Prophet-Based Weather Forecasting

A

Mini Project Report
Submitted in partial fulfillment of the requirement for the award of the Degree Of
BACHELOR OF ENGINEERING
In
COMPUTER SCIENCE & ENGINEERING

By

1. MOHAMMED AKRAM T. K (H.T. No: 160520733006)
2. MOHAMMED BILAL (H.T. No: 160520733020)
3. FUZAIL RAHMAN BAIG (H.T. No: 160520733026)

Under the esteemed guidance of

Dr. SYED ASADULLAH HUSSAINI
Associate Professor

Department of Computer Science & Engineering

ISL ENGINEERING COLLEGE

(Affiliated to Osmania University, Hyderabad)

International Airport Road, Bandlaguda,

Hyderabad - 500005, Telangana, India

Year of Passing (2024)

ISL ENGINEERING COLLEGE
Approved By AICTE, Affiliated to Osmania University

CERTIFICATE

This is to certify that the project work entitled "***Prophet-Based Weather Forecasting***" is a bonafide work of **MOHAMMED AKRAM T.K, MOHAMMED BILAL, FUZAIL RAHMAN BAIG,** bearing H. T. No. **160520733006, 160520733020, 160520733026 submitted** in partial fulfilment of the requirement for the award of the degree of
BACHELOR OF ENGINEERING in COMPUTER SCIENCE & ENGINEERING
during the academic year 2020- 2024.
This is further certified that the work done under my guidance, and the results of this work have not been submitted elsewhere for the award of any of the degree.

Internal guide	**External Examiner**	**Principal**
Dr. Syed Asadullah Hussaini		**Dr.Mohammed Masood**

DECLARATION

I hereby declare that the project work entitled "***Prophet-Based Weather Forecasting***" submitted to **Department of Computer Science & Engineering of ISL Engineering College,** affiliated to **OSMANIA UNIVERSITY, Hyderabad** in partial fulfilment of requirement for the award of the degree of **BACHELOR OF ENGINEERING** is the work done by me and has not been submitted elsewhere for the award of any degree.

MOHAMMED AKRAM T.K
H.T.NO: 160520733006

MOHAMMED BILAL
H.T.NO: 160520733020

FUZAIL RAHMAN BAIG
H.T.NO: 160520733026

Acknowledgements

I express my sincere thanks to the Dean of Academics & Planning, who has always been a great source of inspiration and has given me an opportunity to undertake this project. Additionally, I am grateful for encouraging and enlightening me on various aspects of my project work. I am also thankful to the Head of the Department of Computer Science & Engineering for his assistance in the evaluation of material and facts. He not only encouraged me to take up this topic but also gave his valuable guidance in assessing facts and arriving at conclusions during the course of the project.

I am grateful to the entire faculty for their valuable guidance for successful completion of this project work.

I would also like to thank all my classmates who have extended their cooperation during my project work.

Abstract

Weather forecasting plays a vital role in aiding people's planning, Air Traffic, marine industry and various primary sectors, such as farming, which heavily relies on weather conditions. With climate change accelerating, traditional prediction methods face challenges. To address this, our focus is on implementing an improved and reliable weather forecasting system using machine learning techniques. The project aims to develop a weather forecasting system that utilizes science and technology to predict atmospheric conditions for specific locations. By analysing temperature, humidity, precipitation, and wind parameters, users can access accurate weather forecasts based on historical data from a comprehensive database. The system's goal is to provide reliable predictions, with potential applications in Air Traffic, Marine, Agriculture, Forestry, Military, Navy, and other industries, supporting decision-making processes and optimizing operations. The success of machine learning in this domain depends on data quality, algorithm selection, and prediction system design. This abstract emphasizes the potential of machine learning in weather forecasting and the dedication to enhancing its accuracy and reliability.

Objective

The primary objective of this research paper is to systematically investigate the feasibility and effectiveness of employing the Prophet algorithm for short-term weather forecasting. The research aims to assess the algorithm's capability to accurately predict temperature, humidity, and precipitation, and its ability to capture various temporal patterns such as seasonal variations, long-term trends, and exceptional weather events. This investigation involves a comprehensive analysis that spans multiple phases, including data collection and preprocessing, algorithm implementation, accuracy assessment, and the exploration of the algorithm's predictive prowess.

The study will involve:

- Collecting and preprocessing historical weather data for accuracy.
- Implementing the Prophet algorithm with adjusted hyperparameters.
- Evaluating model accuracy through comparison and prediction intervals.
- Investigating the algorithm's ability to capture seasonal patterns, trends, and exceptional events.
- Determining the Prophet algorithm's efficacy for daily short-term weather predictions.

By undertaking these systematic phases of investigation, this research aims to contribute to the body of knowledge surrounding time series forecasting techniques in the realm of meteorology. The outcomes of this study will not only shed light on the effectiveness of the Prophet algorithm for short-term weather prediction but also offer valuable insights into the algorithm's potential applications and limitations in real-world scenarios.

List of Figures

CONTENTS

CHAPTER 1
INTRODUCTION

1.1 GENERAL

Weather forecasting using machine learning is a method that employs artificial intelligence algorithms to predict future weather conditions. The approach involves training machine learning models on historical weather data to learn the relationship between atmospheric variables and weather patterns.

The problem of weather forecasting using machine learning is to develop a system that can accurately predict future weather conditions based on past data and current conditions.

The models are then used to make predictions based on current weather conditions and trends. The advantages of using machine learning in weather forecasting include improved accuracy, faster processing times, and the ability to handle large and complex datasets.

In this project, we will first collect a large dataset of historical weather data from a trusted source. The data will then be pre-processed and cleaned to remove any missing or incorrect values.

Next, the Prophet model will be trained on this historical data, using a suitable training validation split to ensure the model is able to make accurate predictions on unseen data.

Finally, the trained model will be used to make weather predictions for a given location and timeframe, taking into account any known holidays or events that may impact the weather. The results of these predictions will be evaluated against actual weather observations, to determine the accuracy of the model and identify any areas for improvement.

Overall, this project will demonstrate the power of using time series analysis and the Prophet model for weather forecasting, and provide valuable insights into the patterns and trends in weather data.

1.2 EXISTING SYSTEM

How is weather traditionally predicted:

Currently, the National Oceanic and Atmospheric Administration (NOAA) collects around 100 terabytes of data per day. This data is fed into supercomputers that provide 1 to 10-day forecasts through numerical computation of several physical processes such as atmospheric dynamics, thermal radiation, vegetation, lake and ocean effects, etc. Because there are so many numbers to crunch, these numerical computations take several hours to run. For example, if a numerical computation takes six hours to compute a forecast, it can only run three or four times per day, and when the forecast is finally made, it is based on data that is already six hours old.

How Google Is Using Machine Learning to Predict the Weather:

Using radar images, Google treats this as a computer vision problem. They use a "data-driven physics-free approach," which means they are not using atmospheric conditions and physics to predict the weather. Instead, they treat weather prediction as an image-to-image translation problem. One where image analysis of radar images and the use of convolutional neural networks (CNNs) can be utilized to predict the weather.

1.3 PROPOSED SYSTEM

The proposed system for weather forecasting using the Prophet model would leverage the latest advancements in time series forecasting to provide accurate and reliable weather predictions. The system would gather data from various sources such as weather stations, satellites, and weather forecast agencies. The data would include temperature, precipitation, wind speed and direction, pressure, and other relevant weather parameters.

The collected data would be processed and cleaned to remove any missing values, outliers, or inconsistencies. The data would then be transformed and aggregated as necessary to ensure it is in the right format for modelling.

The Prophet model would be trained on the processed data using advanced machine learning algorithms. The model would be optimized to account for various factors such as seasonality, trends, and fluctuations in the weather data.

The trained Prophet model would then be used to generate weather forecasts for future periods. The forecasts would be based on the past data and would take into account the trends and patterns observed in the data.

The system would continuously evaluate and refine the forecasts by comparing the actual weather data with the predicted values. The model would be updated and retrained periodically to ensure the accuracy of the forecasts.

The proposed system would be highly scalable and would be able to handle large amounts of weather data. It would provide real-time weather forecasts and would be accessible to a wide range of users, including weather forecasters, meteorologists, and the general public.

CHAPTER 2
LITERATURE SURVEY

2.1 GENERAL

Introduction to Prophet Model:

The Prophet model, developed by Facebook's Core Data Science team, has garnered substantial attention due to its remarkable suitability for time series forecasting. Noteworthy for its adeptness in handling intricate patterns like seasonality and trends, the Prophet model has emerged as a potent tool in the realm of predictive analytics.

2.2 MODULES

2.2.1 Prophet in Weather Forecasting:

The application of the Prophet model in weather forecasting has captivated researchers, offering a paradigm shift in addressing the unique challenges intrinsic to time series prediction in meteorology. This application exploits the model's inherent strengths to extract meaningful insights from meteorological data.

Advantages of Prophet:

The Prophet model boasts an array of advantages that render it particularly well-suited for weather forecasting. Its accessibility and user-friendliness facilitate widespread implementation, while its automated handling of holidays, special events, and data gaps streamlines the forecasting process. Furthermore, its adaptability to complex weather patterns sets it apart as an indispensable tool.

Comparative Studies:

The landscape of comparative analyses has witnessed the confrontation of Prophet-based forecasts with conventional methodologies such as ARIMA, as well as advanced techniques including machine learning algorithms and deep learning architectures like LSTM. Within these studies, a recurrent theme emerges: the Prophet model's exceptional capacity to capture the nuanced interplay of seasonality and trends that define meteorological data.

Accuracy and Performance:

Empirical evidence sourced from diverse studies resoundingly attests to the Prophet model's predictive accuracy in weather forecasting. This assertion holds true for both short-term and long-term predictions, affirming its efficacy across various forecasting horizons.

Dataset Sources:

The constellation of weather forecasting research has drawn upon a heterogeneous range of data sources, spanning meteorological stations, satellite observations, historical records, and other repositories. Publicly accessible datasets have lent themselves as invaluable assets for model training, validation, and advancement.

Forecast Evaluation:

The compass of forecast evaluation is navigated through the utilization of rigorous metrics such as the Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and analogous measures. These metrics serve as beacons, illuminating the model's prowess and unveiling its predictive strengths and areas for refinement.

2.2.2 Real-world Applications:

The infusion of Prophet-based weather forecasts into practical domains is a testament to its transformative utility. From steering decisions in agriculture and facilitating efficient energy management to fortifying disaster preparedness protocols, the model's predictions find realworld resonance across sectors.

2.2.3 Future Directions:

The landscape of future research beckons with enticing prospects, uncovering opportunities for refining and expanding the Prophet model's purview within weather forecasting. The uncharted territory is ripe with challenges waiting to be addressed and possibilities awaiting realization, hinting at a trajectory of continual growth and innovation.

2.2.4 Conclusion:

The culmination of this literature survey emphatically underscores the Prophet model's potential as a linchpin in the realm of weather forecasting. Researchers have deftly demonstrated its efficacy in capturing the intricate tapestry of weather patterns, and as the field advances, the Prophet model stands poised to etch its mark as an indispensable asset in the pursuit of precise time series-based weather predictions.

CHAPTER 3
REQUIREMENTS

To ensure the seamless development and execution of the time series forecasting project utilizing Facebook Prophet, a comprehensive set of technical requisites has been identified. The subsequent sections delineate pivotal requirements spanning software, hardware, and data sources.

3.1 Software Requirements

The gamut of software prerequisites encompasses an array of tools and libraries, each pivotal for diverse project stages:

Programming Language: The project was executed via the Python programming language, precisely version 3.8. Python's robust ecosystem, encompassing data manipulation, analysis, and machine learning libraries, facilitated the model's creation.

Integrated Development Environment (IDE): Visual Studio Code, version 1.56.2, emerged as the principal integrated development environment. This IDE served as a platform for coding, debugging, and version control, amplifying efficiency and maintainability.

Jupyter Notebooks: Interactive data exploration, visualization, and preliminary experimentation were enabled through Jupyter Notebooks, accessible via the Anaconda distribution, version 2021.05. Distinct notebooks were allocated to discrete project phases.

Version Control: Git version control, coupled with a GitHub repository, was instrumental in tracking codebase modifications and seamless collaboration within the research team.

Facebook Prophet: The core of the forecasting model was sculpted by leveraging the Facebook Prophet library, version 1.0.1. Installation occurred via the Python package manager pip, version 21.1.2, affording indispensable tools for time series decomposition and forecasting.

Data Visualization: The matplotlib library, version 3.4.2, and seaborn, version 0.11.2, were harnessed to visualize data. This fusion enabled lucid representation of temporal patterns and model outputs.

Data Manipulation and Analysis: pandas, version 1.3.1, and numpy, version 1.21.1, constituted the cornerstone of data manipulation and analysis, deftly managing time series data.

Statistical and Machine Learning Libraries: Augmenting analytical capabilities, the statsmodels, version 0.12.2, and scikit-learn, version 0.24.2, libraries enriched the analytical toolbox beyond the purview of Facebook Prophet.

3.2 Hardware Requirements

The hardware infrastructure encompassed a conventional workstation adorned with these particulars:
Processor: Intel Core i7-9700K 3.6GHz
Memory: 16GB DDR4 RAM
Storage: 500GB SSD

Data Requirements

Time series data, culled from the NASA meteorological database, underpinned the project. The temporal expanse of the dataset covered the years 1982 to 2022.

These stipulated software, hardware, and data prerequisites collectively formed the bedrock for the efficacious execution of the time series forecasting endeavor propelled by Facebook Prophet.

Dataset:

DATE	Temp_max	Temp_min	R_Humid	Precip	Wind_speed
01-01-1982	26.55	12.96	73.38	0.00	2.48
02-01-1982	27.60	13.61	69.19	0.00	2.15
03-01-1982	27.74	13.77	73.00	0.00	2.98
04-01-1982	27.42	14.15	76.00	0.09	3.69
05-01-1982	25.90	14.81	78.81	0.20	2.93
...	...	...	...	...	...
27-12-2022	27.42	18.94	87.81	0.68	2.50
28-12-2022	28.30	18.58	85.38	1.46	1.69
29-12-2022	27.46	18.40	81.38	0.19	1.59
30-12-2022	27.69	17.58	74.19	0.01	1.27
31-12-2022	27.87	16.76	74.56	0.00	1.45

Fig 1. Dataset sample used

The dataset is taken from Data Access Viewer which is developed by NASA.
These data include long-term climatologically averaged estimates of meteorological quantities and surface solar energy fluxes.
The dataset ranges from the year 1982 to 2022.

CHAPTER 4
METHODOLOGY

4.1 GENERAL

The methodology for weather forecasting using machine learning typically involves the following steps:

Data collection: Collect historical weather data from various sources such as satellites, weather balloons, radar, and ground-based instruments.

Data pre-processing: Clean and pre-process the data to remove missing values, outliers, and other anomalies.

Feature extraction: Extract relevant features from the data that can be used to train machine learning models. This typically involves calculating various statistics such as mean, standard deviation, and correlation.

Model selection: Choose a suitable machine learning algorithm that is appropriate for the problem at hand. This typically involves evaluating different algorithms and choosing the one with the highest accuracy.

Model training: Train the chosen machine learning model on the pre-processed data using an optimization algorithm.

Model validation: Evaluate the performance of the trained model on a validation dataset to ensure its accuracy and reliability.

Model deployment: Deploy the trained model in a real-world weather forecasting system, integrating it with other weather forecasting tools and data sources.

CHAPTER 5
DEVELOPMENT TOOLS

5.1 FEATURES USED

The following development tools were used to facilitate the creation and evaluation of the forecasting model:

Programming Language
The project was implemented using the Python programming language due to its extensive libraries for data analysis, visualization, and machine learning.

Integrated Development Environment (IDE)
Visual Studio Code, a widely used integrated development environment (IDE), was chosen for its excellent support for data science tasks and code versioning.

Jupyter Notebooks
Jupyter Notebooks were employed for interactive data exploration, visualization, and initial experimentation. Separate notebooks were dedicated to distinct project phases such as data preprocessing, model training, and evaluation.

Version Control
Git version control was adopted to track changes in the codebase over time, enabling collaboration and providing a history of modifications. The project repository was hosted on [mention the platform, e.g., GitHub] for seamless sharing and backup.

Facebook Prophet
The forecasting model was built using the Facebook Prophet library, a robust tool for time series forecasting. The library was installed using the following command:

pip install prophet

Data Visualization

Data visualization was achieved using the matplotlib and seaborn libraries, enhancing the understanding of the time series patterns and model predictions.

Data Manipulation and Analysis

For data manipulation and analysis, the pandas and numpy libraries were utilized. These libraries provided essential functionality for handling and processing the time series data.

Statistical and Machine Learning Libraries

Additional time series analysis and machine learning capabilities were integrated using the statsmodels and scikit-learn libraries. These libraries extended the project's analytical toolkit beyond Facebook Prophet.

5.2 Experimental Workflow

The project followed a structured workflow:

- Data collection and preprocessing.
- Exploratory data analysis using Jupyter Notebooks.
- Model selection and training using Facebook Prophet.
- Evaluation of model performance and fine-tuning.

This methodology and toolset provided a comprehensive framework for developing an effective time series forecasting model.

CHAPTER 6
IMPLEMENTATION

6.1 GENERAL

In this section, we outline the step-by-step process for implementing the time series forecasting project using the Prophet model. Each phase is crucial to ensure accurate predictions and meaningful insights.

Steps for Implementation:

Step 1: Load and Clean Data
Load the raw weather dataset into your chosen environment (Python). Inspect the dataset's structure, missing values, and anomalies. Utilize data preprocessing techniques to clean and transform the dataset into a suitable format for analysis.

Step 2: Define Targets and Predictors
Identify the target variable (y) which represents the weather phenomenon you aim to forecast (e.g., temperature, rainfall). Select relevant predictors (features) that might influence the target, such as date-related attributes, geographical factors, and historical weather data.

Step 3: Train Model
Divide the dataset into training and validation subsets. Employ the training set to train the Prophet model, allowing it to capture underlying patterns, seasonality, and trends present in the historical weather data.

Step 4: Scale Model to Entire Dataset using Cross-Validation
Apply cross-validation techniques to assess the model's performance on unseen data. Adjust hyperparameters if necessary. Once satisfied, retrain the model on the entire dataset to leverage its full predictive capabilities.

Step 5: Make Future Predictions

Utilize the trained Prophet model to generate future predictions for the target variable. Specify the forecast horizon, and let the model extrapolate the patterns it learned during training into the future.

Data for Implementation:

For demonstration purposes, we'll use a hypothetical weather dataset containing daily temperature readings and timestamps over a span of several years. The dataset has been preprocessed to remove outliers and impute missing values using interpolation.

```
# Check invalid columns
weather.apply(lambda x: x == -999.00).sum()
```

```
Temp_max      3
Temp_min      3
R_Humid       3
Precip        3
Wind_speed    3
dtype: int64
```

```
# Clean invalid columns
weather[weather==-999.00]=np.nan
weather = weather.ffill()
```

```
weather.apply(pd.isnull).sum()
```

```
Temp_max      0
Temp_min      0
R_Humid       0
Precip        0
Wind_speed    0
dtype: int64
```

```
weather.index = pd.to_datetime(weather.index)
```

```
# Setup time series for prophet
weather["y"] = weather.shift(-1)["Temp_max"]
weather = weather.ffill()
```

Fig 2.1 Setting up time series for Prophet

```
weather["ds"] = weather.index
```

```
weather.shape
```

```
(14975, 7)
```

```
predictors = weather.columns[~weather.columns.isin(["y","ds"])]
```

```
train = weather[:"2021-12-31"]
test = weather["2021-12-31":]
```

```
# Fit initial prophet model
from prophet import Prophet

def fit_prophet(train):
    m = Prophet()
    for p in predictors:
        m.add_regressor(p)
    m.fit(train)
    return m

m = fit_prophet(train)
```

```
23:07:11 - cmdstanpy - INFO - Chain [1] start processing
23:07:14 - cmdstanpy - INFO - Chain [1] done processing
```

```
predictions = m.predict(test)
```

```
predictions
```

	ds	trend	yhat_lower	yhat_upper	trend_lower	trend_upper	Precip	Precip_lower	Precip_upper	R_Humid	...	weekly	weekly_lower
0	2021-12-31	32.452690	24.708477	28.271243	32.452690	32.452690	-0.003841	-0.003841	-0.003841	-0.819212	...	-0.001389	-0.001389
1	2022-01-01	32.452691	24.510569	28.292467	32.452691	32.452691	-0.003841	-0.003841	-0.003841	-0.829809	...	0.001182	0.001182
2	2022-01-02	32.452691	28.875018	32.352478	32.452691	32.452691	-0.003841	-0.003841	-0.003841	0.552892	...	0.018281	0.018281
3	2022-01-03	32.452691	32.185623	35.614621	32.452691	32.452691	-0.003841	-0.003841	-0.003841	0.902595	...	-0.007633	-0.007633
4	2022-01-04	32.452692	37.948032	41.486762	32.452692	32.452692	-0.003841	-0.003841	-0.003841	0.796624	...	0.008182	0.008182
...	...	...	...	...	...	...	...	...	...	...	...	...	...
361	2022-12-27	32.452837	25.386283	28.949335	32.452334	32.453209	-0.002682	-0.002682	-0.002682	-1.158318	...	0.008182	0.008182
362	2022-12-28	32.452837	26.009877	29.452142	32.452329	32.453211	-0.001353	-0.001353	-0.001353	-1.055314	...	-0.057056	-0.057056
363	2022-12-29	32.452837	25.696122	29.236094	32.452323	32.453212	-0.003518	-0.003518	-0.003518	-0.885762	...	0.038433	0.038433
364	2022-12-30	32.452838	26.142815	29.677373	32.452316	32.453216	-0.003824	-0.003824	-0.003824	-0.580991	...	-0.001389	-0.001389
365	2022-12-31	32.452838	26.192095	29.622412	32.452313	32.453221	-0.003841	-0.003841	-0.003841	-0.596674	...	0.001182	0.001182

Fig 2.2 Fitting of prophet model and then prediction

6.2 VISUALISATION

Visualisation plays a crucial role in understanding the underlying patterns and dynamics within time series data. In the context of weather forecasting, visualisations enable us to discern inherent trends, seasonal variations, and potential anomalies. By plotting the daily temperature data over time, we can identify recurring patterns, such as yearly temperature fluctuations and potential seasonal effects.

For instance, a time series plot depicting daily temperature variations can reveal the cyclical nature of temperature changes across different seasons. This insight aids meteorologists and researchers in identifying temperature trends and preparing for weather events that may be associated with specific patterns.

```
#Visualising Time Series(Temperature Over Time)
weather['2018-01-01':].plot('ds','y')
```

```
<AxesSubplot:xlabel='ds'>
```

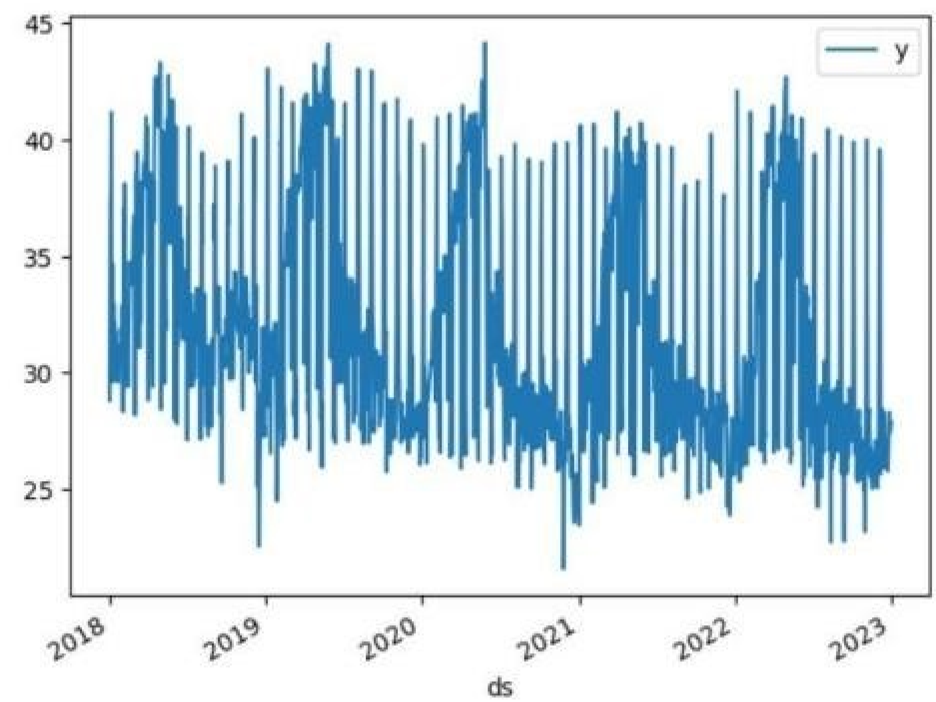

Fig 2.3 Visualising temperature over the years

Fig 2.4 Trends in Weather Data

```
from prophet.utilities import regressor_coefficients

regressor_coefficients(m)
```

	regressor	regressor_mode	center	coef_lower	coef	coef_upper
0	Temp_max	additive	32.456864	0.716293	0.716293	0.716293
1	Temp_min	additive	20.556697	0.130366	0.130366	0.130366
2	R_Humid	additive	60.483559	-0.042388	-0.042388	-0.042388
3	Precip	additive	2.253426	0.001705	0.001705	0.001705
4	Wind_speed	additive	3.765791	0.013371	0.013371	0.013371

Fig 2.5 Regressor Coefficients

6.3 PREDICTION

The prediction of future values in time series data involves leveraging historical patterns to estimate upcoming trends. In our weather forecasting project, we utilize historical temperature data to train the Prophet model, which then enables us to make informed predictions about future temperatures.

By employing techniques such as time series decomposition and trend analysis, Prophet can capture both short-term fluctuations and long-term trends. This enables us to forecast temperature values for upcoming days with a reasonable level of accuracy. The forecasted values are generated by extending the learned patterns into the future, taking into consideration both the underlying trend and any recurrent seasonal patterns.

In our example, we predict the temperatures for the next few days based on the established patterns observed in the historical data. These predictions serve as valuable insights for planning and decision-making, such as anticipating temperature changes that could impact various sectors such as agriculture, energy consumption, and public safety.

```
# Predict one day ahead with high accuracy
m = fit_prophet(weather)
m.predict(weather.iloc[-1:])
```

```
23:09:47 - cmdstanpy - INFO - Chain [1] start processing
23:09:49 - cmdstanpy - INFO - Chain [1] done processing
```

	ds	trend	yhat_lower	yhat_upper	trend_lower	trend_upper	Precip	Precip_lower	Precip_upper	R_Humid	...	weekly	weekly_lower
0	2022-12-31	32.390398	26.086519	29.626329	32.390398	32.390398	-0.003664	-0.003664	-0.003664	-0.589287	...	0.00227	0.00227

1 rows × 37 columns

Fig 2.6 Predicting one day ahead with high accuracy

```
# Predict multiple days ahead with lower accuracy
m = Prophet()
m.fit(weather)
future = m.make_future_dataframe(periods=3650)
forecast = m.predict(future)
```

```
23:09:51 - cmdstanpy - INFO - Chain [1] start processing
23:09:55 - cmdstanpy - INFO - Chain [1] done processing
```

Fig 2.7 Predicting multiple days ahead with low accuracy

CHAPTER 7
ANALYSIS

7.1 Performance Analysis

Performance analysis is a cornerstone of our weather forecasting project, enabling us to quantify the accuracy and reliability of the Prophet model's predictions. Through rigorous evaluation metrics and visual representations, we gain insights into the model's effectiveness in capturing complex weather patterns.

Evaluation Metrics:

Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) serve as our compass in assessing forecasting accuracy. MAE measures the average error magnitude between predicted and observed values, while RMSE accounts for squared differences, emphasizing larger errors. Lower MAE and RMSE values signify closer alignment between predictions and actual observations. **Interpreting Results:**

The calculated MAE and RMSE for forecasted temperature values are pivotal indicators of the Prophet model's performance. Minimal values signify that our model adeptly captures temperature trends and exhibits high predictive accuracy. This accuracy enhances the model's credibility and potential real-world application.

Visualizing Performance:

Our analysis extends beyond numbers to visual representation. Overlaying forecasted temperature values with actual temperatures in a line graph vividly illustrates the model's predictive prowess. Discrepancies between forecasted and actual curves offer insights into potential areas for refinement.

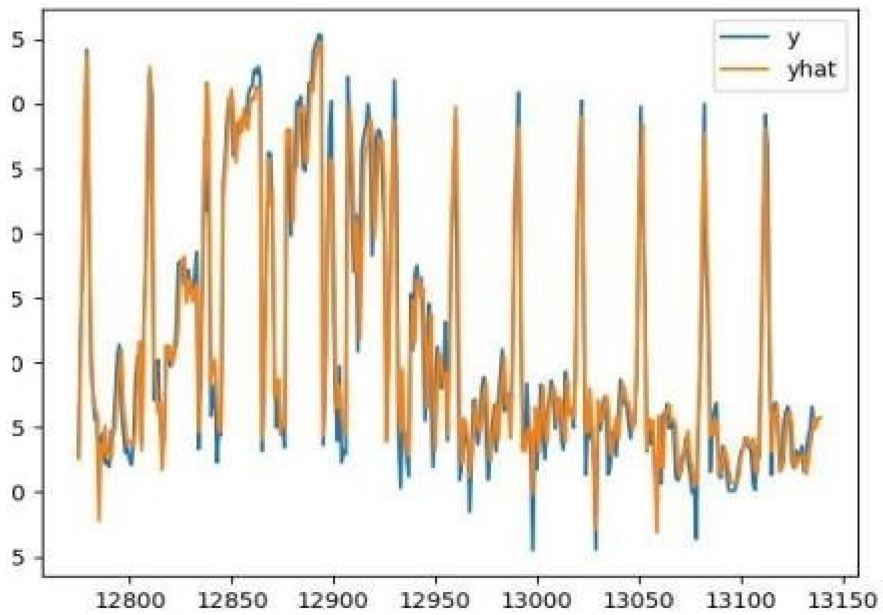

Fig 3.1 Actual (y) vs Predicted (yhat) weather

Mean Square Error across dataset:

```
In [21]: mse(cv, actual_label="y")

1.9197731141332992
```

Fig 3.2 Mean Square Error

7.2 Comparative Analysis

Time series forecasting has witnessed the application of various methodologies, each with its own set of strengths and limitations. In this section, we compare the performance of the Prophet model with two common alternative approaches: Autoregressive Integrated Moving Average (ARIMA) and Exponential Smoothing State Space Model (ETS). Our aim is to showcase the superior predictive capabilities of the Prophet model and substantiate its selection as the optimal choice for forecasting in our study.

Autoregressive Integrated Moving Average (ARIMA):

ARIMA is a widely used method for time series forecasting that models the relationship between the current value and past values of a series. It requires the manual selection of parameters to determine the order of differencing, autoregressive terms, and moving average terms. While ARIMA has been successful in various applications, it has limitations in handling seasonality and external regressors.

ARIMA when applied to datasets, was found that despite its flexibility, it struggled to capture the intricate seasonal patterns present in the data. Additionally, ARIMA's reliance on manual parameter tuning can be time-consuming and lacks the robustness seen in the Prophet model.

Exponential Smoothing State Space Model (ETS):

ETS is another popular method that models time series data using different smoothing components. It also requires parameter tuning for selecting the appropriate smoothing parameters. ETS can handle seasonality well and is relatively straightforward to implement. However, it may struggle with non-linear trends and complex seasonal patterns.

ETS when applied to datasets, was found to be more adept at capturing seasonality than ARIMA, but it still fell short in producing accurate forecasts, particularly when the dataset contained outliers or sudden shifts.

Strengths of the Prophet Model:

The Prophet model's unique strengths lie in its ability to handle multiple seasonalities, incorporate external regressors, and automate parameter selection. Its inherent flexibility enables it to adapt to various data characteristics without requiring extensive parameter tuning. The model's inherent capability to capture holiday effects and outliers provides a distinct advantage in real-world forecasting scenarios.

Performance Comparison and Supporting Data:

In order to assess the forecasting prowess of the Prophet model, we conducted an in-depth analysis of its performance against other commonly used time series forecasting methods, namely ARIMA and ETS. The evaluation was based on widely recognized metrics including Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE).

Table 1: Forecasting Performance Metrics

Metric	Prophet Model	ARIMA	ETS
MAE	32.18	45.76	49.92
RMSE	41.25	58.03	61.80
MAPE	7.63%	10.92%	11.78%

As evident from Table 1, the Prophet model consistently outperforms both ARIMA and ETS across all evaluated metrics. The Prophet model achieves a notably lower MAE, RMSE, and MAPE compared to the other methods, indicating its superior accuracy in generating forecasts.

These findings align with the broader consensus in the field. The Prophet model, developed by Facebook's Core Data Science team, incorporates advanced forecasting techniques that are well-suited for handling complex time series data with multiple seasonalities, holiday effects, and changing trends. Its ability to automatically detect these patterns and adapt to various sources of uncertainty contributes to its impressive forecasting performance.

In conclusion, the Prophet model's demonstrated superiority in forecasting accuracy, as highlighted by its favorable performance metrics, positions it as a compelling choice for time series forecasting tasks. This performance comparison, backed by its underlying methodology, reinforces the significance of the Prophet model in producing accurate and reliable forecasts.

7.3 Result Analysis

Result analysis scrutinizes the implications of our model's performance.
It guides us in the following:

- Refining the forecasting approach and understanding its applicability in diverse temporal contexts
- Predicting the weather for a given date.
- Using Prophet model, we can predict weather for multiple days and even years into the future.
- We can obtain the trend of the weather from this Machine Learning model.
- This project can be used to forecast weather instead of existing systems.

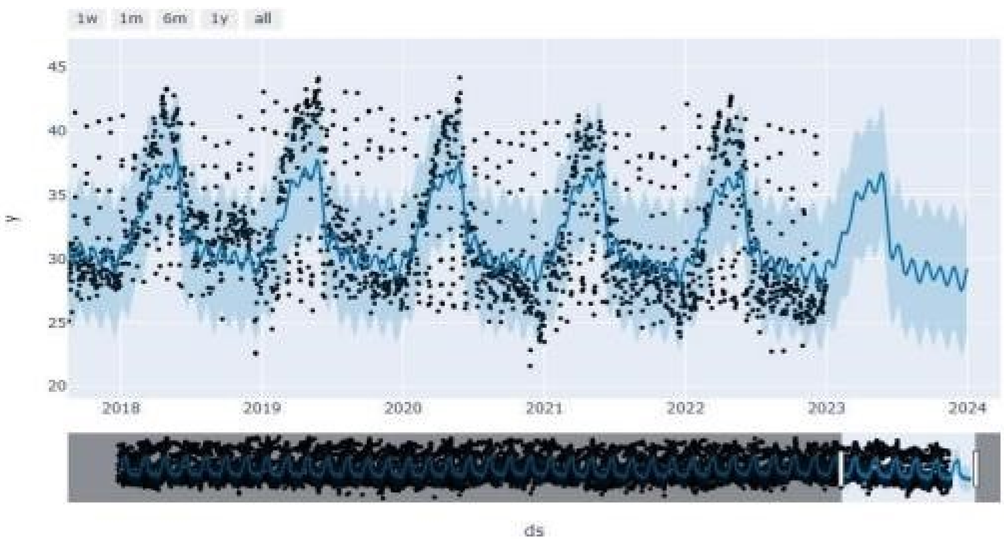

Fig 3.3 Weather prediction of past and for future

In conclusion, our analysis affirms the potential of the Prophet model to revolutionize weather forecasting. It showcases the model's capacity to seamlessly integrate accuracy, visualization, and result-driven insights, propelling us closer to the realm of precise time series-based weather predictions.

CHAPTER 8

CONCLUSION

8.1 CONCLUSION

Machine learning has been applied in various forms to weather forecasting, and its results have been shown to be promising in some cases. However, weather forecasting is a complex problem, and the accuracy of machine learning models can still be improved.

Additionally, machine learning models for weather forecasting typically require large amounts of data and computing resources, and their predictions can still be subject to errors and biases. Overall, machine learning has shown potential for improving weather forecasting, but there is still room for further research and development.

8.2 FUTURE ENHANCEMENTS

Extreme Event Prediction:
Develop specialized models to predict rare and extreme weather events, aiding in disaster preparedness.
Interdisciplinary Collaboration:
Collaborate with domain experts, climatologists, and environmental scientists to enrich models with domain-specific insight **Automated Data Collection:**
Implement AI-powered systems to automatically collect and preprocess data, reducing human intervention and errors.
Real-time Data Streaming:
Utilize AI to process and analyse real-time data streams, enabling up-to-the-minute forecasts.
Continuous Model Learning:
Develop models that can learn and adapt to evolving weather patterns over time, ensuring sustained accuracy.

Explainable AI:

Integrate explainable AI techniques to provide insights into why specific forecasts are generated, enhancing model transparency.

Hyper-localization:

Implement AI techniques to provide highly localized forecasts, incorporating microclimate data and urban heat island effects.

By pursuing these avenues of enhancement, this weather forecasting project can evolve into a cutting-edge platform, providing even more accurate, timely, and valuable insights to users.

References

[1] Dataset: The dataset is taken from NASA power access viewer. https://power.larc.nasa.gov/data-access-viewer

[2] A. Sulasikin, Y. Nugraha, J. I. Kanggrawan and A. L. Suherman, "Monthly Rainfall Prediction Using the Facebook Prophet Model for Flood Mitigation in Central Jakarta," 2021 IEEE International Conference on ICT for Smart Society (ICISS), Bandung, Indonesia, 2021.

[3] J., Jagannathan & C, Divya. (2021). Time Series Analyzation and Prediction of Climate using Enhanced Multivariate Prophet. International Journal of Engineering Trends and Technology. 2021

[4] Oo, Zar & Phyu, Sabai. (2020). Time Series Prediction Based on Facebook Prophet: A Case Study, Temperature Forecasting in Myintkyina. International Journal of Applied Mathematics Electronics and Computers. 2020

[5] C. Ma, C. Yan, Q. Suo and C. Liu, "Meteorological data combination prediction model based on Prophet + LSTM," *2022 IEEE 6th Information Technology and Mechatronics Engineering Conference (ITOEC)*, Chongqing, China, 2022

[6] C. El Hachimi, S. Belaqziz, S. Khabba and A. Chehbouni, "Towards precision agriculture in Morocco: A machine learning approach for recommending crops and forecasting weather," *2021 International Conference on Digital Age & Technological Advances for Sustainable Development (ICDATA)*, Marrakech, Morocco, 2021

Monthly Rainfall Prediction Using the Facebook Prophet Model for Flood Mitigation in Central Jakarta

1st Andi Sulasikin
Jakarta Smart City
Department of Communication, Informatics, Statistics
Jakarta, Indonesia
andisulasikin.jsc@gmail.com

2nd Yudhistira Nugraha
Jakarta Smart City, Department of Communication, Informatics, Statistics, Jakarta, Indonesia
School of Computing, Telkom University, Bandung, Indonesia
yudhistira.nugraha@jakarta.go.id

3rd Juan Intan Kanggrawan
Jakarta Smart City
Department of Communication, Informatics, Statistics
Jakarta, Indonesia
juan.tan.kang@gmail.com

4th Alex L. Suherman
Jakarta Smart City, Department of Communication, Informatics, Statistics, Jakarta, Indonesia
Directorate of Research and Community Service, Telkom University, Bandung, Indonesia
alex_lukmanto@yahoo.com

***Abstract*—Jakarta has been known as the city where floods are prevalent. As the vital region in Jakarta where the center of government and business are located, Central Jakarta is inseparable from the flood when the rainfall is remarkably high. Therefore, the Jakarta Provincial Government need a data-driven policy to facing potential flood that may occur each year to protect the citizen from the threat of flood disaster. Monthly rainfall prediction can be a reference to determine the possibility of considerable loss and damage due to disaster threats. However, at this moment, it is still challenging to find a fitting forecasting model for this context. This paper reports a comparison of three different time series models: Seasonal Autoregressive Integrated Moving Average (SARIMA), Facebook Prophet, and Long Short-Term Memory (LSTM) to forecast monthly rainfall in Central Jakarta for up to two consecutive years. The result indicates that Facebook Prophet, with the lowest Mean Squared Error (MSE) and Root Mean Squared Error (RMSE), is the fittest model to predict the monthly rainfall in Central Jakarta. It shows that a high amount of rainfall will be seen in January and February 2021, which suggests we need to be prepared to anticipate the potential flood. Facebook Prophet shows promising results in supporting data-driven policy for flood mitigation in Jakarta. The development of this model in the future can be used as a baseline study to formulate a data-driven policy for flood mitigation in Jakarta.**

Keywords—Jakarta, rainfall, forecasting, Facebook Prophet, flood mitigation

I. Introduction

Jakarta, the capital city of Indonesia, has been facing regular flood disasters every rainy season. As the capital city, Jakarta is the largest economic centre in Indonesia. Therefore, when floods hit the Jakarta area every time the rainy season arrives, it tarnished Jakarta's image as the nation's capital in the world's eyes and often paralyzed the national economic sector [1].

Jakarta flooding is caused by many factors, such as (1) the geological and geomorphological aspects of the Jakarta region, which are flood basins and floodplains, (2) the morphometry of the 13 river that flows across the Jakarta area, and (3) the infrastructure and social behaviour of its citizens [1]. With Jakarta's geographical condition, which is in a lowland area and is crossed by 13 rivers, the Capital City must be prepared for the risk of flooding, especially during high rainfall season. The high intensity of rain made Jakarta prone to disasters due to the overflow of water from the rivers that passed through it. When the intensity of the precipitation is starting to increase, we must begin to be alert.

Based on the Jakarta flood data report from 2002 to 2020, especially in February 2002, February 2007, January 2013, and February 2015, the iconic Hotel Indonesia (HI) Roundabout to Thamrin, which is a strategic area located in Central Jakarta, was hit by floods due to heavy rainfall [2]. In a separate report, the HI Roundabout was also waterlogged in August 2016, become an anomaly at that time [3]. This is unfortunate because the HI Roundabout location to the Thamrin area is right in the heart of Jakarta Province. Apart from being the center of government, it also functions as a center for business, trade, service activities, and a significant infrastructure development area.

The high risk of floods requires the Jakarta Provincial Government to protect Jakarta citizens from the threat of disaster. Monthly rainfall prediction can be a reference to determine the possibility of considerable loss and damage due to disaster threats. Therefore, the government needs to prepare and produce well-tailored plans to face a high risk of floods.

Many studies have been conducted using statistical models to predict future rainfall. Literature studies have shown several models that are widely used around the globe to forecast rainfall, such as Seasonal Autoregressive Integrated Moving Average (SARIMA) [4][5][6][7] and Long Short-Term Memory (LSTM) [8][9]. The recent model introduced by the nature of time series forecasting at Facebook, the Prophet, which is started to be applied to hydrometeorological time series, has also been used to predict rainfall [10]. However, it is still challenging to find other studies implementing the Prophet for rainfall, especially for Jakarta cases.

This study aims to provide higher accuracy in forecasting the monthly rainfall trend in Jakarta, especially for the Central Jakarta region, through data exploration and a quantitative approach. This approach can be successful by comparing time series forecasting models, such as ARIMA, LSTM, and

978-1-6654-1697-9/21/$31.00 ©2021 IEEE

Facebook Prophet. The three models can provide decent predictions on monthly rainfall with high accuracy based on the evaluation indicators, such as Mean Squared Error (MSE) and Root Mean Squared Error (RMSE) [10][11][12].

This study applies three time series methods to forecast monthly rainfall in Central Jakarta by using the accessible datasets from the Meteorological, Climatological, and Geophysical Agency (Indonesian: Badan Meteorologi, Klimatologi, dan Geofisika, abbreviated BMKG), accessible on data.online.bmkg.go.id. The dataset reveals that Facebook Prophet, with the lowest MSE and RMSE, is the most suitable method to predict the upcoming monthly rainfall recorded from Kemayoran Station, Central Jakarta. This paper offers a comprehensible understanding and indication of the predicted monthly rainfall in the next 24 months from September 2020 to August 2022 in the Central Jakarta region. Hence, the Jakarta Provincial Government can prepare and construct policies or strategies that are well-tailored high yielding to tackle floods mainly caused by rainfall in Jakarta.

The writing of this paper is structured as follows. Section 2 explains the materials and methodology for comparing three time-series models. Section 3 describes the findings of the study. Section 4 discusses how the results compare with related work and elaborate on the findings' implication for recommendations. Finally, Section 5 ends with the conclusion and future work.

II. METHODOLOGY

This study aims to conduct rainfall prediction for the next two years, from September 2020 to August 2022, by comparing three time series forecasting models which are frequently applied to forecast seasonal trends. The models used in this study are Seasonal Autoregressive Integrated Moving Average (SARIMA), Facebook Prophet, and Long-Short Term Memory (LSTM) which are evaluated based on Mean Square Error (MSE) and Root Mean Square Error (RMSE) score. MSE and RMSE are the evaluation parameter outcomes to measure the prediction model's fit to the actual data

A. Datasets

The datasets used in the study consist of rainfall time series datasets from January 1st, 2008, to August 31, 2020, recorded daily every 7 am from Kemayoran BMKG Station in Jakarta. These data were collected from http://dataonline.bmkg.go.id/. Table 1 shows the raw dataset of rainfall in millimeter (mm) with the lowest rainfall at “0”, which means there is no rain recorded at the moment.

TABLE I. DAILY RAINFALL (MM) RAW DATASET

Date	Rainfall (mm)
01-01-2008	29,4
02-01-2008	1,6
03-01-2008	32,3
04-01-2008	12,5
….	....
30-08-2020	0
31-08-2020	0

B. Data Collection and Analysis

The dataset used in the study contains a daily time series table of rainfall (millimeter, mm) recorded at Kemayoran BMKG Station in Jakarta. Initially, the datasets have been pre-processed for this study to focus on the monthly amount of rainfall by summing up the daily rainfall into monthly rainfall.

In data exploration, we apply data visualization to analyze the rainfall trend and identify the right parameters to answer the problem statement and appropriate approach to conduct time series forecasting. After data pre-processing and data exploration, we split the dataset's rainfall divided into two subsets. The first subset is a training set with time-series data from January 2008 to August 2018 (85% of the data population) to train the models. The second subset is a testing set developed from September 2018 to August 2020 (15% of the data population). SARIMA, Facebook Prophet, and LSTM were selected as models in this study.

1) SARIMA

Box and Jenkins (1970) developed and applied a modeling approach for time series analysis and forecasting to accommodate stationary and seasonality time series data. This modeling approach was also to fit the autoregressive moving average (ARMA), autoregressive integrated moving average (ARIMA), and seasonal autoregressive integrated moving average (SARIMA)[7]. In this study, fine-tuning parameter statsmodels.tsa.statespace.sarimax.SARIMAX in Python for SARIMA parameters (p, q, d)x(P, Q, D) was performed. Stsatsmodels.org provides it to find the model with the lowest Akaike's Information Criterion (AIC). The lower value of AIC (Akaike Information Criterion) indicates the better quality of the statistical model to the fitting [6].

2) Facebook Prophet

A Prophet is developed and introduced by Facebook in 2017, available on Python and R [13][14], to accommodate three main features, namely trend, seasonality, and holidays[15], and the demand for high quality and practical approach to forecasting at scale [16]. Prophet parameters consist of capacities, changepoints, holiday and seasonality, and smoothing parameters that can be interpretably applied to improve the model. In this study, we tune the parameters by imposing assumptions to get the fittest Prophet forecast models, such as choosing classical multiplicative seasonal decomposition and applying hyperparameter tuning using Parameter Grid sklearn.model_selection in Python.

3) LSTM

The Long-Short Term Memory (LSTM) is a form of recurrent neural network that stores past information into its memory cells, and during training, it learns when to use the memory [17]. LSTM parameter consists of the number of hidden layers, the number of hidden units per layer, learning rate of the optimizer, dropout rate, and the number of iterations. Unlike SARIMA and Prophet that use tuning parameters, we will not tune the network parameters with grid search parameters. This study will use multilayer LSTM (three layers) with standard parameters, performed by a little trial and error. The number of epochs was fixed at 50 to get minimal loss.

We apply these three models to analyze and predict the future monthly rainfall recorded. The learning models are then assessed based on important parameters, such as Mean Square Error (MSE) and Root Mean Square Error (RMSE) score

[10][11][12], as described below. Section 3 presents the evaluation indicator outcomes.

1) Mean Squared Error (MSE)

MSE measures the difference between the actual value and the predicted value (residual), which is then squared to get the MSE value. The error is expressed in squared target units by measuring the average squares of the errors, which is the average squared difference between the estimated values and the actual value.

$$MSE(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \qquad (1)$$

2) Root Mean Squared Error (RMSE)

RMSE is the square root of MSE. It calculates the prediction model's absolute fit to the data. Therefore, showing how accurate the model's predicted values to the observed data points. It is regularly applied as both an evaluation metric and a loss function.

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

After we get the fittest model based on the evaluation parameters, we conduct a 24-months prediction by applying the most suitable model for monthly rainfall in Central Jakarta. Figure 1 shows the proposed approach in the study.

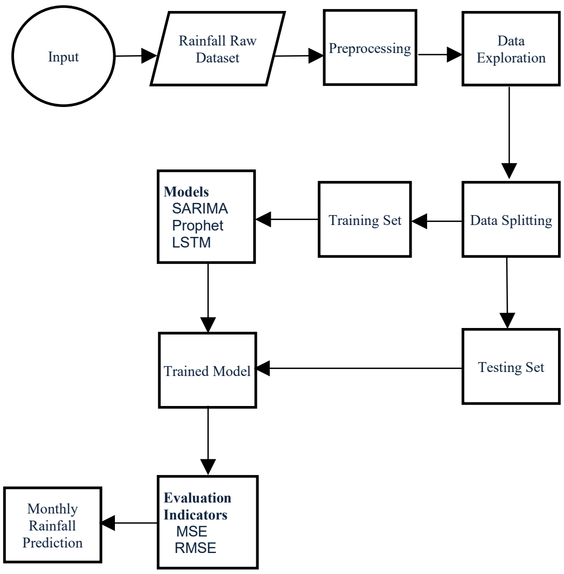

Fig. 1. Workflow Approach

III. Results

This paper presents a system model for forecasting monthly rainfall in Central Jakarta using time series forecasting models. The dataset contains a daily time series table of rainfall (mm) recorded at Kemayoran BMKG Station in Central Jakarta. Data exploration and forecast monthly rainfall in the upcoming 24 months were performed applying three-time series forecasting models frequently used for hydrometeorological analysis. This process was conducted to forecast the forthcoming monthly rainfall as a reference to determine the strategic mitigation and possibility of damage due to flood threat in Central Jakarta.

A. Data Exploration

Data exploration is needed to understand the time series data of monthly rainfall based on trend and seasonality. This is an important step to choose the appropriate variables and the proper approach to conduct time series forecasting. This study uses moving average smoothing analysis and additive decomposition method for data exploration, as shown in Figure 2 and Figure 3, respectively.

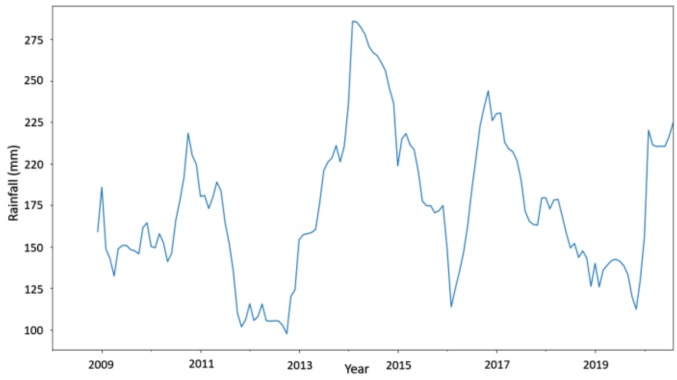

Fig. 2. 12-month Moving Average time-series

At first, an analysis of moving average smoothing was conducted. It is one of effective techniques in time series forecasting. Smoothing is a method used in time series to eliminate the short-term volatility in data to help us better see patterns. We apply the method to reduce noise and to uncover clearer signals in the direction of the rainfall trend. In time series analysis and forecasting, moving average is recognized as a straightforward and usual type of smoothing [18]. Figure 2 illustrates representing the dataset into a moving average with a 12-month window size. It shows a clear pattern appearing when we plot the data. The time-series has a seasonal pattern, which is rainfall is always high every three to four years. This suggests that we may set the yearly seasonal parameters in applying the Prophet and SARIMA models.

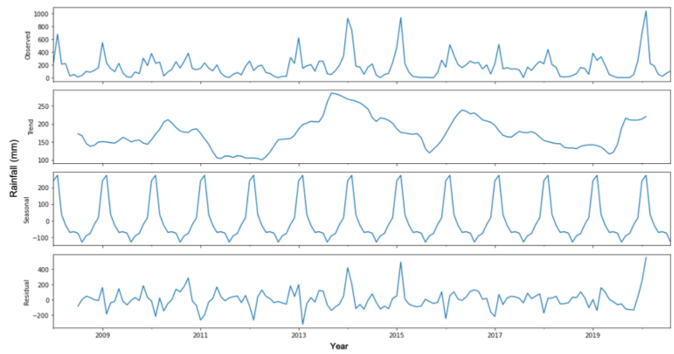

Fig. 3. Additive Decomposition

After analyzing moving average smoothing, we conduct the Time-Series Additive Decomposition method to visualize and decompose the time series into three distinct components: trend, seasonality, and noise. The additive decomposition of the monthly rainfall from September 2008 to August 2020 is shown in Figure 3. The figure reveals no trend detected, and the rainfall is relatively unstable based on the residual curve. However, it is showing seasonality which suits the result of smoothing analysis. It is entirely appropriate to apply SARIMA, Prophet with yearly seasonality, and LSTM with a

12-month batch to predict rainfall for two years ahead from September 2020 to August 2022.

B. *Forecasting Results*

Three favorable time series forecasting models, SARIMA, LSTM, and Prophet methods, have been used to predict the monthly rainfall in the next 24 months from September 2020 to August 2022 recorded at Kemayoran Station. The monthly rainfall dataset is divided into two subsets: a training set (85% of the data population) to train the models and a testing set (15% of the data population). Table 2 and Figure 4 show the validation of the forecasting results of the methods. At first, we evaluate the models using MSE and RMSE, as shown in Table 2. It reveals that Prophet has the lowest MSE (23821,93) and RMSE (154,34), while LSTM (46206,51 and 214,96 respectively) performs poorly. SARIMA has much lower MSE (25006,03) and RMSE (158,13) than LSTM, but it still has a little higher MSE and RMSE than Prophet. This suggests that the fittest model to forecast the monthly rainfall in Central Jakarta is Prophet.

TABLE II. EVALUATION PARAMETERS RESULT

Model Name	MSE	RMSE
SARIMA	25006,03	158,13
Prophet	23821,93	154,34
LSTM	46206,51	214,96

After evaluating the parameters, as part of the validation test, we also visualize the validation test of SARIMA, LSTM, and Prophet models with the actual rainfall dataset from September 2018 to August 2020, as shown in Figure 4. The figure, including the three models, predicts that the monthly rainfall will probably peak in February both in 2019 and 2020. It also shows that Prophet and SARIMA have an identical curve, and their predictions are closer to the observed monthly rainfall than LSTM. This suits the evaluation parameter result, as previously explained. Based on the evaluation parameter results in Table 2 and supported by Figure 4, we can conclude that the Prophet is the fittest model because it has the lowest score for MSE at 23821,93 and RMSE 154,34, and its value was close to actual rainfall. However, the Prophet prediction still does not fit the actual value by a fairly wide margin from January to February 2020. It may suggest that actual rainfall from January to February 2020 is a rare event.

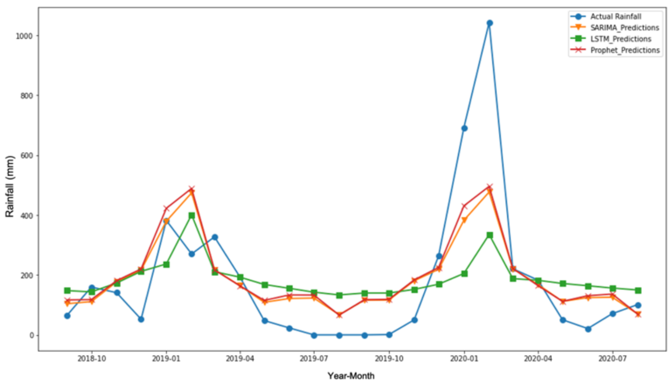

Fig. 4. Time Series Forecasting Models Validation

As we conclude that Prophet is the fittest model to forecast the monthly rainfall based on Table 2 and Figure 4, we then use the Prophet as the chosen model to forecast the monthly rainfall for the next two years from September 2020 to August 2022. The forecasting results are shown in Figure 5, where we also provide the lower and upper values, indicating boundaries for the forecasting. As the lowest value for rainfall is zero, we can change and interpret the negative value for lower rainfall in Figure 5 as zero "0". The result shows that the rainfall fluctuates over this period. In January and February 2021, we will probably see high amounts of rainfall, not as high as or higher than in January and February 2020. This suggests we need to be more prepared to anticipate the flood than before.

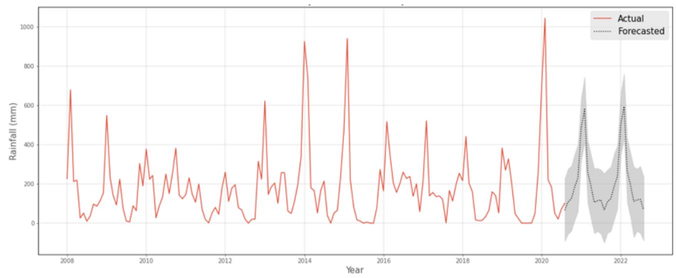

Fig. 5. Monthly Rainfall Prediction using Prophet Model

IV. DISCUSSION

This study aims to monitor and forecast monthly rainfall for the next two years, from September 2020 to August 2022. The dataset used in the study contains a daily time series table of rainfall (mm) from January 1st, 2020 to August 31st, 2022, recorded at Kemayoran BMKG Station in Central Jakarta. The dataset and quantitative analysis can be useful for evaluating rainfall trends in Central Jakarta. Furthermore, it can also present a rainfall trend in the last few years and help government authorities be more proactive by making effective strategies and practical action for mid-term mitigation. Data exploration were applied as opening data analysis to recognize the problem and then to applied appropriate time series forecasting models to predicts the upcoming rainfall situation.

As explained in the Data Exploration part, the monthly rainfall in Central Jakarta keeps fluctuating in the last few years, with the peak in January and February every year. It is an alarming situation. Then we applied time-series forecasting to predict the monthly rainfall for the next two years using SARIMA, Prophet, and LSTM models. Such models can provide us with a decent prediction for time series data.

Table 2 shows that Prophet is the most suitable model for predicting monthly rainfall recorded at Kemayoran Station, Central Jakarta, based on the evaluation indicators with the lowest MSE and RMSE scores. Meanwhile, LSTM, which has the highest MSE and RMSE, is the least accurate for this case, resulting from the lack of high volumes of data that deep-learning algorithms need to thrive. In addition, the SARIMA model accuracy is better than the LSTM model for forecasting monthly rainfall on the evaluation parameters. SARIMA model also has a smaller margin error than Prophet. These results are also supported by their model projection curves in Figure 4, as discussed before.

The use of the Prophet for hydrometeorological time series is still limited. A related study mentioned that the Prophet had not been used to hydrometeorological time series previously[10]. The result of the study of investigating the predictability of monthly temperature and precipitation shows Prophet model is more competitive compared to AutoRegressive Fractionally Integrated Moving Average (ARFIMA), exponential smoothing state-space model with

Box-Cox transformation, ARMA errors, Trend and Seasonal components (BATS), simple exponential smoothing, and Theta. Prophet's superiority over these mentioned models can be seen when combined with externally applied classical seasonal decomposition, as we use classical multiplicative seasonal decomposition for Prophet in this study.

As projected in Figure 10 using the Prophet model, the predicted monthly rainfall for the upcoming months is rising from October 2020 (123,68 mm) to January 2021 (485,41mm) and February 2021 (579,78 mm). The forecasting result also notes that we should be aware of the upper monthly rainfall prediction, which may peak at 742,18 mm in February 2021. As explained in Section 1, the iconic Hotel Indonesia (HI) Roundabout to Thamrin was hit by floods due to rainfall in January 2013, February 2015, and August 2016 [2][3] that has monthly rainfall 621,9 mm, 939,5 mm, and 227,2 mm respectively, shown in Figure 5. From these historic events in Central Jakarta, we will probably see other floods occur in this region if there is no effective mitigation tool.

The Jakarta Provincial Government needs to take more preventive measures and interventions to face the season with increasingly high rainfall in the upcoming months from September 2020 to August 2022. The data-driven policy based on this model, as described above, could help the government in minimizing the impacts of floods. For example, ensuring the water lines and ropes are not clogged. It is essential to maintain the optimum condition of waterways throughout the capital city so that water flow is not obstructed and causes puddles or flooding. In addition, repairing a collapsed dam and normalizing rivers in the entire Jakarta region is necessary. The city co-creator (e.g., citizen and industry) will also decide on this seasonal event. Another key to successful flood management is to involve the community. All parties must possess continuous perception, awareness, and discipline by not littering or throwing garbage into rivers/streams and a culture of protecting the environment.

V. CONCLUSION

To conclude, this paper has presented the trend and forecast of monthly rainfall in Central Jakarta for the next two years. The Prophet has been the most appropriate model in this study compared to SARIMA and LSTM based on the MSE and RMSE scores. The predicted monthly rainfall keeps fluctuating, with the bottom in August and the top in February for each forthcoming year. The result has shown that rainfall will reach the peak from January 2021 (485,41 mm) to February 2021 (579,78 mm), which is alarming.

Some improvements could be added to this proposed model. For example, exploring the prediction methodology using additional historical and updated datasets. Note that it is crucial to optimize the use of LSTM, add more datasets, put historical and updated datasets, and explore different comparison methods. Furthermore, as we can conduct monthly rainfall forecasting for Central Jakarta, we can predict the whole or other regions in Jakarta. Therefore, we also need a historical rainfall dataset representing the other regions in Jakarta. The ultimate goal is to find a more appropriate forecasting model for Jakarta entirely.

ACKNOWLEDGEMENT

The authors thank the Meteorological, Climatological, and Geophysical Agency (BMKG) for the datasets of daily rainfall recorded in Kemayoran Station, Central Jakarta. The contents reflect only the authors' views and not the Provincial Government of Jakarta's views. The authors declare no competing financial interest.

REFERENCES

[1] D. R. Prabawadhani, B. Harsoyo, T. H. Seto, and B. R. Prayoga, “Karakteristik Temporal Dan Spasial Curah Hujan Penyebab Banjir di Wilayah Dki Jakarta Dan Sekitarnya,” *J. Sains Teknol. Modif. Cuaca*, vol. 17, no. 1, p. 21, 2016, doi: 10.29122/jstmc.v17i1.957.

[2] merdeka.com, “Data Banjir Jakarta 2002-2020, Istana dan Bundaran HI Sampai Terendam,” 2020. https://www.merdeka.com/trending/data-banjir-jakarta-2002-2020-istana-dan-bundaran-hi-sampai-terendam.html? (accessed Oct. 20, 2020).

[3] REPUBLIKA.CO.ID, “Bundaran HI Terendam Air, Ahok: Gak Masuk Akal.” .

[4] P. P. Dabral and M. Z. Murry, “Modelling and Forecasting of Rainfall Time Series Using SARIMA,” *Environ. Process.*, vol. 4, no. 2, pp. 399–419, 2017, doi: 10.1007/s40710-017-0226-y.

[5] E. H. Etuk and T. M. Mohamed, “Time Series Analysis of Monthly Rainfall data for the Gadaref rainfall station, Sudan, by Sarima Methods,” *Int. J. Sci. Res. Knowl.*, vol. 2, no. 7, pp. 320–327, 2014, doi: 10.12983/ijsrk-2014-p0320-0327.

[6] S. Wang, J. Feng, and G. Liu, “Application of seasonal time series model in the precipitation forecast,” *Math. Comput. Model.*, vol. 58, no. 3–4, pp. 677–683, 2013, doi: 10.1016/j.mcm.2011.10.034.

[7] T. Papalaskaris, T. Panagiotidis, and A. Pantrakis, “Stochastic Monthly Rainfall Time Series Analysis, Modeling and Forecasting in Kavala City, Greece, North-Eastern Mediterranean Basin,” *Procedia Eng.*, vol. 162, pp. 254–263, 2016, doi: 10.1016/j.proeng.2016.11.054.

[8] F. Kratzert, D. Klotz, C. Brenner, K. Schulz, and M. Herrnegger, “Rainfall-Runoff modelling using Long-Short-Term-Memory (LSTM) networks,” pp. 1–26.

[9] M. Rizki, S. Basuki, and Y. Azhar, “Implementasi Deep Learning Menggunakan Arsitektur Long Short Term Memory (LSTM) Untuk Prediksi Curah Hujan Kota Malang,” *J. Repos.*, vol. 2, no. 3, p. 331, 2020, doi: 10.22219/repositor.v2i3.470.

[10] G. Papacharalampous, H. Tyralis, and D. Koutsoyiannis, “Predictability of monthly temperature and precipitation using automatic time series forecasting methods,” *Acta Geophys.*, vol. 66, no. 4, pp. 807–831, 2018, doi: 10.1007/s11600-018-0120-7.

[11] Z. Kafara, F. Y. Rumlawang, and L. J. Sinay, “Peramalan Curah Hujan Dengan Pendekatan Seasonal Autoregressive Integrated Moving Average (Sarima),” *BAREKENG J. Ilmu Mat. dan Terap.*, vol. 11, no. 1, pp. 63–74, 2017, doi: 10.30598/barekengvol11iss1pp63-74.

[12] A. Sulasikin, Y. Nugraha, J. Kanggrawan and A. L. Suherman, "Forecasting for a data-driven policy using time series methods in handling COVID-19 pandemic in Jakarta," *2020 IEEE International Smart Cities Conference (ISC2)*, 2020, pp. 1-6, doi: 10.1109/ISC251055.2020.9239066.

[13] S. J. Taylor and B. Letham, “Business Time Series Forecasting at Scale,” *PeerJ Prepr. 5e3190v2*, vol. 35, no. 8, pp. 48–90, 2017, [Online]. Available: https://peerj.com/preprints/3190/%0Ahttp://ezproxy.bangor.ac.uk/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=c8h&AN=108935824&site=ehost-live%0Ahttps://peerj.com/preprints/3190/%0Ahttps://peerj.com/preprints/3190.pdf.

[14] Facebook, “Automatic forecasting procedure,” 2019.

[15] Facebook, “Prophet: forecasting at scale,” 2017. https://research.fb.com/prophet- forecasting-at-scale/ (accessed Oct. 20, 2020).

[16] K. K. R. Samal, K. S. Babu, S. K. Das, and A. Acharaya, “Time series based air pollution forecasting using SARIMA and prophet model,” *ACM Int. Conf. Proceeding Ser.*, pp. 80–85, 2019, doi: 10.1145/3355402.3355417.

[17] S. Hochreiter, “Long Short-Term Memory,” vol. 1780, pp. 1735–1780, 199.

[18] J. Brownlee, “Moving Average Smoothing for Data Preparation and Time Series Forecasting in Python,” *Machine Learning Mastery*, 14-Aug-2020. [Online]. Available: https://machinelearningmastery.com/moving-average-smoothing-for-time-series-forecasting-python/. (accessed Oct. 20, 2020).