hourly, daily and monthly intervals, and it contains a large variety of variables including temperature, humidity, precipitation amount, and wind speed. Because hourly temperature is the most significant parameter affecting HVAC system performance, the team extracted hourly temperature readings for further processing while discarding all other fields.

With access to weather data, the team took a closer look at temperature-related parameters. This was achieved using multivariate visualizations to map and understand interactions between different fields in the data. For example, the relationships between discharge air temperature, mixed air temperature, and return air temperature within the HVAC systems can be visualized in Figure 8. The interaction between the system (indoor) air temperature and outdoor air temperature can be observed in Figure 9. It can be shown that the indoor spaces are controlled by the VAV boxes to stay at a relatively constant temperature regardless of outdoor conditions. Similarly, multivariate visualization was performed on the damper data to show the relationship between air flow, damper position, damper position command, and fan speed. This is shown in Figure 10.

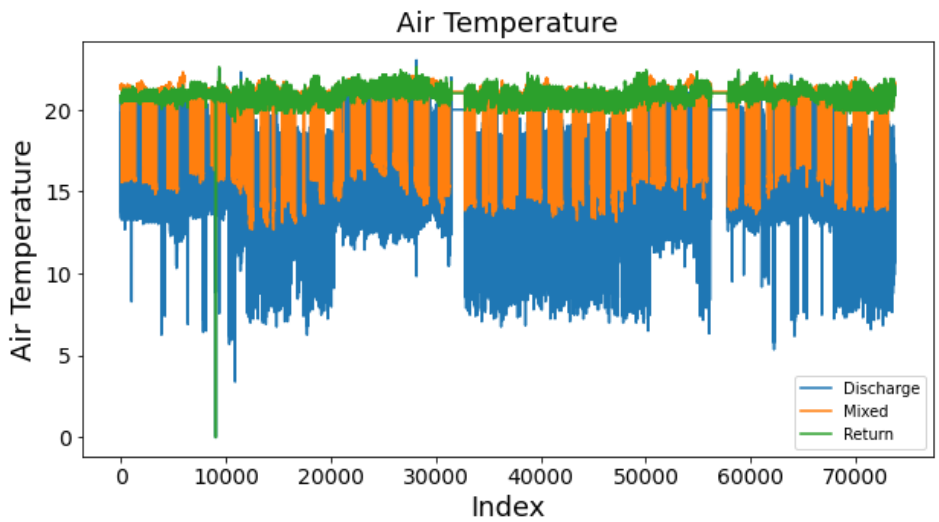


Figure 8: Relationship between discharge air temperature, mixed air temperature, and return air temperature

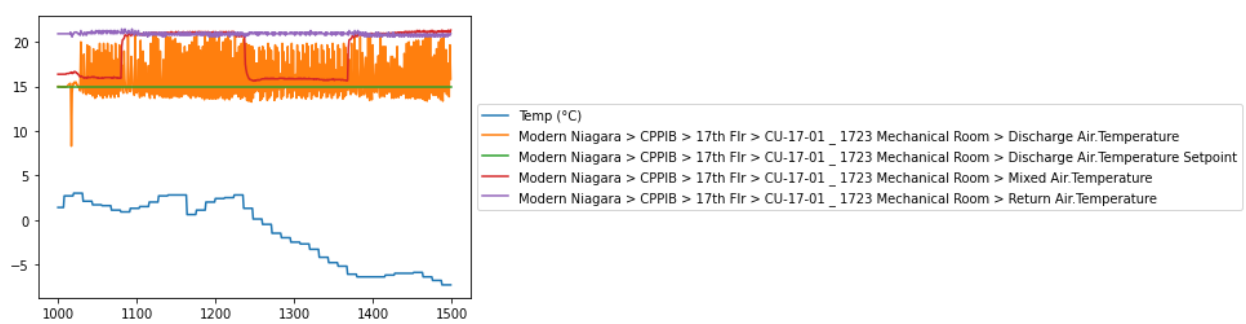


Figure 9: Relationship between system (indoor) air temperature and outdoor air temperature

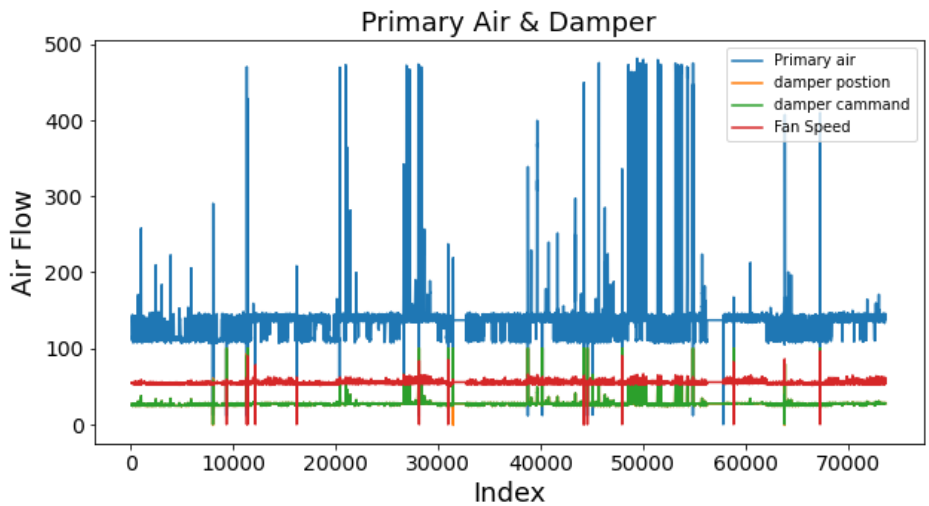
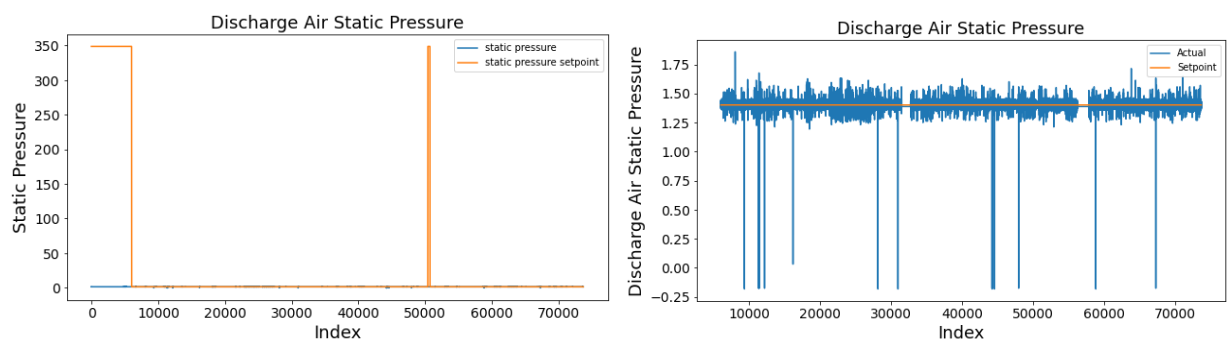


Figure 10: Relationship between air flow, damper position, damper command, and fan speed

6.2 Data Cleaning and Wrangling

Now that the team has a better understanding of the data, the next step was to perform data cleaning on the raw datasets. After communicating with the client regarding system setpoints, the team identified and removed outliers that are caused by human intervention or sensor error which are not part of the normal system behaviour. For example, the data points with an abnormally high static pressure setpoint of 350 shown in Figure 11a were removed, and the resulting figure is shown in Figure 11b. Similarly, the data points with a cooling and heating setpoint of 0 was removed from Figure 12. After confirming with the client, the team believes that these unrealistic setpoints are artificial and do not represent true system behaviour. Similarly, the 0 reading in zone air temperature as shown in Figure 13 is likely a result of sensor error because it is physically impossible for the temperature to drop to zero in such a short time.



(a) (b)

Figure 11: Removal of extreme outliers in static pressure setpoint

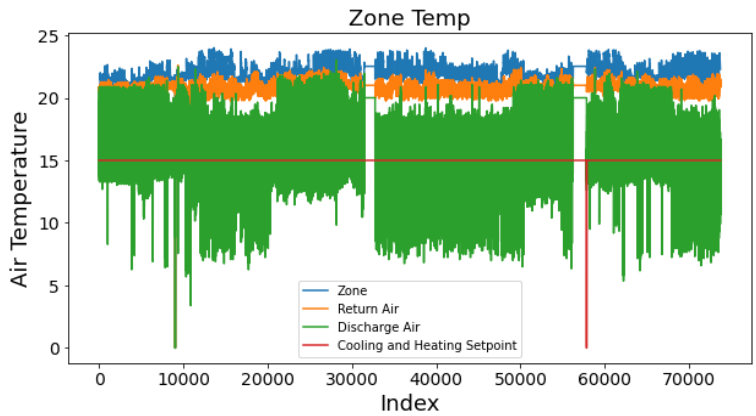


Figure 12: Extreme outliers in cooling and heating setpoint

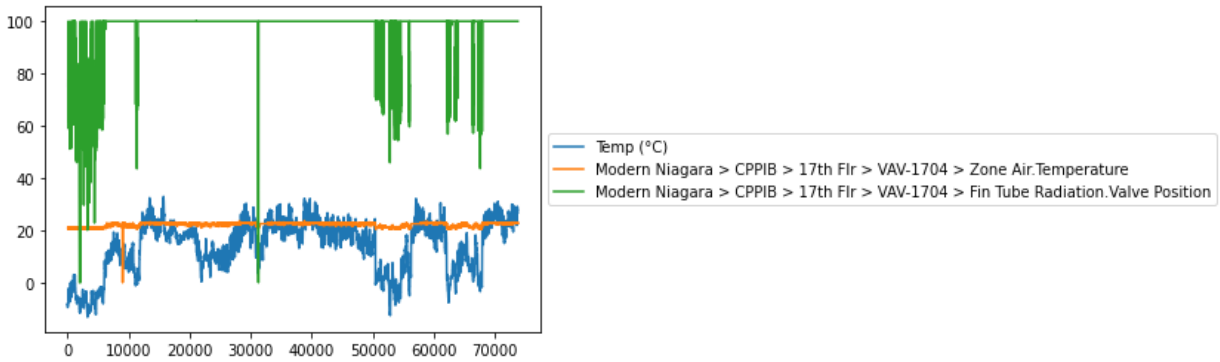


Figure 13: Extreme outliers in zone air temperature

On the other hand, the team also observed that there is some discrepancy between airflow and airflow setpoint (Figure 14). After taking a closer look at the airflow data and plotting it against damper position as well as static pressure (scaled), it appears that the occurrences of 0 airflow coincide with changes in damper position and near 0 static pressure (see Figure 15 and Figure 16). Therefore, these occurrences are systematic errors because they represent a specific state of the physical system (such as a system shutdown) rather than something random such as sensor malfunction. As a result, the team chose to keep these zero values as opposed to discarding them. Finally, after data cleaning, the sensor data is merged with the weather data into a single Dataframe.

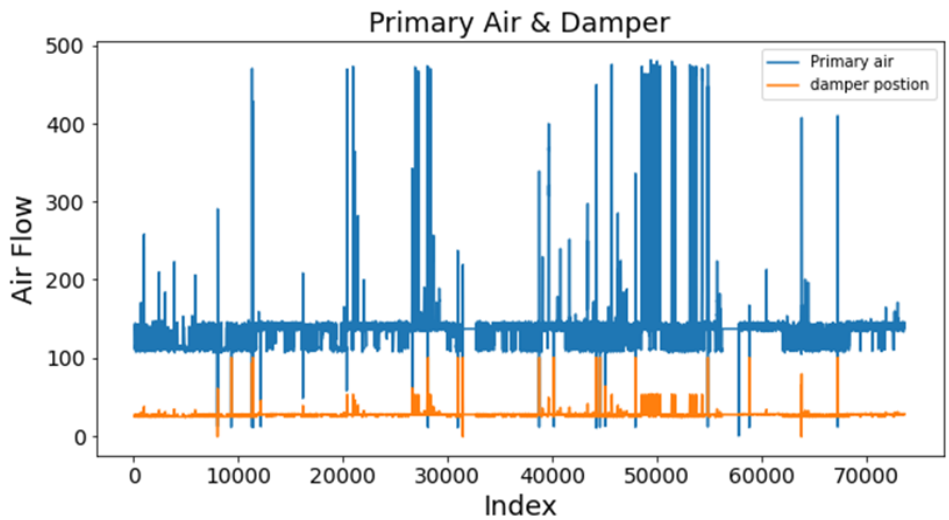


Figure 14: Air flow vs damper position

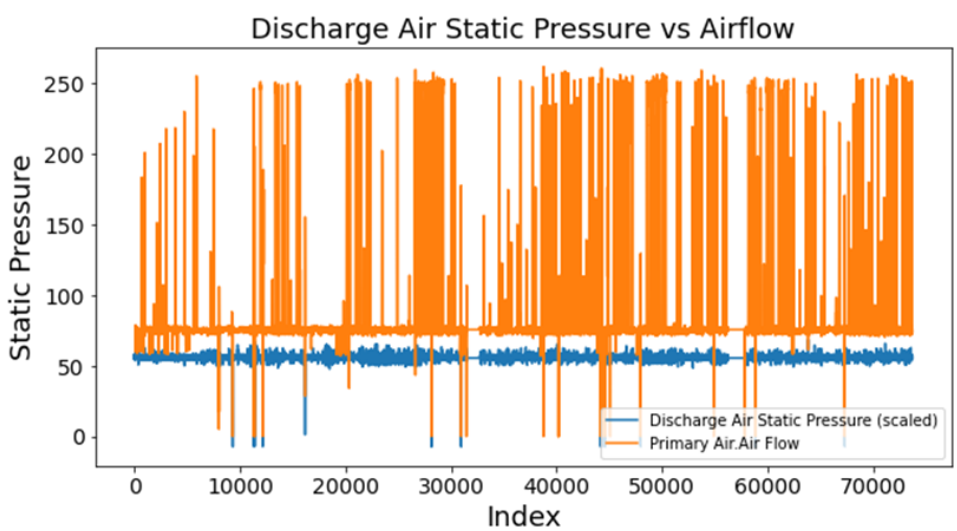


Figure 15: Air flow vs. static pressure

6.3 Model and Feature Selection

To create a digital representation of the AHU and VAV boxes, the values of variables that represent the state of the system must be predicted. The team considered all variables in the dataset provided by MNG to decide which to predict in order to create the digital twin representation of the system.

The variables that were identified as setpoints, such as *Discharge Air.Temperature Cooling Setpoint* and *Zone Air.Temperature Cooling Setpoint,* were automatically disregarded as the value of these variables at a given time is known and thus there is nothing to predict. Other variables,

such as *Supply Fan.Speed* and *Primary Air Damper.Damper Position*, were disregarded as the timestep of the dataset (5 minutes) is too large to generate accurate predictions for these variables. This is because these variables change continuously in response to other variables. For example, *Supply Fan.Speed* changes continuously to maintain the static pressure in the CU. Some variables, such as *Discharge Air.Static Pressure,* had little variation historically and matched the setpoint. As a result, there would be little value in predicting these variables.

The team arrived at two variables that held merit for prediction, *Primary Air. Air Flow* and *Fin Tube Radiation.Valve Position.* The prediction of these variables can be separated into different tasks. The following section details the methodology of feature selection. Each task and its inputs and outputs are then explained.

The inputs and outputs considered are those that exist in the datasets provided by MNG. Thus, if there is an input that affects an output specified by one of the tasks below, but this input is not tracked by MNG, it is not discussed. The designed solution accounts for these missing variables.

6.3.1 Feature Selection Methodology

When determining which input variables to use for each task, the team utilized domain knowledge and the Spearman Correlation Coefficient. The Spearman Correlation Coefficient measures the strength and direction of association between two variables [14]. It is a number ranging from -1 to 1 [14]. A Spearman Correlation Coefficient close to 1 indicates a very strong positive relationship between two variables, while a Coefficient close to 0 indicates no or a negligible relationship [14]. A heat map of the Spearman Correlation Coefficient for the variables in the dataset is shown in Figure 16. For both *Primary Air. Air Flow* and *Fin Tube Radiation.Valve Position,* the team used the Spearman Correlation Coefficient to determine the relationship between the variable in question and all other variables in the MNG dataset. For variables that had high Spearman Correlation Coefficients, the team used domain knowledge to determine if there was a causation relationship between the variables or if it was only a correlation relationship. Domain knowledge was also used to identify any other variables in the dataset that would be useful as inputs.

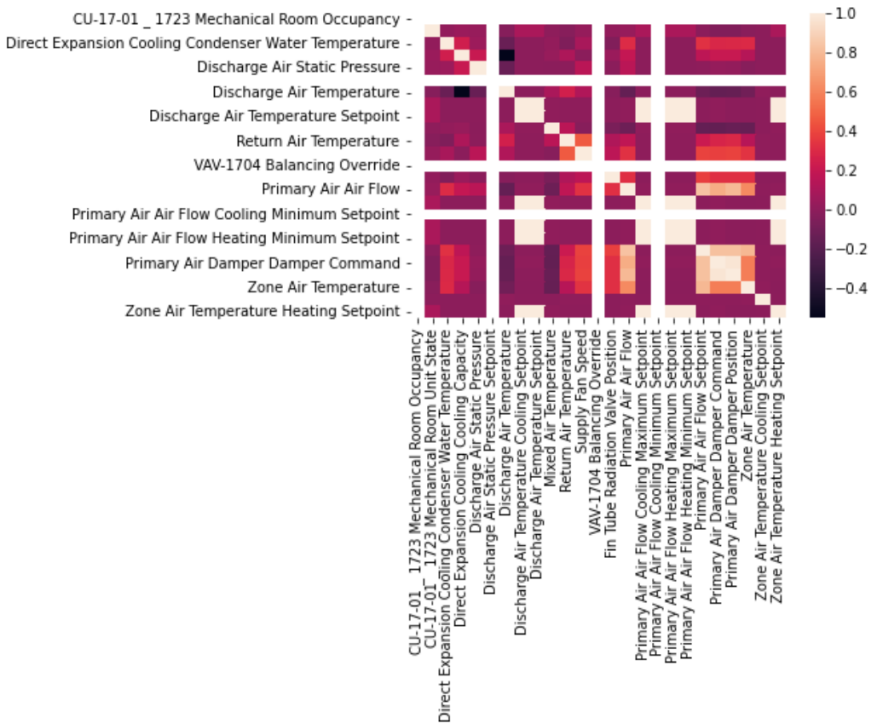


Figure 16: Spearman Correlation Coefficient Heat Map

6.3.2 Fin Tube Radiation.Valve Position Prediction

The fin tube functions to increase the temperature of air leaving each VAV box. There is a fin tube located in each VAV box, so the temperature of the air from the CU can be increased to meet the local heating/cooling demands of the zone each VAV box services. Because the air leaving the CU is at a constant temperature (15 degrees), if a room requires heating, the fin tube will be enabled. The valve position of the fin tube determines the amount of heating applied to the air leaving each VAV box, and is a value ranging from 0-100.

The Spearman Correlation Coefficient between *Fin Tube Radiation.Valve Position* and all other variables in the MNG dataset was calculated and analyzed. Table 3 shows the variables that are highly correlated to the valve position.

	Fin Tube Radiation. Valve Position	Air Flow	Air Flow Setpoint	Zone Air Temperature	Outdoor Temp
Valve Position	1.000	0.277	0.278	0.553	0.597

Table 3: Spearman Correlation Coefficients for Valve Position Inputs

Zone Air. Temperature and *Primary Air.Air Flow* are highly correlated to the *Fin Tube Radiation. Valve Position*. This supports domain knowledge about fin tubes. The temperature of the air leaving the VAV (determined by the valve position) is controlled by a feedback loop that takes into account the current zone air temperature and desired zone air temperature. The air flow out of a VAV box is also determined by this feedback loop.

In addition to system-related parameters, the team further aggregated the original timescale of 5-minute intervals into hour of day, day of week, and week of year, then used them as separate input parameters for training. The inclusion of different timescales allowed the model to capture both global trends such as seasonal weather conditions, as well as local trends such as the level of occupation throughout the day.

In terms of the weather feature, the team further divided outdoor temperature into current temperature and temperature difference in the next 2 hours (according to temperature forecast). The team believed that the inclusion of weather forecasts allows the model to further capture system responses to outdoor temperature change and make more accurate predictions.

A summary of the inputs for the Valve Position model are shown in Figure 17.

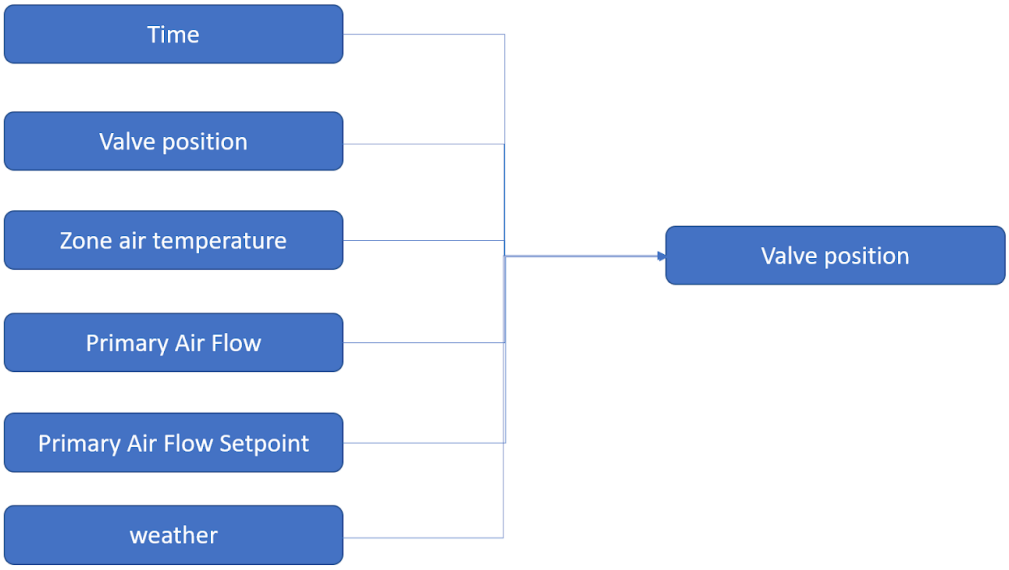


Figure 17: Valve Position Inputs

6.3.3 Primary Air.Air Flow Prediction

If the static pressure matches the static pressure setpoint, *Damper Position* is the only factor affecting *Primary Air Flow*. As a result, predicting air flow is sufficient instead of predicting both air flow and damper position.

The Spearman Correlation Coefficient between *Primary Air.Air Flow* and all other variables in the MNG dataset were calculated and analyzed. Table 4 shows the variables that are highly correlated to the valve position.

	Condenser Water Temp	Cooling Capacity	Return Air Temp	Supply Fan Speed	Valve Position	Air Flow	Air Flow Setpoint	Damper Command	Damper Position	Zone Air Temp
Air Flow	0.3038	0.1944	0.1816	0.3294	0.3260	1	0.8335	0.7507	0.7982	0.6225

Tabel 4: Spearman Correlation Coefficients for Air Flow Inputs

Some inputs highly correlated to air flow that the team did not consider initially when relying only on domain knowledge for feature selection are *Condenser Water Pressure*, *Return Air Temperature*, *Supply Fan speed*, and *Value Position*. The first three parameters will not directly affect airflow and merely show correlation, because there are confounding variables such as outdoor temperature that affect both these input variables and airflow. Since causality is what we need to construct a model for fault detection and diagnosis, we eliminated these variables in our model. In addition, it is worth noting that we excluded *Cooling Capacity* because it contains many 0 labels, which led to numerical instability in the model. Furthermore, similar to the valve position model, aggregated datetime parameters at the hourly, weekly, and monthly level are included as features. The final input parameters for the air flow model are shown in Figure 18.

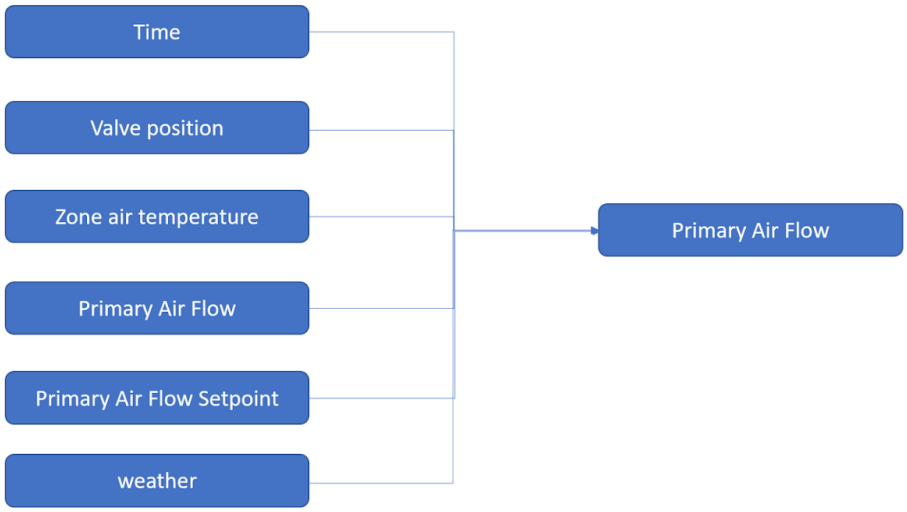


Figure 18: Air Flow Model Inputs

7.0 ML design

7.1 Model Selection

After the team confirmed the desired inputs and outputs, the next step is to establish a suitable model. After extensive literary review, the team selected a statistical method called vector-autoregression as our baseline model and two RNN models (LSTM and GRU) as our main prediction models.

7.1.1 Baseline Model: Vector Autoregression (VAR)

After the team confirmed the desired inputs and outputs, the next step is to establish a suitable baseline model. After extensive literary review, the team selected a statistical method called vector-autoregression, or VAR. Simply put, VAR captures the long-term relationship of multiple time-based quantities such as the sensor data we received from MNG. This regression model also allows for a multivariate output, which is useful if future iterations of the design include more than one output variable. The formula for two-variable VAR with a two-step lag is shown in equation 1 [15]:

$$
\begin{aligned}
Y_t &= \alpha + \beta_{11} Y_{t-1} + \beta_{12} Y_{t-2} \\
&\quad + \gamma_{11} X_{t-1} + \gamma_{12} X_{t-2} + \varepsilon_{1t} \\
X_t &= \alpha_2 + \beta_{21} Y_{t-1} + \beta_{22} Y_{t-2} \\
&\quad + \gamma_{21} X_{t-1} + \gamma_{22} X_{t-2} + \varepsilon_{2t}
\end{aligned} \tag{1}
$$

Vector autoregression (VAR) is used to model the relationship between multiple time-series variables as well as their past values (self, or autoregression). It is appropriate to be used where all variables affect each other, and move together in time [15]

1. **Testing for stationary series**
   VAR is to be used on stationary series, where none of the variables have an upwards to downwards trend over time. In both the Air Flow and the Valve models, all variables were shown to be stationary through use of the Augmented Dickey Fuller test (ADF)[2].

2. **Cointegration Analysis**
   Similar to correlation in linear regression, cointegration is used to assess whether the selected variables move together over time, and have long-term relationships. Significant cointegration behavior is confirmed in both the Air Flow and Valve models, meaning each identified variable is affected by the other variables and VAR is suitable. Notice below how the setpoints are not cointegrated with the other variables even though there is correlation, because setpoints do not move synchronously with the other features.

```
Name   ::  Test Stat > C(95%)    =>   Signif
 ----------------------------------------
Temp (°C) ::  8138.77   > 60.0627   =>   True
Primary Air Air Flow ::  1847.02   > 40.1749   =>   True
Zone Air Temperature ::  521.72    > 24.2761   =>   True
Fin Tube Radiation Valve Position ::  22.11     > 12.3212   =>   True
Primary Air Air Flow Setpoint ::  0.0       > 4.1296    =>   False
```

Figure 19. Cointegration Analysis of Air Flow models (95%) confidence

```
Name   ::  Test Stat > C(95%)    =>   Signif
 ----------------------------------------
Temp (°C) ::  8138.77   > 60.0627   =>   True
Fin Tube Radiation Valve Position ::  1847.02   > 40.1749   =>   True
Zone Air Temperature ::  521.72    > 24.2761   =>   True
Primary Air Air Flow ::  22.11     > 12.3212   =>   True
Primary Air Air Flow Setpoint ::  0.0       > 4.1296    =>   False
```

Figure 20. Cointegration Analysis of Valve model (95%) confidence

3. **Determining the lag parameter**

Although the regression coefficients are fitted on the entire training dataset, the VAR model requires users to specify the order of the VAR, or how many past time steps each current value is affected by (lag, or look-back). Unlike in the neural-network based model, it is costly and impractical to lag as much as 48 steps in a simple regression model. Thus, we look at when the Akaike Information Criterion (AIC), which estimates prediction error and therefore model

quality, begins to plateau. In both cases, we see that this happens around VAR of order 5, meaning that each prediction includes information from the last 5 time steps.

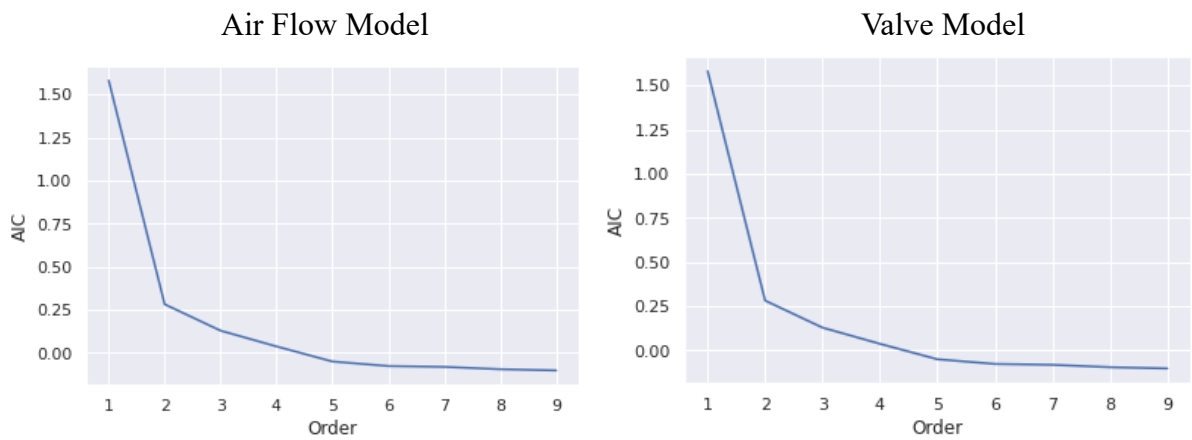


Figure 21. AIC vs Order of VAR model

The VAR model, due to its order and linear nature, is less robust than the other models, so using more complex neural-networks is anticipated to be more appropriate for our prediction task.

7.1.2 Recurrent Neural Networks (RNNs)

In addition to the baseline model, the team also explored other machine learning models that would be suitable for time-series prediction. Recurrent Neural Networks (RNNs) are a type of Artificial Neural Network (ANN) with "loops" that allow learned information to persist, which is not the case in traditional neural networks. The "loop" of the RNN can be visualized as a chain as shown in Figure 22. This chain-like nature makes RNNs suitable for use with sequences like time-series data.

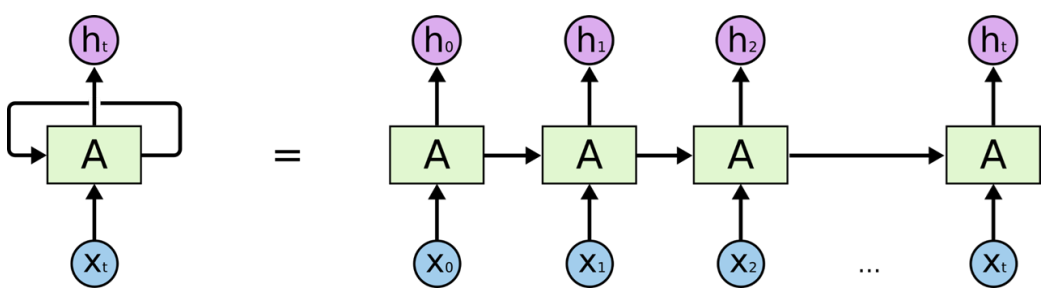


Figure 22: Chain-like nature of RNNs

One type of RNN is Long Short-Term Memory (LSTM), shown in Figure 23. LSTM models are capable of learning the long-term dependencies between variables. They are suitable for time-series data that may have lags of unknown duration between significant events. This makes the LSTM model a good candidate for HVAC sensor data which spans over months.

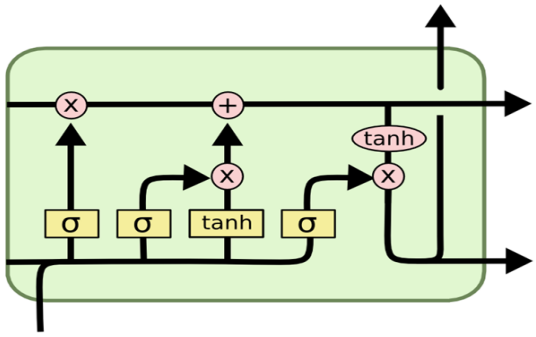


Figure 23: LSTM model

Another type of RNN the team examined was the Gated-Recurrent Unit (GRU), shown in Figure 24. GRU is a type of RNN that is very similar to LSTM. The GRU model has a slight difference in model architecture (it has fewer gates and therefore fewer parameters) which makes it slightly more computationally efficient.

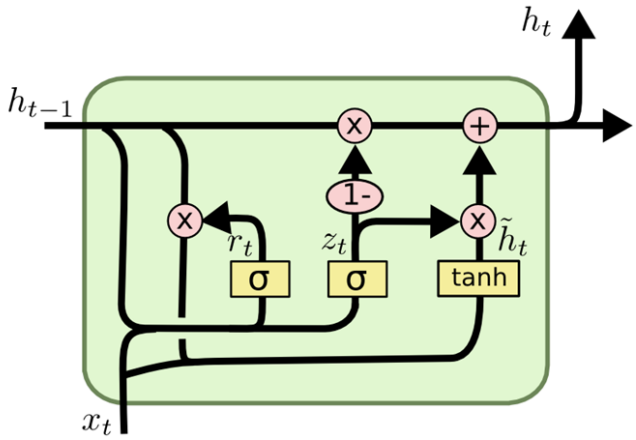


Figure 24: GRU model

7.2 ML Workflow

After selecting the appropriate models, the team built a full machine learning pipeline that takes pre-processed data and trains, validates, and evaluates the predictions. The predictions from the model are then used as inputs to the anomaly detector, which flags any possible outliers in the next 2 hours and generates visualizations.

7.2.1 Prediction

After data preparation and model selection, the team designed a machine learning workflow to train the models. Out of the 70,000 collected data samples, 55,000 are used for training, 5000 are used for validation, and 10,000 are used for testing as shown in Figure 25. Each data sample contains a sequence length of 48 as features and a prediction length of 24 as labels. This enables the model to learn a sequence of 48 data points (i.e., 4 hours) and use this sequence to predict the next 24 data points (i.e., 2 hours). The samples are shuffled to increase randomness, as shown in Figure 26.

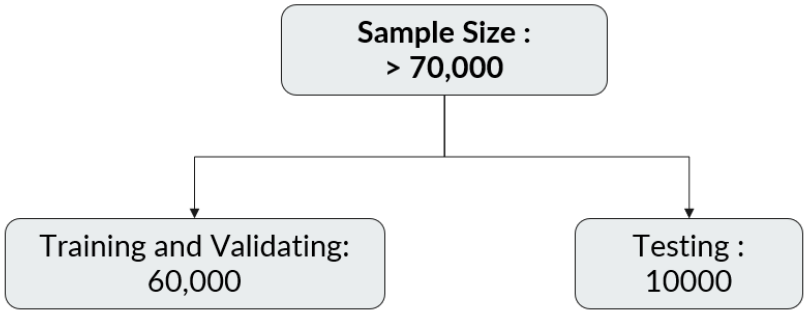


Figure 25: Training, validation, and testing split

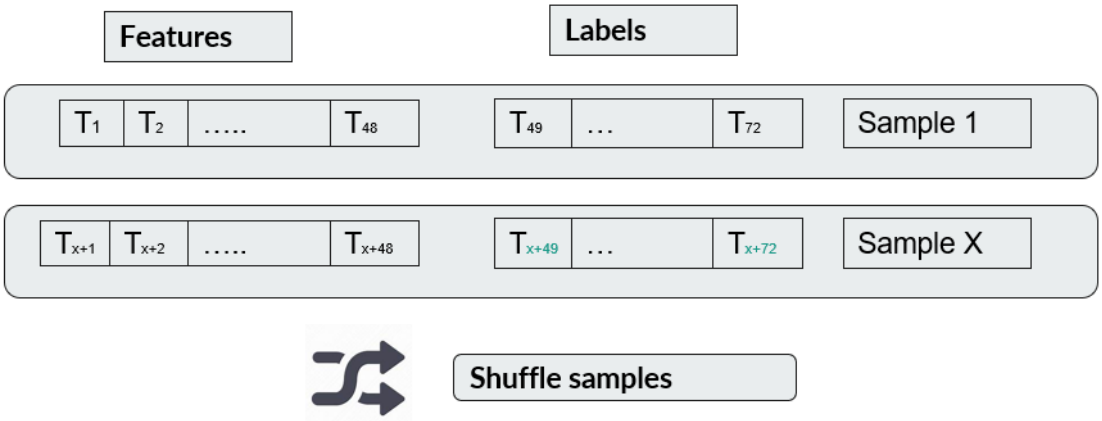


Figure 26: Organization of data samples

After shuffling, the features of each sample were fed into a Neural Network Model (GRU and LSTM in this case), and the model will generate a prediction. Subsequently, the predicted values were compared with the actual values to compute the Huber loss, which was then used to update the parameters for the NN model. The training process is illustrated in Figure 27.

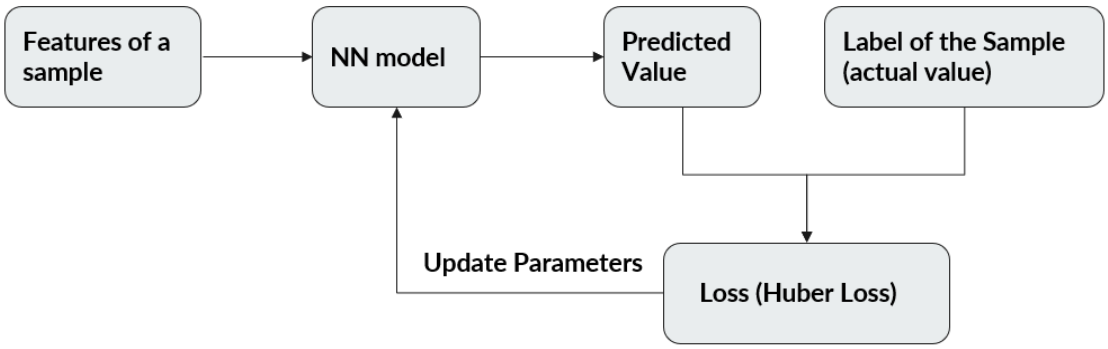


Figure 27: Flowchart showing the model training process

Huber Loss is a common loss function used in regression problems. Compared with Mean Squared Error (MSE), Huber Loss is less sensitive to outliers. When the loss surpasses a certain threshold, the quadratic term is replaced by a linear term which makes the Huber Loss a combination of both MSE and Mean Absolute Error (MAE). The sub-function for large errors (e.g., outliers) is the absolute error function, which avoids the excessive sensitivity to large errors that characterize MSE [16]. The sub-function for small errors is the squared error making the entire function continuous and differentiable, which overcomes MAE's convergence issues. Although the Huber loss combines the advantages of the MSE and MAE, its main disadvantage is the associated complexity. In order to maximize model accuracy, the hyperparameter δ will need to be optimized which increases the training requirements. The mathematical definition of Huber Loss is shown in equation (2).

$$L_\delta(y, f(x)) = \begin{cases} \frac{1}{2}(y - f(x))^2 & \text{for} |y - f(x)| \leq \delta, \\ \delta\,|y - f(x)| - \frac{1}{2}\delta^2 & \text{otherwise.} \end{cases} \tag{2}$$

After training, the model is evaluated on the test data using MAE. MAE measures the average of the squared difference between the estimated values and the actual value. In other words, MAE represents the average magnitude of error produced by the model i.e., how far the model's predictions deviate from their corresponding outputs on average. The mathematical definition of MAE is shown in equation (3).

$$MAE = \frac{1}{n}\sum_t |A_t - F_t| \tag{3}$$

7.2.2 Anomaly Detection

The second stage of the model pipeline is to use the prediction models to generate forecasts for the next time horizon, and then monitor this prediction for any anomalies. The anomaly detection task is unsupervised, since we do not know whether the given CPPIB data is normal, or contains malfunctions. Thus, the team explored two state-of-the-art outlier detection methods, which are detailed in the following sections. Both methods selected are global outlier detection methods, meaning that the entire history is considered when we are deciding whether a point is anomalous. The team decided against locally-based methods because looking at neighboring points to classify anomalies is susceptible to data drift, which would prevent us from detecting progressive degradation of the mechanical system. In addition to the output variable, we accounted for temporal variables such as hour and day of the week, as well as the weather. Since outlier detection methods depend on distances and densities, a preprocessing step was taken to normalize the inputs to account for differences in scale and to minimize skew [17]. Both algorithms are implemented using the PyOD python package [18].

Statistical Approach - Histogram-based Outlier Score (HBOS)

To begin, we fit a Histogram-based Outlier Score (HBOS) model [19] to the dataset and classify each data point as anomalous (score = 1) or not. For each input dimension, the HBOS algorithm

constructs a histogram, representing the density in each bin, or range, of values. The HBOS of each point is the sum of the log(density value) for all output features. The mathematical definition of the HBOS is shown in equation (4).

$$HBOS(p) = \sum_{i=0}^{d} log(\frac{1}{hist_i(p)}) \quad (4)$$

The HBOS assumes that each input dimension is independent and assumes a gaussian distribution of frequencies. Statistical density-based values are simple to calculate, and HBOS is very effective if the aforementioned assumptions hold true [20]. However, due to these distributional assumptions, the algorithm is not robust in real-life conditions, which can often have extreme or non-gaussian behaviour and dependencies between variables.

Tree-based Approach - Isolation Forest (iForest)

Instead of building a profile of normal instances like in HBOS, the iForest method seeks to isolate anomalous points through sub-sampling [21]. This approach assumes two characteristics of anomalous points:

(i) There are few of them

(ii) They have attribute values that are very different from other points

Since anomalies are "few and different," they are easy to isolate - when we split on attributes, they will be close to the root of the tree whereas normal points will be deeper in the tree. An iForest is an ensemble of iTrees that extract instances with short average path length (this path length is the anomaly score). Each tree has a small sub-sample size, which prevents "swamping," which is when a normal instance is misclassified as anomalous because it is surrounded by anomalous points, or "masking" which is the inverse[22]. It also does not depend on density or distance calculations, which makes it faster and less sensitive to the way normalization or scaling is performed. Other ML-based methods such as KNN, SVMs, etc are impractical for the large number of data points we have and the frequency of new data, but iForests run in linear time. Despite being very efficient and more sophisticated than HBOS, this tree-based method is less explainable and hard to visualize. Additionally it suffers from a bias as a result of the parallel cuts that are made during branching:

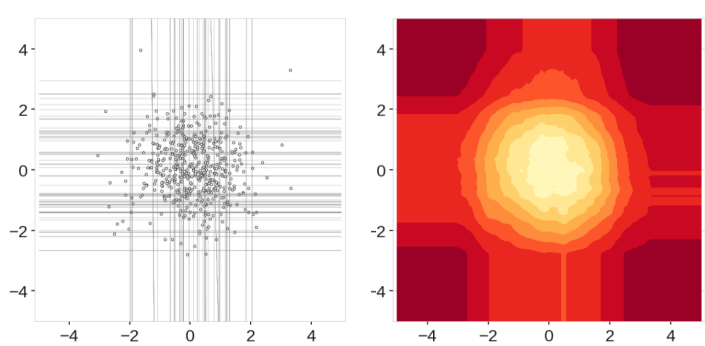


Figure 28: Tree Cuts and heatmap of anomaly score (darker = more anomalous) [23]

As we can see in Figure 28, each split divides the plane in half, so the corners are more likely to be classified as anomalous than the four lighter areas at the middle of each axis.

8.0 Model Implementation

Both the LSTM and the GRU model have 1 hidden layer with 7 features in the hidden state. was trained over 1200 epochs with a learning rate of 0.008. There are 55000 samples in the training dataset, 5000 samples in the validation dataset, and 10000 samples in the test dataset. The sequence length for the training samples is 48 datapoints (4 hours), and the sequence length for the predicted samples is 24 data points (2 hours). The hyperparameter δ for Huber loss is set as 0.03 to maximize performance.

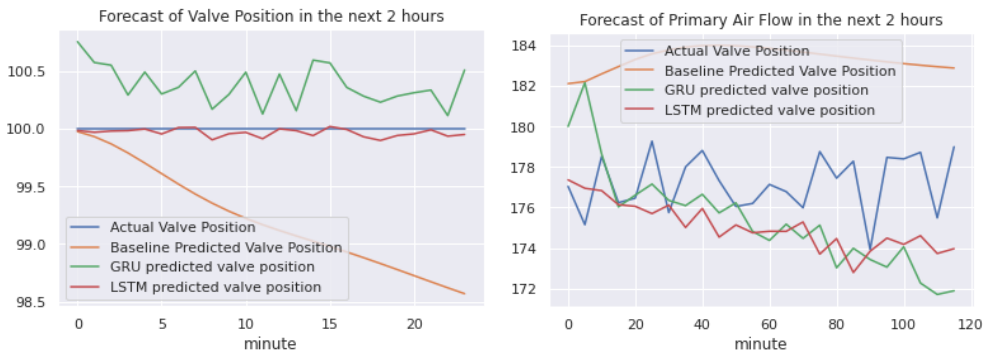
8.1 Prediction Results

Table 5 summarizes the prediction results in terms of MAE for airflow and valve position (note that MAE is not a percentage). These results are used as a sense check as to whether our more complex neural net models are working. It appears that the LSTM model works better for both the airflow model and the valve position model. In addition, the valve position model performs better than the airflow model partly because the valve position is often set to a stationary value, so the prediction would not have as much variance. It is worth noting that the errors mostly come from the fact the predicted behaviour fluctuates more than the actual system behaviour, but their mean values are quite similar.

Model	**MAE**	
	Airflow	Valve Position
VAR	23.05	1.32
LSTM	17.27	1.14
GRU	17.55	1.31

Table 5: MAE achieved for task and model

Some sample results are displayed in Figure 29 to Figure 32. Figure 29 shows the air flow prediction by the baseline model. Figure 27 shows the valve position prediction by the baseline model. Figure 31 shows the air flow prediction by LSTM and Figure 32 shows the valve position prediction by LSTM. Specific points of error will be discussed in the following sections.



Forecast of Valve *Forecast of AirFlow*

Figure 29: Forecast of Valve Position

8.2. Error Analysis

When deploying ML models in the real world and performing performance evaluation, it is important to recognize that model accuracy may not be uniform across subgroups of data (Figure 33). In other words, there might exist input conditions for which the model fails more often, and such failures may cause direct consequences related to lack of reliability, unfairness, or more broadly lack of trust in machine learning altogether [24]. It is difficult to convey a detailed story on model behavior with a single number and use aggregate metrics such as MAE to score models on a whole benchmark.

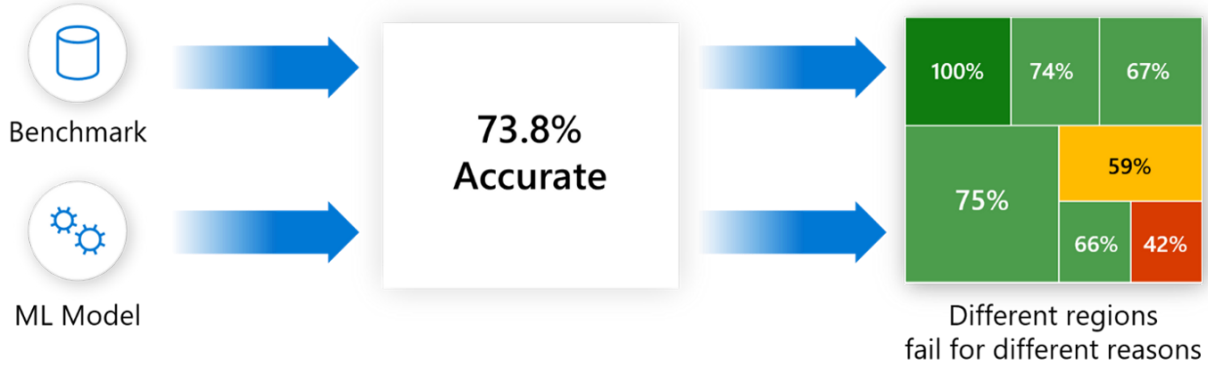


Figure 30: Illustration of model accuracy varying across subgroups of data

8.2.1 Error Analysis in Airflow Model

As a result, the team took a deeper dive into the input feature space and examined prediction results for different time periods. For the air flow model, prediction errors show several types of behavior. One type of error occurs when the prediction fails to catch up with the actual data when it is rapidly changing, as shown in Figure 34. This indicates that we may need to shorten the prediction horizon to accurately predict system behavior in later time steps.

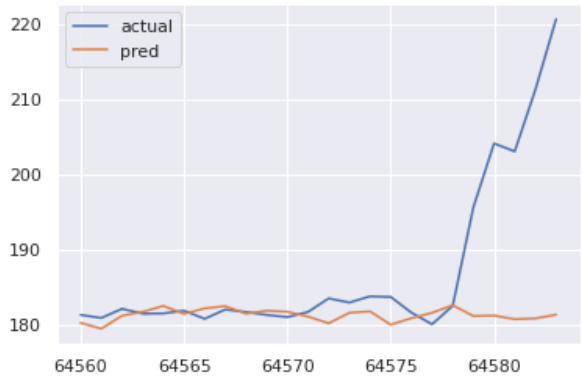


Figure 31: Error - Prediction Failing to Catch Up to Changes in Actual Data

In addition to issues with time-lag, the team also faced issues with overestimation and underestimation as shown in Figures 35a and 35b.. The client has reassured the team that the differences are small and likely do not make a difference in the overall system performance.

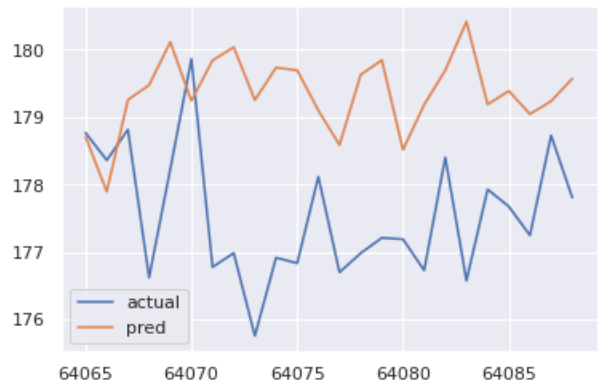


Figure 32a (left): Overestimation *Figure 32b (right): Underestimation*

The team also observed that the variance in the predicted values (i.e., how much the values fluctuate) is often smaller than the actual variance, as shown in Figure 36. Again, the client believes that this difference is small and does not impact the overall system performance. However, this may be an indication that more training data is required to accurately capture the behaviour of the air flow system.

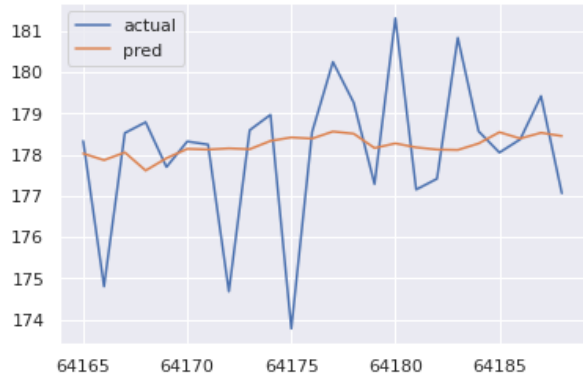


Figure 33: Error - Variance

8.2.2 Error Analysis in Valve Position Model

As shown in Figure 37 and 38, the valve position is frequently 100, with occasional deviations ranging from 40 - 100, as well as the occasional 0 value.

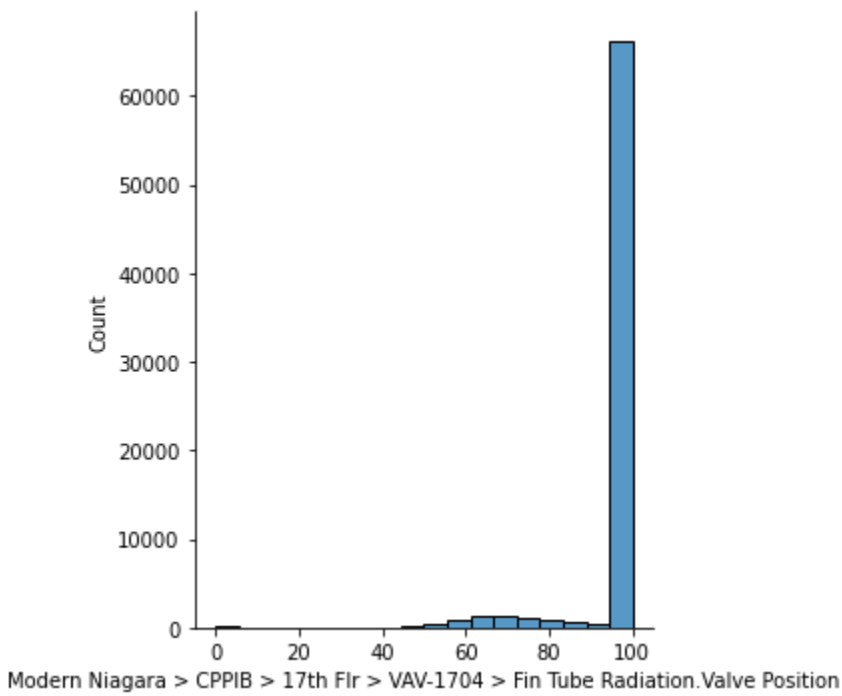


Figure 34: Valve Position histogram

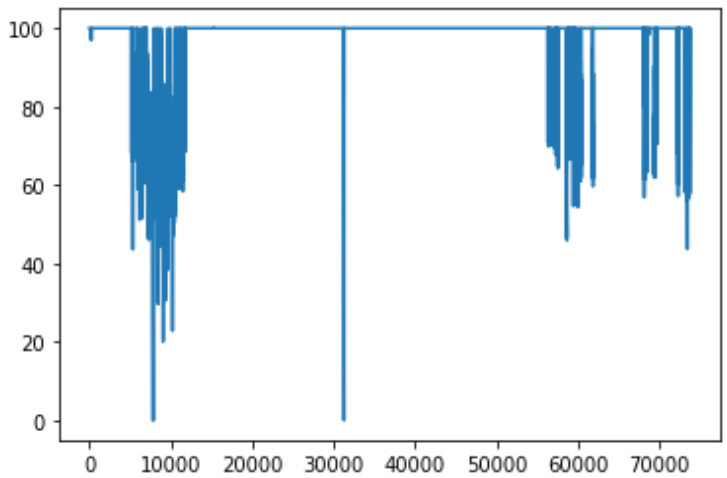


Figure 35: Valve Position over time

The model is biased towards this valve position value and does not perform as accurately when it deviates from 100. This can be seen in Figures 39 and 40. In Figure 39, when the valve position is actually 100, the predicted value closely follows the actual value. In Figure 40, although the valve position deviates from 100, the predicted value remains close to 100, which results in a lower model accuracy. However, it should be noted that the overall error of the model remains low because the frequency of the valve position deviating from 100 is low.

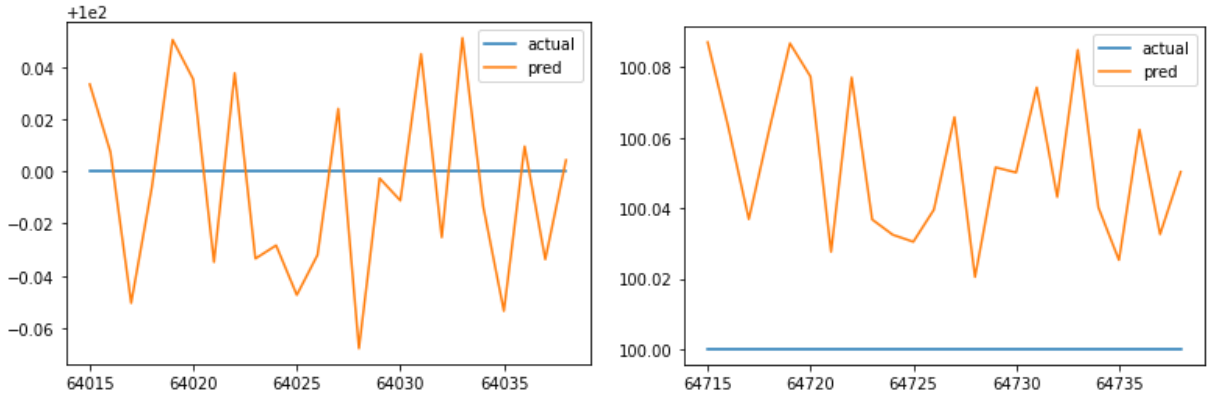


Figure 36: Valve Position prediction when valve position is actually 100

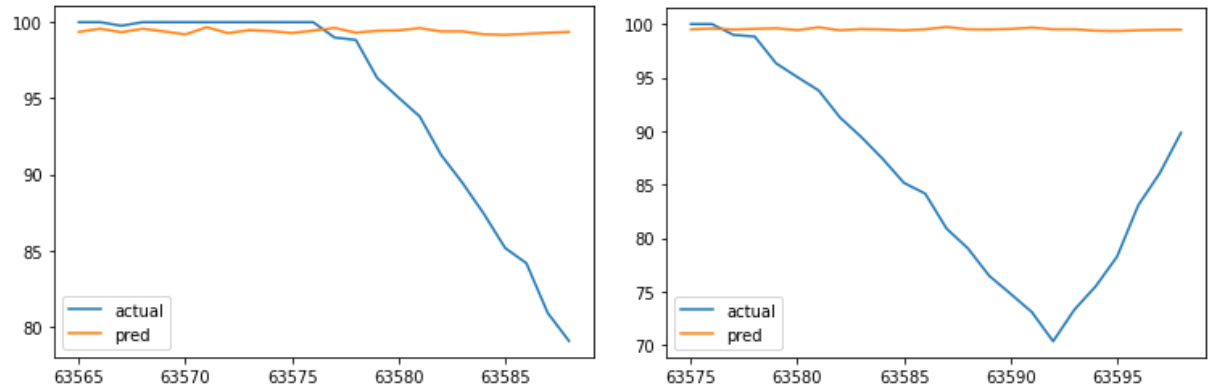


Figure 37: Valve Position prediction when valve position deviates from 100

The team previously believed that including future output temperature forecasts would improve the model performance. Since the valve position is highly correlated to the outdoor temperature, this could give the model the additional information it requires to predict when the valve position will deviate from 100. However, this does not appear to be the case as shown in Figure 41. The temperature difference within 2 hours is generally consistent and ranges between -2 to 2 degrees. On the other hand, the valve position deviates from 100 regardless of the consistent temperature difference. As a result, the team concluded that the valve behavior is not a result of rapidly changing future weather conditions but a result of other factors not captured in our model.

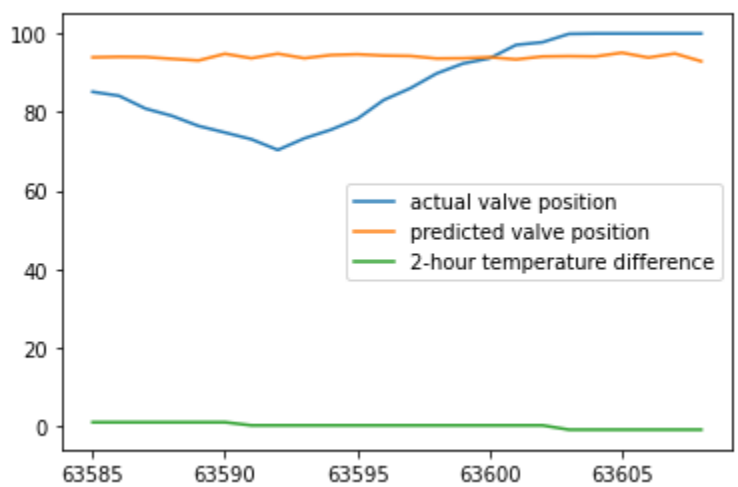


Figure 38: Actual vs. predicted valve position when 2 hr temperature difference is negligible

8.3 Anomaly Detection Results and Discussion

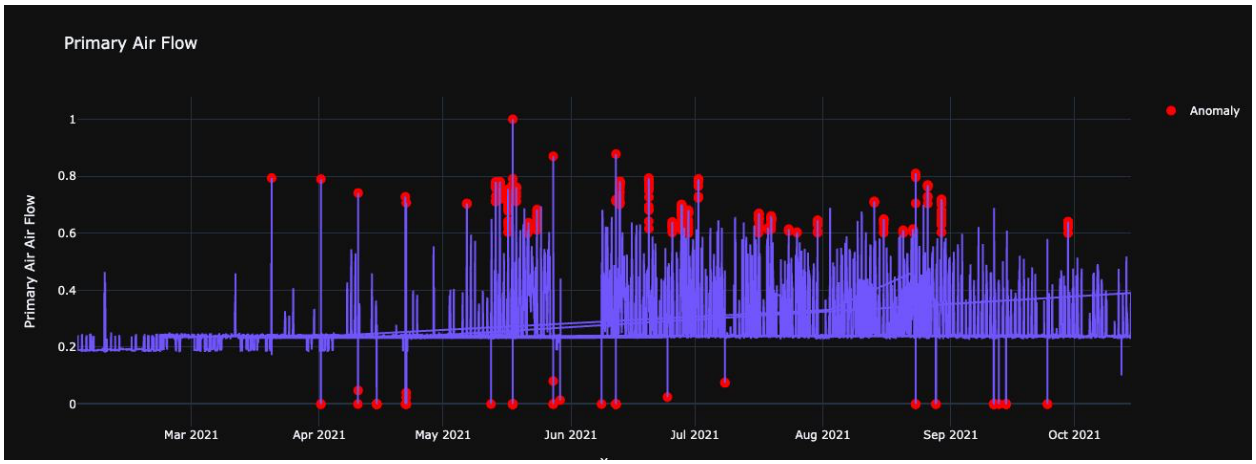
8.3.1 Setting the Contamination Fraction

A primary drawback of not having labelled anomaly data is that we have to make certain assumptions about the fraction of outliers that are present in the original dataset. This is called the fraction of contamination, which is a necessary parameter for each unsupervised learning method. As shown below, the number of identified anomaly points depends largely on this

number, which we can only set as an arbitrary value if we do not have prior knowledge.

Results for varying contamination fraction settings are for HBOS as follows:

Figure 39: Assume 0.1% are outliers for Air Flow (Top) Valve (Bottom), outliers in red



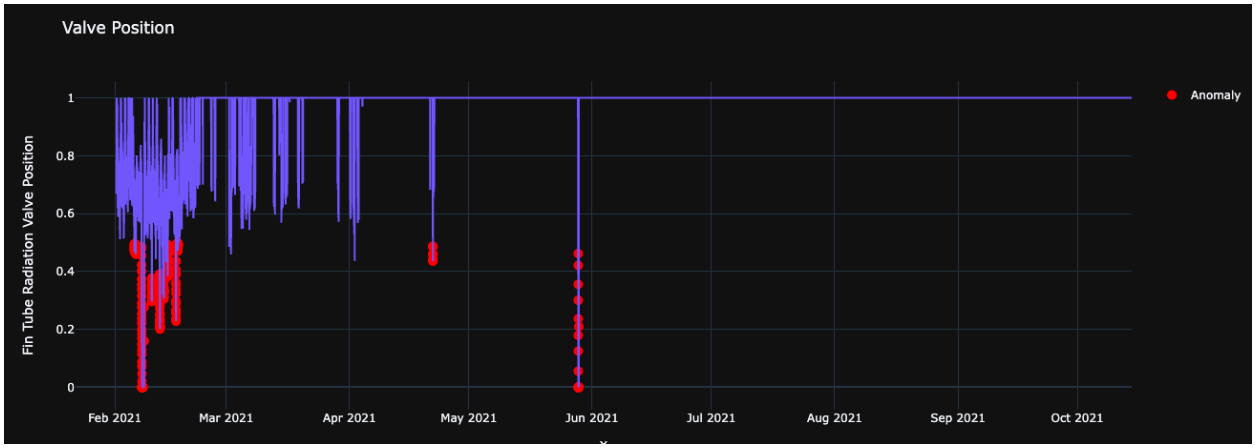


Figure 40: Assume 0.5% are outliers for Air Flow (Top) Valve (Bottom), outliers in red

8.3.2 Comparison of Methods and Results Discussion

Figure 41. Valve Position AD at Fraction = 0.5%

For the valve position, we see that anomalous points are detected at different parts in the dataset. In the isolation forest, we see that the anomalies (in red) occur at the *onset* of values that are very different from what we have previously seen. For example, the first time that the valve position dips down is marked in red, as well as the first time it dips to extremely low values that are close to zero. In the HBOS, it seems like there is a cutoff below values of around 50, and those are classified as anomalous. In this case, iForest is better suited to identifying when irregular behaviour begins to happen, since HBOS just tells us whenever the valve position has a low value. However, low values aren't synonymous with anomalies - it could just be a systematic error such as when the system starts up or shuts down or changes modes.

Figure 42: Air Flow AD at Fraction = 0.5%

In the airflow model, we again see that the iForest and HBOS categorize anomalies at different points. As highlighted on the figure above, there are four key differences to discuss:

1. In the first circle, the isolation forest did not pick up on the first spike in the time series while the HBOS did
2. The isolation forest classified the last spike in the time series as anomalous, but it is not greatly different from surrounding values
3. HBOS does not classify the highest value in the timeseries as anomalous, but instead only labels the values around it as anomalous
4. HBOS labels ‘0’ points as anomalous while iForest doesn’t

For (1) and (2), we know that iForest measures outliers by average path length and the sub-sampling is done using random selection. Thus, depending on the subsamples chosen, what is defined as being “different enough” from the other points to be classified as an anomaly could change. We also know that when there are a lot of anomalous points in a sub-sample, they could aggregate in a cluster and be hard to detect.

For (3), we know that the HBOS is very sensitive to the selection of number of bins and the width, as well as how the normalization is done (in this case we use min-max normalization), all of which could affect which values are flagged as anomalous. Thus, hyperparameter tuning and exploring alternative normalization methods could help pinpoint what parameters affect which points.

For (4), HBOS is flags the points that are irregular with respect to the empirical distribution, so the ‘0’ points may fall outside of the normal range, whereas in iForest, each anomalous point has to be sufficient infrequent and different. The ‘0’ values are not uncommon, due to it being caused by system shut-offs, so would not be classified as anomalous by iForest. It is also important to note that since HBOS is a density-based method, it assumes that the underlying distribution is gaussian, which is not the case as seen in 8.2. This could also lead to skewing of the As we can deduce from the airflow model, further hyperparameter tuning is necessary to make a definitive statement about the performance of both outlier detection algorithms. However, hyperparameter tuning would require prior knowledge about which points should be outliers, which is currently unavailable to us.

9.0 UI Design

Following the design performance of our model designs the MNG showed interest in the development of a User Interface (UI) for the different models we have developed. We developed a tool that can take data inputs (currently in the form of a static .csv file) and display predicted results, show the detected anomalies, and generate reports.

9.0.1 Prototyping and Lean Evolution

We designed a first round of Balsamiq Wireframe sketches (Figure 46) to layout possible interfaces with features. We used the wireframe to walkthrough possible feature options (Table#) for the UI with MNG. For the full list of questions and feedback received, please refer to Appendix B. For MNG their most important request was being able to see the plotting of ML models and the ability to export that data. So in our final design, we wanted to meet the goals of MNG.

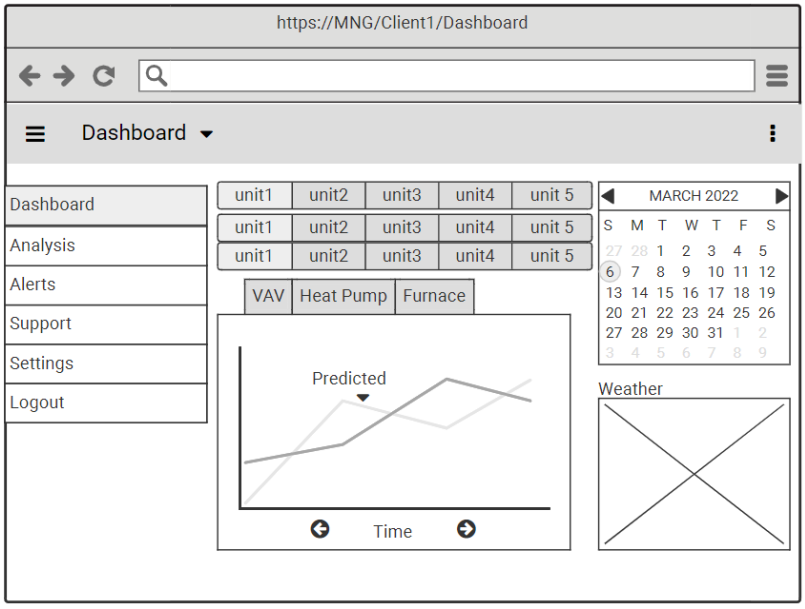


Figure 43: Round one of Wireframing

Features	Description
Input Window	Showing the static data
Filters	Ability select/navigate to a date and time
Output window	Ability to download predictions
Interactive Plots	The ability to navigate the plots that have been generated.

Table 6: Ideation of possible features

9.1 Platform Selection

Following prototyping, we wanted to determine a platform that can support and display our ML models. We conducted testing on JupyterLab Viola but found the interface did not support some

of the features we wanted to incorporate like input and export/download options. Following that assessment, we tested our platform and ML models on Streamlit.

Streamlit provided us with more flexibility and a more interactive way to incorporate our current ML models, but also provided features for user/client engagement. We conducted a walk-through of the design using Streamlit on the Anomaly Detection Screen with the MNG.

9.2 Page for Training Prediction Model

The input screen, shown in Figure 47, takes in the input data and trains a new model based on the user selections. The user chooses which model to train, as well as the prediction horizon, and creates a new .pth file which can then be used for anomaly detection. This separate interface gives the user the ability to modify a limited set of model parameters and allows the user to separately predict and detect anomalies without having to retrain a new model each time.

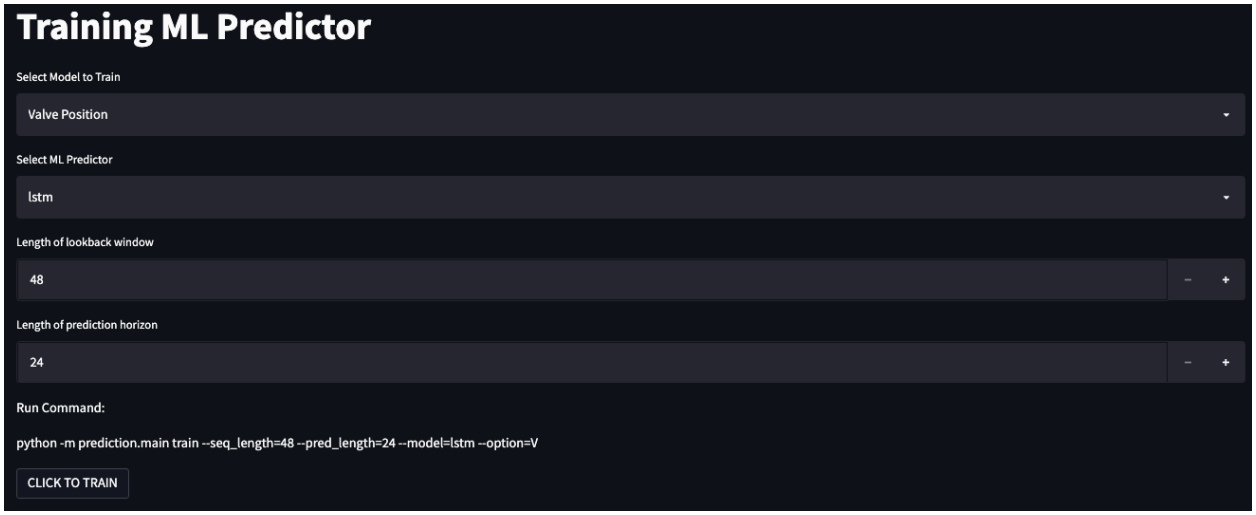
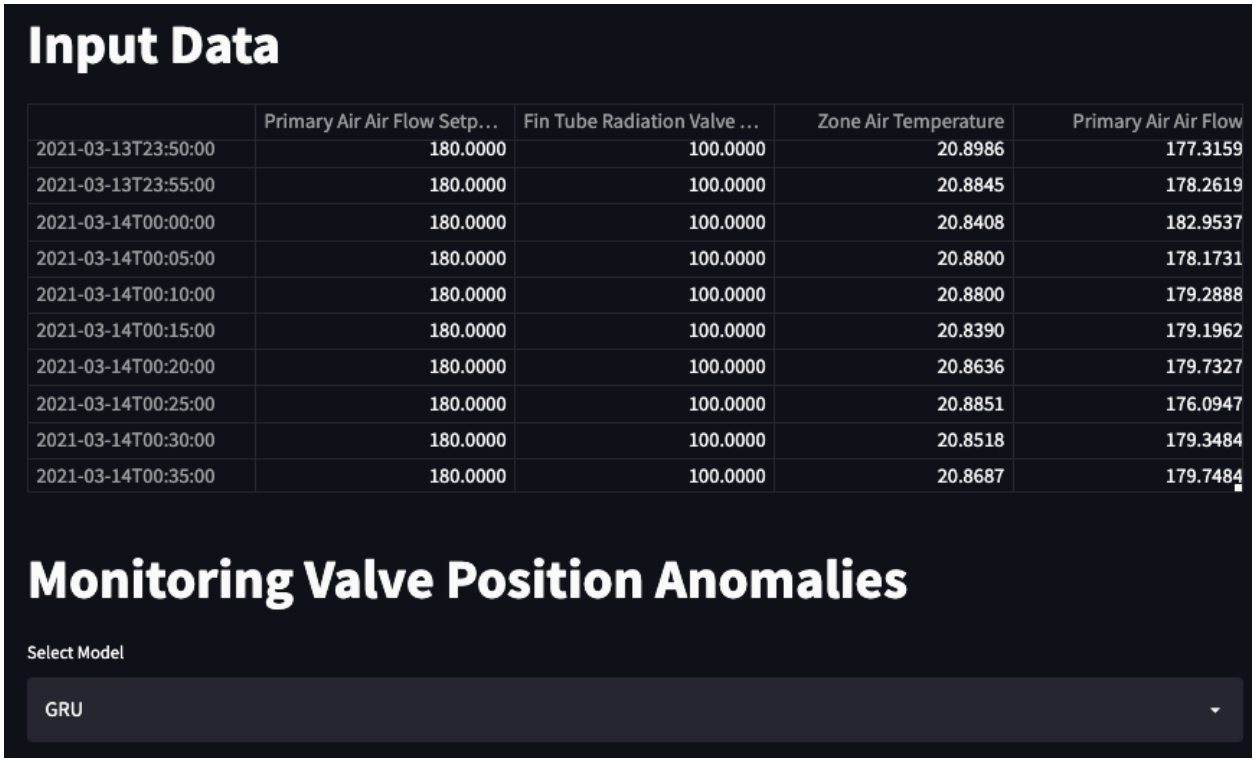


Figure 44: ML Prediction Interface

9.3 Anomaly Detection Interface

The anomaly detection interface is shown in Figure 47. First, input data is shown and users can sort using each column and scroll through the input parameters. Then, the user should select their desired prediction model.



Input Data

	Primary Air Air Flow Setp…	Fin Tube Radiation Valve …	Zone Air Temperature	Primary Air Air Flow
2021-03-13T23:50:00	180.0000	100.0000	20.8986	177.3159
2021-03-13T23:55:00	180.0000	100.0000	20.8845	178.2619
2021-03-14T00:00:00	180.0000	100.0000	20.8408	182.9537
2021-03-14T00:05:00	180.0000	100.0000	20.8800	178.1731
2021-03-14T00:10:00	180.0000	100.0000	20.8800	179.2888
2021-03-14T00:15:00	180.0000	100.0000	20.8390	179.1962
2021-03-14T00:20:00	180.0000	100.0000	20.8636	179.7327
2021-03-14T00:25:00	180.0000	100.0000	20.8851	176.0947
2021-03-14T00:30:00	180.0000	100.0000	20.8518	179.3484
2021-03-14T00:35:00	180.0000	100.0000	20.8687	179.7484

Monitoring Valve Position Anomalies

Select Model

GRU

Figure 45: Input Data Pane

For the Valve and Airflow models, the UI page displays prediction results for the next 2 hours, as shown in Figure 48.

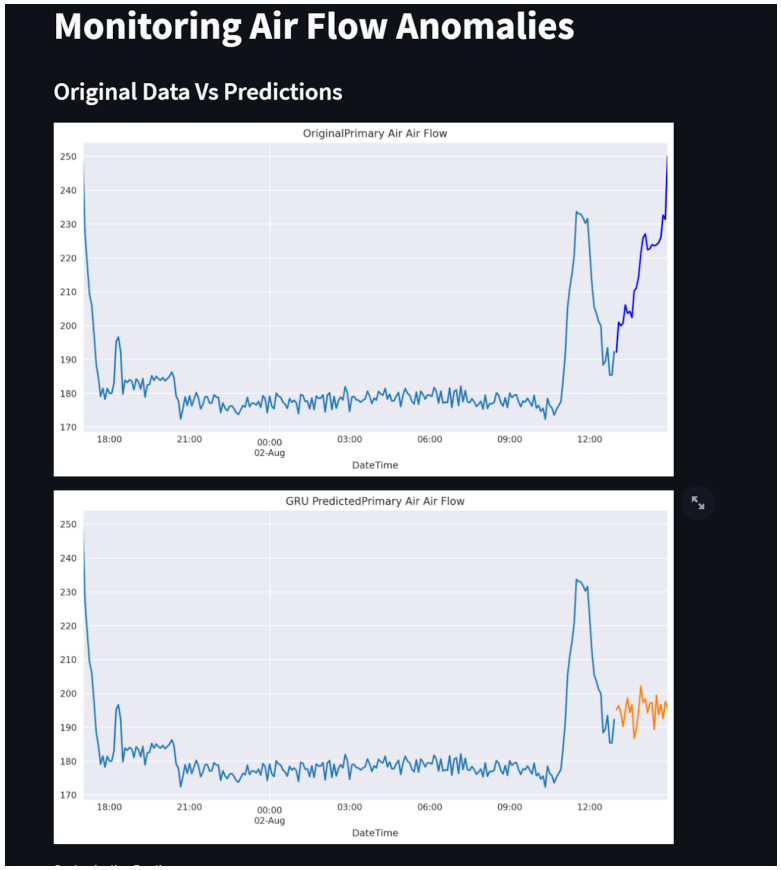
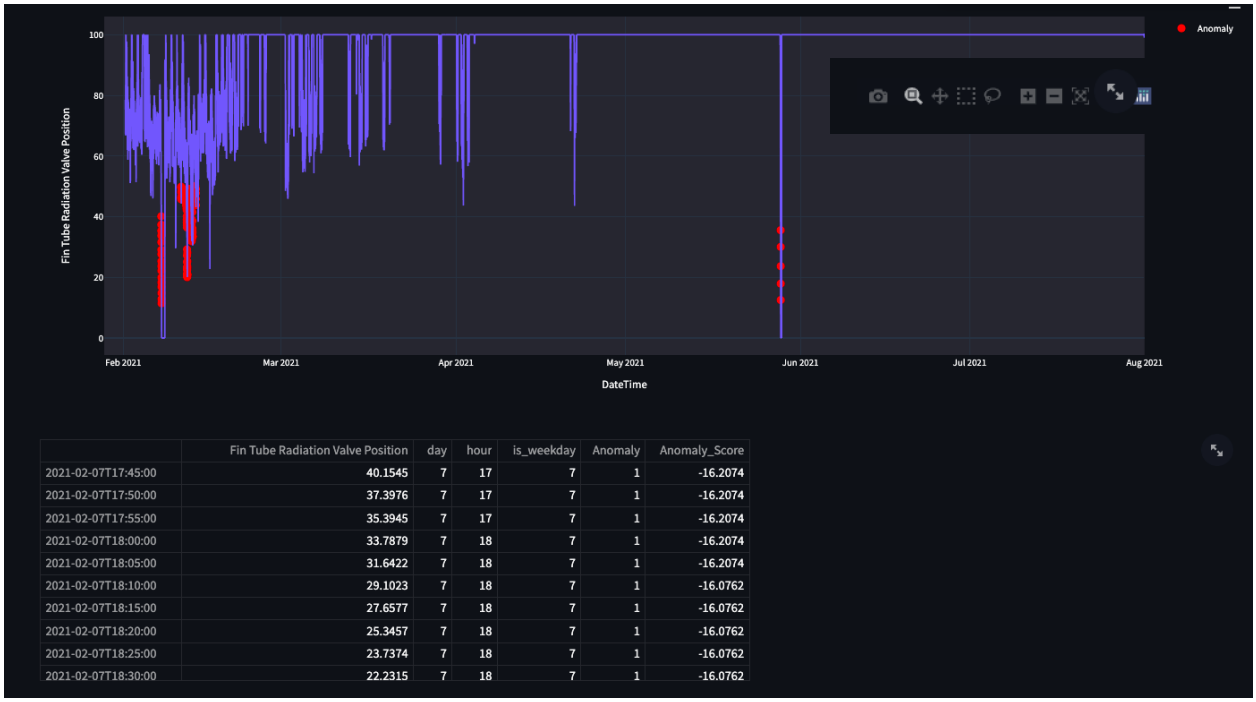


Figure 46: Prediction (Orange) and Original (Blue) in the next 24 time steps (2h)

For each Valve and Airflow model, the UI page displayed Anomaly Diagnosis with 2 different technologies, Isolation Forest and global HSBO. The prediction is then joined with the rest of the time series and fed as an input into the anomaly detector, along with date-related features and the projected 2-hour temperature difference. The anomaly plot allows the user to hover over each point to see its information, as well as zoom, pan, and save screen captures. The anomalies are also displayed in a tabular format so the operator can log anomalous behaviour. This can be seen in Figure 50.



	Fin Tube Radiation Valve Position	day	hour	is_weekday	Anomaly	Anomaly_Score
2021-02-07T17:45:00	40.1545	7	17	7	1	-16.2074
2021-02-07T17:50:00	37.3976	7	17	7	1	-16.2074
2021-02-07T17:55:00	35.3945	7	17	7	1	-16.2074
2021-02-07T18:00:00	33.7879	7	18	7	1	-16.2074
2021-02-07T18:05:00	31.6422	7	18	7	1	-16.2074
2021-02-07T18:10:00	29.1023	7	18	7	1	-16.0762
2021-02-07T18:15:00	27.6577	7	18	7	1	-16.0762
2021-02-07T18:20:00	25.3457	7	18	7	1	-16.0762
2021-02-07T18:25:00	23.7374	7	18	7	1	-16.0762
2021-02-07T18:30:00	22.2315	7	18	7	1	-16.0762

Figure 57: Anomaly reports

10.0 Proposed Future Iterations

10.1 Improving the Dataset

The given dataset is from a single building and has issues that were discussed previously, such as irregularities and uncertainty surrounding anomalous points. In order to generalize the machine learning model, a dataset of good quality is important. In the future, a potential improvement that could be made is training with a dataset of increased length and diversity, since this dataset only spans one year. In practice, we would also want a continuous stream of data rather than a static dataset. Additionally, expanding the availability of sensor and building information would allow us more input features with which to predict outputs. We could then build a wider variety of models rather than just air flow and valve position, which does not cover the range of system failures that could occur.

In the future, repairs and failures should be logged in order to establish a labeled anomaly dataset. This could be used to better apply preprocessing techniques such as scaling and can enable us to use supervised learning methods to classify anomalies instead. We would also be able to set a contamination fraction with more confidence, which will give practical significance to the anomalous points detected.

10.2 Improving the Machine Learning Model

The goals for improving the machine learning model are:

a. Generalize the model and make the model be able to do multivariate forecasting
b. Increase prediction accuracy

To achieve the goals above, we could apply Neural Ordinary Differential Equations [25] into hidden states of RNN, which outperforms in the cases of irregular datasets. Although Neural ODE is a new concept proposed in 2018, it has achieved good performance of time series multivariate forecasting, compared with traditional LSTM or GRU, in the field of agriculture application [26]. In the previous successful practice of Neural ODE [26], they were also limited to computational resources and thus only experimented with partial autoencoders. And they don't have enough external factors to refer to, thus combining RNNs with Neraul ODE in their practice only achieved slight advantages over RNNs, compared with the performance demonstrated by full autoencoders of Neraul ODE [25] which is the idea they borrow from. In the future, if more computational resources are available, we could also try applying Neural ODE, in order to generalize the model and improve the accuracy.

As mentioned in the midterm design proposal, the team had conceptualized several other configurations of the machine learning model such as regressor chains and multi-task learning. Since we have only identified two output variables, it would not have added value to further integrate the two models, but shared features and shared learning could improve the task performance of more complex digital representations.

11.0 Measure of Success

The following table outlines the five objectives initially defined through collaboration with the client. Each objective has an associated metric defined that allows the team to measure the success of the design. As shown, the design created by the team meets all objectives specified except for the generalizable objective. This objective calls for the design to have the capacity to model different buildings with a similar performance accuracy. Unfortunately, the team was unable to obtain another dataset from the client and it was agreed that this objective would be eliminated.

Objective	***Metric***	***Design***	***Pass / Fail***
Autonomous	Number of human interventions required	0	Pass
Reliable	Accuracy achieved	Airflow: 17.27 Valve: 1.14	Pass
Computationally Efficient	RAM required	Valve: 3.51 GB Airflow: 2.05 GB	Pass
	Interval of prediction	5 minutes	Pass
Generalizable	Accuracy achieved on the dataset of another building	N/A	N/A
Modularity	Number of modular units	2	Pass

12.0 Conclusion

Utilizing the current MNG analytics we created two model frameworks for predictive maintenance. The predictive maintenance tool with further development will create efficiencies for MNG in reducing service calls and downtime. The digital twin of the VAV system will improve the operational performance of the target system with continued development. The benefit of this approach is that when MNG starts collecting more time-series data and documenting service calls that align with sensor data, they can tune the proof-of-concept model for further accuracy and reliability.

The next steps for MNG would be to further iterate and expand the HVAC systems included in the dataset. Our intention is that this tool with further enhancements can improve the comfort level of tenants and revolutionize the way we think about building maintenance. With further development, this tool can help predict the wear and tear of the building system and help business owners plan ahead for updates and maintenance checks.

13.0 Project Plan

Following the first iteration of the design and the Design Review and Critique in January we amended our milestones to reflect the new design objectives specified by MNG. Following the critique, we were prompted by the client to develop a UI platform for the two models and the anomaly detection process we had designed. Following the critique, we altered some of our design objectives to incorporate three streams of work that we believe we have fulfilled to the need and specifications of MNG:

Error diagnosis: The team analyzed the model performance in different input situations (i.e. highly frequent inputs or highly irregular inputs). This allowed the team to evaluate how the model performed in different input situations, and how we could accommodate for these different situations.

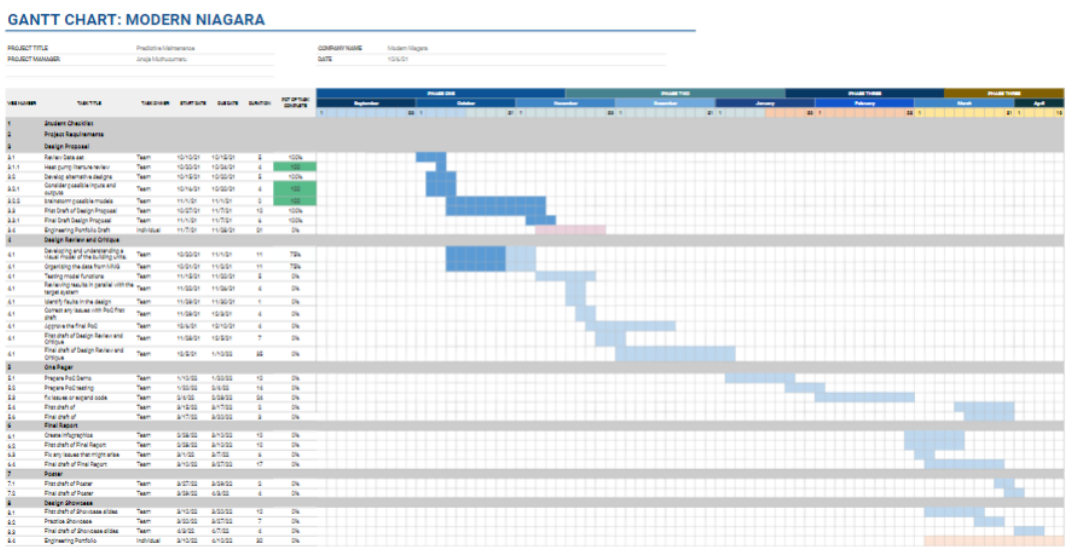
Model Improvements: Although Design 1 results are acceptable for MNG, we further trained and developed the model to improve the accuracy.

Connecting Code & Dashboard Implementation: We designed a prototype of a dashboard connecting the code we developed to a user-friendly dashboard.

13.1 Milestones

The milestones we have completed are tracked on a Gantt chart linked below. The chart was updated to reflect the ongoing progress of the work. The chart was available to MNG and supervisors to continue to maintain and develop transparency and trust between all stakeholders.

Click to view the full chart



13.2 Final deliverables

Our approach to the project was iterative, requiring trial and error to identify reliable inputs and accurate outputs, ML models and UI Design. MNG was consulted throughout every phase of the work plan and was a primary source of information for datasets and for identifying expert knowledge within the building and system. MNG also worked with the team to determine which building is most suitable for the proof of concept given the constraints and available resources.

Program supervisors offered support by providing expert pedagogical knowledge and guidance when requested or when they felt it was warranted.

Deliverable	**Deadline**
~~Student checklist~~	~~09/23/2021~~
~~Safety training complete~~	~~10/03/2021~~
~~Project requirements~~	~~10/10/2021~~
~~Design Proposal~~	~~11/07/2021~~
~~Engineering Portfolio Draft~~	~~11/28/2021~~
~~Design Review and Critique~~	~~01/10/2022~~
~~One-Pager due~~	~~03/20/2022~~
Final Report	03/27/2022
Poster	04/03/2022
Design Showcase	04/07/2022
Engineering Portfolio due	04/10/2022

Table 7: The team course deliverables and deadlines.

13.3 Responsibilities

The team has designated primary and secondary roles for each member, as shown in Table 8. The project managers, Anoja and Elizabeth, were responsible for ensuring the work was done correctly and on time. The communicators, Shirley and Anoja, were responsible for setting up meetings, sending out reminders, and communicating with the client and advisors on behalf of the team. The team leaders, Sherry and Chi, were in charge of ensuring the team had what it needed to succeed. The editors, Elizabeth and Shirley, provided a style guide for formatting as well as led the final check for consistency between sections. There are also skill-specific roles for the UX and ML portion of the project. The UX design was led by Anoja and Chi and the ML design was led by Chi and Sherry.

Role	**Primary**	**Secondary**
Project Manager	Anoja Muthucumaru	Elizabeth Chelmecki
Communications	Shirley Zhang	Anoja Muthuucmaru

Skill Specific	UX: Anoja Muthucumaru ML: Chi Zhang	UX: Chi Zhang ML: Sherry Zuo
Team Leader	Sherry Zuo	Chi Zhang
Editor	Elizabeth Chelmecki	Shirley Zhang

Table 8 Primary and secondary roles of the design team

References

[1] “Mechanical, electrical, HVAC & Plumbing,” Modern Niagara, 22-Oct-2021. [Online]. Available: https://modernniagara.com/. [Accessed: 27-Mar-2022].

[2] "RentSafeTO for Building Owners", City of Toronto, 2021. [Online]. Available: https://www.toronto.ca/community-people/housing-shelter/rental-housing-tenant-information/rental-housing-standards/apartment-building-standards/rentsafeto-for-building-owners/?accordion=new-bylaw-requirements-from-april-1. [Accessed: 09- Oct- 2021].

[3] Susto, G. A., Schirru, A., Pampuri, S., McLoone, S., & Beghi, A. (2015). Machine Learning for Predictive Maintenance: A Multiple Classifiers Approach. IEEE Transactions on Industrial Informatics, 11(3), 812-820 https://doi.org/10.1109/TII.2014.2349359

[4] “Contact us,” CPP Investments. [Online]. Available: https://www.cppinvestments.com/contact-us. [Accessed: 27-Mar-2022].

[5] Nejklakmb, “What is building automation,” What is Building Automation - Learning the basics. [Online]. Available: http://www.controlservices.com/learning_automation.htm. [Accessed: 08-Nov-2021].

[6] Security of Information Act, R.S.C. 1985, c. O-5. https://laws-lois.justice.gc.ca/eng/acts/o-5/. Retrieved on 2021-10-09

[7] Ontario.ca, "Freedom of Information and Protection of Privacy Act, R.S.O. 1990, c. F.31," 31 August 2021. [Online]. Available: https://www.ontario.ca/laws/statute/90f31/v46. [Accessed 5 October 2021].

[8] G. Eleftheriadis and M. Hamdy, “Impact of building envelope and mechanical component degradation on the whole building performance: A review paper,” *Energy Procedia*, vol. 132, pp. 321–326, 2017.

[9] IBM, "What is a digital twin?," IBM, 2021. [Online]. Available: https://www.ibm.com/topics/what-is-a-digital-twin. [Accessed 4 11 2021].

[10] N. Ben-Zvi, “A 2021-ready deep learning hardware guide,” Medium, 05-Dec-2020. [Online]. Available: https://towardsdatascience.com/another-deep-learning-hardware-guide-73a4c35d3e86. [Accessed: 08-Nov-2021].

[11] National Energy Code of Canada for Buildings, Canadian Commission on Building and Fire Codes, National Research Council of Canada,2017. [Online]. Available: https://publications.gc.ca/collections/collection_2018/cnrc-nrc/NR24-24-2017-eng.pdf

[12] Office of the Privacy Commissioner of Canada. The Personal Information Protection and Electronic Documents Act. [Online]. Available: https://www.priv.gc.ca/en/privacy-topics/privacy-laws-in-canada/the-personal-information-protection-and-electronic-documents-act-pipeda/. [Accessed: 7 November 2021].

[13] E. and C. C. Canada, "Government of Canada / gouvernement du Canada," *Climate*, 02-Mar-2022. [Online]. Available: https://climate.weather.gc.ca/historical_data/search_historic_data_e.html. [Accessed: 27-Mar-2022].

[14] "Spearman's rank-order correlation," Spearman's Rank-Order Correlation - A guide to when to use it, what it does and what the assumptions are. [Online]. Available: https://statistics.laerd.com/statistical-guides/spearmans-rank-order-correlation-statistical-guide.php. [Accessed: 27-Mar-2022].

[15] Chris Sims. Macroeconomics and Reality, Econometrica. 1980.

[16] "Blog: Machine learning: Loss functions," *Evergreen Innovations | Energy Storage & Renewable Energy Innovation*, 25-Aug-2020. [Online]. Available: https://www.evergreeninnovations.co/blog-machine-learning-loss-functions/. [Accessed: 27-Mar-2022].

[17] "Importance of scaling in unsupervised distance-based ..." [Online]. Available: http://www.imeko2009.it.pt/Papers/FP_517.pdf. [Accessed: 28-Mar-2022].

[18] Markus Goldstein, Andreas Dengel. Histogram-based Outlier Score (HBOS): A fast Unsupervised Anomaly Detection Algorithm. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.401.5686

[19] Zhao, Y., Nasrullah, Z. and Li, Z., 2019. PyOD: A Python Toolbox for Scalable Outlier Detection. Journal of machine learning research (JMLR), 20(96), pp.1-7. https://github.com/yzhao062/pyod

[20] Y. Feng, W. Cai, H. Yue, J. Xu, Y. Lin, J. Chen, and Z. Hu, "An improved X-means and isolation forest based methodology for network traffic anomaly detection," *PLOS ONE*, vol. 17, no. 1, 2022.

[21] “Isolation Forest - NJU.EDU.CN.” [Online]. Available: https://cs.nju.edu.cn/zhouzh/zhouzh.files/publication/icdm08b.pdf. [Accessed: 28-Mar-2022].

[22] A. Young, “Isolation Forest is the best anomaly detection algorithm for Big Data Right Now,” *Medium*, 25-May-2021. [Online]. Available: https://towardsdatascience.com/isolation-forest-is-the-best-anomaly-detection-algorithm-for-big-data-right-now-e1a18ec0f94f. [Accessed: 27-Mar-2022].

[23] “Isolation Forest: Anomaly detection with Isolation Forest,” *Analytics Vidhya*, 26-Jul-2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/07/anomaly-detection-using-isolation-forest-a-complete-guide/. [Accessed: 27-Mar-2022].

[24] B. Nushi, “Responsible machine learning with error analysis,” *Medium*, 05-Mar-2021. [Online]. Available: https://towardsdatascience.com/responsible-machine-learning-with-error-analysis-a7553f649915. [Accessed: 27-Mar-2022].

[25] Chen, R. T. Q., Rubanova, Y., Bettencourt, J., and Duvenaud, D. Neural Ordinary Differential Equations. 2018. https://arxiv.org/abs/1806.07366.

[26] Wang, Yuchen and Chan Chee, Matthieu and Edher, Ziyad and Hoang, Minh Duc and Fujimori, Shion and Bettencourt, Jesse. Forecasting Black Sigatoka Infection Risks with Latent Neural ODEs. ICML 2021 Workshop on Tackling Climate Change with Machine Learning. 2021. https://www.climatechange.ai/papers/icml2021/26.

[27] N. Sharma, "7 Tools to Build Proof-of-Concept Pipelines for Machine Learning Applications," 19 July 2021. [Online]. Available: https://neptune.ai/blog/tools-proof-of-concept-pipelines-for-machine-learning.

Appendices

Appendix A: Github repository address

---

github.com/chuiyunjun/aps490

Appendix B: UI Interview and Wireframe Walkthrough with Pranjal De

---

1) What about your current platform do you like?
Generation of the Alert.

2) What actionable tasks do you typically complete after using the platform?
Show alerts and analysis and logging of past alerts

3) What actionable tasks would you like to typically complete after using the platform?
CSV (use new data set), Show alerts and analysis, Log of past alerts, Report of analysis of data sheet,

4) Ideally what type of information is and will be important to display?
What makes it an anomaly? How long is the anomaly, is it 5 minutes or a longer duration. A trend report (data, unit, equipment, timeframe). The graphs need to be included, different percentages, and what got ignored.

6) What is your wish list for a new dashboard?
Answered in above questions.

7) Who is the typical user of the current dashboard?
Pran/Shaun (Engineering & Shared Services)

<u>Summary of Wireframe</u>
MNG is keen to have a simple UI that takes in inputs, plots those inputs and wants a way to export the plotted predictions. Currently, they do not need many features and filters but want an easy-to-navigate screen for plotted predictions and alerts.