IoT System Development Lab

Project Report

Early Warning System for Hydro Power Plant

विद्या तत्व ज्योतिसमः

M.Tech CSE

Team Members:

1. Sidhant Moza (23303026)
2. Ehtesham Ashraf (23003021)
3. Jaivardhan Singh (23303014)
4. Noor Mohammad (23303003)

Table of Contents

Abstract

Flooding poses a severe threat to hydro power plants, especially along the Alaknanda River in Vishnuprayag, emphasizing the need for an Early Flood Warning System. This study presents a comprehensive solution utilizing Machine Learning (ML) and Internet of Things (IoT) technologies. The proposed system integrates a machine learning model for flood prediction and a Node RED flow monitoring system. Through MQTT communication, real-time data streams are processed, enabling prompt alarm generation. Results indicate robust performance, with the LSTM model achieving a training accuracy of 96% and real-time alerts ensuring effective flood management.

Introduction

Flooding remains a recurrent and devastating natural disaster, causing substantial economic losses and endangering human lives across the globe. The Alaknanda River, particularly in the vicinity of Vishnuprayag, presents a significant risk due to its susceptibility to flooding. The presence of a dam approximately 20 kilometers from the observation site adds complexity to the situation, as timely closure of the dam gates is critical to mitigating the impact of floods downstream. However, relying solely on human observation and intervention introduces the potential for errors, which could lead to catastrophic consequences.

In response to this challenge, our project aims to develop an Early Flood Warning System utilizing a combination of cutting-edge technologies, including Machine Learning (ML), Internet of Things (IoT), MQTT (a lightweight messaging protocol), and data flow management mechanisms. By harnessing these technologies, we endeavor to provide timely and accurate warnings to authorities responsible for managing the dam, thereby enabling them to take proactive measures to minimize the impact of flooding.

Literature Review

Sr. No.	Title of the paper	Methodology & Result
1	IoT-Enabled Flood Severity Prediction via Ensemble Machine Learning Models (2020) [2]	● Data for a period of 31 years (4214 instances) were collected from the Environment Agency website. It has three classes (normal, abnormal and dangerous water level). ● Ensemble learning using the Long-Short Term memory model and random forest outperformed individual models with a sensitivity, specificity and accuracy of 71.4%, 85.9%, 81.13%, respectively.
2	Smart Early Flood Monitoring System Using IoT (2022) [1]	● An LCD display was used in this project to show all of the detection in the environment, namely temperature, humidity, rainfall, water level, and water level increase rate. Temperature appears first on the LCD monitor, followed by humidity, rainfall detection, water level, and water increase rate. The water level is the main parameter that makes the difference between flood and no flood. Then the flood prediction is done using IoT enabled sensor data and machine learning with the help of MATAB software. ● The curve shows the expected humidity and water level on November 19: 80% and 35 cm, respectively. That means there is a high chance of flooding in that area.
3	Designing Early Warning Flood Detection and Monitoring System via IoT (2020) [4]	This product utilizes an ultrasonic sensor to detect water levels, transmitting the data to a mobile app via IoT. GPS connectivity enables accurate location tracking of flooding. NodeMCU manages water levels, activating a solenoid valve to pump out excess water at warning levels and sending notifications to users. At critical levels, users are alerted to take precautionary measures before flooding worsens.
4	Smart IoT Flood Monitoring System (2019) [3]	Ultrasonic HC-SR04 module sensor was used to detect the water level of the river. The NXP LPC1768 is a microcontroller that collects all the data in this system. Buzzer will act as an alarm to alert the public and authority when there is an upcoming flood and data updates to the web server. The function of LCD and LED are to display and indicate the water level.

Problem Statement

The urgency for an Early Flood Warning System arises from the critical necessity to address inherent challenges posed by a very short reaction time after occurrence of flood events at hydro power plants, particularly along the Alaknanda River in Vishnuprayag. Current reliance on manual observation methods and inadequate data recording infrastructure exacerbates risks, leading to potential human error and delays in decision-making. The lack of timely flood detection mechanisms results in operational disruptions and sometimes even damage to power plant infrastructure, incurring significant financial losses. Leveraging machine learning algorithms, IoT sensors, and real-time data analysis techniques presents a compelling solution to enhance preparedness and response capabilities, ensuring proactive disaster management and minimizing adverse impacts on power plant operations and surrounding communities.

Tools & Techniques

1. **Software Tools**
    - Node-RED
    - Python
    - JavaScript
    - Mosquitto
    - Machine Learning (LSTM)
    - Proximity Sensor

2. **Protocols to be used**
    - Communication Protocol - MQTT
    - Server communication - Mosquitto (publish/subscribe)

3. **Data Storage & Analysis**
    - Cloud (unstructured collection of data)
    - Pickle data files (python checkpoint instances)
    - CSV data files (structured data)
    - JSON (structured objects)

Idea and Design of Project

Idea

1. To train a machine learning model for automated flood prediction and flood detection in Alaknanda River with large reaction time.
2. To create a Node RED flow monitoring and displaying a dashboard and generate alarm in case of danger from flood
3. To generate a simulation of injecting real-time sensor data into flow and predicting probability of flood.

Design

a) Node RED Dashboard and MQTT

The Node-RED flow architecture consists of several components for monitoring and processing incoming data streams.

Here's a breakdown of the flow:

1. MQTT Input Node : Receives incoming data streams from the topic "Incoming data stream".

2. Function Node : Segregates the temperature data from the incoming payload and sends it to various outputs, including a gauge and a line chart. It also sends the temperature data to a debug node.

3. File Node : Appends the received data to a CSV dataset.

4. Function Node : Checks if the temperature or measured features exceeds a certain threshold and raises an alert if it does.

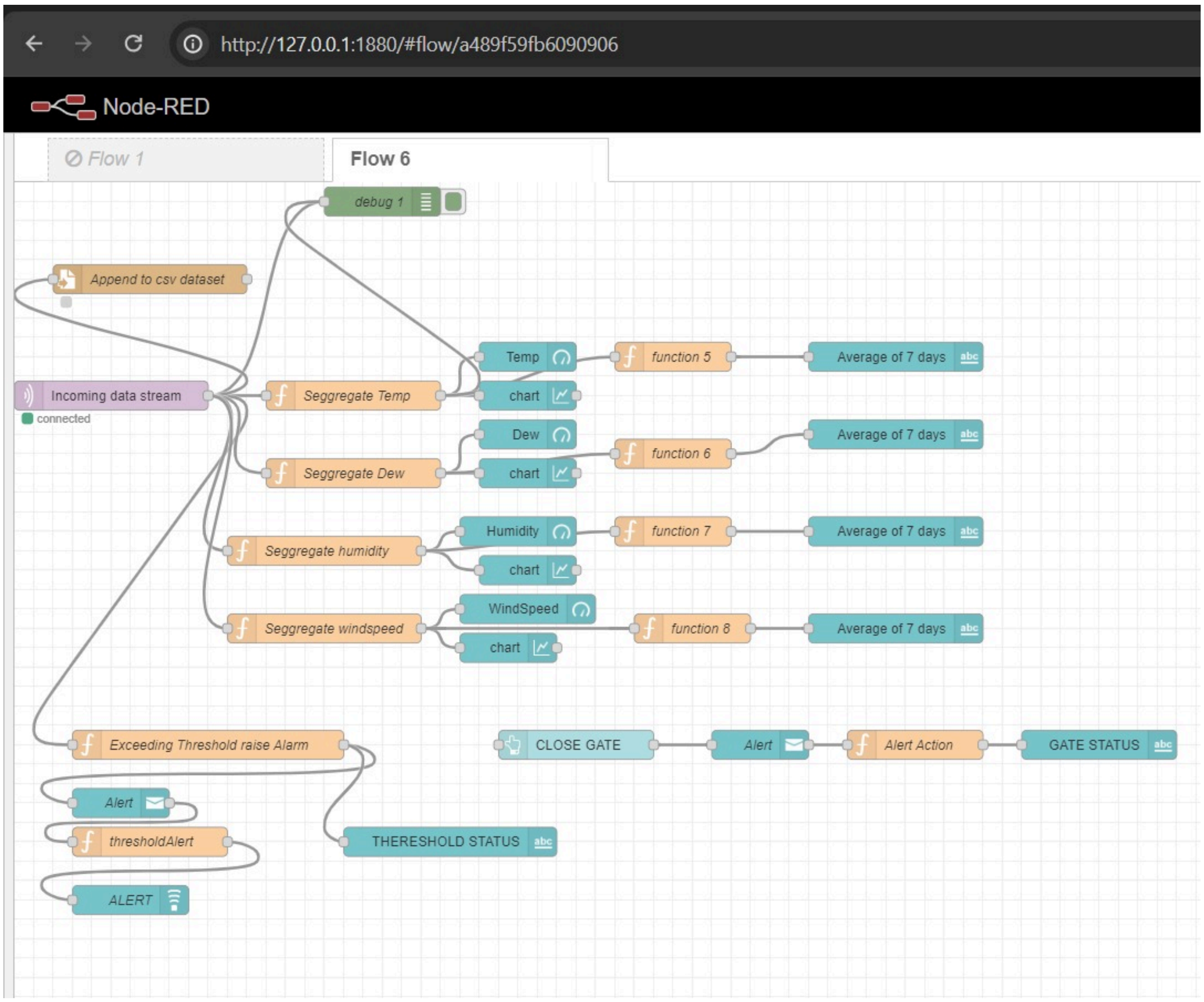
5. Trigger Function Node : Raises an alarm if the threshold is exceeded.

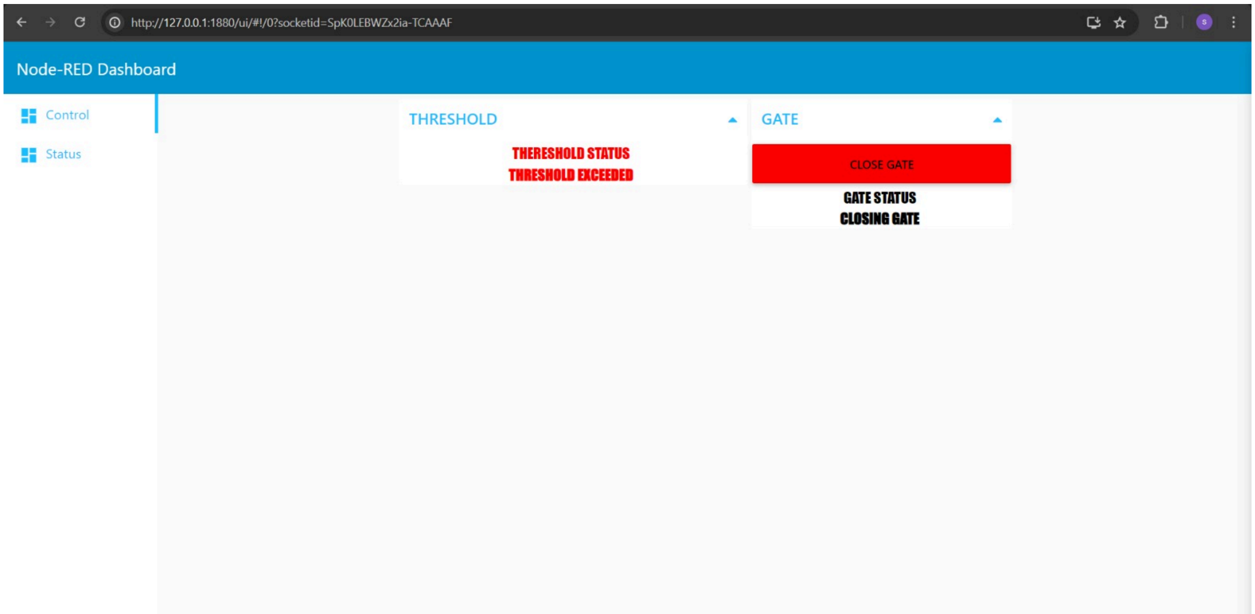
6. UI Toast Node : Displays a dashboard notification for the raised alarm.

7. UI Audio Node : Plays an alert sound in case of an alarm trigger.

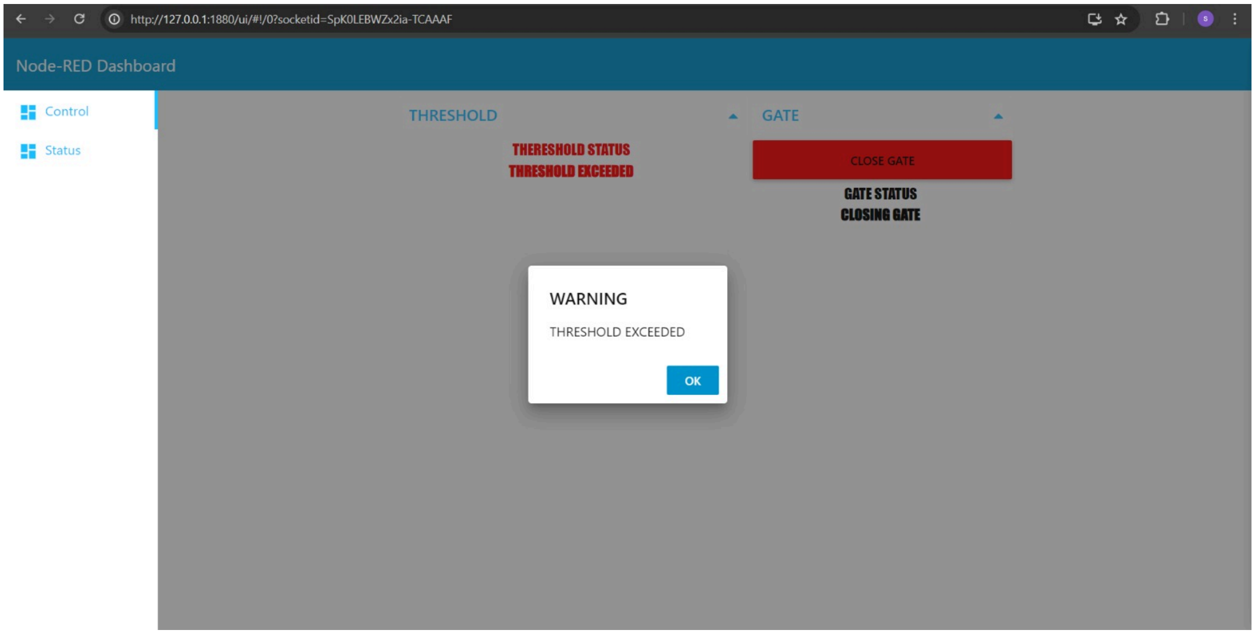
8. UI Button Node : Enables manual closing of a gate empowering with fast response in case of flooding.

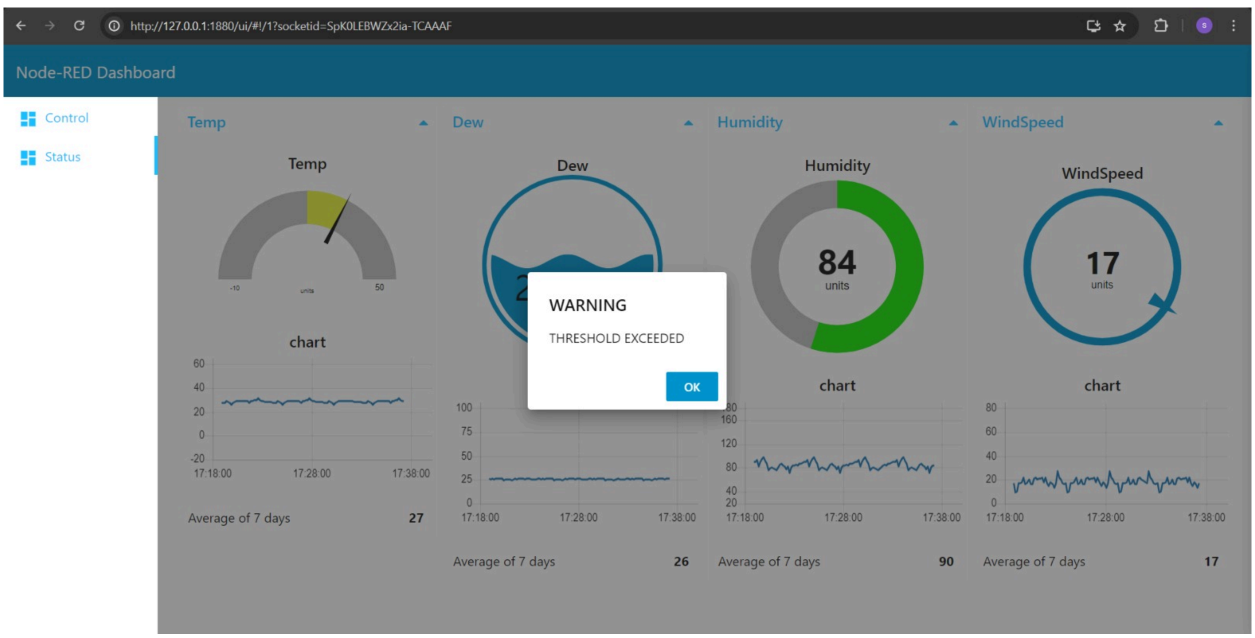
As for the dashboard, it includes various UI components such as gauges, charts, text fields, and buttons. These components visualize real-time data streams and provide controls for manual actions, such as closing a gate. Additionally, the dashboard incorporates toast notifications and audio alerts for important events, such as threshold exceedances. Each UI component is organized into different groups based on their functionality, such as temperature, dew, humidity, wind speed, threshold, and gate status.




http://127.0.0.1:1880/ui/#!/0?socketid=SpK0LEBWZx2ia-TCAAAF
Node-RED Dashboard
Control
Status
THRESHOLD
THERESHOLD STATUS
THRESHOLD EXCEEDED
GATE
CLOSE GATE
GATE STATUS
CLOSING GATE


http://127.0.0.1:1880/ui/#!/1?socketid=SpK0LEBWZx2ia-TCAAAF
Node-RED Dashboard
Control
Status
Temp
Temp
chart
Average of 7 days
27
Dew
Dew
26units
chart
Average of 7 days
26
Humidity
Humidity
85
units
chart
Average of 7 days
90
WindSpeed
WindSpeed
21
units
chart
Average of 7 days
17





b) Machine Learning Model

This phase dealt with the development and training of a machine learning model for predicting the occurrence of floods in proximity to the power plant. The model utilized a range of weather parameters as input features, including temperature, dew point, humidity, sea level pressure, wind speed, wind direction, and solar radiation. The dataset employed for model training spanned five years of daily weather records (2016-2020) from the state of Mumbai.

An extensive exploratory data analysis (EDA) was conducted on the dataset to discern underlying patterns and relationships among the variables. Following EDA, data preprocessing was executed, which encompassed addressing missing values through appropriate imputation techniques and standardizing the data using standard scaling methods to facilitate model convergence.

Subsequently, a Long Short-Term Memory (LSTM) neural network model was deemed suitable for capturing temporal dependencies within the sequential weather data. The LSTM architecture was initialized with specifications as detailed in the provided model summary with 100K parameters. Notably, the input sequences were structured to encapsulate information from the preceding 20 days, thereby enabling the model to forecast the probability of flood occurrence on the subsequent day.

```
Model: "sequential_3"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 bidirectional_2 (Bidirectio  (None, 20, 128)          38912
 nal)

 dropout_7 (Dropout)         (None, 20, 128)           0

 lstm_8 (LSTM)               (None, 20, 64)            49408

 dropout_8 (Dropout)         (None, 20, 64)            0

 lstm_9 (LSTM)               (None, 32)                12416

 dropout_9 (Dropout)         (None, 32)                0

 dense_3 (Dense)             (None, 1)                 33

=================================================================
Total params: 100,769
Trainable params: 100,769
Non-trainable params: 0
_________________________________________________________________
```

To mitigate overfitting during model training, an early stopping callback mechanism was integrated, configured with a patience parameter set to 15 epochs. The choice of loss function and optimizer was governed by the nature of the task, with 'binary_crossentropy' serving as the appropriate loss metric and the 'adam' optimizer being employed to optimize model parameters.

Finally, the trained LSTM model was serialized and deployed within the Node RED application infrastructure. This deployment facilitated real-time prediction of flood occurrences utilizing weather data sourced from various simulated sensors integrated within the Node RED environment, thereby enhancing the proactive monitoring capabilities of the hydro power plant's early warning system.

Results

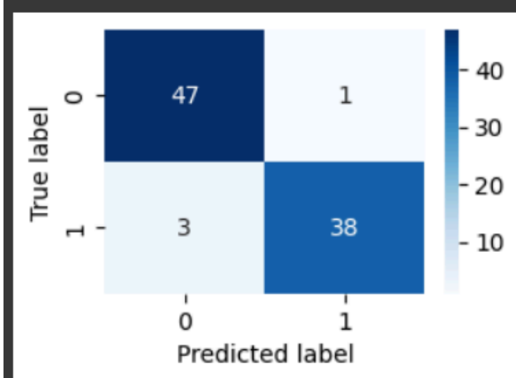
Training of the LSTM model was executed over 1000 epochs, resulting in commendable performance metrics. The model achieved a training accuracy of 96% and a validation accuracy of 91%. Upon evaluation on a hold-out test dataset, the model exhibited robust performance, attaining an accuracy of 95.5%. The classification report and confusion matrix of the testing results is shown below.

```
Classification Report:
              precision    recall  f1-score   support

           0       0.94      0.98      0.96        48
           1       0.97      0.93      0.95        41

    accuracy                           0.96        89
   macro avg       0.96      0.95      0.95        89
weighted avg       0.96      0.96      0.95        89

Accuracy : 95.50561797752809%
```



The results obtained indicate that the IoT setup, orchestrated through Node-RED, functions seamlessly. Utilizing machine learning model predictions, alarms are promptly raised when necessary. The system demonstrates effective operation across all functions. Crucially, the communication framework between the dashboard, machine learning model, and the connected devices operates on a publish/subscribe model, ensuring efficient data exchange. Throughout the process, communication links are consistently established successfully. Updates are transmitted to subscription channels timely, affirming the system's reliability in delivering real-time information and actions.

Conclusion and Future Scope

Conclusion:

Summary of Findings: The research endeavors addressed the critical necessity for an Early Flood Warning System, particularly crucial along the Alaknanda River in Vishnuprayag due to the very short reaction time post-flood events. Manual observation methods and inadequate data recording infrastructure exacerbate risks, resulting in potential human error, delays in decision-making, and operational disruptions at hydro power plants. Leveraging machine learning algorithms, IoT sensors, and real-time data analysis techniques offers a compelling solution. Through the implementation of a machine learning model for flood prediction and detection, coupled with a Node-RED flow monitoring system and dashboard, the research successfully demonstrated the effective operation of an Early Flood Warning System.

Limitations: Despite the successful implementation of the Early Flood Warning System, certain limitations were encountered during the research process. These include methodological constraints in data preprocessing and model training, as well as the need for further refinement in addressing real-time data challenges and optimizing the system's performance in diverse environmental conditions.

Practical Applications: The research findings have significant practical implications for disaster management authorities, hydro power plant operators, and policymakers. The developed Early Flood Warning System offers a valuable tool for proactive decision-making, enabling timely response to flood events and minimizing operational disruptions. Additionally, the system's communication framework and real-time monitoring capabilities can be adapted for use in various other disaster management scenarios, contributing to overall resilience and sustainability.

Concluding Remarks: In conclusion, the research underscores the importance of integrating advanced technologies and data-driven approaches in disaster management strategies. By addressing the inherent challenges of timely flood detection and response, the developed Early Flood Warning System offers a scalable solution with broad implications for enhancing resilience and minimizing the adverse impacts of natural disasters on critical infrastructure and communities.

Future Scope:

The future scope of the project holds promising avenues for further enhancement and integration, particularly with the inclusion of Arduino-based implementations. Below mentioned are some potential directions:

1. Hardware setup with Arduino: Integrate Arduino microcontrollers to interface with various sensors and actuators. Arduino's extensive ecosystem offers a wide range of modules for measuring

temperature, humidity, wind speed, and controlling devices such as gates or alarms. This expansion allows for a more diverse and comprehensive IoT setup.

2. Real-time Data Processing: Utilize Arduino's real-time processing capabilities to perform rapid data analysis on the edge. For example, implement algorithms to detect anomalies or patterns in sensor data directly on the Arduino boards, enabling faster response times for critical events.

4. Wireless Communication Protocols: Implement wireless communication protocols such as Bluetooth, Zigbee alongside MQTT for efficient data transmission between Arduino nodes and the central Node-RED server. This enables the deployment of distributed sensor networks over large areas with minimal infrastructure requirements.

By integrating Arduino-based implementations into the project, it opens up opportunities to enhance functionality, improve reliability, and extend the reach of the IoT system. The combination of Node-RED's flexibility with Arduino's hardware capabilities forms a powerful foundation for building scalable and adaptable IoT solutions.

Future research endeavors should focus on refining the machine learning model for flood prediction, incorporating additional environmental variables and refining the communication framework to enhance system reliability and responsiveness. Furthermore, exploring the integration of advanced sensor technologies and predictive analytics could further enhance the accuracy and efficiency of flood detection mechanisms.

References

[1] M. E. Deowan, S. Haque, J. Islam, M. Hanjalayeamin, M. T. Islam and R. Tabassum Meghla, "Smart Early Flood Monitoring System Using IoT," 2022 14th Seminar on Power Electronics and Control (SEPOC), Santa Maria, Brazil, 2022, pp. 1-6

[2] M. Khalaf et al., "IoT-Enabled Flood Severity Prediction via Ensemble Machine Learning Models," in IEEE Access, vol. 8, pp. 70375-70386, 2020

[3] Zahir, S. B., Ehkan, P., Sabapathy, T., Jusoh, M., Osman, M. N., Yasin, M. N. M., Wahab, Y. A., Hambali, N. a. M. A., Ali, N., Bakhit, A. S. A., Husin, F., Kamil, M., & Jamaludin, R. (2019). Smart IoT Flood Monitoring System. *Journal of Physics. Conference Series*, *1339*(1), 012043. https://doi.org/10.1088/1742-6596/1339/1/012043

[4] Hadi, M. I., Yakub, F., Fakhrurradzi, A., Hui, C. X., Najiha, A., Fakharulrazi, N. A., Harun, A. N., Rahim, Z. A., & Azizan, A. (2020). Designing early warning flood detection and monitoring system via IoT. *IOP Conference Series. Earth and Environmental Science*, *479*(1), 012016. https://doi.org/10.1088/1755-1315/479/1/012016