Applying Machine Learning for Geographic Forecasts in Singapore's Maritime and Aviation Industries

Team ID: Lab P13 - 2

Ang Xuan Yu Pamela	Crystal Ng Jing Jing	Elton Tan	Justin Tan	Kok Zi Xin
2401870	2401398	2401545	2402149	2402212
2401870@sit.singaporetech.edu.sg	2401398@sit.singaporetech.edu.sg	2401545@sit.singaporetech.edu.sg	2402149@sit.singaporetech.edu.sg	2402212@sit.singaporetech.edu.sg

***Abstract*—In Singapore, real-time weather data is critical in ensuring the safety of both aviation and maritime operations. Current weather forecast data lacks the necessary precision for region-specific decision-making, particularly concerning wind speed. This project aims to develop a data visualization platform that integrates real-time weather data with geographic data. The interactive interface will help professionals in the aviation and maritime sectors analyze region-specific weather conditions.**

Keywords—Wind Speed, Weather Forecast, Data Analysis, Data Visualisation, Singapore, Machine Learning, Catboost Regression, Root Mean Square Error

Introduction

In Singapore, real-time wind speed predictions and weather forecasts are vital for the safety of the maritime and aviation industry. Wind and weather conditions, such as strong gusts and heavy thunderstorms can impact areas in these industries, like aircraft takeoffs. In contrast, strong wind and adverse weather conditions can make sea navigation hazardous. However, current geographic forecasting requires more details and regional specificity to address the geographic variability across the island. This limitation results in insufficient data for localized geographic-forecast-based decision-making. This project proposes developing a data visualization platform to provide a comprehensive and user-friendly solution for weather monitoring with the integration of real-time wind speed, weather forecasts, and geographic data, providing regional insights to enhance the industry's safety

Related works

This project will be built on top of the current weather data platform, with the enhancement of real-time interactive maps for specific regions in Singapore. The enhanced platform will include detailed weather data, with a key focus on wind speed, and real-time alerts for extreme weather.

Proposed approach and Methodology

The development of the proposed platform follows a structured methodology which consists of 5 key approaches:

1. Data Scraping: The data used for the platform is sourced from data.gov.sg, an open portal that provides access to real-time data. Using automated processing, relevant data is gathered such as wind speed and other necessary geographic parameters that are required for our system

2. Data Cleaning: Raw data obtained from scraping undergoes data cleaning to ensure consistency and relevance, by removing incomplete data and maintaining consistency across different datasets by ensuring formats and structures are standardized for integration and analysis in subsequent stages

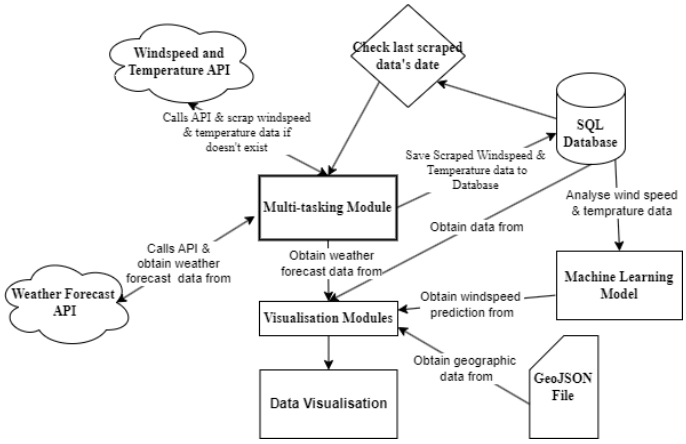
3. Data Storage: After cleaning, the data is stored in an SQL Database, structured for efficient querying and integration with other geographic data. This ensures persistent data can be easily accessed for real-time monitoring, machine-learning model training, and historical analysis.

4. *Model Training*: The data will be retrieved from the database and used by machine learning models to make forecasts on geographic data. Results will be evaluated based on their error rates, and parameters will be fine-tuned to ensure models make predictions as accurately as possible.

5. Data Visualization: The platform integrates with the GeoJSON files to generate an interactive map. Regions with normal weather conditions are represented in green while regions that expect extreme weather conditions will be alerted by highlighting in red.. Airports and seaports are annotated with icons on the map for quick reference and professionals in aviation and maritime can also view forecasted wind speed through an interactive line chart. When hazardous weather is forecasted, alerts and safety warnings will be triggered in the proximity of affected regions. Users can also zoom in on the map and click to view detailed reports on the weather.

The combination of real-time data, accurate geographic mapping, and under-friendly visualization reduces the time required for maritime and aviation professionals to assess and respond to weather conditions. By centralizing weather forecasts and geographical predictions in a visualization, it enables accurate decision making which can improve operational safety.

A. System Architecture Overview



The system architecture is designed to ensure efficient collection, storage, analysis, and visualization of weather data, focusing on wind speed and temperature. The system integrates multiple key modules that work in sync to automate the end-to-end process from data collection to visualization, ensuring real-time monitoring and insights.

1. Multi-tasking Module: Efficient Data Scraping

The system starts with the **Multi-tasking Module**, which manages the data scraping process by interacting with external APIs. This module starts by checking the **SQL Database** for the timestamp of the last successfully scraped data and compares it with the current time to identify any gaps. For any missing data points, the module retrieves updated wind speed and temperature data from the **Wind Speed and Temperature APIs**.

- **Handling Missing Data**: By continuously comparing timestamps and fetching any missing data, the Multi-tasking Module ensures the data set remains up-to-date.
- **API Integration**: The module is responsible for interacting with APIs, including the **Wind Speed and Temperature API** for real-time weather data and the **Weather Forecast API** for predictive data. The scraped data is then stored in the SQL Database for further processing.

2. SQL Database: Persistent Data Storage using MySQL Connector

The **SQL Database** is a crucial component of the system architecture, used for persistent storage of all collected weather data. This database stores historical wind speed and temperature data with corresponding timestamps. The system uses **MySQL Connector**, a Python library, to facilitate communication between the Python environment and the MySQL database.

- **MySQL Connector for Data Interaction**: The **MySQL Connector** library allows the Multi-tasking Module to efficiently store the scraped weather data in structured tables. It also enables the **Machine Learning Model** to retrieve historical data from the SQL Database for analysis and prediction purposes. Through efficient query handling, the connector ensures quick and reliable access to large amounts of weather data, enabling real-time decision-making.
- **Scalable Data Storage**: The SQL Database is scalable, making it ideal for storing large volumes of data collected over time, ensuring the system can handle high-frequency data without performance issues.

3. Machine Learning Model: Predictive Analysis

Once the data is stored in the SQL database, the **Machine Learning Model** accesses this historical data to perform predictive analysis. By analyzing past weather patterns, the model generates predictions for future wind speeds and temperature changes, which are critical for forecasting.

- **Training and Prediction**: The model continuously retrains itself on newly scraped and stored data, improving its forecasting accuracy over time. This allows it to predict future wind speeds based on both recent and long-term historical data.
- **Output for Visualization**: The predictions made by the machine learning model are passed to the **Visualization Module**, where they are represented alongside actual data trends.

4. Visualization Module: Interactive Data Representation with Dash and Matplotlib

After the prediction process is complete, the **Visualization Module** uses **Dash** and **Matplotlib** to convert raw weather data and machine learning predictions into intuitive visual formats that are easy for end-users to interpret. This module is responsible for generating both static and interactive visualizations that provide insights into current and future weather patterns.

- **Dash for Interactive Dashboards**: **Dash**, a Python framework for creating web-based dashboards, allows users to interact with the data in real time. Users can filter the data by date ranges, locations, or weather metrics, and view dynamic updates on visualizations without refreshing the page. For example, wind speed and temperature trends are displayed on line charts, and users can toggle between viewing historical data or predictions.
- **Map Integration**: Using Dash's ability to incorporate geographic data from **GeoJSON** files, the module can also display wind speed patterns on maps, highlighting specific regions (e.g., coastal areas)

where wind conditions might be more critical. This feature is useful for visualizing spatial variations in weather data.

- **Matplotlib for High-Quality Static Visuals**: **Matplotlib**, a powerful plotting library, is used to create static visualizations, such as line graphs and bar charts, that can be exported and included in reports or presentations. Matplotlib allows for precise customization of graphs, helping users generate visually appealing plots that clearly show trends in wind speed, temperature, and other key metrics.
- **Combining Data for Visualization**: The **Weather Forecast API** and **GeoJSON** files containing geographic boundary data are also fed into the Visualization Module, where Dash and Matplotlib integrate both real-time data and machine learning predictions into cohesive, meaningful visuals.

5. Real-time Data Visualization and Insights

Once the data is processed and visualized, users can interact with it through an intuitive dashboard powered by **Dash**. The system enables end-users to view both real-time and forecasted weather data, making it easy to monitor weather conditions and trends dynamically.

Key Components in the Architecture:

1. **Multi-tasking Module**: Efficiently scrapes data from APIs and manages the detection of missing data.
2. **SQL Database**: Provides persistent storage for historical data and supports data access for real-time analysis.
3. **MySQL Connector**: Facilitates communication between the Python environment and the SQL database, ensuring smooth data storage and retrieval.
4. **Machine Learning Model**: Analyzes historical weather data and makes predictions about future wind speed and temperature trends.
5. **Dash**: Creates interactive, real-time dashboards that allow users to explore data through an intuitive interface.
6. **Matplotlib**: Generates high-quality static visualizations that can be used for reports or presentations, providing a clear representation of trends in the weather data.

B. Data Collection and Discussions

- Weather Forecast Data: 24-Hour Weather Forecasts, including wind speed from data.gov.sg. It is presented in five regions: North, South, East, West, and Central of Singapore in JSON format. Possible forecasts for each region are mapped according to its risk level are shown in TABLE I [1]:

TABLE I: Possible Geographic Forecasts with mapped risks

Forecast Category	Possible Forecasts	Risks
Fair Weather Conditions	Fair Fair (Day) Fair (Night) Fair and Warm	Low Risks
Partly Cloudy to Overcast	Partly Cloudy Partly Cloudy (Day) Partly Cloudy (Night) Cloudy	
Mild Weather Conditions	Hazy Slightly Hazy Windy Mist Fog	Moderate Risks
Moderate Rainfall Conditions	Light Rain Moderate Rain Passing Showers Light Showers Showers	
Heavy rain and Storms	Heavy Rain Heavy Showers Thundery Showers Heavy Thundery Showers Heavy Thundery Showers with Gusty Winds	High Risks

- Wind Speed Data: Real-time wind speed data measured in meters per second (m/s) is captured minute by minute from five wind stations located in the North, South, East, West, and Central regions of Singapore [2]. However, only hourly wind speed averages are collected and stored to prevent data overload.
- Temperature Data: Real-time temperature data, measured in degrees celsius (°C), captured minute by minute from five regions: North, South, East, West, and Central of Singapore [3]. However, only hourly temperature records are collected and stored to prevent data overload.
- GeoJSON Data: Geographic Data of Singapore are used for mapping and visual representation of the weather [4].

C. Algorithms and Features design

To optimize the efficiency of our data scraping process, we designed a comparative analysis between two concurrency modules: **Multithreading** and **Asyncio**. Both modules were evaluated based on their performance in handling I/O-bound operations, specifically making API calls and storing the

retrieved data into a database. Referencing [5], given that API calls are I/O-bound, we anticipated that an asynchronous approach would minimize idle time by allowing other tasks to be executed while waiting for responses. Asyncio's event-driven design, which allows multiple tasks to run concurrently without the overhead of creating new threads, was expected to outperform multithreading. Multithreading, by contrast, relies on creating separate threads for each task, which incurs additional overhead due to context switching between threads.

Machine Learning (ML) algorithms are adopted to analyze historical data and use their analysis in the prediction of future wind speed. In determining the most suitable ML algorithm for our wind speed predictions, a comparison was made amongst several algorithms based on their **Root Mean Square Error (RMSE)** performance. It's calculation is as follows:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

$n = number\ of\ observations$
$y_i = actual\ value\ for\ the\ i^{th}\ data\ point$
$\hat{y}_i = predicted\ value\ for\ the\ i^{th}\ data\ point$

As the difference between the actual and predicted values are squared, this ensures all errors are positive and larger deviations and errors are emphasized. Therefore, as RMSE penalizes large deviations between predicted and actual results [6], we will rate the predictive ability of our machine learning models on this aspect as even small deviations in predicted wind speed can lead to major safety risks in the aviation and maritime industries.

To improve prediction accuracy of future wind speed, a **multivariate prediction** was adopted as an approach which incorporates historical wind speed and temperature data as input features. Using these two variables, the model can capture the complex relationship between environmental factors [7]. This allowed the model to leverage on historical wind speed patterns while accounting for temperature variations which can influence future wind speed as well.

The predictive ability of the different machine learning algorithms was further enhanced through the implementation of several techniques designed to improve accuracy while preventing **overfitting**. Specifically, we incorporated **cross-validation, lagging features, and rolling statistics** into the model:

- **Cross-validation** was employed to ensure the robustness of the model by evaluating its performance across multiple subsets of the dataset. This method helps to minimize overfitting by testing the model on unseen data, ensuring that the algorithm generalizes well to new data which helps in preventing overfitting from occurring[8].
- **Lagging features** were introduced to leverage historical wind speed and temperature data, which are critical in time-series prediction tasks. By including previous time steps as predictors, the model was able to capture temporal dependencies, improving its ability to forecast future wind speed accurately.
- **Rolling features** such as moving averages and rolling standard deviations were utilized to smooth out short-term fluctuations and capture long-term trends in the data. This approach helps in identifying patterns over time and mitigates the effects of noise or sudden variations in wind speed, further enhancing the model's predictive power.

The combined use of cross-validation, lagging, and rolling features allowed the CatBoost model to learn from temporal and environmental patterns, which is expected to result in more accurate and reliable wind speed predictions.

Different machine learning models were evaluated to ensure the most model with the highest predictive accuracy would be used for the project. These models are: **Gradient Boosting Regression**, **Random Forest Regression**, and **CatBoost Regression**. Ultimately, **CatBoost Regression was chosen** due to its superior performance in handling both historical wind speed and temperature data, which were key features in our model which aids in increased accuracy in the prediction of future wind speed.

Several factors discussed in [9] influenced our decision to choose CatBoost Regression Model:

1. **Handling Multiple Features**: CatBoost effectively integrates both historical wind speed and temperature data as input features. It excels at capturing complex interactions and non-linear relationships between these variables, which are critical for accurate wind speed predictions.
2. **Flexibility with Non-Stationary Data**: Unlike traditional time series models, CatBoost does not require the data to be stationary, making it ideal for datasets with time-varying patterns, such as seasonal variations in wind speed and temperature.
3. **Robustness to Non-Linearity**: Wind speed is influenced by complex, non-linear interactions with temperature and other environmental variables. CatBoost's gradient boosting framework is particularly adept at modeling these non-linear relationships, leading to more accurate predictions.
4. **Ease of Feature Engineering**: While effective feature engineering enhances the performance of most models, CatBoost simplifies this process by automatically handling categorical variables and learning feature interactions without requiring extensive manual preprocessing.

Based on these advantages, CatBoost was selected as the most appropriate model for our wind speed prediction task.

Experimental Evaluation and Result Analysis

Our comparative analysis between Asyncio and Multi-Threading demonstrated that **Asyncio consistently outperformed in responsiveness** compared to multithreading as from TABLE II:

Table II: Comparison of Data Scraping Speed Between Multi-Threading And Asyncio

Rows of Data Collected	*Time taken by* ***Multi-Threading***	*Time Taken by* ***Asyncio***
24	2.6s	2.1s
72	6.1s	5.2s
168	12.4s	10.8s
336	**19.2s**	**15.5s**

Time measured for data to be collected, cleaned and stored in database

Furthermore, as the number of scraped data increases, the time difference between Multithreading and Asyncio becomes larger, with Asyncio performing over **20% faster in responsiveness** compared to Multi-threading when scraping 336 rows of data. Therefore, Asyncio has shown to be quicker in scraping both smaller and larger sets of data as compared to multi-threading. This performance boost was attributed to Asyncio's ability to manage I/O-bound tasks more efficiently by handling additional API requests and performing database write operations while awaiting responses. In contrast, multithreading introduced higher overhead and lacked the same level of efficient task management, leading to increased idle time and lower overall performance. Given the significant improvement in efficiency, **Asyncio was selected as the optimal solution** for our data scraping and storage processes

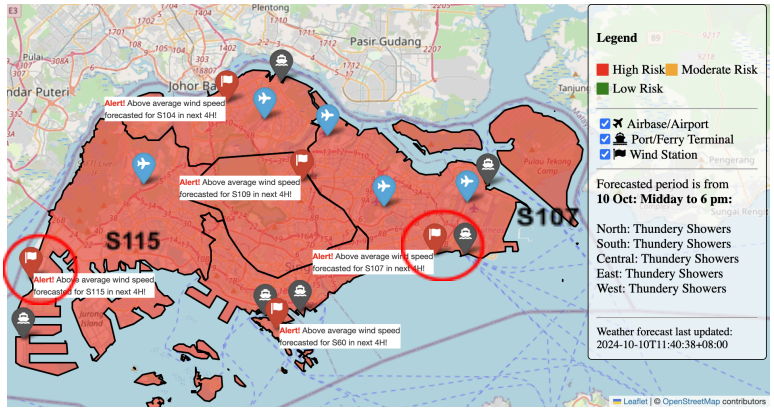
The **RMSE** and corresponding **RMSE Standard Deviation** for five different weather stations, each predicted by three machine learning models: **CatBoost Regression**, **Gradient Boosting Regression**, and **Random Forest Regression** is represented by TABLE III:

Table III: RSME and its Standard Deviation among different Machine Learning Models

ML Models	**Prediction Result**		
	Station ID	***RMSE***	***RMSE Standard Deviation***
CatBoost Regression	**S109**	**0.3945**	**± 0.0862**
	S60	**0.2789**	**± 0.0878**
	S115	**0.6299**	**± 0.1560**
	S104	**0.4336**	**± 0.0501**
	S107	**0.6036**	**± 0.1244**
Gradient Boosting Regression	S109	1.8562	± 0.3026
	S60	1.6479	± 0.2127
	S115	3.0377	± 0.1114
	S104	2.0728	± 0.1961
	S107	3.5567	± 0.4254
Random Forest Regression	S109	1.7842	± 0.0786
	S60	1.6875	± 0.1332
	S115	3.1139	± 0.1233
	S104	2.0209	± 0.1186
	S107	3.6929	± 0.3680

Models Evaluated using 3-Fold Cross-Validations & Sci-kit Learn Library

It is observed through the result that S60 in general, has the best prediction result across all models while S115 and S107 tends to have higher than average RMSE which reflected higher difficulty in predicting the wind speed for the station. When referencing the visualization, S115 and S107 weather stations are placed much closer to the coastal areas as compared to the other weather stations



The sea breezes in these coastal areas caused by topographical differences could cause a higher variability in wind speed, which could explain the poorer performances of these 3 machine learning models for S115 and S107.

The standard deviation (SD) of RMSE represents the variability of dispersion of RMSE across different subsets of data used for testing the ML model. It indicates the model's consistency in its prediction across different sets of data and can be derived as such:

$$SD = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(RMSE_i - \overline{RSME})^2}$$

$RMSE_i$ = *RMSE for each dataset*

$\overline{RSME}$ = *Mean RMSE across all dataset*

n = *number of RMSE values*

Overall, CatBoost Regression has outperformed the other ML models in both prediction accuracy and stability as evidenced by the lower RMSE and its standard deviation. Through the formula:

$$Mean\ RMSE = \frac{\sum_{Station_i}^{n} [RMSE(Station_i)}{n}$$

$station_i = individual\ stations$
$n = number\ of\ stations$

We can conclude from the results that the aggregated mean RMSE using Catboost Regression is **0.4681** across the 5 different stations which indicates that the error in prediction is roughly 0.47m/s, pitting it against **industry standards at 0.52m/s** [10]**.** For Singapore, with an average wind speed between 1.5 to 9m/s [2][11], the error margin is fairly small, indicating accurate predictions were made by Catboost regression.

Potential Challenges and Future Works

According to [12], wind speeds during thunderstorms can even exceed 10 m/s. Despite occasional thunderstorms, our model managed to achieve low RMSE, indicating its high prediction accuracy in both thundering and fair weather conditions, validating the effectiveness of the model for the safety of the aviation and maritime industry. However, it is important to note that the model could face challenges when predicting squall,categorized by sudden increase in wind speed to above 20m/s.[13] Due to its data sparsity, low occurrences of squall in Singapore leads to lack of training data for the model which could affect the model's ability to generalize during such extreme conditions.

Furthermore, the model is reliant on real time data gathered from weather and wind stations in making accurate geographic forecasts. However, malfunctions or disruptions in these stations such as maintenance lapses and sensor failures can lead to potential data gaps and inaccurate reading, affecting the model's performance.

Future works can be carried out to tackle these potential challenges, such as incorporation of real-time satellite and radar observations that can observe events like squall, which often travel from country to country. Using satellite observations allow us to to pre-empt such events, especially when the squalls are near the proximity of Singapore. Data imputation techniques can also be adopted in the case of weather stations malfunctioning. Machine learning algorithms can be implemented such as K-Nearest-Neighbour to fill the gaps in data if the weather stations are unable to obtain a geographic reading. [14]

Conclusion

Overall, this project provides more precise weather forecasting in Singapore, which is crucial for the safety of aviation and maritime operations. By using ML algorithms such as the CatBoost regression model, it helps us to predict future wind speed and temperature with high accuracy. The use of Asyncio for data scraping also improves efficiency compared to multithreading. This ensures a faster and more responsive data collection process, which is crucial for real-time monitoring. Our project interactive map and charts will provide a user-friendly interface for users to quickly assess weather conditions and make informed decisions. Alerts and notifications of hazardous weather can also enhance the safety and efficiency of the aviation and maritime industries in Singapore. While the model provides high accuracy in predicting typical weather conditions, future work can address potential challenges related to predicting events like squalls.

References and Citations

[1] "24-hour weather forecast API," NEA (National Environment Agency), Data.gov.sg, Updated: Oct. 2024. [Online]. Available: https://data.gov.sg/datasets?topics=environment&page=1&query=weather+forecast&resultId=d_ce2eb1e307bda31993c533285834ef2b. [Accessed: Sep 10, 2024]

[2] "Wind Speed across Singapore," NEA (National Environment Agency), Data.gov.sg, Updated: Oct. 2024. [Online]. Available: https://data.gov.sg/datasets?topics=environment&page=1&query=wind+speed&resultId=d_7677738484067741bf3b56ab5d69c7e9#tag/default/GET/environment/air-temperature. [Accessed: Sep. 13, 2024].

[3] "Air Temperature across Singapore," NEA (National Environment Agency), Data.gov.sg, Updated: Oct. 2024. [Online]. Available: https://data.gov.sg/datasets?topics=environment&page=1&query=temperature&resultId=d_66b77726bbae1b33f218db60ff5861f0#tag/default/GET/environment/air-temperature. [Accessed: Sep. 13, 2024].

[4] "Region Level 1 GIS Data," MapOG, [Online]. Available: https://gisdata.mapog.com/singapore/Region%20level%201. [Accessed: Sep. 10, 2024].

[5] R. Jain, H. Tang, A. Dhruv, J. A. Harris, and S. Byna**,** "Accelerating flash-x simulations with asynchronous I/O," presented at *Argonne National Laboratory, Lawrence Berkeley National Laboratory, Oak Ridge National Laboratory*, 2023. [Online]. Available: https://ieeexplore-ieee-org.singaporetech.remotexs.co/stamp/stamp.jsp?tp=&arnumber=10026923.

[6]T. Chai and R. R. Draxler**,** "Root mean square error (RMSE) or mean absolute error (MAE)? – Arguments against avoiding RMSE in the literature," *Geosci. Model Dev.*, vol. 7, no. 3, pp. 1247-1250, 2014. [Online]. Available: https://gmd.copernicus.org/articles/7/1247/2014/gmd-7-1247-2014.pdf.

[7]B. Nguyen-Thai, V. Le, N. T. Tieu, T. Tran, S. Venkatesh, and N. Ramzan, "Learning evolving relations for multivariate time series forecasting," *Appl. Intell.*, vol. 53, no. 2, pp. 1-15, Mar. 2024. [Online]. Available:

https://link.springer.com/content/pdf/10.1007/s10489-023-05220-0.

[8] Z. Jia, "Controlling the overfitting of heritability in genomic selection through cross-validation," *Scientific Reports*, vol. 7, no. 1, Oct. 2017. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5651917/pdf/41598_2017_Article_14070.pdf.

[9] P. Sharma and B. J. Bora**,** "A review of modern machine learning techniques in the prediction of remaining useful life of lithium-ion batteries," *Batteries*, vol. 9, no. 1, pp. 13, Jan. 2023. [Online]. Available: https://www.mdpi.com/2313-0105/9/1/13.

[10] X. Liu, Z. Li, and Y. Shen, "Study on downscaling correction of near-surface wind speed grid forecasts in complex terrain," *Atmosphere*, vol. 15, no. 9, article 1090, Sep. 2023. [Online]. Available: https://doi.org/10.3390/atmos15091090

[11] National Environment Agency, "Wind speed data," Singapore Meteorological Service, 2024. [Online]. Available: https://www.weather.gov.sg. [Accessed: Oct. 4, 2024].

[12] E. C. C. Choi**,** "Extreme wind characteristics over Singapore – an area in the equatorial belt," *J. Wind Eng. Ind. Aerodyn.*, vol. 83, no. 1-3, pp. 61-74, 1999. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167610599000616.

[13]F. Mujibah, "ST explains: What's a Sumatra squall?," *The Straits Times*, Sep. 19, 2024. [Online]. Available: https://www.straitstimes.com/singapore/st-explains-what-s-a-sumatra-squall.

[14]C. Clifton, E. J. Hanson, K. Merrill, and S. Merrill, "Differentially Private k-Nearest Neighbor Missing Data Imputation," in *Proc. 2022 ACM Asia Conf. on Computer and Communications Security*, pp. 1724-1737, 2022. [Online]. Available: https://dl-acm-org.singaporetech.remotexs.co/doi/pdf/10.1145/3507952.