Zhejiang University/University of Illinois Urbana-Champaign Institute

Senior Design Individual Report

MACHINE LEARNING BASED WEATHER FORCASTING SYSTEM ON ARDUINO

By Zheyu Fu

Individual Report for Senior Design, Spring 2023

Sponsor: Cristoforo Demartino

TA: Wang Yi

22 May 2023

Project No. 26

Abstract

As a whole team, we have developed a machine learning-based weather forecasting system that provides accurate and timely predictions for our surrounding areas. By analyzing data from various sources, including weather sensors and historical records, our system generates real-time weather forecasts. Incorporating advanced techniques such as pattern recognition and artificial neural networks, we have created a comprehensive and accurate weather forecasting model. Our system continuously learns from new data, improving prediction accuracy and adapting to changes in weather patterns. Our objective is to overcome the limitations of traditional weather stations and provide more reliable, location-specific forecasts. We envision our system as an accessible and user-friendly tool, with a web application that allows users to access weather forecasts on the go and make informed decisions based on up-to-date weather information.

This report is individual, so it will mostly contain the work that I am involved in. To be more specific, I will introduce the design, verification as well as conclusion of the power supply module and wireless transmission module for our project.

Keywords: machine learning, weather forecast, power supply.

Contents

1. Introduction (Whole Project)

The introduction part will introduce the whole project that is done by our whole group.

1.1 Background

We develop a weather forecasting system that leverages the power of machine learning to produce accurate and timely weather predictions for our surrounding areas. Our system will analyze vast amounts of data from various sources, including weather sensors and historical weather data, to generate real-time weather predictions. We also incorporate advanced techniques such as pattern recognition and artificial neural networks to create a comprehensive and accurate weather forecasting model. The system can continuously learn from the new data to improve the accuracy of the predictions, making it a self-adaptive system that can handle changes in weather patterns over time.

A weather forecast system can be created by using a few different hardware components and software tools. Our solution mainly consists of two parts: weather stations and forecasts. As to the weather measurement and data collection part, temperature, humidity, barometric pressure, and rain sensors are considered the main components. For the other part, a Machine Learning (ML)-based algorithm is to be applied for data analysis and weather predictions. Also, we visualize our timely weather prediction on a website that largely promotes the practicality of our product.

Hardware part: Due to the complexity of weather conditions, our system incorporates the following weather indicators and their corresponding collectors: a humidity and temperature sensor, a barometric pressure sensor, a rain sensor, a light sensor, a PM2.5 Air Quality sensor, a UV detector, and an anemometer for wind speed. The equipment will be integrated into Arduino, and covered with a waterproof enclosure. All the sensed indicators will be used for weather prediction and be transmitted to our database by our data transmission module. The data transmission module consists of wireless transmission from outdoor Arduino and indoor Arduino and wired serial transmission from indoor to personal computer via serial port. Besides, there is an OLED display screen to showcase the real-time sensed weather data. Our power supply for the weather station consists of batteries, a voltage booster, solar panels as well as a solar panel charging circuit.

Software part: A practically usable weather forecast system is supposed to make reliable predictions for real-world multi-variable weather conditions. We apply Machine Learning techniques to suffice such generalization to unseen data. To this end, a high-quality dataset for training and evaluating the

Machine Learning model is required, and a specially designed Machine Learning model would be developed on such a dataset. After a preliminary investigation, we have downloaded Haining's weather datasets for the most recent 40 years from the OpenWeather platform. And we believe auto-regressive models could serve as practical solutions for our model. Once a well-trained machine-learning model is obtained, we will deploy the model on portable devices with easy-to-use APIs. Moreover, we develop a web application with an elegant display and user-friendly designs to demonstrate the real-time sensed data as well as our weather prediction. Our website is based on Vue as front-end and Django as back-end.

1.2 Objective

Great Weather Forecasting Accuracy and Accessibility: The accuracy of weather forecasting plays a crucial role in making informed plans and preparations, especially in regions where traditional weather stations may not be available or their predictions are unreliable due to distance and altitude factors. Our primary objective is to build a machine learning-based system that overcomes these limitations and provides more reliable and location-specific weather forecasts for our community. By leveraging advanced algorithms and data analysis techniques, we aim to enhance the accuracy of our predictions and deliver reliable weather information to users.

Great Accessibility and User-Friendliness: By developing a user-friendly OLED display screen, we aim to cater to the needs of users and provide them with intuitive tools to interact with the weather data effectively. We also develop a web application as part of our system. This application will allow users to access weather forecasts on the go, providing them with up-to-date and real-time weather information. By incorporating user-friendly features and a responsive design, our objective is to empower users to make informed decisions based on the most current weather data available.

1.3 High Level Requirements

1. The weather measurement prototype with sensors should be able to accurately collect the temperature, humidity, and barometric pressure. etc. To be more specific, our collected weather parameters will be compared to real-time data from a meteorological station and should be within a reasonable margin of error. For example, temperature error should be within 5\%, humidity error, and barometric error within 10\%.

2. A machine learning algorithm should be successfully trained to make predictions on the weather conditions: rainy, sunny, thunderstorm, etc. This is a multi-classification problem and we hope our macro-average precision could reach 0.75 or higher.
3. Our system can collect and forecast the weather in Haining, in real-time, and/or longer-period forecasts. To be specific, our hardware subsystem could collect and upload real-time weather information to the software subsystem every 30 minutes. And our software subsystem could predict the following 24 hours' weather conditions based on the recently inputted 48 hours weather parameters. For longer-period prediction, our system will be able to predict the weather up to 7 days later, but it could be a great challenge to predict accurately in this case as you can imagine.
4. The forecast weather information could be demonstrated elegantly through some UI interface. A display screen would be a baseline, and a web application on a phone or PC would be extra credit if time permitted.

2 Design (My Own Parts)

2.1 [Block Diagram]

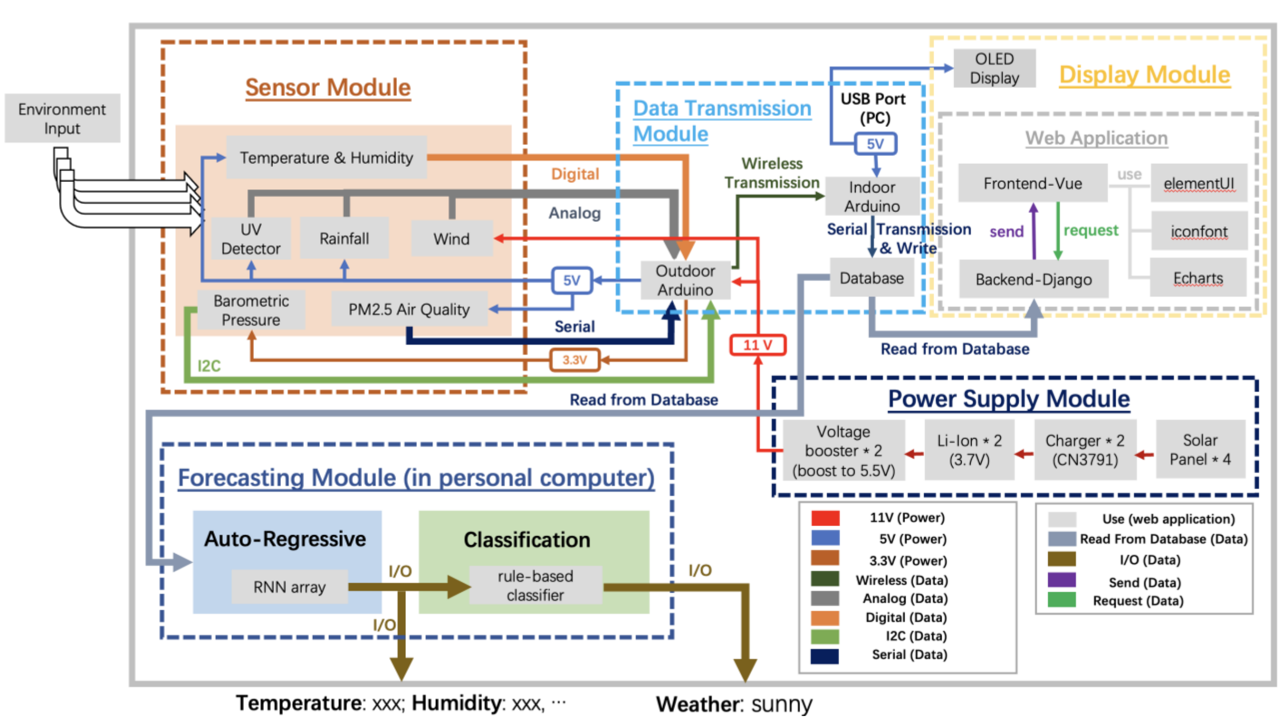

Figure 1: block diagram for our entire system

This is the block diagram for our whole system. Our design consists of two parts: a data collection system and forecasting. The power supply module consists of a lithium-ion battery with a boost converter and a solar-powered battery charger. Five different sensors are used to retrieve environmental data, and those data will be transmitted to our data integration sub-system through an analog signal. We use Arduino to integrate those incoming data and then transmit them via a wireless communication module to the indoor Arduino receiver. Finally, the receiver uses a serial interface to send those data to a personal computer for further training and forecasting. As for our forecasting system, we leverage an RNN-based model to process time series and make predictions, and a final weather condition would be determined by a rule-based classifier given the predictions. Details are shown in subsequent sections.

As for my own contribution, I take the charge of power supply module and the wireless transmission module.

2.2 [Power Supply Module]

There are two objectives for this module. Firstly, to power up the weather station with stable voltage output, and secondly to make use of solar energy to make our system sustainable.

For our outdoor weather station, the power source of the Arduino Board is 2 * 3.7V 4000mAh batteries with a DC-to-DC Converter Module to boost the voltage from 3.7V to 5.5V to finally support stable 11V (2 * 5.5V) for our outdoor Arduino and anemometer. Other sensors will be directly powered by outdoor Arduino. Arduino provides three voltage levels, including 3.3V for the water-level sensor, and 5V for the remaining components. Battery-power supply is shown at the top left of figure 3. For the indoor receiver, Arduino will be charged with the USB cable.

To ensure that our system won't suffer from failure because of a dead battery, a solar panel charging system is designed to ensure the sensor module with the Arduino is powered continuously. The circuit is shown below in figure2. The Lithium-Ion battery will be charged through a charger, powered by the solar panels to make our system sustainable.

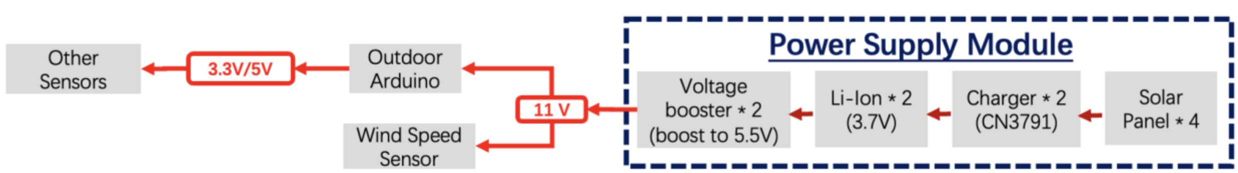

Figure 2: block diagram for power supply module

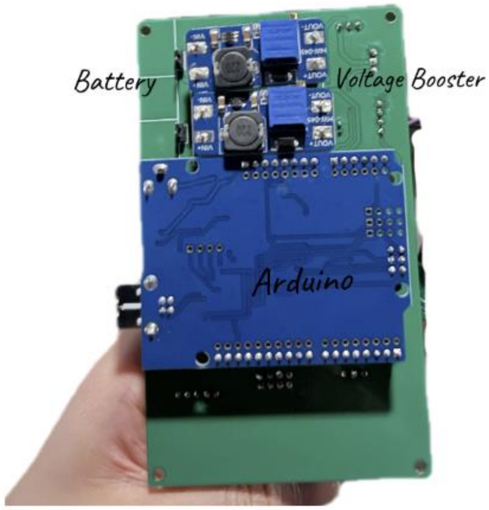

Figure 3: PCB board and voltage booster

In my first design for our solar panel charging system, I choose to use TP4059 as the controller. However, the charging performance is not so good, with only about 80mA when sunny. So I choose to design another version of charging circuit using another controller, CN3791, and it turns out to work well, with about 260mA of charging current in cloudy days and about 400mA in sunny days.

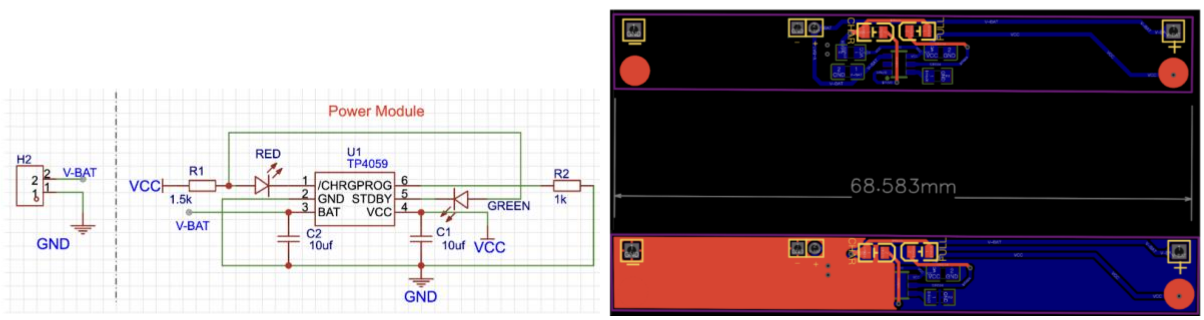

Figure 4： First version circuit diagram and PCB layout

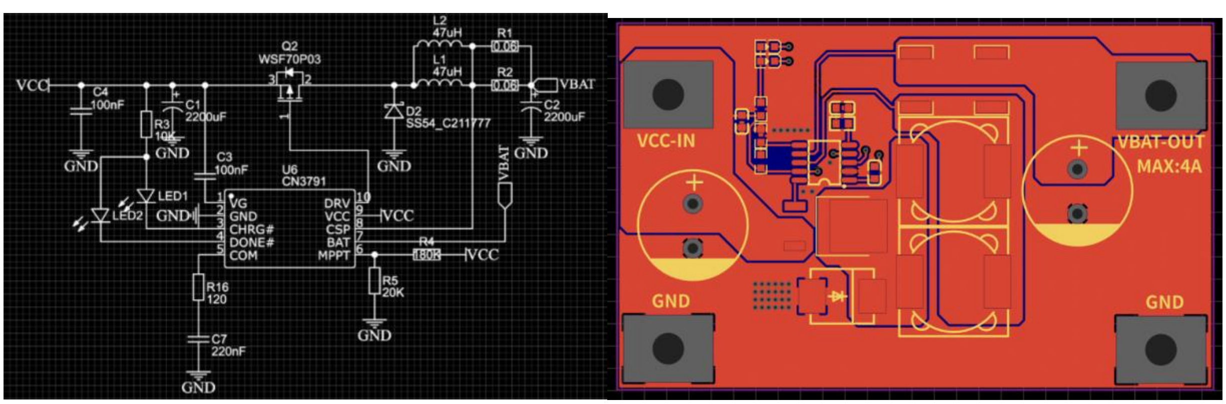

Figure 5: second version circuit diagram and PCB layout

2.3 [Wireless Transmission Sub-Module] (corporate with Xuanyu Chen)

Weather data collected by sensors needs to be transmitted to our personal computers for machine-learning weather prediction. Since our sensors will be placed in an outdoor environment that probably gets caught in rain, we decide to use wireless communication (not traditional wire connection) to let the data come indoors. Also, we would like our sensing information could be demonstrated through an OLED screen real-time. Consequently, we design our data transmission module to be two sub-modules, wireless transmission sub-module and wired transmission sub-module. And my work is mainly in wireless transmission sub-module.

As Figure 6 demonstrates, I designed to use NRF24L01, an SPI-connected digital transceiver, to wirelessly transmit weather data from the left Arduino, which is placed outdoors with weather sensors, to the right Arduino which is placed indoors near our personal computer.

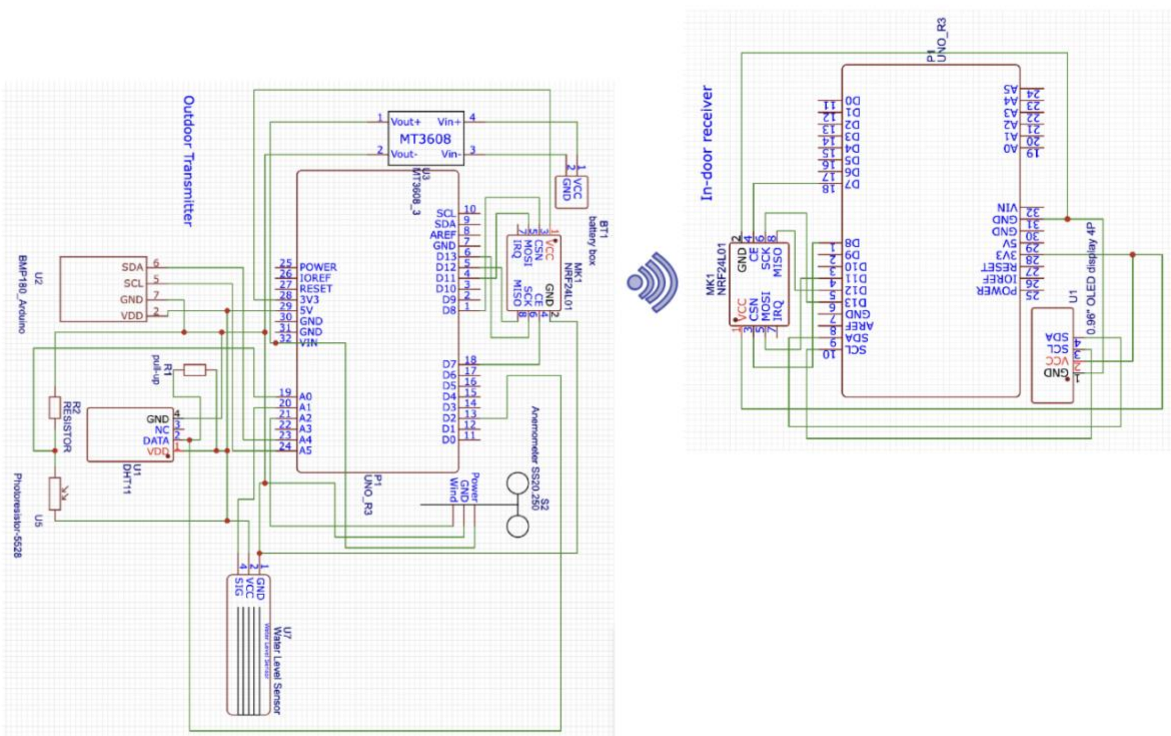

Figure 6: Wireless communication schematic

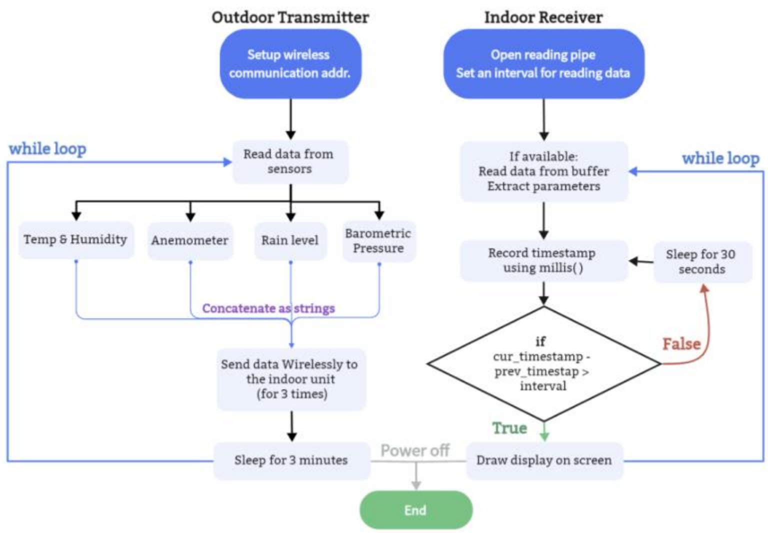

Figure 7: Workflow of weather data collection and transmission

3. Design Verification (My Own Parts)

3.1 [Verification of Power Supply Module]

In our design, the whole power supply module should support the sensing module at a stable voltage that is near 11V. In the final demo, we measure the voltage difference between VCC and ground pins on our main PCB board and it shows a stable 10.98V that is very close to 11V. So, this requirement is met.

Another objective for this module is that it should make use of solar energy to make our system sustainable. Firstly, we need to know the working current or working power of our weather station. We did a theoretical analysis of the working power of our weather station as Table 1 shows, which reveals a working power of nearly 1W, and the working current flowing through each battery to be nearly 135mA. We also manually test the working current that flows out of each battery, which is nearly 100mA. Secondly, we test the charging current that flows into each battery and find that the charging current when in sunny is about 400mA with our second version solar charging circuit. Assuming there are 12 hours of sunlight each day, the charging current should be at least 200mA to make our system sustainable, and our charging current is enough actually. So, this objective is also met.

Components	Power
Arduino Uno	Power: $\approx 450mW$ from Youtube [2]
Wind speed sensor	Power: $\leq 300mW$ from its data sheet [3]
Water level sensor	Working Voltage: $5V$ Working Current: $\leq 20mA$ Power: $\approx 100mW$
DTH11: Temperature and Humidity Module	Working Voltage: $5V$ Working Current: $0.3mA$ when measuring, $60\mu A$ when standby Power: $\approx 0.3mW$
BMP180 Digital pressure sensor	Working Voltage: $3.3V$ Working Current: $5\mu A$ Power: $\approx 0.02mW$
Light detector sensor	Working Voltage: $5V$ Working Current: $\leq 5\mu A$ Power: $\approx 0.03mW$
nRF24L01 Wireless Module	Working Voltage: $3.3V$ Working Current: $\leq 13.5\mu A$ Power: $\approx 44.55mW$
Other auxiliary resistors	Power: $\leq 100mW$

Table 1: Theoretical Working Power Analysis

Conditions	Theoretical WC	Real WC	CC with Version 1 charger	CC with Version 2 charger
Current	around $135mA$	around $100mA$	around $80mA$	around $400mA$

Table 2: Working/Charging current that flows out of/into each battery. WC stands for working current, and CC stands for charging current. Charging currents are tested under median strong sunlight on sunny days.

3.2 [Verification of Wireless Transmission Sub-Module]

We expected our wireless transmission distance to be at least 8 m without any barrier in our design document. To our excitement, the maximum wireless transmission distance is much better, reaching at most 28m without any barriers. So, this requirement in our design document is met.

4. Conclusion (Whole Project)

4.1 Accomplishments

With unwavering determination and a collaborative spirit, our team has successfully completed this project that has surpassed all expectations.

One of our key accomplishments was the development of a robust data acquisition module. We designed and implemented a sensor network that collected real-time weather data, including temperature, humidity, wind speed, and atmospheric pressure. These sensor readings served as the foundation for our forecasting models, ensuring that the predictions were based on the most up-to-date and relevant information. Also, we designed a double-layer waterproof shelter to keep our sensors safe and sound.

To facilitate the Machine Learning aspect of our project, we meticulously designed and trained forecasting models using state-of-the-art algorithms. Through a rigorous process of data preprocessing, feature selection, and model training, we developed accurate and efficient prediction models. Our team explored various techniques such as regression analysis, time series forecasting, and ensemble methods to optimize the performance of our models.

Throughout the project, we emphasized the importance of usability and user experience. We developed an intuitive graphical user interface (GUI) that displayed the weather forecasts in a clear and accessible manner. This user-friendly interface allows users to easily interpret the predictions and make informed decisions based on the forecasted weather conditions.

4.2 Uncertainties

One major uncertainty lies in the accuracy of our weather forecasting models. While we have invested considerable effort in training and optimizing our models, there is always the possibility of unforeseen variations or outliers in weather patterns. Factors such as sudden changes in atmospheric conditions, complex local topography, or the emergence of extreme weather events may pose challenges to the accuracy of our predictions. To address this uncertainty, we can consider implementing an ensemble modeling approach, where multiple forecasting models are combined to provide more robust and reliable predictions. By leveraging the collective wisdom of diverse models, we can mitigate the impact of potential inaccuracies in individual models.

Additionally, uncertainties may arise regarding the robustness and reliability of our sensor network. Weather sensors are susceptible to environmental factors such as temperature variations, humidity, and physical damage, which can affect the accuracy and consistency of the collected data. To mitigate these uncertainties, we can implement redundant sensors or periodic calibration procedures to ensure the quality and reliability of the data. Furthermore, regular maintenance and monitoring of the sensor network will be essential to identify and address any issues promptly.

Lastly, uncertainties may arise in the usability and user acceptance of our forecasting system. While we have developed a user-friendly graphical interface, there is always the possibility of users encountering difficulties in interpreting or navigating the system. To mitigate this uncertainty, we can conduct user testing and gather feedback to identify areas of improvement. Incorporating user feedback and making iterative design refinements will help enhance the usability and overall user experience of the system.

4.3 Ethical considerations

According to the IEEE Code of Ethics 1, we should hold paramount the safety, health, and welfare of the public, to strive to comply with ethical design and sustainable development practices. Following this, We are committed to designing our weather forecast equipment to be both practical and ethical. We hope our design could be a next-generation solution for a mini-size meteorological station, and we will try to prevent any form of behavior that may endanger public order with our products.

Also, the IEEE Code of Ethics states that engineers shall maintain confidentiality and protect the privacy of others. We ensure that we use legally available datasets, like Haining's weather conditions from the OpenWeather platform, for training our machine learning models. To avoid ethical breaches, we may implement appropriate measures to safeguard the privacy of data, such as encryption and secure storage.

Moreover, in line with the IEEE Code of Ethics 4, it is important to actively seek, accept, and provide sincere critiques of technical work while acknowledging and rectifying any mistakes. We should be truthful and practical when making statements or predictions based on available data and give appropriate recognition to the contributions of others.

4.4 Future work

The prospects of our project lie in its integration and interoperability with other intelligent systems. Currently, our system's output consists of five variables, which we envision as being utilized as inputs for other systems.

Several potential applications of our system have been considered, all of which involve the integration of our system with smart devices. One such scenario pertains to the situation where an individual departs from their residence in the morning and is unexpectedly confronted with rainfall in the afternoon. In this case, our Where the Focus system exhibits a remarkable ability to accurately predict the occurrence of rainfall and autonomously initiate the closure of windows. Distinguishing itself from existing products, which possess the capacity to solely monitor weather conditions and react accordingly, our innovative forecasting system proactively mitigates potential losses by promptly taking appropriate measures.

Alternatively, it is a widely acknowledged fact that larger buildings necessitate a lengthier cooling-down period during the summer season. Within our library, the issue of temperature regulation frequently elicits complaints from patrons, and the response time for adjusting the temperature settings following these complaints is excessively protracted. In addressing this challenge, our system adeptly leverages predicted data to modulate the output power, thereby eradicating the delay and ensuring expeditious and efficient temperature control.

References

[1] IEEE. “IEEE Code of Ethics.” (2016), [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html (visited on 03/07/2023).