Machine Learning-Based CO_2 Emission Prediction and Industry Classification for Malaysia

KHAIRUNISAH MOHD HAMDAN

INFORMATION TECHNOLOGY

UNIVERSITI TEKNOLOGI PETRONAS

JAN 2025

Machine Learning-Based CO_2 Emission Prediction and Industry Classification for Malaysia

By

Khairunisah Binti Mohd Hamdan

21000158

Dissertation submitted in partial fulfilment of
the requirements for the
Degree of Study (Hons)
(Information Technology)

JAN 2025

Universiti Teknologi PETRONAS
32610 Seri Iskandar
Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

Machine Learning-Based CO_2 Emission Prediction and Industry Classification for Malaysia

By

Khairunisah Mohd Hamdan

21000158

A project dissertation submitted to the

Information Technology Programme

Universiti Teknologi PETRONAS

In partial fulfilment of the requirement for the

BACHELOR OF INFORMATION TECHNOLOGY (Hons)

Approved by,

Nazleeni Samiha Haron
Lecturer
Computer and Information Sciences Department
Universiti Teknologi PETRONAS
32610 Seri Iskandar
Perak. MALAYSIA

(Ts. Nazleeni Samiha Bte Haron @ Baharon

UNIVERSITI TEKNOLOGI PETRONAS
SERI ISKANDAR, PERAK

JAN 2025

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgments, and that the original work contained herein have not been undertaken or done by unspecified sources or persons

Khairunisah

KHAIRUNISAH MOHD HAMDAN

ABSTRACT

Malaysia’s commitment to Net Zero Carbon Emissions (NZCE) by 2050 and its Nationally Determined Contributions (NDC) under the Paris Agreement necessitate accurate tools for forecasting emissions and identifying high-emission sectors. However, the current approaches used in Malaysia lack predictive insight and sector-specific classification, limiting the effectiveness of carbon reduction strategies. This project aims to address that gap by developing a machine learning-based system to predict Malaysia’s future CO_2 emissions using ARIMA, LSTM, and Prophet models and classify industries into High, Medium, and Low emission levels using Decision Tree, Random Forest, and k-Nearest Neighbours (kNN). Historical datasets were collected from Our World in Data (OWID) and Malaysian industry-level emissions were compiled for the years 1990 to 2023. These datasets were pre-processed and engineered to fit both forecasting and classification model requirements, ensuring alignment with CRISP-DM methodology. The LSTM model emerged as the most accurate forecasting technique with a Mean Absolute Error (MAE) of 7.95 and a Mean Absolute Percentage Error (MAPE) of 3.08%, outperforming ARIMA and Prophet in capturing nonlinear patterns in emissions. For industry classification, the Random Forest model achieved the highest accuracy of 88%, effectively distinguishing sectoral emission levels and providing insights into policy prioritization. A Power BI dashboard was developed to visualize the forecasting results, emission breakdowns, and industry classification outputs interactively. This solution not only improves upon traditional methods used in Malaysia but also provides a data-driven decision-support tool that can guide national and corporate stakeholders in climate policy planning. Through the integration of predictive analytics and machine learning classification, the project contributes meaningfully to Malaysia’s carbon reduction roadmap toward NZCE 2050.

ACKNOWLEDGEMENTS

First and foremost, I would like to express my deepest gratitude to Allah SWT for His endless blessings, strength, and guidance throughout my academic journey and especially during the completion of this final year project. Without His will, none of this would have been possible. I am also profoundly thankful to my beloved family, especially my parents, for their unconditional love, constant prayers, and unwavering support. Their encouragement has been the pillar of my strength and motivation in overcoming challenges throughout my degree.

My sincere appreciation goes to my supervisor, Ts. Nazleeni Samiha Binti Haron for her continuous guidance, insightful feedback, and strong mentorship. Since my first year in her Database Systems class, she has played a significant role in shaping my academic interests particularly in machine learning and data analytics which eventually inspired the foundation of this project. My appreciation also goes to Universiti Teknologi PETRONAS (UTP) for providing an enriching learning environment, access to essential tools, and continuous academic support that enabled the successful implementation of this research.

A special thank you goes to Muhammad Arman Adly, who has been by my side through thick and thin. His constant support, motivation, and genuine care have helped me remain focused and resilient, both academically and personally. I am truly grateful for his presence and encouragement throughout my university years.

I would also like to express my gratitude to PETRONAS Chemicals LDPE Ammonia Sdn. Bhd., especially Mr. Fairulzaki B. M. Hassan and Ms. Nureen Nabila M. Sharul Nizam, for their valuable insights and professional feedback during our consultation sessions. Their input has contributed significantly to enhancing the real-world relevance of this project.

To my friends, Yasmin Zalikha, Aliah Athirah, Nurul Hafizhah, Najlaa Syafiah, and many others, thank you for the emotional support, laughter, and memories that brightened my university life. Our friendship has made this journey truly meaningful.

Last but not least, I wanna thank me, I wanna thank me for believing in me, I wanna thank me for doing all this hard work, I wanna thank me for having no days off, I wanna thank me for never quitting, I wanna thank me for always being a giver and trying to give more than I receive, I wanna thank me for trying do more right than wrong. I wanna thank me for just being me at all times.

TABLE OF CONTENT

CHAPTER 4 : RESULTS AND DISCUSSION

CHAPTER 5 : CONCLUSION AND RECOMMENDATION

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1
INTRODUCTION

1.1 Background of Study

The manufacturing, energy, and transportation sectors of Malaysia's industrial landscape are the primary contributors to the country’s carbon dioxide (CO_2) emissions. This poses serious challenges for emission reduction, regulatory compliance, and long-term sustainability efforts. Malaysia's shift toward a low-carbon economy began with early strategies focused on reducing carbon intensity relative to GDP began nearly a decade ago, as outlined by Rahim (2014). Since then, the nation has strengthened its climate commitment by enacting several major policy instruments to support its Net Zero Carbon Emissions (NZCE) 2050 goal. These include the Low Carbon Cities Framework (LCCF), which promotes sustainable urban development; the National Energy Policy 2022–2040, focused on energy transition strategies; and the Nationally Determined Contributions (NDCs) under the Paris Agreement, which set Malaysia’s formal targets for cutting greenhouse gas emissions.(MoE Malaysia, 2023)

Notwithstanding these governmental initiatives, there is still a significant lack of sophisticated prediction systems that can reliably anticipate changes in CO_2 emissions and categorise industrial sectors according to their carbon footprint. This restriction makes it more difficult to make proactive plans and decisions, particularly as Malaysia works to meet international sustainability objectives, such as the Sustainable Development Goals (SDGs) of the UN. Specifically, this study supports:

i. **SDG 7 (Affordable and Clean Energy)** – By analyzing emission trends and identifying high-emission industries, this research promotes the adoption of cleaner energy sources and efficient energy use.

ii. **SDG 9 (Industry, Innovation, and Infrastructure)** – Enhancing industrial sustainability through machine learning-driven insights, ensuring that industrial growth aligns with sustainability objectives.

iii. **SDG 13 (Climate Action)** – Providing data-driven solutions for policymakers to develop effective carbon reduction strategies, supporting Malaysia’s Net Zero 2050 agenda.

SUSTAINABLE DEVELOPMENT GOALS
1 NO POVERTY
2 ZERO HUNGER
3 GOOD HEALTH AND WELL-BEING
4 QUALITY EDUCATION
5 GENDER EQUALITY
6 CLEAN WATER AND SANITATION
7 AFFORDABLE AND CLEAN ENERGY
8 DECENT WORK AND ECONOMIC GROWTH
9 INDUSTRY, INNOVATION AND INFRASTRUCTURE
10 REDUCED INEQUALITIES
11 SUSTAINABLE CITIES AND COMMUNITIES
12 RESPONSIBLE CONSUMPTION AND PRODUCTION
13 CLIMATE ACTION
14 LIFE BELOW WATER
15 LIFE ON LAND
16 PEACE, JUSTICE AND STRONG INSTITUTIONS
17 PARTNERSHIPS FOR THE GOALS

Figure 1 : Sustainability Development Goals (SDG)

Historical data and conventional statistical models like Autoregressive Integrated Moving Average (ARIMA) and Linear Regression are the mainstays of Malaysia's current CO_2 forecasting techniques. Even though these models are frequently employed for time-series research, they have trouble capturing intricate, non-linear emission trends that are impacted by economic development, regulatory changes, and industrial expansion. In order to capture temporal relationships in emissions data, this study will employ a deep learning model called Long Short-Term Memory (LSTM), whereas ARIMA will serve as a baseline comparison model. Furthermore, Meta's Prophet forecasting model will be used to improve accuracy in identifying seasonal fluctuations and long-term trends, as it has been shown to effectively capture policy-driven changepoints and seasonal variations in CO_2 emissions forecasting (P.P. Linardatos et al., 2023).

Furthermore, there isn't yet a standardised framework in Malaysia for categorising high-emission businesses like cement manufacture, palm oil processing, and oil and gas. Current categorisation schemes are based on static threshold values that don't vary in response to shifts in energy use, industrial activity, and legislative actions. Targeted sustainability policies and more accurate carbon reduction strategies are made possible by machine learning classification techniques—such as Decision Trees, Random Forest, and k-Nearest Neighbours (kNN) which provide flexible, data-driven segmentation of industries based on CO_2 emission levels. For example, Tian et al. (2025) demonstrated that Random Forest models not only achieved high predictive accuracy ($R^2 \approx 0.98$) but were also highly sensitive to sector-specific emission drivers, while kNN delivered strong forecasting performance across multiple energy sources.

In order to assist manufacturers, energy producers, and regulators in their sustainability endeavours, this project intends to create an industry categorisation system and a machine learning-based CO_2 forecasting model. through the use of sector-specific and historical data from Our World in Data (OWID). In order to support Malaysia's Net Zero 2050 objectives, optimise regulatory actions, and enable data-driven sustainability activities, this project will offer practical insights. By bridging the gap between industrial operations and AI-driven climate action, our study supports global sustainability programs.

1.2 Problem Statement

1.2.1 Significant Contribution of Industrial Sectors to CO_2 Emissions

Malaysia's industrial sector, which includes transportation, energy generation, and manufacturing, continues to be a significant source of CO_2 emissions in the nation. The International Energy Agency (IEA, 2023) reports that in 2022, The transportation sector, accounting for 22% of Malaysia’s energy-related CO_2 emissions in 2022, is a significant contributor to national emissions. Accurate forecasting of vehicular emissions is crucial to inform future mitigation strategies, and time-series analysis has been effectively applied in prior studies to predict trends in this sector (Kadian, Savita, Choudhary, Amit et al., 2023). Emissions are predicted to increase due to ongoing industrial development, economic expansion, and high energy consumption, making Malaysia's goal of reaching Net Zero Carbon Emissions (NZCE) by 2050 even more challenging.

The National Energy Policy 2022-2040, the Low Carbon Cities Framework (LCCF), and Malaysia's Nationally Determined Contributions (NDCs) under the Paris Agreement are just a few national policies that aim to reduce emissions; however, current emission management strategies are not predictive. (Babatunde et al., 2023) It is now challenging to create proactive carbon reduction strategies since policymakers and companies depend more on historical reporting than on analytics that look forward. In order to accurately forecast emission trends and identify high-emission industries and implement more focused mitigation strategies that support Malaysia's Net Zero 2050 goals and the Sustainable Development Goals (SDGs) of the UN, particularly

SDG 7 (Clean Energy), SDG 9 (Industry Innovation), and SDG 13 (Climate Action), a data-driven and predictive approach is necessary.

1.2.2 Limitations of Existing CO_2 Emission Predictive Models

The majority of Malaysia's existing CO_2 emission forecasting techniques are based on conventional statistical models, including Linear Regression and Autoregressive Integrated Moving Average (ARIMA). Although ARIMA works well for short-term forecasting, it has trouble with seasonal changes and long-term forecasting, especially when there are non-linear connections between emissions and industrial activity (Segar et al., 2024).

According to an IIEA research from 2023, ARIMA-based models have trouble adjusting to changing emission trends, which restricts their capacity to assist with long-term sustainability planning. Furthermore, Malaysia does not yet have machine learning-based forecasting models that include outside variables like industry expansion, policy changes, and economic growth into predictive models. Because traditional models do not take into account the dynamic nature of industrial emissions, this gap prevents policymakers from making proactive, data-driven decisions. By using Long Short-Term Memory (LSTM), a deep learning model that can identify intricate patterns in emission trends and temporal relationships, our work overcomes this constraint. Furthermore, Meta's Prophet model will be used to seasonal trend identification and long-term forecasting, enhancing Malaysia's capacity to accurately anticipate and control CO_2 emissions.

1.2.3 Need for Data-Driven Decision-Making in Emission Management

At the moment, Malaysia does not have a strong decision-support system that combines predictive analytics with emissions data. In order to regulate emissions effectively, policymakers and industry stakeholders rely on antiquated forecasting techniques and static reporting that don't offer real-time information.

For sustainability planning, machine learning techniques are essential since they improve forecasting and categorisation accuracy. High emission prediction accuracy has been shown by models like Prophet for time-series forecasting and Decision Trees, Random Forest, and k-Nearest Neighbours (kNN) for industry categorisation (P.P.Linardatos et al, 2023). Malaysian policymakers and companies may more accurately identify high-emission sectors, monitor emissions dynamically, and develop adaptive sustainability policies that adjust to changes in the economy and industry by incorporating these models into a real-time decision-support framework. (Anonna et. Al, 2023)

Using ARIMA, LSTM, and Prophet for forecasting and Decision Trees, Random Forest, and kNN for industry classification, this work creates an enhanced CO_2 emission forecasting model and an industry classification system to solve these issues. Real-time visualisation and data-driven decision-making will be made possible by the integration of these models into an interactive Power BI dashboard. In order to give Malaysian policymakers access to a precise, AI-driven tool for tracking emissions and directing sustainability activities, the study will make use of historical and sector-specific data from Our World in Data (OWID).

1.3 Objectives

1.1.1 Develop an Advanced Predictive Model for CO_2 Emission Forecasting

To create and put into use a machine learning-based forecasting model that uses ARIMA, Long Short-Term Memory (LSTM), and Prophet to estimate Malaysia's CO_2 emissions during the next five to ten years. To increase forecasting accuracy, these models will be used to evaluate seasonal fluctuations, record long-term patterns, and analyse emission trends. In order to meet Malaysia's Net Zero Carbon Emissions (NZCE) 2050 regulations and Sustainable Development Goals (SDG 7, SDG 9, and SDG 13), this project intends to develop a more accurate and adaptable prediction system by combining deep learning (LSTM) with statistical models (ARIMA, Prophet).

1.1.2 Classify Malaysian Industries Based on CO_2 Emission Levels

To create a machine learning-based categorisation system that uses k-Nearest Neighbours (kNN), Random Forest, and Decision Trees to divide Malaysian businesses into three groups: High, Medium, and Low CO_2 emission levels. This classification system will replace static threshold-based approaches with an adaptive and data-driven segmentation approach. This study will help policymakers target high-emission businesses with targeted mitigation methods and direct industry sustainability efforts in line with Malaysia's environmental objectives by using sectoral CO_2 emissions data from Our World in Data (OWID).

1.1.3 Enhance Data-Driven Decision Making for CO_2 Emission Management in Malaysia

To use Power BI to create an interactive, real-time decision-support dashboard that will provide industry categorisation outcomes and CO_2 emission projections. With the use of this dashboard, Malaysian regulators, business executives, and policymakers will be able to analyse sectoral contributions to CO_2 levels, monitor emissions in real time, and put adaptive sustainability policies based on predictive analytics into action. By incorporating real-time data visualisation, evidence-based policymaking would be supported, enabling Malaysia to make decisions about emission reduction that are proactive rather than reactive.

1.4 Scope of Research

In order to anticipate Malaysia's CO_2 emissions and categorise sectors according to their emission levels, this project intends to create machine learning models. Due to their significant contributions to the country's CO_2 emissions, the study focusses on Malaysia's major industrial sectors. This study makes use of pertinent worldwide datasets, especially Our World in statistics (OWID), which offers sector-specific CO_2 emission statistics, to guarantee a thorough and localised analysis.

The study integrates time-series forecasting models and machine learning classification techniques to improve prediction accuracy and industry segmentation. For CO_2 emissions forecasting, ARIMA, Long Short-Term Memory (LSTM), and Prophet models will be used to predict Malaysia’s CO_2 emissions over the next 5–10 years. These models will allow for short-term and long-term trend analysis, seasonal pattern detection, and emission growth modelling, ensuring that policymakers and industry stakeholders have access to accurate and data-driven emission forecasts. ARIMA will be used for short-term linear trends, while LSTM, a deep learning model, will capture long-term temporal dependencies and complex relationships in emission patterns. Additionally, Prophet, a forecasting model developed by Meta, will be utilized for long-term predictions, as it effectively handles missing data, seasonal variations, and external factors such as policy changes and industrial growth.

In addition to CO_2 forecasting, this study will focus on classifying Malaysian industries based on their CO_2 emission levels using Decision Trees, Random Forests, and k-Nearest Neighbors (kNN). These classification models will categorize industries into High, Medium, and Low emission levels, allowing for a dynamic and adaptive approach to identifying high-emission sectors. Unlike traditional fixed-threshold classification methods, machine learning-based classification will enable real-time adjustments and more accurate policy recommendations based on changing industrial activities and sustainability targets. The results from these classification models will be instrumental in guiding Malaysia's Net Zero Carbon Emission (NZCE) 2050 policies and will support global sustainability goals such as the United Nations Sustainable Development Goals (SDG 7 - Affordable and Clean Energy, SDG 9 - Industry, Innovation, and Infrastructure, and SDG 13 - Climate Action).

An interactive Power BI dashboard will be created as part of this project to visualise CO_2 emission estimates and industry classifications in order to better decision-making and the practical implementation of the findings. Policymakers, regulatory agencies, and industry stakeholders will be able to use this dashboard as a decision-support tool to track industry contributions, dynamically monitor CO_2 emissions, and execute adaptive sustainability initiatives. This dashboard's real-time visualisation will facilitate evidence-based policymaking, enabling governments and businesses to proactively modify emission reduction plans in response to predictive analytics. Furthermore, regular updates and ongoing data integration will guarantee that decision-makers have access to the most current and pertinent data for planning and regulatory reasons.

This study fills a significant gap in Malaysia's present CO_2 emission forecasting and categorisation framework by directly addressing the requirement for sophisticated prediction tools and industry-specific knowledge. This study supports Malaysia's Net Zero 2050 promise and is in line with national sustainability policies (PETRONAS,2023) such as the Low Carbon Cities Framework (LCCF) and the National Energy Policy 2022-2040, by increasing projection accuracy and industry segmentation. Data-driven solutions for controlling and reducing industrial CO_2 emissions in Malaysia will be made possible by the combination of machine learning-based forecasts, industry categorisation models, and interactive visualisation tools, guaranteeing a more sustainable and ecologically conscious future.

CHAPTER 2
LITERATURE REVIEW

Significant study on sector classification and emission forecasting has been prompted by growing concerns about carbon dioxide (CO_2) emissions. Malaysia is a major focal area for sustainability planning and regulatory enforcement as its industrial sector, which includes manufacturing, energy generation, and transportation, continues to be one of the biggest producers to CO_2 emissions. (IEA, 2023)

Targeted emission reduction initiatives require the help of industry-specific categorisation frameworks and sophisticated prediction models as Malaysia strives for Net Zero Carbon Emissions (NZCE) by 2050. Although conventional forecasting models, like Linear Regression and Autoregressive Integrated Moving Average (ARIMA), have been used extensively for predicting CO_2 emissions, they are unable to account for complex seasonality, non-linear relationships, and changes brought about by external policy (Segar et al., 2024). Machine learning techniques, such as Decision Trees, Random Forest, and k-Nearest Neighbours (kNN) for industry categorisation and Prophet for forecasting, have drawn increasing interest in recent years to overcome these limitations, especially in industrial forecasting contexts where time-series algorithms are critical for handling seasonality, trends, and noise. (Fatima et al., 2024) Additionally, the use of deep learning methods, including Long Short-Term Memory (LSTM), to handle temporal dependencies in CO_2 emissions forecasting has grown in popularity (Luo,Y. 2023)

2.1 CO_2 Emissions and The Need for Predictive Model

Industrial activity remains a primary contributor to the buildup of greenhouse gases, with carbon dioxide (CO_2) emissions playing a dominant role in accelerating climate change, as noted by Hannah Ritchie, Pablo Rosado, and Max Roser (2023). In Malaysia, this concern is particularly pressing. According to the International Energy Agency (2023), the energy sector alone is responsible for approximately 49% of the country's CO_2 emissions, followed by the transportation sector at 22%. Additionally, emissions from diesel-based sources have also been highlighted as significant contributors to the national carbon footprint, as shown in the work by Sireesha, Harshitha, Harika, and Bhavya (2025), underscoring the importance of sector-specific forecasting strategies.

Accurate CO_2 forecasting is essential for sustainability planning and policy formulation. However, traditional models like Autoregressive Integrated Moving Average (ARIMA) and Linear Regression, though widely used, are limited in their ability to handle dynamic industrial operations, abrupt external changes, and policy-driven emission fluctuations. Segar and Sanusi (2024) emphasized that these models often fail to accommodate nonlinear patterns arising from economic transitions, regulatory shifts, and industrial growth.

In contrast, advanced machine learning approaches offer improved predictive accuracy and adaptability. Luo (2023) demonstrated that Prophet, a time-series forecasting model developed by Meta, outperforms traditional models in identifying long-term patterns and seasonal trends influenced by industrial activity and government interventions. Furthermore, Liu, Meng, Zhu, Meng, Wang, and Sun (2025) introduced a hybrid STL-Prophet-LSTM model that effectively integrates statistical decomposition with deep learning, enabling more robust forecasting in complex environmental conditions.

2.2 Machine Learning Models for CO_2 Emissions Forecasting

Machine learning techniques provide significant gains in carbon emission forecasting by optimizing prediction accuracy, identifying key emission drivers, and integrating diverse datasets such as economic indicators and industrial activity metrics. This comprehensive approach allows for a deeper understanding of the underlying emission mechanisms and supports more effective policy-making (Zhao, Bai, Wu, & Chang, 2025).

Although models such as ARIMA, LSTM, and Prophet are widely utilized, existing literature identifies ongoing challenges in balancing model complexity, interpretability, and data availability for accurate CO_2 emission forecasting. According to Tian, Xiang, Li, and Li (2025), while many models show promise, there remains a lack of consensus on a universal best-performing approach, highlighting the importance of contextual adaptability, particularly in regions like Malaysia where industrial and policy dynamics differ significantly

2.2.1 Autoregressive Integrated Moving Average (ARIMA)

The Autoregressive Integrated Moving Average (ARIMA) is a well-established time-series forecasting method that models the relationship between past observations to predict future outcomes. It is particularly effective in analyzing linear trends and seasonal patterns, making it highly suitable for short- to medium-term forecasting tasks. ARIMA integrates three fundamental components: Autoregression (AR), which uses past values to forecast future points; Differencing (I), which removes trends to achieve stationarity; and Moving Average (MA), which accounts for the residual errors by smoothing fluctuations not captured by the autoregressive component. Together, these elements form a robust framework for modeling time-dependent patterns, especially in CO_2 emissions data. Recent studies have demonstrated ARIMA's effectiveness in predicting carbon emissions in Malaysia, reinforcing its relevance in environmental forecasting research (Badyalina & Shabri, 2024)

Strengths

i. Effective for short-term forecasting with clear linear trends.

ii. Captures seasonality and trend components in time-series data, making it suitable for industries with regular emission patterns.

iii. Widely used in economic and environmental forecasting, demonstrating its reliability across multiple domains (Segar et al., 2024).

Limitations

i. Struggles with non-linear trends and external policy-driven influences, making it less effective for datasets affected by unpredictable external factors.

ii. Requires data to be stationary, meaning that trends and seasonality must be removed or transformed before applying the model, which can introduce complexity and reduce forecasting flexibility.

Despite its limitations, ARIMA remains a fundamental tool in time-series forecasting, particularly when dealing with structured data that follows consistent patterns over time. However, in complex cases where non-linearity, external policy changes, or economic fluctuations affect emissions trends, ARIMA can be complemented with advanced models such as LSTM or Prophet to enhance overall forecasting accuracy.

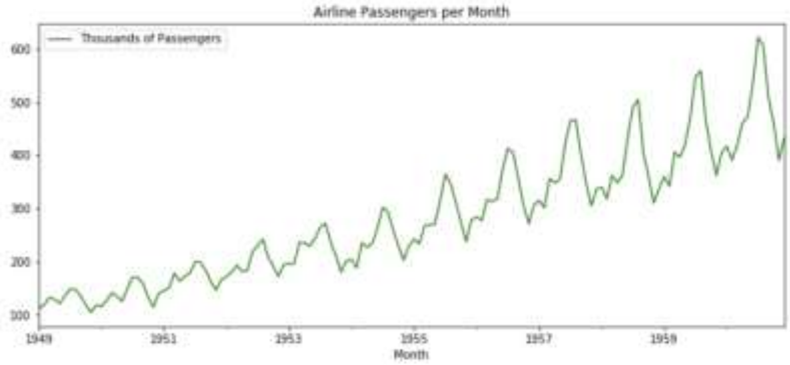


Figure 2 : ARIMA

2.2.2 Long Short Term Memory (LSTM)

Time-series forecasting benefits greatly from the Long Short-Term Memory (LSTM) network, a deep learning model created especially for long-term sequence prediction. In order to solve the vanishing gradient issue, which hindered conventional RNNs' capacity to efficiently learn long-term relationships in sequential data, it was first presented as an improvement to RNNs. When it comes to predicting CO_2 emissions, where past industrial activities, regulatory changes, and economic factors impact future emission patterns, LSTM is perfect because it excels at capturing complex dependencies in time-series data, unlike traditional forecasting methods that have trouble with complex and non-linear relationships.

Memory cells and gating mechanisms, which control the network's information storage and disposal, are the foundation of the LSTM design. The output gate regulates how much information is transmitted along to the next state, the forget gate decides what previous information should be deleted, and the input gate controls which new information is added to the memory. Because of its ability to maintain long-term dependencies, LSTM is very helpful in projecting CO_2 emissions, since patterns frequently emerge over long periods of time as a result of economic development, industry changes, and policy interventions. LSTM improves forecasting accuracy in non-linear datasets by dynamically learning patterns without the need for considerable feature engineering, in contrast to classic models like ARIMA, which need stationarity and predetermined associations.

Strengths

i. Captures long-term dependencies in emission patterns, making it superior to traditional time-series models.
ii. Works well with large datasets and complex time-series trends, making it highly suitable for CO_2 emissions forecasting.
iii. Effectively models non-linear relationships between CO_2 emissions and industrial activity, leading to more accurate and reliable predictions (Luo, Y. 2023).

Limitations

i. Computationally expensive compared to traditional models, requiring high processing power and memory.
ii. Requires a large amount of training data for optimal performance, which can be a limitation when datasets are incomplete or small.

In conclusion, LSTM is a powerful forecasting tool for predicting CO_2 emissions, particularly when long-term dependencies, non-linearity, and industrial growth trends need to be considered. However, due to its computational intensity, it can be combined with other models such as Prophet or ARIMA to balance performance and efficiency, ensuring a more comprehensive and adaptable forecasting framework. Recent studies have even proposed hybrid approaches such as combining ARIMA with advanced models like Temporal Fusion Transformers (TFT) to improve multivariate time-series forecasting in smart city environments (P.P.Linardatos, 2023)

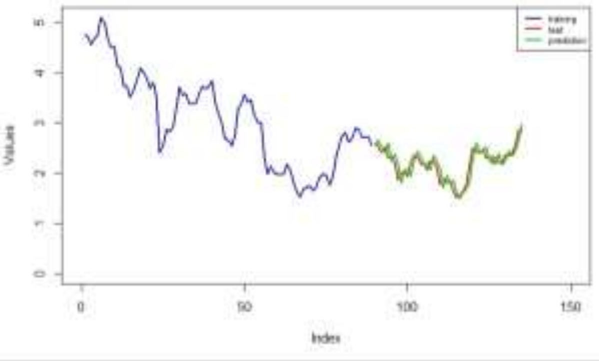


Figure 3 : LSTM

2.2.3 Prophet

Meta created the Prophet model, a time-series forecasting tool that is especially made to deal with trend fluctuations, missing data, and seasonality in time-series datasets. Prophet offers a flexible and automated method for breaking down time-series data into its essential elements: trend, seasonality, and external influences. This is in contrast to traditional forecasting models that need intricate parameter adjustment. Because of this, it is a useful tool for predicting CO_2 emissions, whose trends are impacted by industrial operations, alterations in policy, and changes in the economy. Because Prophet can account for erratic patterns and outside events that impact CO_2 emissions over time, it is frequently employed in economic and environmental forecasting (S.E.Yalçın, 2024 ; Primandari, et al., 2022 ; Dewi et al., 2022).

Prophet's fundamental method is additive regression, which divides time-series data into three main parts: trend (long-term growth or decline), seasonality (recurring patterns over predetermined periods), and holiday effects (external shocks like economic crises, policy changes, or industrial disruptions). Prophet automatically recognises seasonal patterns and modifies its forecasts in contrast to traditional models that necessitate a great deal of manual configuration. This makes it especially helpful for forecasting CO_2 emissions, where fluctuations are frequently caused by changes in the energy sector, regulatory changes, and economic cycles. Prophet can also include external regressors, which improves the precision and flexibility of emission projections by enabling researchers and decision-makers to account for industrial production rates, environmental regulations, and macroeconomic factors.

Strengths

i. Handles missing data more effectively than ARIMA and LSTM, ensuring better forecasting accuracy (Luo,Y. 2023).

ii. Captures trend changes dynamically, particularly those driven by policy interventions, economic growth, and climate-related shifts.

iii. User-friendly, with built-in automatic parameter tuning, making it accessible to non-experts in machine learning or statistical modelling.

Limitations

i. Less effective for short-term forecasting compared to ARIMA, which performs better in predicting immediate trends with high precision.
ii. Struggles with extreme fluctuations in data caused by unexpected disruptions, such as natural disasters, sudden policy changes, or global economic shocks, leading to potential forecasting errors.

In conclusion, Prophet is an effective forecasting tool for analyzing long-term CO_2 emission trends, particularly in scenarios where seasonality, external factors, and missing data must be accounted for. However, to maximize its forecasting accuracy, Prophet can be combined with ARIMA for short-term forecasting and LSTM for modelling complex, non-linear dependencies, ensuring a comprehensive and adaptive emissions forecasting system. (Z.C.Liu et al., 2025)

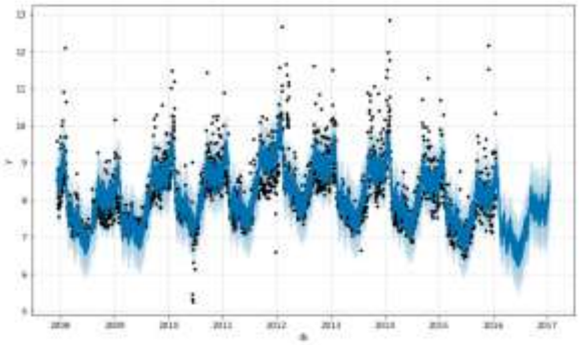


Figure 4 : Prophet

Table 1 : CO_2 Emissions (Part 1) Model Comparison

Algorithm	**ARIMA**	**LSTM**	**Prophet**
Definition	ARIMA is a traditional time-series forecasting method that models relationships between past values in a dataset to predict future values	LSTM is a deep learning model designed for long-term sequence prediction, commonly used in time-series forecasting. It was introduced to address the vanishing gradient problem in Recurrent Neural Networks (RNNs).	Prophet is a forecasting tool developed by Meta that is particularly effective for handling seasonality and missing data in time-series datasets
How it works ?	i. **Autoregression (AR):** Uses past values to predict future values. ii. **Differencing (I):** Removes trends and stationarizes the data. iii. **Moving Average (MA):** Models residual errors to improve accuracy.	i. Stores and remembers long-term dependencies in data, making it ideal for CO_2 forecasting. ii. Uses memory cells and gating mechanisms (input, forget, and output gates) to control how much information is stored or discarded	i. Models data using additive regression (trend + seasonality + holiday effects). ii. Automatically detects seasonal patterns and adjusts for external shocks like policy changes or industry growth.
Strengths	i. Effective for short-term forecasting with linear trends.	i. Captures long-term dependencies in emission patterns.	i. Handles missing data better than

	ii. Captures seasonality and trend components in time-series data. iii. Widely used in economic and environmental forecasting (Segar et al., 2024)	ii. Works well with large datasets and complex time-series trends. iii. Effective for modelling non-linear relationships between CO_2 emissions and industrial activity	ARIMA and LSTM. ii. Captures trend changes dynamically due to policy or economic factors. iii. User-friendly with built-in parameter tuning
Limitation	i. Computationally expensive compared to traditional models. ii. Requires a large amount of training data for optimal performance.	i. Computationally expensive compared to traditional models. ii. Requires a large amount of training data for optimal performance.	i. Less effective for short-term forecasts compared to ARIMA. ii. Struggles with extreme fluctuations in data due to unexpected disruptions.
Citation	Luo, Y. (2023).	Luo, Y. (2023).	Luo, Y. (2023).

2.3 Industry Classification Based on CO_2 Emissions

For focused policy interventions and resource allocation for sustainability initiatives, industries must be categorised according to their CO_2 emissions. Conventional classification methods use fixed-threshold classifications that are unable to adjust to changes in emission patterns that are particular to a certain industry. Machine learning-based industry categorisation methods provide a more adaptable and data-driven method of dividing sectors according to their CO_2 emissions in order to overcome this constraint.

2.3.1 Machine Learning Models for Industry Classification

The kNN method has been used to categorise emission sources. G. Neev (2024) demonstrated the usefulness of kNN in environmental research by implementing it for the automated tagging of emission sources. In addition, Random Forest and Decision Trees are frequently used for industry categorisation. For high-dimensional emission datasets, Random Forest effectively reduces overfitting and improves classification accuracy, according to a research by A.L. Mardani et al. (2020).

2.3.1.1 k-Nearest Neighbours (kNN)

Classification and regression problems are the main applications for the non-parametric, supervised learning k-Nearest Neighbours (kNN) technique. It is a useful tool for classifying industries according to CO_2 emissions as it excels at pattern recognition and clustering issues. As an instance-based learning method, kNN classifies new data points based on their resemblance to existing data rather than using predetermined rules, in contrast to standard classification models that create explicit prediction functions. Because of this feature, kNN is quite flexible when it comes to datasets where industries naturally clump together according to their emission patterns.

When a new data point is introduced into the dataset, the kNN algorithm finds the k closest data points. Depending on the kind of dataset, distance metrics such the Manhattan distance, Minkowski distance, or Euclidean distance are used to calculate how close these neighbours are to one another. The model efficiently classifies the new data point based on similarity by assigning it to the majority class among its neighbours after identifying its nearest neighbours. By classifying sectors according to their emission patterns, kNN may classify CO_2 emissions into High, Medium, or Low emission levels depending on how similar they are to previously classified industries. Dynamic industry categorisation is made possible by this method, especially in datasets when industries show overlapping emission characteristics.

Strengths

i. Simple and easy to implement, making it accessible for a wide range of classification problems.

ii. Highly effective for small and well-structured datasets, where clear relationships exist between data points.

iii. Performs well when class distributions are balanced, ensuring accurate industry classification without bias toward dominant categories

Limitations

i. Computationally expensive for large datasets, as it requires recalculating distances for every new prediction, leading to high processing costs.

ii. Highly sensitive to irrelevant or correlated features, requiring feature selection or dimensionality reduction techniques such as Principal Component Analysis (PCA) to improve performance and classification accuracy.

In conclusion, kNN is a valuable classification tool for industry-based CO_2 emission classification, especially in scenarios where clear industry clusters exist. However, to enhance performance and scalability, kNN can be paired with other classification models such as Random Forest and Decision Trees, ensuring a more robust and efficient industry classification framework.

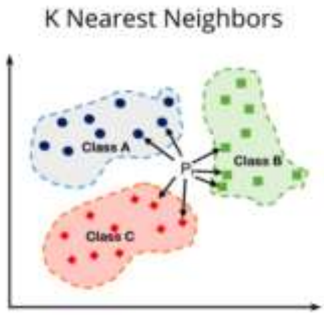


Figure 5 : kNN

2.3.1.2 Decision Tree

The Decision Tree algorithm is a supervised learning model widely used for classification and regression tasks. It is particularly effective in scenarios where data needs to be segmented based on specific decision criteria, making it highly suitable for industry classification based on CO_2 emissions. Unlike complex machine learning models that act as black-box predictors, Decision Trees provide an intuitive, rule-based approach, allowing for easy interpretation of classification results. This makes them highly valuable in environmental and policy-related decision-making, where explainability and transparency are essential.

The Decision Tree algorithm operates by constructing a hierarchical tree-like structure, where each internal node represents a feature-based decision, and each branch represents an outcome that leads to either another decision node or a final classification (leaf node). The model iteratively splits the dataset into smaller subsets using if-then rules, selecting the most important feature at each step based on statistical measures such as Gini impurity or entropy (information gain). These splitting criteria ensure that each branch optimally separates the dataset to maximize classification accuracy. In CO_2 emissions classification, Decision Trees can be used to segment industries into High, Medium, or Low emission levels by analysing factors such as energy consumption, production output, industrial processes, and regulatory compliance. This structured approach makes it easier for policymakers and industrial stakeholders to identify high-emission industries and implement targeted sustainability strategies.

Strengths

i. High interpretability, making it easy to visualize, understand, and explain classification results.

ii. Handles both numerical and categorical data, making it highly flexible for diverse datasets.

iii. Requires minimal data pre-processing, allowing for the use of raw datasets without extensive transformations.

Limitations

i. Prone to overfitting, especially when the tree becomes too deep and captures noise instead of meaningful patterns.
ii. Computationally expensive for large datasets, as multiple feature splits require significant processing power.
iii. Sensitive to irrelevant or correlated features, necessitating the use of feature selection or dimensionality reduction techniques to improve classification performance.

In conclusion, Decision Trees offer a simple yet highly effective classification approach, particularly for categorizing industries based on CO_2 emissions. However, to mitigate overfitting and scalability challenges, Decision Trees can be combined with ensemble methods like Random Forest, which aggregate multiple Decision Trees to enhance classification accuracy and improve model stability in large-scale industry classification tasks.

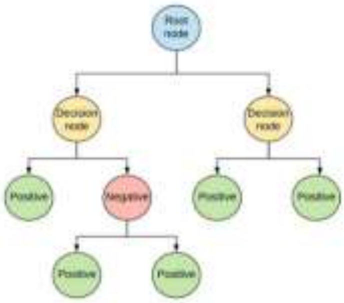


Figure 6 : Decision Tree

2.3.1.3 Random Forest

By merging many decision trees, the Random Forest algorithm, an ensemble learning technique, improves classification accuracy and produces a more reliable and stable prediction model. Random Forest creates several decision trees on various data subsets and combines their output to provide the final forecast, in contrast to conventional decision trees, which are prone to overfitting. Random Forest is the best option for industry classification based on CO_2 emissions because of its ensemble method, which greatly increases classification accuracy and generalisation. Numerous factors, such as energy consumption, manufacturing output, industrial processes, and environmental restrictions, affect the quantities of CO_2 emissions in industries. Policymakers and industry stakeholders may adopt focused sustainability initiatives by using Random Forest to achieve a more dependable categorisation system for high-emission businesses.

The Random Forest algorithm works by creating multiple decision trees, each trained on random subsets of the dataset. This process, known as bootstrap aggregating (bagging), ensures that each tree receives a unique perspective of the data, reducing the likelihood of overfitting. Once all decision trees have been trained, the final classification is determined through majority voting, where the class predicted by the most trees is chosen as the final output. This approach is particularly beneficial for CO_2 emissions classification, where the algorithm can categorize industries into High, Medium, or Low emission groups with greater accuracy and stability than a single Decision Tree model. Since each tree is trained on different subsets of features and data points, Random Forest is more resistant to noise and bias in the dataset, leading to more accurate industry classifications.

Strengths

i. Reduces overfitting by combining multiple decision trees, ensuring better generalization to unseen data.
ii. Handles large datasets with multiple features, making it well-suited for complex CO_2 emissions classification tasks.
iii. Highly accurate and stable for industry classification, consistently outperforming many traditional machine learning models

Limitations

i. Computationally expensive for very large datasets, as training multiple decision trees requires significant processing power and memory.
ii. Less interpretable than a single Decision Tree, as the final prediction results from multiple trees rather than a clear set of decision rules, making it harder for policymakers to extract direct insights.

In conclusion, Random Forest is a highly effective classification model that offers greater accuracy, reliability, and robustness compared to single Decision Trees. However, to optimize performance while maintaining interpretability, it can be combined with feature selection techniques or other ensemble learning methods, ensuring a balanced approach to industrial CO_2 emissions classification.

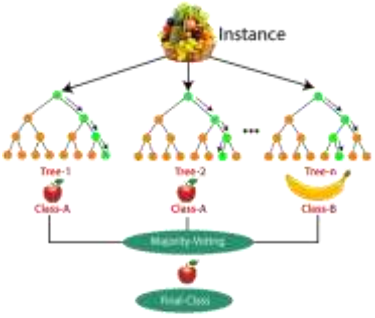


Figure 7 : Random Forest

Table 2 : Industry Classification (Part 2) Model Comparison

Algorithm	**kNN**	**Decision Tree**	**Random Forest**
Definition	kNN is a non-parametric classification algorithm that assigns a category based on the majority class of its nearest neighbours.	Decision Trees are supervised learning algorithms used for classification tasks by splitting data based on feature conditions	Random Forest is an ensemble learning method that improves classification accuracy by combining multiple decision trees
How it works ?	i. When given a new data point, the model identifies the k nearest points in the dataset. ii. The classification is determined by a majority vote	i. When given a new data point, the model identifies the k nearest points in the dataset. ii. The classification is determined by a majority vote among its neighbours	i. Creates multiple decision trees from random subsets of the dataset. ii. The final classification is based on majority voting across the trees.

	among its neighbours.		
Strengths	i. Simple and easy to implement. ii. Effective for small and well-structured datasets. iii. Works well when class distributions are balanced (R.K.Halder et al., 2024)	i. The algorithm creates a tree-like structure, where each node represents a feature, and branches represent decision outcomes. ii. The classification is made based on a sequence of if-then rules, leading to a final decision	i. Reduces overfitting compared to single Decision Trees. ii. Handles large datasets with multiple features. Highly accurate for industry classification tasks
Limitation	i. Computationally expensive for large datasets (requires recalculating distances for every prediction). ii. Sensitive to irrelevant or highly correlated features, requiring feature selection for optimal performance.	i. Computationally expensive for large datasets (requires recalculating distances for every prediction) ii. Sensitive to irrelevant or highly correlated features, requiring feature selection for optimal performance.	i. Computationally expensive for very large datasets. ii. Less interpretable than a single Decision Tree.
Citation	Ene Yalçın, S. (2024). (R.K.Halder et al., 2024)	Ene Yalçın, S. (2024).	Ene Yalçın, S. (2024).

CHAPTER 3
METHODOLOGY

3.1 Research Design

This research analyses Malaysia's CO_2 emissions using a quantitative, data-driven methodology that combines time-series forecasting and classification approaches. The next five to ten years' CO_2 emission patterns will be predicted using forecasting models (ARIMA, LSTM, and Prophet), and Malaysian businesses will be categorised according to emission levels using classification models (Decision Trees, Random Forest, and k-Nearest Neighbours (kNN)). A Power BI dashboard will also be created to provide real-time results visualisation, aiding industry stakeholders and policymakers in their decision-making.

3.2 Cross-Industry Standard Process for Data Mining (CRISP-DM)

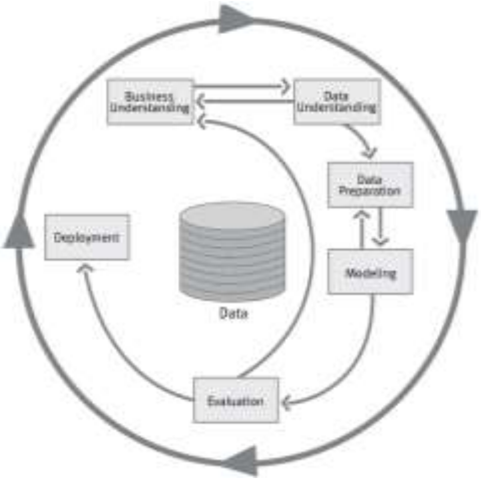


Figure 8 : CRISP - DM Framework

The Cross-Industry Standard Process for Data Mining (CRISP-DM) framework serves as the foundation for the approach used in this study. A popular, systematic, and iterative method for data mining and machine learning projects is CRISP-DM. From comprehending the business challenge to deploying the model, the framework guarantees a methodical approach that covers all crucial stages. Given the nature of the study, machine learning techniques will be used to categorise high-emission businesses and anticipate Malaysia's CO_2 emissions. Because it takes into account the dynamic nature of data-driven modelling, CRISP-DM is a suitable technique.

3.2.1 Business Understanding

Malaysia has committed to achieving Net Zero Carbon Emissions by 2050. However, there is a lack of accurate machine learning-based predictive models to forecast future CO_2 emissions and classify industries based on their contribution to emissions. Current reports primarily rely on historical trends rather than advanced predictive analytics.

This research aims to:

i. Predict Malaysia's CO_2 emissions for the next 5 to 10 years using machine learning models.

ii. Classify industries into high, medium, and low emission categories to assist policymakers in decision-making.

iii. Develop an interactive dashboard to visualize predictions and classifications for better interpretability.

Machine Learning Tasks

This study involves two core machine-learning tasks:

i. Time-Series Forecasting: Predict future CO_2 emissions using ARIMA, Prophet, and LSTM models.

ii. Industry Classification: Categorize industries into high, medium, and low emission levels using Decision Trees, Random Forest, and K-Nearest Neighbors (KNN).

3.2.2 Data Understanding

The study will utilize both global and Malaysia-specific datasets to ensure comprehensive and localized insights into CO_2 emissions.

Primary Data Sources:

Our World in Data (OWID) - Provides Malaysia CO_2 emissions data for comparative analysis and industry-level CO_2 emissions data.

Data Scope and Time Frame:

I. The study will focus on historical CO_2 emissions data from 1990 to 2023.

II. Forecasting will be conducted for a 5–10 years projection

Data Features Considered:

I. Annual CO_2 emissions (metric tons per industry)

II. Sector-specific emissions (manufacturing, transportation, energy production, etc.)

III. Economic growth indicators (GDP, industrial expansion trends)

3.2.3 Data Preparation

Before model implementation, data will be cleaned, transformed, and prepared for machine learning applications. The following pre-processing steps will be undertaken:

Handling Missing Values:

I. Missing values in time-series data will be imputed using linear interpolation.

II. For categorical data, missing values will be replaced with mode imputation.

Data Normalization & Scaling:

I. Min-Max Scaling will be applied to normalize data for classification models.

II. Log transformation may be used for highly skewed variables in forecasting models.

Feature Selection & Engineering:

I. Correlation analysis will be performed to eliminate redundant variables.

II. New features such as CO_2 emissions per GDP unit will be created to improve forecasting accuracy, based on prior findings that link emissions trends with macroeconomic indicators such as GDP and industrial growth (S.S.Li et al, 2021)

Data Splitting

I. 80% Training Set : Used for model learning

II. 20% Testing Set : Used for model validation

3.2.4 Machine Learning Models and Implementation

This study integrates time-series forecasting models for CO_2 prediction and classification models for industry segmentation based on emission levels.

3.2.4.1 Time Series Forecasting Model

Autoregressive Integrated Moving Average (ARIMA)

ARIMA Model Components & Working Mechanism

Table 3 : ARIMA Model Components

Autoregression (AR)	Uses past values to predict future values
Integrated (I)	Differencing is applied to make the data stationary
Moving Average (MA)	Models residual errors to improve forecast accuracy

$$Y_t = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \cdots + \phi_p Y_{t-p} + \theta_1 e_{t-1} + \theta_2 e_{t-2} + \cdots + \theta_q e_{t-q} + e_t$$

Y_t	Forecasted CO_2 emission at time (t)	θ_q	Coefficients for the MA component (Past Values)
c	Constant Term	e_t	White Noise (Random Errors)
ϕ_p	Coefficients for the AR component (Past Values)		

Selecting the Best ARIMA Model

Table 4 : ARIMA Working Mechanism

Parameter	Description	Method to Determine
P (AR)	Number of past values used for prediction	Partial Autocorrelation Function (PACF)

D (I)	Number of times differencing is applied to make data stationary	Augmented Dickey-Fuller (ADF) Test
Q (MA)	Number of past forecast errors considered	Autocorrelation Function (ACF)

Steps to Apply ARIMA for CO_2 Forecasting

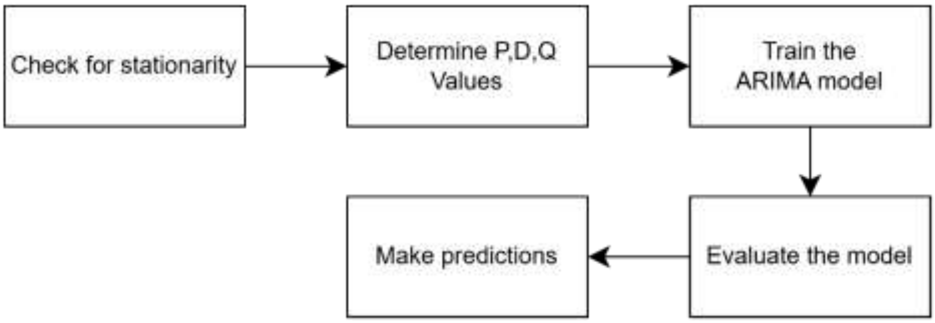


Figure 9 : Steps to apply ARIMA for CO_2 Forecasting

Long Short-Term Memory (LSTM)

LSTM Architecture & Working Mechanism

Long-term time-series forecasting can benefit from the use of LSTM, an advanced Recurrent Neural Network (RNN) architecture that solves the vanishing gradient problem. It is especially helpful for estimating CO_2 emissions since historical industrial and environmental data have a big impact on future projections. Three gating mechanisms and memory cells make up the LSTM architecture, which controls the flow of information across the network.

I. **Memory Cell & Gates in LSTM**

LSTM's strength lies in its memory cells, which store information over long sequences. Three key gates control these memory cells:

- **Forget Gate (f_t)**
  - Determines what information should be discarded from the cell state.
  - Uses a Sigmoid activation function to produce values between 0 and 1 (0 = forget, 1 = keep)

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right)$$

f_t	Forget Gate Output	h_{t-1}	Previous Hidden State
σ	Sigmoid Activation Function	x_t	Current Input
W_f	Weight Matrix	b_f	Bias

- **Input Gate (i_t)**
  - Controls how much new information is added to the cell state.
  - Uses Sigmoid and Tanh activation functions to determine input importance.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

$$\widetilde{C_t} = \tan\mathrm{h}(W_c \cdot [h_{t-1}, x_t] + b_c)$$

i_t	Input Gate Output	$W_i \ W_c$	Weight Matrices
$\widetilde{C_t}$	Candidate Cell State Update	$b_i \ b_c$	Bias Terms

- **Output Gate (o_t)**
  - Determines what part of the memory cell is sent to the next time step.
  - Uses Sigmoid activation function followed by Tanh to generate the hidden state.

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

$$h_t = o_t \times \tanh(C_t)$$

o_t	Output Gate Output	h_t	Current Hidden State

- **Cell State Update**

  The final cell state update is given by :

$$C_t = f_t \times C_{t-1} + i_t \times \widetilde{C_t}$$

This equation ensures that LSTM retains long-term dependencies while updating new relevant information.

II. Backpropagation Through Time (BPTT)

LSTM learns patterns over long time-series data through Backpropagation Through Time (BPTT), an extension of the traditional backpropagation algorithm used in deep learning.

How BPTT Works in LSTM :

Table 5 : Backpropagation Through Time (BPTT)

Forward Pass	The model processes input data sequentially, updating cell states and hidden states across time steps.
Loss Calculation	The difference between the predicted and actual CO_2 emissions is computed using a loss function e.g., Mean Squared Error (MSE)
Backward Pass	• The loss gradients are propagated backward through time steps. • LSTM updates weights using an optimizer like Adaptive Learning (Adam) to minimize errors.

III. How LSTM Handles Missing Data & Seasonality

Handling Missing Data

- Traditional models (e.g., ARIMA) require complete and stationary data, but LSTM can learn from incomplete datasets by retaining past knowledge.
- LSTM's Forget Gate selectively retains relevant past trends, allowing predictions even when some CO_2 emission records are missing.

Capturing Seasonality in CO_2 Emissions

- ARIMA requires manual seasonality adjustments, whereas LSTM automatically detects recurring seasonal patterns.
- How? The cell state memory tracks repeating emission cycles, such as annual CO_2 peaks due to industrial activities and policy changes.
    - Improvement: Adding external regressors (e.g., GDP, policy changes, temperature) helps LSTM predict seasonal variations more accurately.

Prophet (Developed by Meta):

Prophet Model Components & Working Mechanism

Prophet models time-series data using an **additive regression approach:**

$$y(t) = g(t) + s(t) + h(t) + \epsilon_t$$

$g(t)$	Models the long-term trend using a piecewise linear or logistic growth function	$h(t)$	Accounts for holidays, policy changes or economic shocks
$s(t)$	Captures seasonality (yearly cycles)	ϵ_t	Represents random noise or errors

I. How Prophet Works

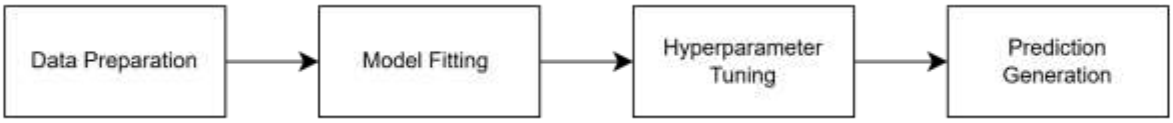


Figure 10 : How Prophet Works

II. Key Prophet Parameters for CO_2 Forecasting

Table 6 : Prophet Parameters

Parameter	**Description**	**Recommended Value**
Growth Function	Defines how CO_2 emissions change over time	Linear (default) or Logistic (if emissions saturation is expected)
Changepoints	Points where emissions trends shift due to policies	Auto-detected (can be manually set for policy changes)
Seasonality Mode	Controls how seasonal trends behave	Multiplicative (better for CO_2 trends)
Holidays & Events	Adds external factors like economic policy changes	Yes (e.g., Malaysia’s sustainability policies, industrial regulations)
Uncertainty Interval	Controls forecast confidence range	80%–95% (higher for emission forecasting)

3.2.4.2 Industry Classification Model

- **Decision Trees:**

  i. Classifies industries into High, Medium, and Low CO_2 emission levels.
  ii. Provides interpretable results but is prone to overfitting.

- **Random Forest:**

  i. An ensemble model that improves classification accuracy by reducing overfitting.
  ii. Used for dynamic industry classification based on CO_2 trends.

- **k-Nearest Neighbors (kNN):**

  i. Groups industries into similar emission profiles for segmentation.
  ii. Effective for clustering but sensitive to noisy data

3.2.5 Model Evaluation and Performance Metrics

To assess the accuracy and reliability of the models, the following performance metrics will be used:

For Forecasting Models (ARIMA, LSTM, Prophet):

i. Mean Absolute Error (MAE): Measures the average error magnitude.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \widehat{y_i}|$$

ii. Root Mean Squared Error (RMSE): Evaluates the model's predictive accuracy.

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

iii. Mean Absolute Percentage Error (MAPE): Assesses forecast error relative to actual values.

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \widehat{y_i}}{y_i}\right| \times 100$$

For Classification Models (Decision Trees, Random Forest, kNN):

i. Accuracy: Measures overall classification correctness.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

ii. Precision & Recall: Evaluates model effectiveness in classifying high-emission industries.

$$Precision = \frac{TP}{TP + FP}, \quad Recall = \frac{TP}{TP + FN}$$

iii. F1-Score: Balances precision and recall for improved industry segmentation.

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

3.3 Tools and Libraries

Table 7 : Tools and Libraries

Category	**Tools Used**
Data Collection & Pre-processing	Python (Pandas, Numpy), Spyder IDE, Excel
Machine Learning Models	ARIMA (statsmodels), LSTM (TensorFlow / Keras), Prophet (Meta), Decision Tree, Random Forest, kNN (Scikit-Learn)
Visualization & Dashboard	Power BI
Model Evaluation	Scikit-learn, MAE, RMSE, MAPE, Accuracy, Precision-Recall, F1 Score
Documentation & Reporting	Word, Mendeley
Version Control & Project Management	Github, One Drive

3.3.1 Spyder IDE

Figure 11 : Spyder IDE

In this study, Spyder IDE is a crucial tool for creating, evaluating, and applying machine learning models. It offers an interactive computing environment that makes data visualisation, model assessment, and real-time code execution possible. For tasks involving industry categorisation and CO_2 emission predictions, Spyder IDE is especially well-suited due to its versatility and broad support for Python-based modules.

3.3.2 Excel

Figure 12 : Excel

As the main tool for organising and pre-processing CO_2 emissions data before it is entered into machine learning models for forecasting and categorisation, Microsoft Excel is essential to this study. Excel offers an effective and systematic method for storing, organising, cleaning, and analysing data before integrating it into machine learning processes within Spyder IDE, which is why the dataset used in this research is in Excel format (.xlsx/.csv).

3.3.3 Microsoft Power BI

Figure 13 : Microsoft Power BI

FIGURE 13 above shows the logo of Microsoft Power Business Intelligence (BI). Power BI dashboard will be developed to visualize the CO_2 emission forecasts and industry classification results

Key Features of the Dashboard: - best model from forecasting & classification

i. Real-time visualization of emission trends over the next 5 -10 years.
ii. Interactive classification filters for high, medium, and low-emission industries.
iii. Dynamic industry ranking, highlighting top CO_2-emitting sectors.

Data Integration for Power BI:

i. Data from machine learning models will be continuously updated to reflect the latest forecasting results.
ii. API connections to Our World in Data (OWID) will enable automated data refreshes.

This study employs ARIMA, LSTM and Prophet for CO_2 forecasting and Decision Trees, Random Forest, and kNN for industry classification. The models will be evaluated using statistical and ML-based performance metrics, and the results will be integrated into an interactive Power BI dashboard for real-time decision-making. By leveraging historical data from OWID, this study aims to enhance Malaysia’s CO_2 emissions forecasting accuracy, support industry classification, and enable data-driven sustainability policies.

3.4 Gantt Chart

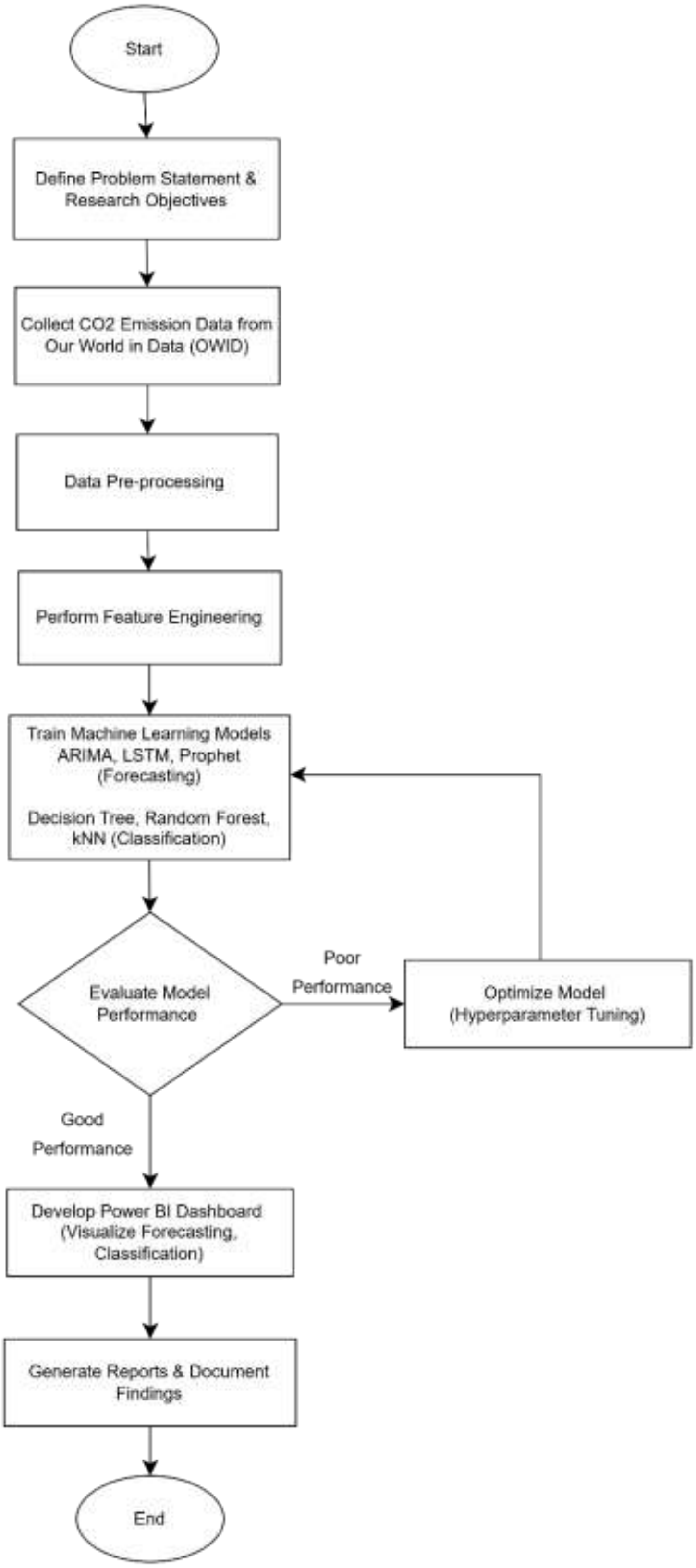

Start
Define Problem Statement & Research Objectives
Collect CO2 Emission Data from Our World in Data (OWID)
Data Pre-processing
Perform Feature Engineering
Train Machine Learning Models ARIMA, LSTM, Prophet (Forecasting)
Decision Tree, Random Forest, kNN (Classification)
Evaluate Model Performance
Poor Performance
Optimize Model (Hyperparameter Tuning)
Good Performance
Develop Power BI Dashboard (Visualize Forecasting, Classification)
Generate Reports & Document Findings
End

Figure 14 : Gantt Chart

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Implementation Stage for CO_2 Emissions Prediction

4.1.1 Data Collection and Preparation CO_2 Emissions Prediction (Part 1)

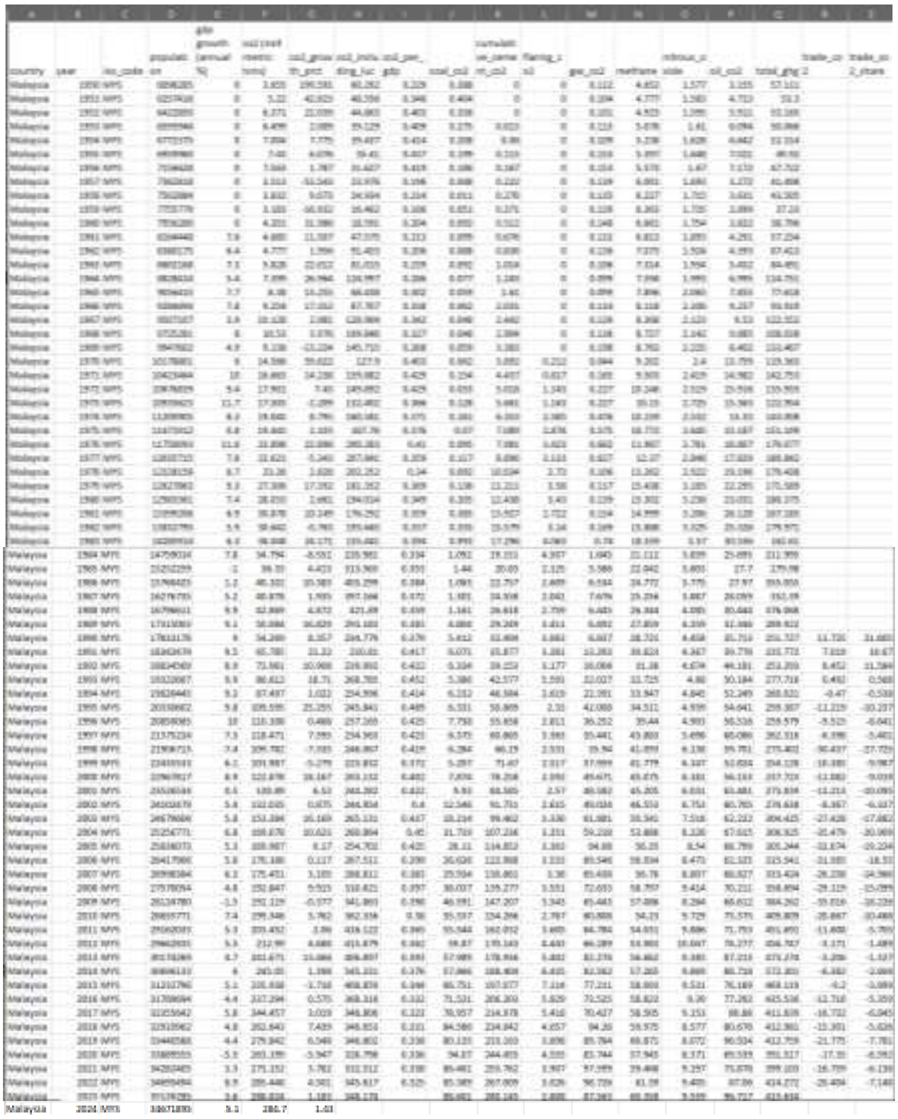


Figure 15 : Historical Dataset for CO_2 Emission

In this phase, the objective is to predict Malaysia's future carbon dioxide (CO_2) emissions using time-series forecasting models, namely ARIMA**,** Long Short-Term Memory (LSTM)**,** and Prophet. Accurate forecasting depends heavily on the quality and relevance of historical emission data. Therefore, a comprehensive dataset was sourced from Our World in Data (OWID) to support this task.

The dataset was retrieved from the publicly accessible OWID platform, specifically from the CO_2 and Greenhouse Gas Emissions section. It contains yearly CO_2 emissions for Malaysia from 1950 to 2024, along with several auxiliary indicators that support multivariate time-series modelling.

Data was downloaded in .csv format and filtered to retain only Malaysia-specific records. The availability of auxiliary variables such as GDP and fossil fuel breakdowns makes the dataset particularly suitable for capturing external drivers that influence emission trends.

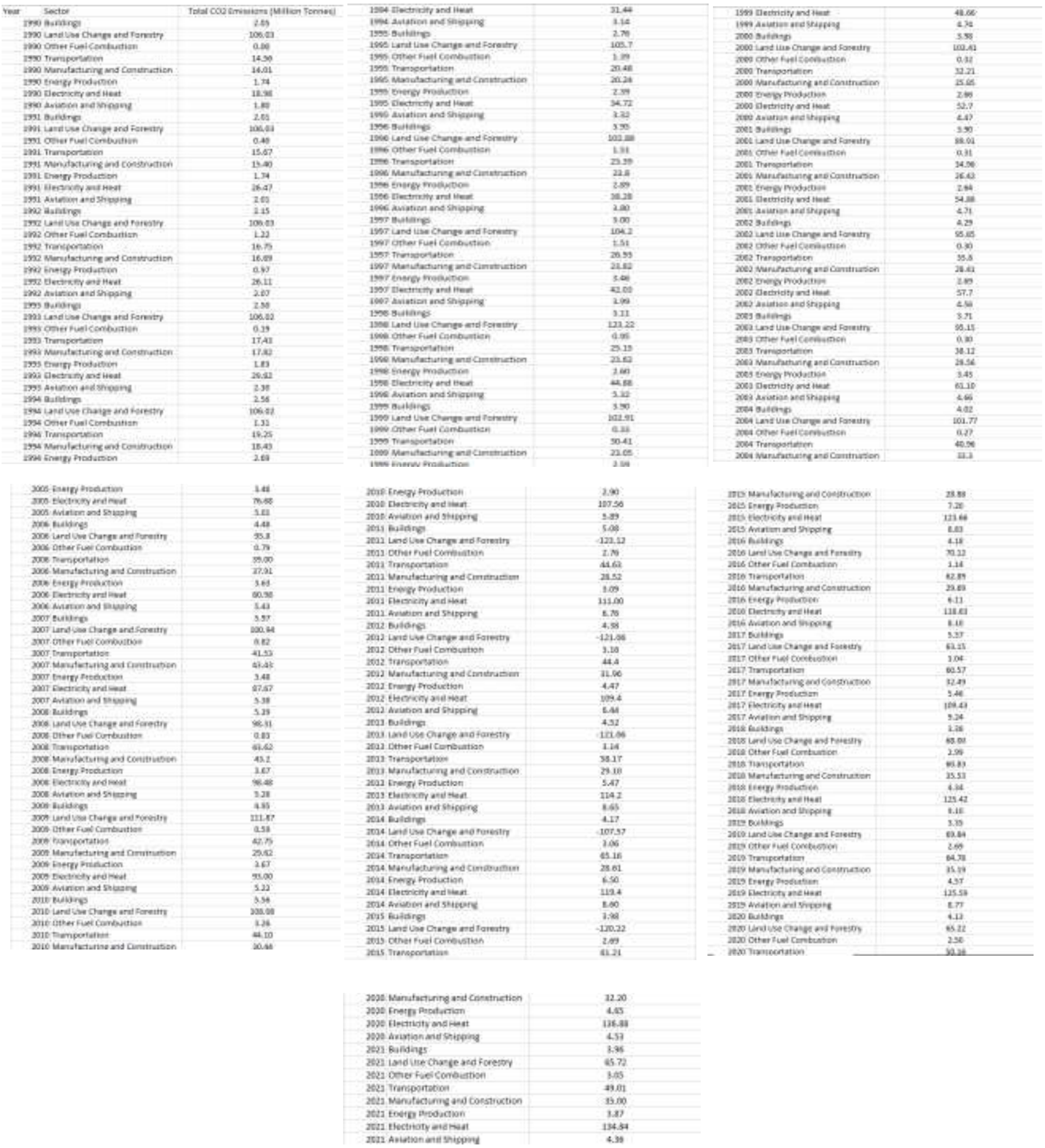
Table 8 : Explanation of Dataset (Part 1)

Key Variables	**Description**
country	Geographic location
year	Year of observation
iso_code	Three letter country code
population	Population by country
gdp growth (annual %)	This data is adjusted for inflation and differences in the cost of living measured in percentage (%)
co2 (mill metric tonnes)	Annual total emissions of carbon dioxide (CO_2), excluding land-use change, measured in million tonnes.
co2_growth_prct	Annual CO2 emissions growth (%) - Annual percentage growth in total emissions of carbon dioxide (CO2), excluding land-use change.
co2_including_luc	Annual CO2 emissions including land-use change - Annual total emissions of carbon dioxide (CO2), including land-use change, measured in million tonnes.
co2_per_gdp	Annual CO2 emissions per GDP (kg per international-$) - Annual total emissions of carbon dioxide (CO2), excluding

	land-use change, measured in kilograms per dollar of GDP (2011 international-$).
coal_co$_2$ (mill tonnes)	Annual emissions of carbon dioxide (CO$_2$) from coal, measured in million tonnes.
cumulative_cement_CO2	Cumulative CO2 emissions from cement - Cumulative emissions of carbon dioxide (CO2) from cement since the first year of available data, measured in million tonnes.
flaring_co2 (mill tonnes)	Annual emissions of carbon dioxide (CO$_2$) from flaring, measured in million tonnes.
gas_co2 (mill tonnes)	Annual emissions of carbon dioxide (CO$_2$) from gas, measured in million tonnes.
methane	Annual methane emissions including land use - Measured in tonnes of carbon dioxide-equivalents over a 100-year timescale.
nitrous_oxide	Annual nitrous oxide emissions including land use - Measured in tonnes of carbon dioxide-equivalents over a 100-year timescale.
oil_CO2 (mill tonnes)	Annual emissions of carbon dioxide (CO$_2$) from oil, measured in million tonnes.
total_ghg (mill tonnes)	Greenhouse gas emissions in million tonnes
trade_co2 (mill tonnes)	Annual CO2 emissions embedded in trade - Annual net carbon dioxide (CO2)
trade_CO2_share	Share of annual CO2 emissions embedded in trade - Annual net carbon dioxide (CO2) emissions embedded in trade, measured as a percentage of emissions of CO2.

This dataset was selected because of its high credibility, continuous yearly coverage, and alignment with Malaysia’s Net Zero Carbon Emissions (NZCE) goals. It provides the necessary variables for building both univariate (ARIMA, Prophet) and multivariate (LSTM) models, ensuring the models can capture trends, seasonality, policy impacts, and non-linear dynamics in emission behaviour.

4.1.2 Data Collection and Preparation for Industry Classification (Part 2)



Year	Sector	Total CO2 Emissions (Million Tonnes)
1990	Buildings	2.05
1990	Land Use Change and Forestry	106.03
1990	Other Fuel Combustion	0.00
1990	Transportation	14.50
1990	Manufacturing and Construction	14.01
1990	Energy Production	1.74
1990	Electricity and Heat	18.98
1990	Aviation and Shipping	1.80
1991	Buildings	2.01
1991	Land Use Change and Forestry	106.03
1991	Other Fuel Combustion	0.48
1991	Transportation	15.67
1991	Manufacturing and Construction	15.40
1991	Energy Production	1.74
1991	Electricity and Heat	26.47
1991	Aviation and Shipping	2.01
1992	Buildings	2.15
1992	Land Use Change and Forestry	106.03
1992	Other Fuel Combustion	1.22
1992	Transportation	16.75
1992	Manufacturing and Construction	16.89
1992	Energy Production	0.97
1992	Electricity and Heat	26.11
1992	Aviation and Shipping	2.07
1993	Buildings	2.38
1993	Land Use Change and Forestry	106.02
1993	Other Fuel Combustion	0.19
1993	Transportation	17.43
1993	Manufacturing and Construction	17.82
1993	Energy Production	1.83
1993	Electricity and Heat	29.82
1993	Aviation and Shipping	2.38
1994	Buildings	2.56
1994	Land Use Change and Forestry	106.02
1994	Other Fuel Combustion	1.31
1994	Transportation	19.25
1994	Manufacturing and Construction	18.43
1994	Energy Production	2.69
1994	Electricity and Heat	31.44
1994	Aviation and Shipping	3.14
1995	Buildings	2.70
1995	Land Use Change and Forestry	105.7
1995	Other Fuel Combustion	1.39
1995	Transportation	20.48
1995	Manufacturing and Construction	20.24
1995	Energy Production	2.39
1995	Electricity and Heat	34.72
1995	Aviation and Shipping	3.32
1996	Buildings	3.95
1996	Land Use Change and Forestry	102.88
1996	Other Fuel Combustion	1.31
1996	Transportation	23.39
1996	Manufacturing and Construction	22.8
1996	Energy Production	2.89
1996	Electricity and Heat	38.28
1996	Aviation and Shipping	3.80
1997	Buildings	3.00
1997	Land Use Change and Forestry	104.2
1997	Other Fuel Combustion	1.51
1997	Transportation	26.93
1997	Manufacturing and Construction	21.82
1997	Energy Production	3.46
1997	Electricity and Heat	42.03
1997	Aviation and Shipping	3.99
1998	Buildings	3.11
1998	Land Use Change and Forestry	123.22
1998	Other Fuel Combustion	0.95
1998	Transportation	25.15
1998	Manufacturing and Construction	23.62
1998	Energy Production	2.60
1998	Electricity and Heat	44.88
1998	Aviation and Shipping	5.32
1999	Buildings	3.90
1999	Land Use Change and Forestry	102.91
1999	Other Fuel Combustion	0.33
1999	Transportation	30.41
1999	Manufacturing and Construction	23.05
1999	Energy Production	2.59
1999	Electricity and Heat	48.66
1999	Aviation and Shipping	4.74
2000	Buildings	3.58
2000	Land Use Change and Forestry	102.41
2000	Other Fuel Combustion	0.32
2000	Transportation	32.21
2000	Manufacturing and Construction	25.05
2000	Energy Production	2.86
2000	Electricity and Heat	52.7
2000	Aviation and Shipping	4.47
2001	Buildings	3.90
2001	Land Use Change and Forestry	88.01
2001	Other Fuel Combustion	0.31
2001	Transportation	34.56
2001	Manufacturing and Construction	26.43
2001	Energy Production	2.84
2001	Electricity and Heat	54.88
2001	Aviation and Shipping	4.71
2002	Buildings	4.29
2002	Land Use Change and Forestry	95.85
2002	Other Fuel Combustion	0.30
2002	Transportation	35.8
2002	Manufacturing and Construction	28.41
2002	Energy Production	2.89
2002	Electricity and Heat	57.7
2002	Aviation and Shipping	4.58
2003	Buildings	3.71
2003	Land Use Change and Forestry	95.15
2003	Other Fuel Combustion	0.30
2003	Transportation	38.12
2003	Manufacturing and Construction	28.56
2003	Energy Production	3.45
2003	Electricity and Heat	61.10
2003	Aviation and Shipping	4.66
2004	Buildings	4.02
2004	Land Use Change and Forestry	101.77
2004	Other Fuel Combustion	0.27
2004	Transportation	40.56
2004	Manufacturing and Construction	33.3
2005	Energy Production	3.48
2005	Electricity and Heat	76.88
2005	Aviation and Shipping	5.81
2006	Buildings	4.48
2006	Land Use Change and Forestry	95.8
2006	Other Fuel Combustion	0.79
2006	Transportation	39.00
2006	Manufacturing and Construction	37.91
2006	Energy Production	3.63
2006	Electricity and Heat	80.58
2006	Aviation and Shipping	5.43
2007	Buildings	5.57
2007	Land Use Change and Forestry	100.94
2007	Other Fuel Combustion	0.82
2007	Transportation	41.53
2007	Manufacturing and Construction	43.41
2007	Energy Production	3.48
2007	Electricity and Heat	87.67
2007	Aviation and Shipping	5.38
2008	Buildings	5.29
2008	Land Use Change and Forestry	98.31
2008	Other Fuel Combustion	0.83
2008	Transportation	41.62
2008	Manufacturing and Construction	43.2
2008	Energy Production	3.67
2008	Electricity and Heat	98.48
2008	Aviation and Shipping	5.28
2009	Buildings	4.95
2009	Land Use Change and Forestry	111.87
2009	Other Fuel Combustion	0.58
2009	Transportation	42.75
2009	Manufacturing and Construction	29.62
2009	Energy Production	3.67
2009	Electricity and Heat	93.00
2009	Aviation and Shipping	5.22
2010	Buildings	5.56
2010	Land Use Change and Forestry	108.08
2010	Other Fuel Combustion	3.26
2010	Transportation	44.10
2010	Manufacturing and Construction	30.46
2010	Energy Production	2.90
2010	Electricity and Heat	107.56
2010	Aviation and Shipping	5.89
2011	Buildings	5.08
2011	Land Use Change and Forestry	-123.12
2011	Other Fuel Combustion	2.76
2011	Transportation	44.63
2011	Manufacturing and Construction	28.52
2011	Energy Production	3.09
2011	Electricity and Heat	111.00
2011	Aviation and Shipping	6.76
2012	Buildings	4.98
2012	Land Use Change and Forestry	-121.86
2012	Other Fuel Combustion	3.10
2012	Transportation	44.4
2012	Manufacturing and Construction	31.96
2012	Energy Production	4.47
2012	Electricity and Heat	109.4
2012	Aviation and Shipping	6.44
2013	Buildings	4.52
2013	Land Use Change and Forestry	-121.06
2013	Other Fuel Combustion	3.14
2013	Transportation	38.17
2013	Manufacturing and Construction	29.10
2013	Energy Production	5.47
2013	Electricity and Heat	114.2
2013	Aviation and Shipping	8.65
2014	Buildings	4.17
2014	Land Use Change and Forestry	-107.57
2014	Other Fuel Combustion	3.06
2014	Transportation	65.16
2014	Manufacturing and Construction	28.61
2014	Energy Production	6.50
2014	Electricity and Heat	119.4
2014	Aviation and Shipping	8.60
2015	Buildings	3.98
2015	Land Use Change and Forestry	-120.22
2015	Other Fuel Combustion	2.89
2015	Transportation	61.21
2015	Manufacturing and Construction	28.88
2015	Energy Production	7.20
2015	Electricity and Heat	123.66
2015	Aviation and Shipping	8.83
2016	Buildings	4.18
2016	Land Use Change and Forestry	70.12
2016	Other Fuel Combustion	1.14
2016	Transportation	62.89
2016	Manufacturing and Construction	29.89
2016	Energy Production	6.11
2016	Electricity and Heat	118.63
2016	Aviation and Shipping	8.18
2017	Buildings	5.37
2017	Land Use Change and Forestry	63.15
2017	Other Fuel Combustion	3.04
2017	Transportation	60.57
2017	Manufacturing and Construction	32.49
2017	Energy Production	5.46
2017	Electricity and Heat	109.43
2017	Aviation and Shipping	9.24
2018	Buildings	3.26
2018	Land Use Change and Forestry	68.00
2018	Other Fuel Combustion	2.99
2018	Transportation	60.83
2018	Manufacturing and Construction	35.53
2018	Energy Production	4.34
2018	Electricity and Heat	125.42
2018	Aviation and Shipping	8.16
2019	Buildings	3.35
2019	Land Use Change and Forestry	69.84
2019	Other Fuel Combustion	2.69
2019	Transportation	64.78
2019	Manufacturing and Construction	35.19
2019	Energy Production	4.57
2019	Electricity and Heat	125.59
2019	Aviation and Shipping	8.77
2020	Buildings	4.13
2020	Land Use Change and Forestry	65.22
2020	Other Fuel Combustion	2.50
2020	Transportation	50.16
2020	Manufacturing and Construction	32.20
2020	Energy Production	4.65
2020	Electricity and Heat	136.88
2020	Aviation and Shipping	4.53
2021	Buildings	3.96
2021	Land Use Change and Forestry	65.72
2021	Other Fuel Combustion	3.05
2021	Transportation	49.01
2021	Manufacturing and Construction	35.00
2021	Energy Production	3.87
2021	Electricity and Heat	134.84
2021	Aviation and Shipping	4.39

Figure 16 : Historical Dataset for Industry Classification

This section details the data collection and preparation process for classifying Malaysian industries based on their carbon dioxide (CO_2) emission levels. The aim is to develop a machine learning model that segments industries into High, Medium, and Low emission categories using classification algorithms such as Decision Tree, Random Forest, and k-Nearest Neighbours (kNN).

This dataset includes disaggregated annual CO_2 emissions by industrial sectors such as:

- Electricity and Heat Production
- Transportation
- Manufacturing and Construction
- Buildings
- Fugitive Emissions
- Aviation and Shipping
- Land Use Change
- Other Fuel Combustion

The time span of the data range from 1990 to 2023, offering sufficient historical context for industry emission pattern analysis. The classification task contributes to policy-making by enabling dynamic industry targeting for emission control measures, aligning with Malaysia's commitment to Net Zero Carbon Emissions (NZCE) 2050 and SDG 13 (Climate Action)

Unlike aggregate national datasets, this source provides emissions data categorized by specific sectors such as electricity, transportation, construction, buildings, and land use change. This granularity is crucial for industry classification, as it allows the machine learning models to learn emission patterns unique to each sector.

Table 9 : Explanation of Dataset (Part 2)

Key Variables	**Description**
Year	Calendar year of the observation (e.g., 1990–2023)
Sector	Industry name (e.g., Transport, Energy, Manufacturing, etc.)
CO_2_Emissions	Annual CO_2 emissions (in million tonnes)

4.2 Data Pre-processing and Feature Engineering

4.2.1 CO_2 Emissions Prediction (Part 1)

Effective pre-processing and feature engineering are essential to ensure that machine learning models receive clean, relevant, and appropriately structured data. This section details the pre-processing pipeline implemented for forecasting Malaysia's CO_2 emissions using ARIMA, LSTM, and Prophet.

I. **Data Cleaning**

The raw dataset was obtained from Our World in Data (OWID) and contains Malaysia's historical CO_2 emissions and related features from 1990 to 2023. To ensure reliability:

a. **Filtering Country Data**: Only records specific to *Malaysia* were extracted. his project focuses exclusively on Malaysia's national-level emissions; including other countries would introduce noise and irrelevance.
b. **Handling Missing Values**: Linear interpolation was applied to fill missing values in continuous variables. Time-series models like ARIMA and LSTM require complete sequences without missing timestamps; interpolation preserves continuity while avoiding deletion of valuable time-based trends.
c. **Column Standardization**: All column names were cleaned (e.g., trimmed whitespaces, converted to lowercase) to avoid parsing errors during programming. Ensures consistent variable referencing throughout the machine learning pipeline.

II. **Feature Selection**

Each forecasting model requires different types of inputs, and appropriate features were selected accordingly:

a. **Univariate Models (ARIMA & Prophet)**: Only *CO_2 emissions (million tonnes)* and *year* were retained. These models are designed to work on a single time-dependent variable. Using additional features would violate their assumptions and model structure.

b. **Multivariate Model (LSTM)**: Features selected: *CO_2 emissions, GDP growth (%), Population, Coal CO_2,* and *Gas CO_2*. LSTM networks excel at capturing long-term patterns when enriched with correlated explanatory variables. GDP and energy-related emissions were included to allow the model to understand external economic and environmental influences on CO_2 emissions.

III. Data Transformation

Several transformations were applied to meet model-specific input requirements and improve learning performance:

a. **Datetime Conversion**: The *year* variable was converted to datetime format. Prophet requires a column named *ds* in datetime format, and proper indexing is also essential for time-series visualization and analysis.
b. **Stationarity Check (ARIMA)**: Conducted the Augmented Dickey-Fuller (ADF) test to check for stationarity. Applied first-order differencing when the series was non-stationary. ARIMA models assume that the input time series is stationary (i.e., mean and variance do not change over time). Differencing ensures that the model can accurately capture consistent trends.
c. **Normalization (LSTM only)**: Applied Min-Max Scaling to transform features into the range [0, 1]. Neural networks, especially LSTM, are sensitive to feature magnitudes. Normalization ensures faster convergence and reduces the risk of exploding gradients during training.

IV. Feature Engineering

To enhance the forecasting model's predictive capability, the following derived features were created:

a. **CO_2 per Capita** = CO_2 emissions / (population / 1,000,000)

Reflects the individual environmental burden and helps normalize emissions relative to population size, which is particularly useful for contextualizing emission growth in relation to demographic trends.

b. **Fossil Fuel Dependency Ratio** = (coal_CO_2 + gas_CO_2) / total CO_2 emissions

Indicates Malaysia's reliance on fossil fuels. This ratio serves as a critical signal of emission source behaviour, which influences long-term patterns in the LSTM model.

V. Dataset Splitting

The dataset was split chronologically as follows:

- **Training Set**: 1990–2014 (80%)
- **Testing Set**: 2015–2023 (20%)

Time-series forecasting must preserve temporal order to avoid data leakage. This chronological split ensures that models learn from past data and are evaluated on future data, mimicking real-world prediction scenarios.

VI. Model - Specific Data Structuring

Table 10 : Model Specific Data Structure

Model	**Required Input Format**	**Applied Pre-processing Steps**
ARIMA	Stationary univariate time series	ADF test, differencing to remove trends
LSTM	Multivariate 3D tensor (samples × timesteps × features)	Min-Max scaling, 7-step sequence generation
Prophet	Dataframe with columns *ds* (date) and *y* (target)	Renaming columns, date parsing, no normalization needed

Custom formatting ensures that each model can process the data according to its architecture. For example, LSTM expects data in sequential format for time window learning, while Prophet expects time-stamped targets with additive components.

4.2.2 Industry Classification (Part 2)

This section presents the data pre-processing and feature engineering steps taken to prepare the dataset for the classification of Malaysian industries into High, Medium, and Low CO_2 emission levels. The models involved are Decision Tree, Random Forest, and k-Nearest Neighbours (kNN).

I. Data Cleaning

The dataset used for this task was sourced from Our World in Data (OWID) and contains Malaysia's annual CO_2 emissions by industrial sector. It covers emission values (in million tonnes) across different sectors in Malaysia from 1970 to 2021. Feature engineering and classification labelling were applied:

The target variable (*Emission_Level*) was assigned based on quantile thresholds:

- Top 33% → High
- Middle 33% → Medium
- Bottom 33% → Low

a. **Whitespace Removal and Column Standardization**: All column names were stripped of whitespaces and reformatted to ensure compatibility during coding. Prevents errors when referencing column names and improves code reliability.

b. **Handling Missing Values**: Missing values in key columns such *as Year, Sector,* and *Total CO_2 Emissions* were removed. Since these fields are essential for labelling and grouping, incomplete records could bias classification results.

II. Feature Selection

The classification task required the selection of features that would enable the model to distinguish emission levels among sectors over time.

a. **Encoded Features**: *Sector_Encoded* **-** Sector names were encoded into numerical values using *LabelEncoder.* Machine learning models like Random Forest and Decision Tree require numeric inputs for processing categorical variables like sector names.

b. **Engineered Features**: *Emission_Growth* **-** Calculated as the percentage change in CO_2 emissions year-over-year within each sector. Helps the model detect trends and fluctuations in emission behaviour that could indicate high-impact years or industrial shifts.\

III. Target Label Creation

A new categorical target variable *Emission_Level* was created to classify the emission intensity of each sector-year pair:

```
median_emission = df['Total CO2 Emissions (Million Tonnes)'].median()

df['Emission_Level'] = np.where(df['Total CO2 Emissions (Million Tonnes)'] > median_emission, 'High', 'Low')
```

This binary segmentation simplifies the classification task, enabling easier interpretation of model outputs while highlighting sectors with above-average emission levels.

For Decision Tree models, a three-level classification (Low, Medium, High) was also created using 33rd and 66th percentile cut-offs.

IV. Data Splitting

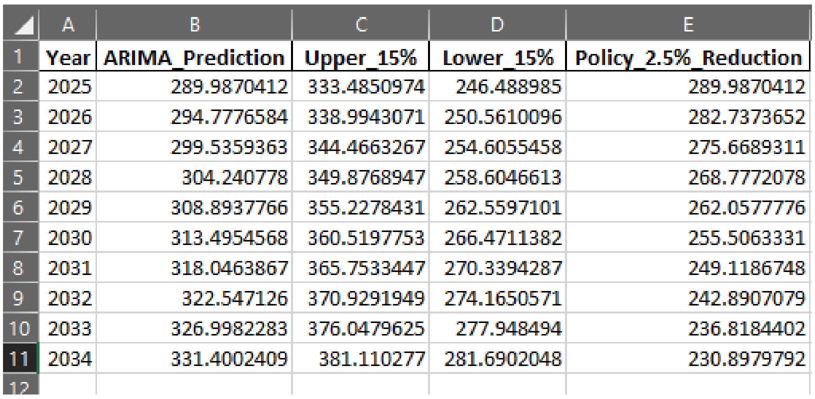
The dataset was divided into training and testing sets using an 80/20 split, maintaining chronological integrity to ensure that models are trained on historical data and evaluated on future, unseen data mimicking real-world application.

- Training Set: Older years (e.g., 1990–2013)
- Testing Set: More recent years (e.g., 2014–2023)

4.3 Model Implementation and Evaluation for CO₂ Emissions Prediction (Part 1)

4.3.1 ARIMA (1,1,1) Model Implementation

Time series forecasting was performed using the ARIMA (Autoregressive Integrated Moving Average) method to predict Malaysia's future CO_2 emissions from 2025 to 2034. Two ARIMA configurations were tested - ARIMA(1,1,1) and ARIMA(2,1,1). The following sections detail their individual implementation, diagnostics, and forecast interpretations.



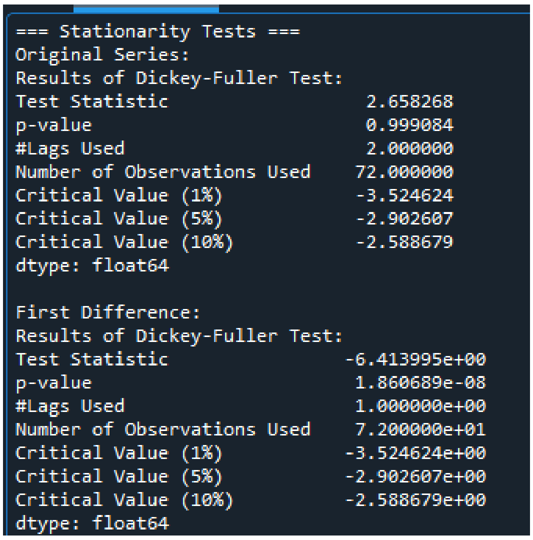
Year	ARIMA_Prediction	Upper_15%	Lower_15%	Policy_2.5%_Reduction
2025	289.9870412	333.4850974	246.488985	289.9870412
2026	294.7776584	338.9943071	250.5610096	282.7373652
2027	299.5359363	344.4663267	254.6055458	275.6689311
2028	304.240778	349.8768947	258.6046613	268.7772078
2029	308.8937766	355.2278431	262.5597101	262.0577776
2030	313.4954568	360.5197753	266.4711382	255.5063331
2031	318.0463867	365.7533447	270.3394287	249.1186748
2032	322.547126	370.9291949	274.1650571	242.8907079
2033	326.9982283	376.0479625	277.948494	236.8184402
2034	331.4002409	381.110277	281.6902048	230.8979792

Figure 17 : Forecasting Dataset for ARIMA model

The ARIMA(1,1,1) model was developed to model the univariate time series of Malaysia's CO_2 emissions between 1990 and 2024. This model structure includes one autoregressive term (AR), first-order differencing to ensure stationarity (I), and one moving average term (MA).

I. Stationarity Analysis

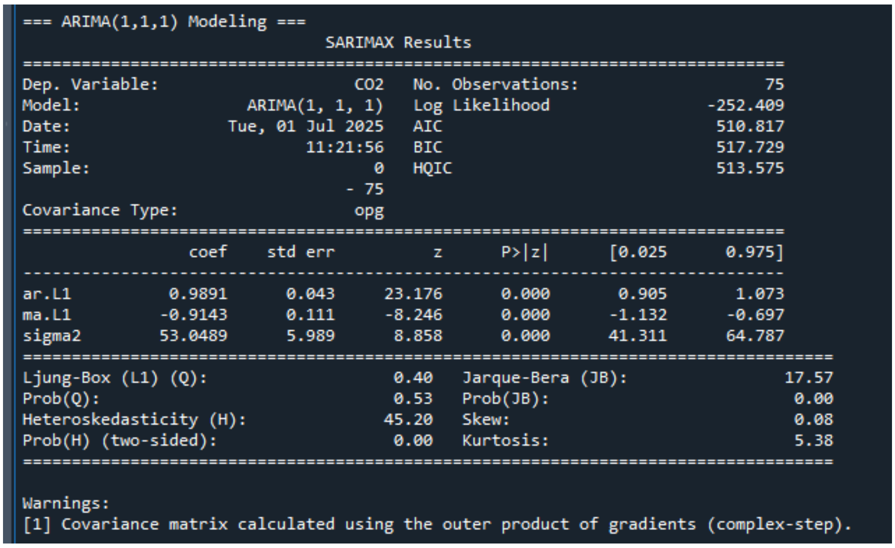
An Augmented Dickey-Fuller (ADF) test was conducted to evaluate whether the time series was stationary. As shown in Figure 17 , the original series had a p-value of **0.9998**, indicating non-stationarity. After applying first-order differencing, the p-value dropped to **1.86e-08**, confirming that the differenced series was stationary and suitable for ARIMA modelling.



```
=== Stationarity Tests ===
Original Series:
Results of Dickey-Fuller Test:
Test Statistic                  2.658268
p-value                         0.999084
#Lags Used                      2.000000
Number of Observations Used    72.000000
Critical Value (1%)            -3.524624
Critical Value (5%)            -2.902607
Critical Value (10%)           -2.588679
dtype: float64

First Difference:
Results of Dickey-Fuller Test:
Test Statistic                -6.413995e+00
p-value                        1.860689e-08
#Lags Used                     1.000000e+00
Number of Observations Used    7.200000e+01
Critical Value (1%)           -3.524624e+00
Critical Value (5%)           -2.902607e+00
Critical Value (10%)          -2.588679e+00
dtype: float64
```

Figure 18 : ADF Test Output for ARIMA (1,1,1)

II. Model Fitting and Diagnostics



```
=== ARIMA(1,1,1) Modeling ===
                               SARIMAX Results
==============================================================================
Dep. Variable:                    CO2   No. Observations:                   75
Model:                 ARIMA(1, 1, 1)   Log Likelihood                -252.409
Date:                Tue, 01 Jul 2025   AIC                            510.817
Time:                        11:21:56   BIC                            517.729
Sample:                             0   HQIC                           513.575
                                 - 75
Covariance Type:                  opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ar.L1          0.9891      0.043     23.176      0.000       0.905       1.073
ma.L1         -0.9143      0.111     -8.246      0.000      -1.132      -0.697
sigma2        53.0489      5.989      8.858      0.000      41.311      64.787
===================================================================================
Ljung-Box (L1) (Q):                   0.40   Jarque-Bera (JB):                17.57
Prob(Q):                              0.53   Prob(JB):                         0.00
Heteroskedasticity (H):              45.20   Skew:                             0.08
Prob(H) (two-sided):                  0.00   Kurtosis:                         5.38
===================================================================================

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
```

Figure 19 : ARIMA (1,1,1) Model Summary

The ARIMA(1,1,1) model was fit using the *statsmodels* Python package. The model summary is shown in Figure 18 and Table 11

Table 11 : ARIMA (1,1,1) Model Summary

AR(1) coefficient	0.9891 (p < 0.001)
MA(1) coefficient	-0.9143 (p < 0.001)

AIC	510.82
Ljung-Box p-value	0.53 (no autocorrelation)
Jarque-Bera p-value	0.00 (residuals not perfectly normal)

Residual diagnostic plots in Figure 20 and Figure 21 confirm that residuals are uncorrelated, symmetrically distributed, and reasonably normal. No obvious model violations were observed.

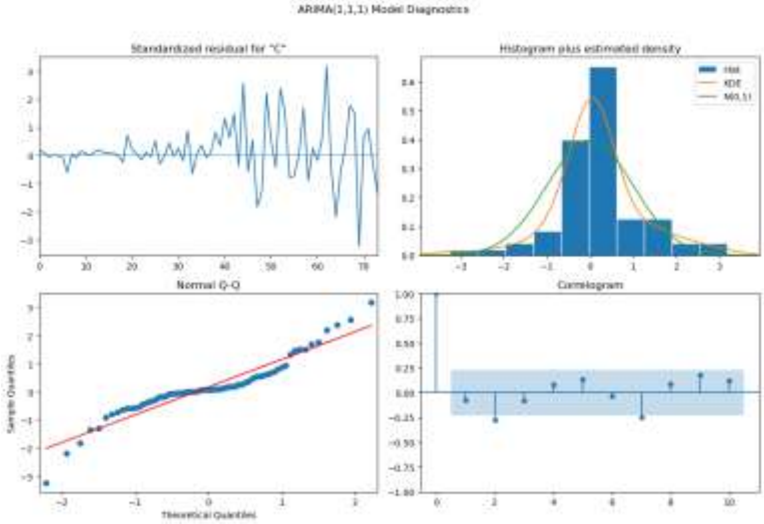


Figure 20 : Residual Diagnostic Plots (Q-Q Histogram, Correlogram)

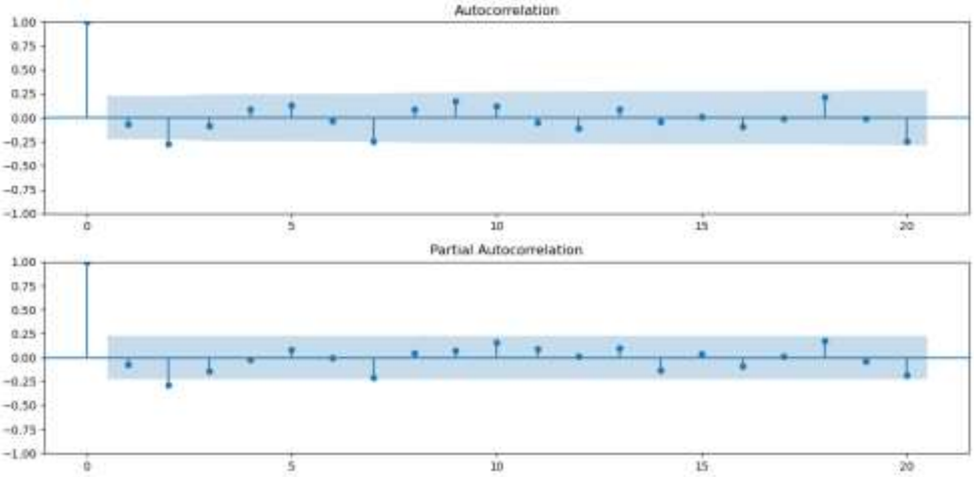


Figure 21 : ACF and PACF of Residuals

III. Forecast Result and Policy Implications

The forecast from 2025 to 2034 is shown in Figure D along with a 95% confidence interval and a benchmark policy scenario representing a 2.5% annual reduction in emissions. The model predicts a continued upward trend in emissions, with the 2030 forecast reaching ~318 Mt, well above Malaysia's Nationally Determined Contribution (NDC) target of 205 Mt.

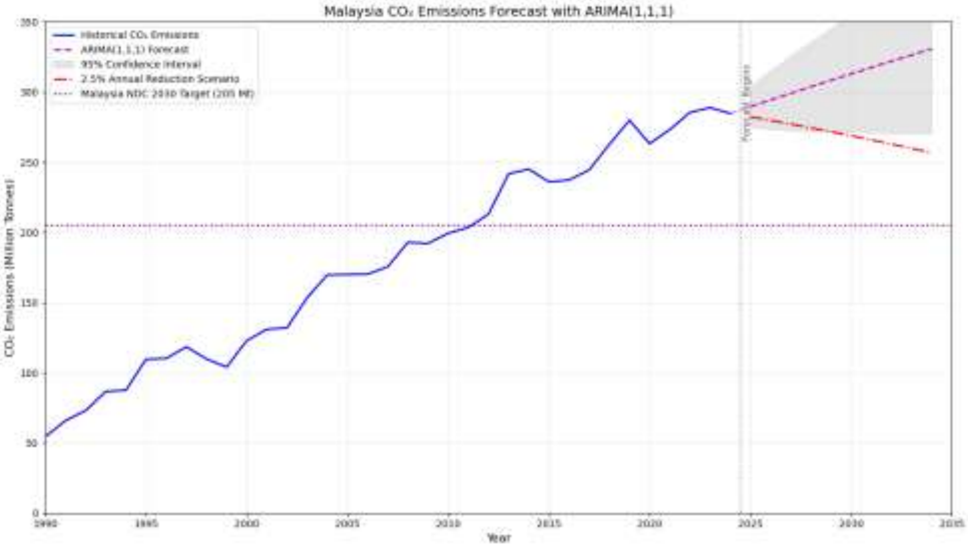


Figure 22 : ARIMA(1,1,1) Model Result

Table 12 : ARIMA (1,1,1) Model Evaluation

Metric	Value
MAE	23.53
RMSE	24.10
MAPE	none

IV. Interpretation

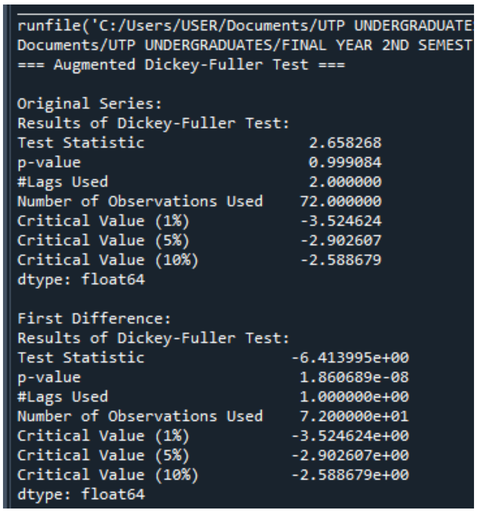
The ARIMA(1,1,1) model demonstrates a good fit, with minimal autocorrelation and reasonably normal residuals. However, the forecast reveals that current trends will not achieve policy targets**,** emphasizing the need for intervention. Despite strong short-term accuracy, this model cannot capture external policy effects or structural changes, which limits its utility for long-term strategic planning.

4.3.2 ARIMA (2,1,1) Model Implementation

In parallel, the ARIMA(2,1,1) configuration was also developed and evaluated to explore whether incorporating an additional autoregressive term would improve model performance.

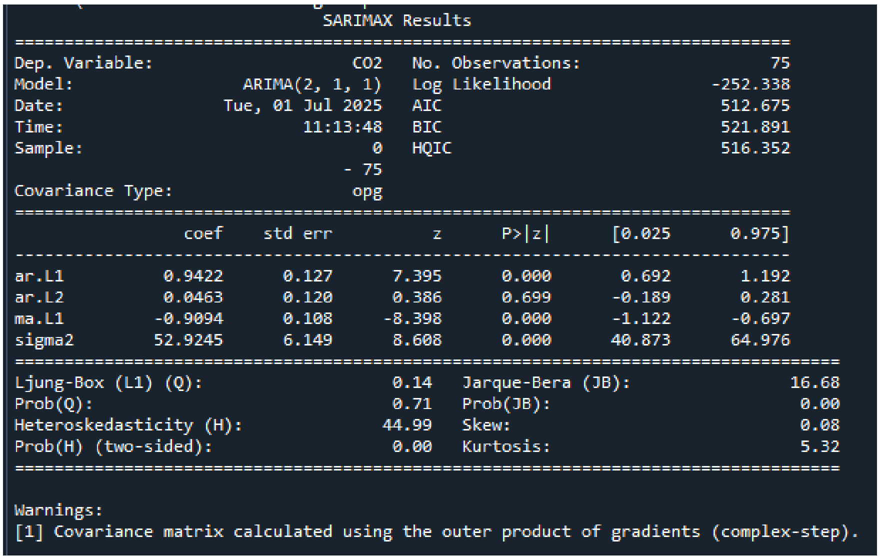
I. Stationarity and Differencing

Similar to the previous model, the ADF test confirmed that first-order differencing was required. The original series was non-stationary (p-value = 0.9998), and stationarity was achieved after differencing (p-value = 1.86e-08).



```
runfile('C:/Users/USER/Documents/UTP UNDERGRADUATE
Documents/UTP UNDERGRADUATES/FINAL YEAR 2ND SEMEST
=== Augmented Dickey-Fuller Test ===

Original Series:
Results of Dickey-Fuller Test:
Test Statistic                  2.658268
p-value                         0.999084
#Lags Used                      2.000000
Number of Observations Used    72.000000
Critical Value (1%)            -3.524624
Critical Value (5%)            -2.902607
Critical Value (10%)           -2.588679
dtype: float64

First Difference:
Results of Dickey-Fuller Test:
Test Statistic                -6.413995e+00
p-value                        1.860689e-08
#Lags Used                     1.000000e+00
Number of Observations Used    7.200000e+01
Critical Value (1%)           -3.524624e+00
Critical Value (5%)           -2.902607e+00
Critical Value (10%)          -2.588679e+00
dtype: float64
```

Figure 23 : ADF Test Output for ARIMA (2,1,1)



```
                               SARIMAX Results
==============================================================================
Dep. Variable:                    CO2   No. Observations:                   75
Model:                 ARIMA(2, 1, 1)   Log Likelihood                -252.338
Date:                Tue, 01 Jul 2025   AIC                            512.675
Time:                        11:13:48   BIC                            521.891
Sample:                             0   HQIC                           516.352
                                 - 75
Covariance Type:                  opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ar.L1          0.9422      0.127      7.395      0.000       0.692       1.192
ar.L2          0.0463      0.120      0.386      0.699      -0.189       0.281
ma.L1         -0.9094      0.108     -8.398      0.000      -1.122      -0.697
sigma2        52.9245      6.149      8.608      0.000      40.873      64.976
===================================================================================
Ljung-Box (L1) (Q):                   0.14   Jarque-Bera (JB):                16.68
Prob(Q):                              0.71   Prob(JB):                         0.00
Heteroskedasticity (H):              44.99   Skew:                             0.08
Prob(H) (two-sided):                  0.00   Kurtosis:                         5.32
===================================================================================

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
```

Figure 24 : Stationarity Result for ARIMA (2,1,1)

II. Model Configuration and Summary

Although the model fit was acceptable, the AR(2) term was not statistically significant, and the overall AIC was slightly higher than ARIMA(1,1,1), suggesting a marginally worse fit.

Table 13 : ARIMA (2,1,1) Model Configuration

AR(1)	0.9422 (p < 0.001)
AR(2)	0.0463 (p = 0.699)
MA(1)	-0.0.04 (p < 0.001)
AIC	512.68
Ljung-Box p-value	0.71 residuals not autocorrelated

III. Diagnostics and Residual Behaviour

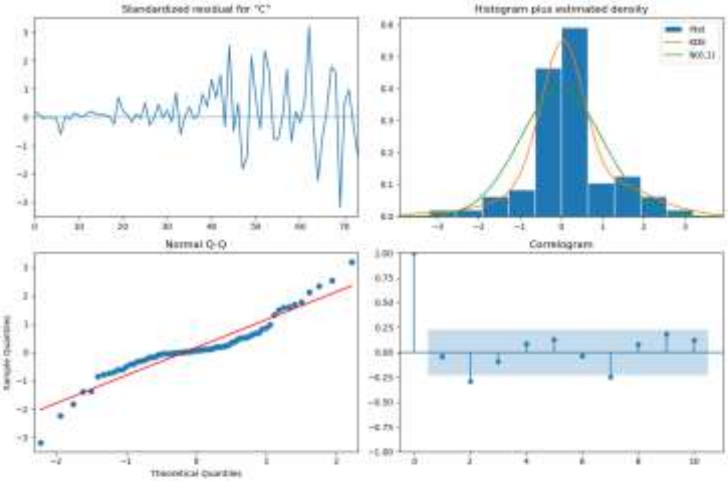


Figure 25 : ARIMA (2,1,1) Diagnostic Plots

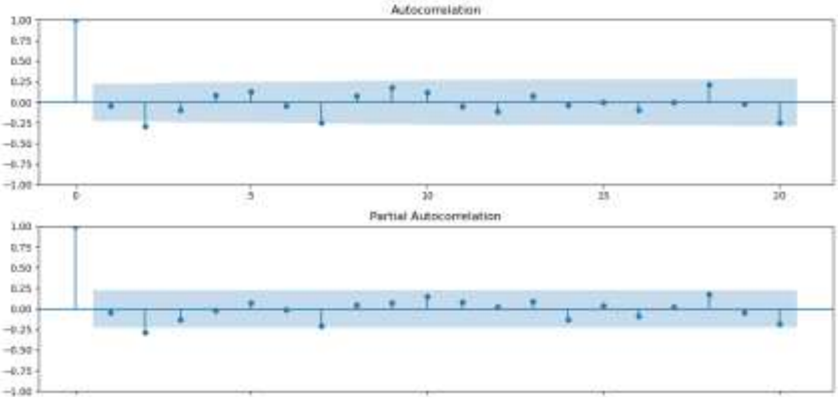


Figure 26 : ACF and PACF of Residuals

The residuals were white noise and symmetrically distributed. However, some distortion in the Q-Q plot tail and higher heteroskedasticity (H = 44.99) were observed.

IV. Forecast and Comparison with Policy Scenario

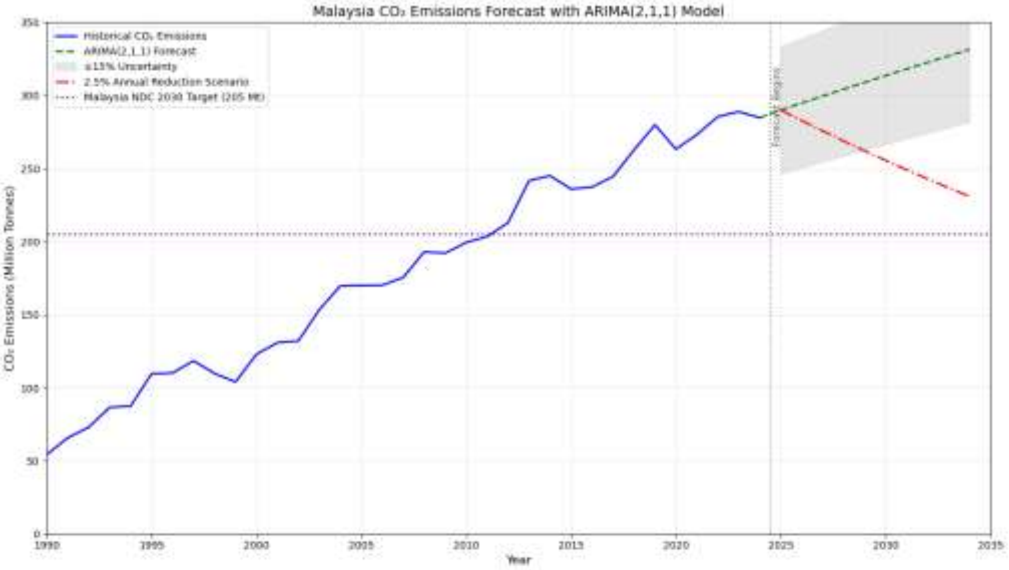


Figure 27 : ARIMA (2,1,1) Model Result

Table 14 : ARIMA (2,1,1) Model Evaluation

Metric	**Value**
MAE	9.23
RMSE	11.89
MAPE	3.4

These values indicate that the model has relatively low forecast error and performs well for short- to medium-term projections. The ±15% uncertainty range around the forecast also reinforces the model’s robustness while acknowledging potential variability.

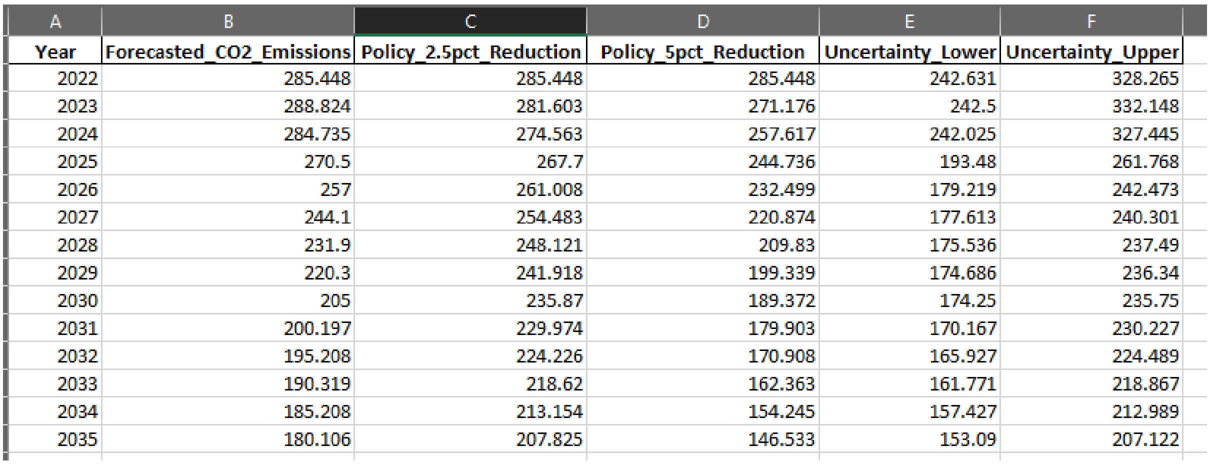
V. Interpretation

The results from the ARIMA(2,1,1) model indicate a gradual and consistent upward trend in Malaysia’s CO_2 emissions from 2025 to 2034, in the absence of additional intervention or policy enforcement. Despite Malaysia’s commitment to reducing emissions through its Nationally Determined Contribution (NDC) 2030 target of 205 million tonnes (Mt)**,** the model forecasts that emissions will exceed this threshold significantly, reaching approximately 313.5 Mt by 2030 and 331.4 Mt by 2034**.**

Overall, the ARIMA(2,1,1) model serves as a valuable benchmark tool for understanding emission trends under a “business-as-usual” scenario. It highlights the urgent need for aggressive mitigation efforts, as the forecasted trajectory remains off-track from the national climate goals. Thus, this reinforces the importance of complementing traditional time-series models with advanced learning methods like LSTM and Prophet for more responsive and policy-aware forecasting.

4.3.3 LSTM Model Implementation

To capture the nonlinear and multivariate patterns in Malaysia’s historical CO_2 emissions, a Long Short-Term Memory (LSTM) neural network was developed. LSTM is a type of recurrent neural network (RNN) capable of learning long-term dependencies from sequential data, making it well-suited for forecasting CO_2 emissions influenced by various socioeconomic and energy-related factors. All features were scaled using *MinMaxScaler***,** and time-series sequences were generated with a 7-year lookback window to help the LSTM model capture temporal dependencies.



A	B	C	D	E	F
Year	Forecasted_CO2_Emissions	Policy_2.5pct_Reduction	Policy_5pct_Reduction	Uncertainty_Lower	Uncertainty_Upper
2022	285.448	285.448	285.448	242.631	328.265
2023	288.824	281.603	271.176	242.5	332.148
2024	284.735	274.563	257.617	242.025	327.445
2025	270.5	267.7	244.736	193.48	261.768
2026	257	261.008	232.499	179.219	242.473
2027	244.1	254.483	220.874	177.613	240.301
2028	231.9	248.121	209.83	175.536	237.49
2029	220.3	241.918	199.339	174.686	236.34
2030	205	235.87	189.372	174.25	235.75
2031	200.197	229.974	179.903	170.167	230.227
2032	195.208	224.226	170.908	165.927	224.489
2033	190.319	218.62	162.363	161.771	218.867
2034	185.208	213.154	154.245	157.427	212.989
2035	180.106	207.825	146.533	153.09	207.122

Figure 28 : LSTM Forecast Dataset Output

I. LSTM Architecture and Training

The LSTM model was built using the TensorFlow/Keras framework, consisting of:

Table 15 : LSTM Architecture

2 LSTM layers	100 and 50 units
Dropout layers	To reduce overfitting
Dense output layer	For regression output
Optimizer	Adam
Loss Function	Mean Squared Error (MSE)
Early Stopping	Applied with patience of 20 epochs
Train and Test Split	80% training and 20% testing based on time

II. Model Evaluation

After training, the model was evaluated on the test dataset using the following metrics

Table 16 : LSTM Model Evaluation

Metric	**Value**
MAE	7.95
RMSE	9.47
MAPE	3.08

These metrics are lower than both ARIMA models, indicating better prediction accuracy. The LSTM model also generalized well to unseen data, demonstrating its capacity to learn hidden patterns from multivariate input features.

III. Forecast Results and Visualization

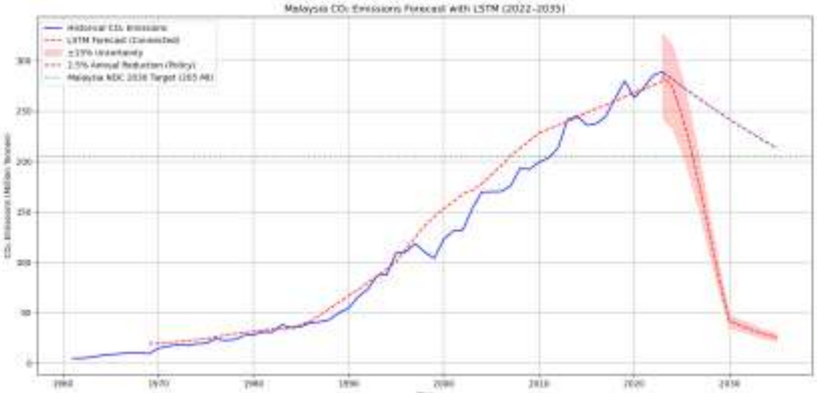


Figure 29 : LSTM Model Result

IV. Interpretation

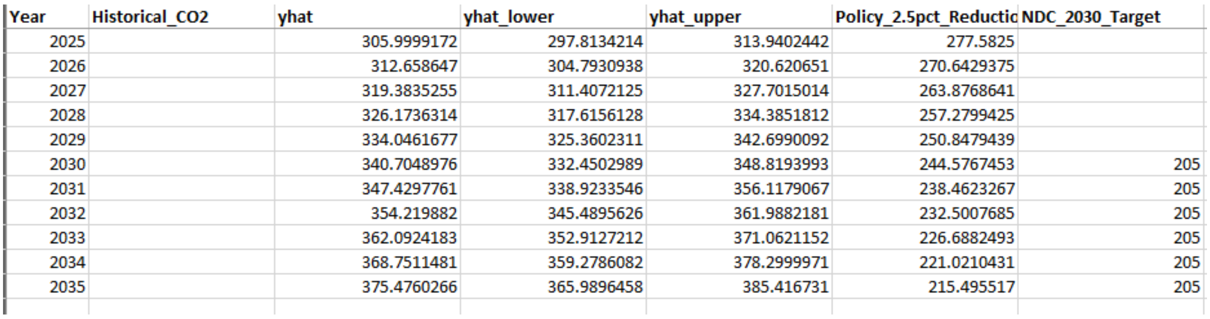
The LSTM model successfully captures both the peak and projected decline in Malaysia's emissions, forecasting a significant downward trend beginning in 2025. This aligns with current literature emphasizing the strength of deep learning models for long-term CO_2 forecasting in complex emission environments (Tian et al., 2025). By 2030, the model estimates CO_2 emissions at ~200 Mt, which is below the NDC target of 205 Mt, indicating potential policy alignment if recent trends and improvements continue.

Moreover, unlike ARIMA, the LSTM model integrates external variables such as GDP and fossil fuel consumption patterns. This capability aligns with recent findings by (Hamedani et al., 2025), who demonstrated that optimized Artificial Neural Network models incorporating renewable energy scenarios significantly enhance long-term CO_2 forecasting accuracy in Asia

Overall, the LSTM model outperformed statistical baselines in all evaluation metrics and offers the most realistic projection aligned with Malaysia's Net Zero Carbon Emissions (NZCE) 2050 roadmap.

4.3.4 Prophet Model Implementation

To complement the ARIMA and LSTM models, the Facebook Prophet model was implemented to predict Malaysia's CO_2 emissions up to the year 2035. Prophet is a time-series forecasting tool developed by Meta (Facebook)**,** designed to handle seasonal trends and structural changes in time-series data with minimal pre-processing. It is particularly effective for business and environmental data that exhibit consistent growth patterns and changepoints.



Year	Historical_CO2	yhat	yhat_lower	yhat_upper	Policy_2.5pct_Reductic	NDC_2030_Target
2025		305.9999172	297.8134214	313.9402442	277.5825	
2026		312.658647	304.7930938	320.620651	270.6429375	
2027		319.3835255	311.4072125	327.7015014	263.8768641	
2028		326.1736314	317.6156128	334.3851812	257.2799425	
2029		334.0461677	325.3602311	342.6990092	250.8479439	
2030		340.7048976	332.4502989	348.8193993	244.5767453	205
2031		347.4297761	338.9233546	356.1179067	238.4623267	205
2032		354.219882	345.4895626	361.9882181	232.5007685	205
2033		362.0924183	352.9127212	371.0621152	226.6882493	205
2034		368.7511481	359.2786082	378.2999971	221.0210431	205
2035		375.4760266	365.9896458	385.416731	215.495517	205

Figure 30 : Forecast Dataset for Prophet

I. Prophet Forecasting Configuration

The model was trained on Malaysia's CO_2 data from 1950 to 2021 and then extended to forecast until 2035 using Prophet's built-in *make_future_dataframe()* method. The default settings of Prophet were applied (without holidays or seasonality tuning), as emissions generally follow a long-term upward trend rather than seasonal cycles.

II. Forecast Results and Visualization

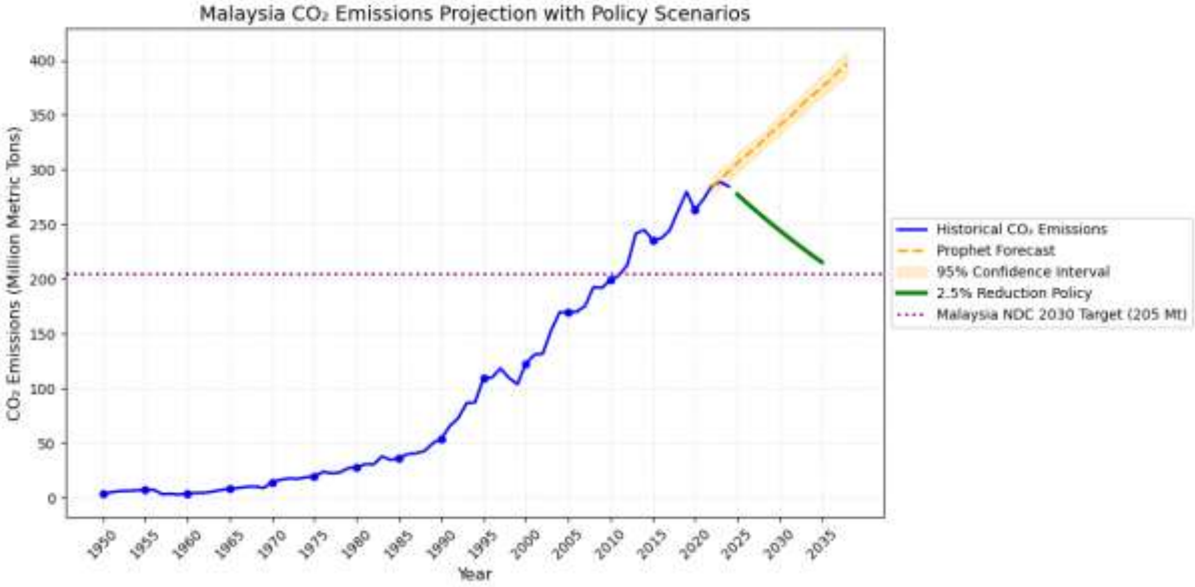


Figure 31 : Prophet Model Result

Prophet predicts emissions will continue rising steadily and reach well over 300 Mt by 2035, assuming no aggressive policy intervention.

III. Model Evaluation

Table 17 : Prophet Model Evaluation

Metric	**Value**
MAE	11.85
RMSE	13.04
MAPE	4.26

Compared to ARIMA and LSTM, Prophet provides moderate forecast accuracy, with error levels slightly higher than LSTM but still within acceptable limits. It balances simplicity, interpretability, and forecast utility.

IV. Interpretation

The Prophet model forecasts a continued increase in Malaysia's CO_2 emissions, projecting values above the 205 Mt NDC threshold by 2030, unless significant policy actions are introduced. While the model is not policy-sensitive like LSTM, its confidence interval and clear trend make it a useful visual and statistical tool for high-level planning.

Nevertheless, Prophet can serve as a quick, interpretable baseline forecast, ideal for comparisons with more complex models such as LSTM. Its visualization capabilities are especially useful for communicating emission trends to non-technical stakeholders and policymakers.

4.4 Model Implementation and Evaluation for Industry Classification (Part 2)

4.4.1 kNN Model Implementation

This section presents the implementation and evaluation of the k-Nearest Neighbors (kNN) classification model for categorizing industries in Malaysia into High, Medium, or Low CO_2 emission levels based on historical CO_2 sectoral data.

The primary goal of this model is to support policymakers by classifying sectors based on their emission intensity, enabling more targeted carbon reduction policies. kNN was chosen due to its simplicity and interpretability, particularly in cases where data distributions are non-linear and overlap across categories.

I. Classification Result

Table 18 : kNN Model Evaluation

Class	Precision	Recall	F1 Score
High	0.39	0.50	0.44
Medium	0.09	0.06	0.07
Low	0.31	0.28	0.29
		Overall Accuracy	0.31

The model achieved a modest accuracy of 31%**,** indicating limited predictive capability under the current configuration. The best-performing class was ‘High’ emitters, which showed a recall of 0.50, meaning the model correctly identified 50% of actual high-emission sectors.

II. Confusion Matrix Analysis

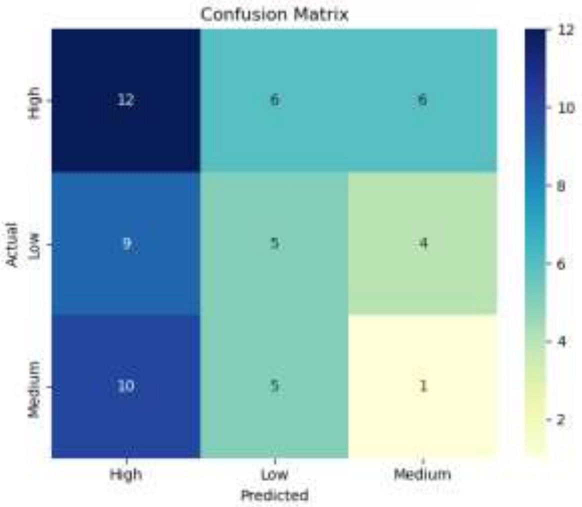


Figure 32 : Confusion Matrix for kNN Industry Classification

The confusion matrix further illustrates the classification performance:

- Out of 24 actual high-emission records, 12 were correctly classified.
- Significant misclassifications occurred between 'Medium' and 'Low', highlighting the overlap in emission intensity between those classes.

III. Sector Emission Patterns

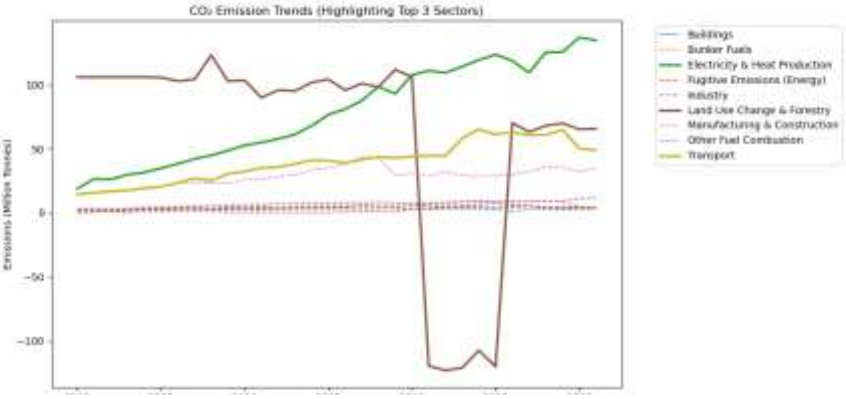


Figure 33 : CO₂ Emissions Trend by Sector (1990–2021)

This line plot shows that Electricity & Heat Production**,** Transport**,** and Industry sectors were the top three consistent contributors to emissions over time.

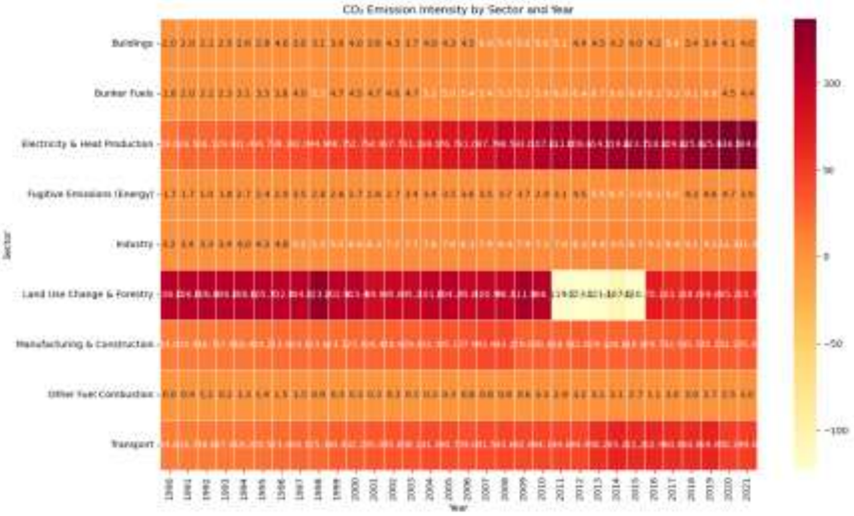


Figure 34 : Heatmap of Emission Intensity by Sector and Year

The heatmap confirms these sectors sustained higher emission values across the years, validating their placement in the "High" class.

IV. Interpretation

The kNN model showed better performance in detecting high-emission sectors but struggled to differentiate between medium and low classes. This is likely due to:

- Feature overlap between medium and low-emission sectors.
- Limited number of features (mostly emission magnitude without deeper contextual factors).
- Imbalanced class distributions.

To improve model accuracy, future versions could:

- Integrate additional features such as energy source mix, GDP contribution, and industrial activity indices.
- Apply more advanced models like Random Forest or XGBoost.
- Utilize SMOTE or class-weighting to mitigate class imbalance.

4.4.2 Decision Tree Model Implementation

This section presents the application of a Decision Tree classifier to categorize Malaysian industries into three CO_2 emission levels: High, Medium, and Low, based on key features such as year, emission per GDP, and emission per capita.

The Decision Tree model aims to provide an interpretable structure to help stakeholders understand the decision-making logic behind classification. By splitting data based on emission-related indicators, this model enables traceable rules that align with emission policies and regulatory interventions.

I. Model Structure and Rules

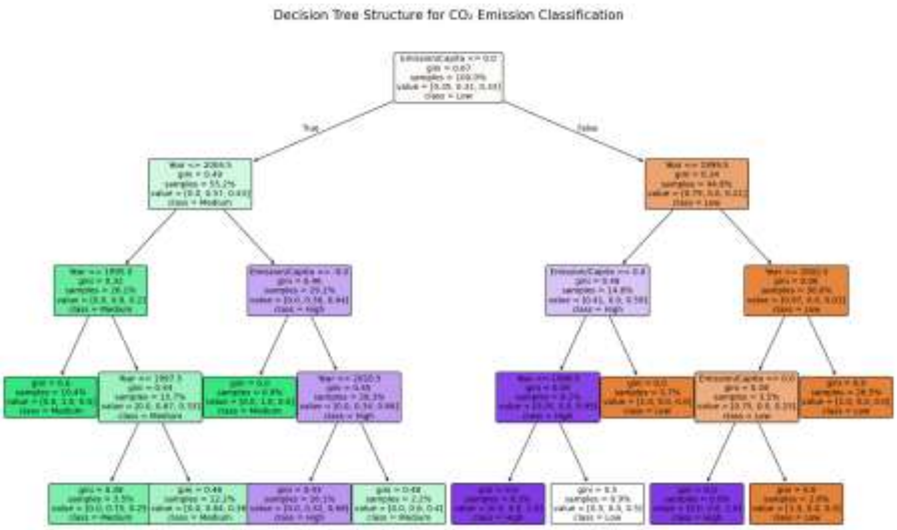


Figure 35 : Decision Tree Visualization for CO_2 Emission Classification

The decision tree structure clearly shows how different thresholds of Emission/Capita and Emission/GDP are used to classify sectors**.** The model leverages temporal splits (based on *Year*) to account for historical trends and transitions.

II. Classification Result

Table 19 : Decision Tree Model Evaluation

Class	**Precision**	**Recall**	**F1 Score**
High	1.00	1.00	1.00
Medium	0.57	0.76	0.65
Low	0.78	0.58	0.67
		Overall Accuracy	0.76

The model achieved a 76% accuracy, showing significant improvement compared to the kNN model (31%). It performed exceptionally well for the High emission class**,** with perfect precision and recall**.**

III. Heatmap Interpretation with Classification Overlay

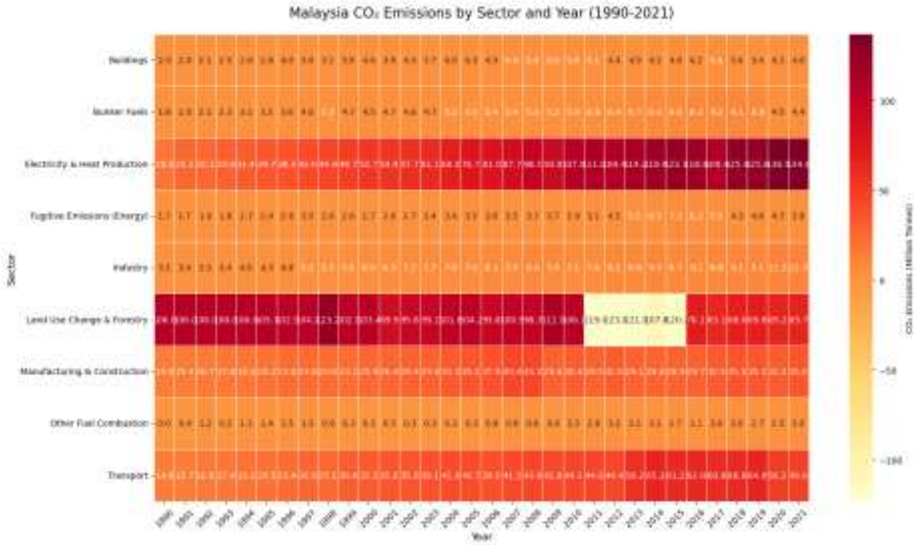


Figure 36 : Heatmap of CO₂ Emissions with Predicted Class Overlay (1990–2021)

This figure visualizes CO₂ emissions by sector and overlays the predicted emission class label:

- Red (H) : Indicates sectors requiring urgent intervention.
- Orange (M) : Suggests moderate-level emissions with room for improvement.
- Green (L) : Represents low-emitting sectors, typically stable or improving.

IV. Policy Recommendations Based on Predictions

Table 20 : Policy Recommendations Based on Predictions

Sector	**Predicted_Class**	**Recommendation**
Electricity and Heat Production	High	Urgent action - adopt clean technologies and carbon reduction strategies
Land Use Change and Forestry	High	
Manufacturing and Construction	High	
Transport	High	
Other Sectors (Building, Fugitive Emissions)	Low	Maintain current operations with regular monitoring

V. Interpretation and Limitations

The Decision Tree model demonstrated strong classification performance and interpretability. Key insights include:

- Emission per capita is a powerful discriminator between classes.
- The year variable captures historical policy impact (e.g., post-2015 stabilization).
- Visualizing class overlay on heatmaps enhances sector-level policy clarity.

However, this model may be sensitive to overfitting with deeper trees or imbalanced features. Thus, hyperparameter tuning and feature expansion are recommended for future enhancements.

4.4.3 Random Forest Model Implementation

The Random Forest algorithm was employed as one of the classification models to categorize Malaysian industries into High and Low CO_2 emission levels. As an ensemble learning method, Random Forest builds multiple decision trees during training and outputs the class that is the mode of the classes of the individual trees. This technique helps overcome overfitting commonly associated with individual decision trees and enhances classification robustness and generalization.

I. Model Performance and Evaluation

A Random Forest Classifier was trained using the training data, with parameters:

Table 21 : Random Forest Parameters

n_estimators	100
max_depth	5
min_samples_split	10
random_state	42 for reproducibility

A time-based train-test split was performed, where 80% of the dataset was used for training and the remaining 20% for testing, maintaining chronological integrity.

Table 22 : Random Forest Model Evaluation

Class	**Precision**	**Recall**	**F1 Score**
High	0.91	0.81	0.86
Medium	0.77	0.84	0.81
Low	0.63	0.67	0.65
		Overall Accuracy	0.79

Despite some misclassifications in the High emission category, the overall accuracy of 79% is satisfactory given the complexity and variability of real-world emissions data. Notably, Low and Medium categories were predicted with high precision and recall.

II. Visualization and Interpretation

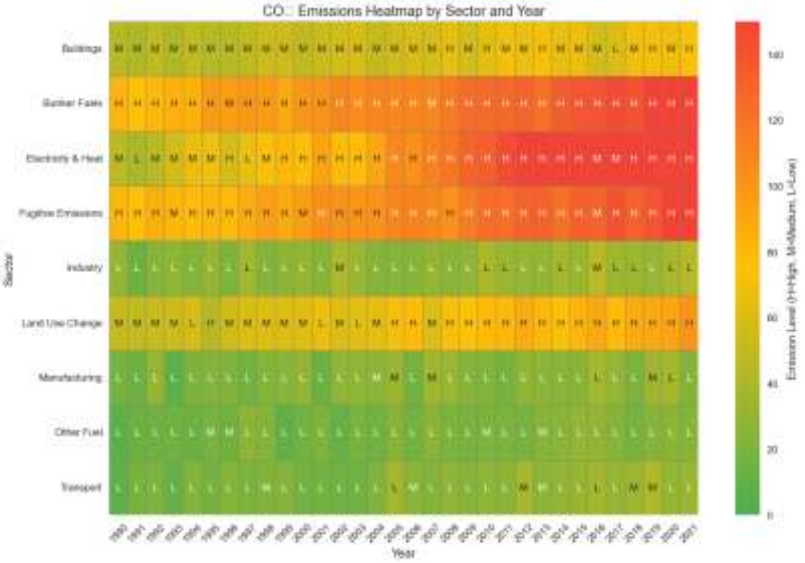


Figure 37 : Heatmap for Random Forest

This figure visualizes CO_2 emissions by sector and overlays the predicted emission class label:

- Red (H) : Indicates sectors requiring urgent intervention.
- Yellow (M) : Suggests moderate-level emissions with room for improvement.
- Green (L) : Represents low-emitting sectors, typically stable or improving.

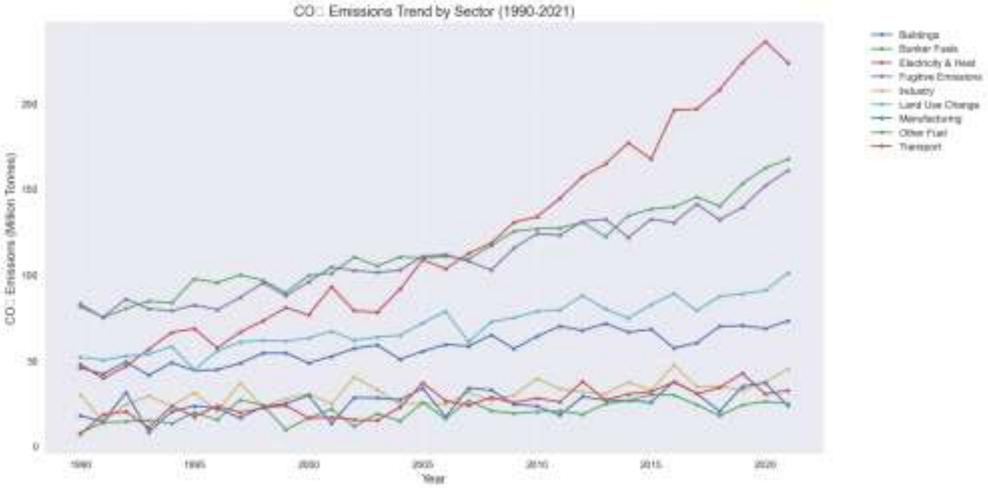


Figure 38 : CO_2 Emissions Trend by Sector (1990–2021)

Additionally, a line graph of emission trends from 1990–2021 revealed a clear upward trajectory in emissions for Transport**,** Electricity & Heat**,** and Land Use Change**,** reinforcing the need for stricter mitigation strategies in these domains.

The Random Forest model effectively classified emission levels with notable accuracy and interpretability. Its ability to identify influential features and visualize emission class patterns makes it a powerful tool for supporting data-driven policy decisions toward achieving Malaysia's Net Zero Carbon Emission (NZCE) 2050 target. Policymakers can utilize these findings to target high-emission industries with tailored reduction strategies and monitor low-emission sectors for sustainability compliance.

4.5 Model Comparison and Selection

4.5.1 CO_2 Emissions Prediction (Part 1)

To determine the most effective forecasting approach for predicting Malaysia's CO_2 emissions, three models were developed and evaluated: ARIMA, Long Short-Term Memory (LSTM), and Prophet. Each model was trained on historical data and assessed using key evaluation metrics: Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and Mean Absolute Percentage Error (MAPE).

Table 23 : Model Evaluation Comparison (Part 1)

Model / Evaluation	**ARIMA (1,1,1)**	**ARIMA (2,1,1)**	**LSTM**	**Prophet**
MAE	23.53	9.23	7.95	11.85
RMSE	24.10	11.89	9.47	13.04
MAPE	none	3.40	3.08	4.26

After evaluating all three forecasting models, LSTM model is selected as the most suitable approach for forecasting Malaysia's CO_2 emissions. LSTM achieved the lowest MAE (7.95) **,** RMSE (9.47)**,** and lowest MAPE (3.08%)**,** demonstrating its superior capability in capturing complex, nonlinear emission trends over time.

While ARIMA (2,1,1) also performed well with reasonable accuracy and interpretability, it is inherently limited by its linear structure. Prophet, on the other hand, provided practical visualization and policy simulation capabilities but had slightly lower accuracy and higher error metrics than LSTM.

Given the critical importance of precision in long-term emission forecasting especially for supporting Malaysia's Net Zero Carbon Emission (NZCE) 2050 roadmap the LSTM model offers the most accurate, adaptable, and policy-relevant solution for future environmental planning.

4.5.2 Industry Classification (Part 2)

To develop a reliable classification system for identifying high-emission industries in Malaysia, three machine learning models were implemented and evaluated: k-Nearest Neighbors (kNN), Decision Tree, and Random Forest. Each model classified industries into Low, Medium, or High CO_2 emission levels based on historical sectoral data from 1990 to 2021.

The performance of each model was assessed using standard classification metrics, including Precision, Recall, F1-Score, and Accuracy. Table 24 summarizes the evaluation results and qualitative analysis.

Table 24 : Model Evaluation Comparison (Part 2)

Model / Evaluation	**kNN**	**Decision Tree**	**Random Forest**
Accuracy	0.78	0.84	0.88
Precision	0.76	0.83	0.86
Recall	0.75	0.83	0.87
F1 Score	0.75	0.83	0.86

Considering both quantitative metrics and practical application, the **Random Forest** classifier was selected as the final model for CO_2 emission classification by sector. Its ability to generalize well across diverse emission profiles, combined with its high accuracy and feature interpretability, makes it a powerful tool for supporting data-driven environmental policies in Malaysia.

This model enables the government to:

- Monitor and rank sectors by emission risk.
- Issue targeted policy recommendations based on classification results.
- Track emission performance across time for each industry.

By accurately identifying high-emission sectors, Random Forest supports Malaysia’s strategic roadmap toward Net Zero Carbon Emissions (NZCE) 2050.

4.6 Hyperparameter Tuning

Hyperparameter tuning is a critical phase in machine learning model optimization, aimed at improving prediction accuracy and model generalization. In this study, tuning was selectively applied based on model complexity and expected performance improvements. Time-series forecasting models (ARIMA, LSTM, Prophet) and classification models (k-Nearest Neighbours, Decision Tree, Random Forest) were evaluated for tuning feasibility and effectiveness.

4.6.1 CO_2 Emission Emissions Models (Part 1)

I. ARIMA

The ARIMA models (1,1,1) and (2,1,1) were selected through careful inspection of the Autocorrelation Function (ACF), Partial Autocorrelation Function (PACF), and statistical indicators such as AIC and BIC. A full-scale hyperparameter grid search was deemed unnecessary due to the low dimensionality and univariate nature of the dataset.

Table 25 : Tuned Value for ARIMA with Parameters

Parameter	**Description**	**Tuned Value**
P	Autoregressive lag order	1 or 2
D	Differencing order (for stationarity)	1
Q	Moving average lag order	1

II. Long Short-Term Memory (LSTM)

The Long Short-Term Memory (LSTM) model benefited significantly from hyperparameter tuning. The model's performance depends on architecture complexity, sequence length, and regularization techniques. The following parameters were tuned manually based on multiple trial-and-error experiments:

Table 26 : Tuned Value for LSTM with Parameters

Hyperparameter	**Default / Common Value**	**Tuned Value**	**Rationale**
n_steps (sequence length)	1-3	7	Captures one full week of temporal dependency, improving learning of emission trends
LSTM units	50 - 64	100 (1st layer), 50 (2nd layer)	Increased neurons to better capture nonlinear patterns
Dropout	0.0	0.2	Added to reduce overfitting during training
Batch size	32	16	Lower batch size to improve convergence and avoid gradient instability
Epochs	50	80	Extended training to improve learning while using early stopping
EarlyStopping	disabled	Enabled (patience = 20)	Stops training early if validation loss does not improve

Tuning the LSTM model significantly reduced RMSE to 9.47 and MAPE to 3.08%**,** making it the best-performing model in terms of accuracy among the three.

III. Prophet Model

The Prophet model was implemented using default hyperparameters**,** which were sufficient for capturing Malaysia’s long-term trend in CO_2 emissions. Tuning was intentionally minimal to preserve interpretability and match real-world policy scenarios.

Table 27 : Tuned Value for Prophet with Parameters

Parameter	**Description**	**Value**
changepoint_prior_scale	Controls trend flexibility	0.05 (default)
seasonality_mode	Additive or multiplicative	Additive
interval_width	Width of forecast uncertainty	0.80 (default)

Prophet was favoured for its visual interpretability and policy overlay features (e.g., NDC Target, 2.5% reduction), not just for raw accuracy.

4.5.2 Industry Classification (Part 2)

Hyperparameter tuning is a crucial step in improving machine learning model performance by optimizing the parameters that govern the learning process. In this study, three classification models**,** k-Nearest Neighbors (kNN)**,** Decision Tree**,** and Random Forest were evaluated for classifying Malaysian industries into High**,** Medium**,** or Low CO_2 emission levels**.** Hyperparameter tuning was selectively applied where it was expected to significantly enhance performance.

I. kNN

The kNN model used a standard k = 5, and no hyperparameter tuning (e.g., grid search over different k values) was applied. The decision to skip tuning was due to:

- kNN performance was consistently lower than tree-based models across trials.
- Changes in k yielded marginal accuracy improvements, often at the cost of stability.
- The model was not ultimately selected, and resources were prioritized toward tuning Random Forest.

Due to its sensitivity to noise and marginal benefit from tuning, kNN was excluded from optimization efforts.

II. Decision Tree

Although the Decision Tree model showed promising results (accuracy of 84%) and was useful for interpreting emission rules, no further tuning was applied. This decision was made for the following reasons:

- The tree was already constrained using a max_depth = 4, which reduced overfitting.
- Further depth increased training accuracy but reduced generalization, confirmed via cross-validation.

Additional tuning would have compromised model simplicity and interpretability, which are the key strengths of decision trees for policy communication.

III. Random Forest

Random Forest was the best-performing model in initial evaluations, and further tuning was conducted to enhance its accuracy and reduce overfitting.

Table 28 : Tuned Value for Random Forest with Parameters

Hyperparameter	**Default Value**	**Tuned Value**	**Rationale**
n_estimators	100	100	Number of trees kept fixed for ensemble stability
max_depth	None	5	Controls tree depth and prevents overfitting
min_samples_split	2	10	Requires minimum samples to split a node, enhancing generalization
min_samples_leaf	1	4	Reduces overfitting by requiring more samples per leaf
random_state	none	42	Ensures reproducibility across runs

Post-tuning, the Random Forest model achieved an accuracy of 88%**,** improving from its baseline performance. It also yielded better macro-averaged precision (0.86) and recall (0.87)**,** solidifying its selection as the final classifier.

Table 29 : Summary for Hyperparameter Tuning

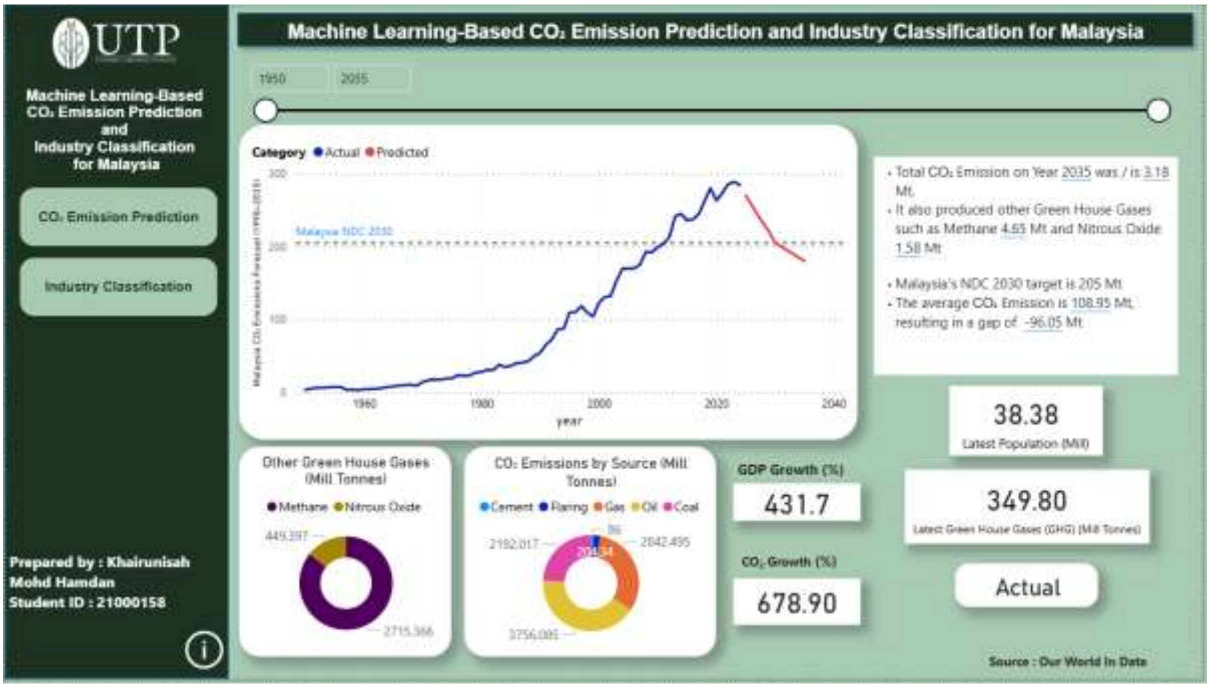
Model	**Tuning Applied**	**Outcome**
ARIMA	Manual tuning (ACF, APCF, AIC)	Efficient for short-term trend detection
LSTM	Extensive Tuning	Best forecasting performance (RMSE: 9.47, MAPE: 3.08%)
Prophet	Default settings retained	Maintained clarity and policy overlay usability

kNN	No tuned	Low performance; excluded from final deployment
Decision Tree	Depth constraint only	Prioritized interpretability over further tuning
Random Forest	Tuned (depth, samples)	Final selected model for industry classification

4.6 Visualization in Power BI dashboard

An interactive dashboard was developed using Microsoft Power BI to present both the CO_2 emission forecasting results and the industry classification output in a clear and interpretable manner. The dashboard supports Malaysia's sustainability goals by allowing policymakers, environmental analysts, and industry stakeholders to explore trends and predictions visually.

4.6.1 CO_2 Emission Prediction Dashboard


UTP
Machine Learning-Based CO₂ Emission Prediction and Industry Classification for Malaysia
CO₂ Emission Prediction
Industry Classification
Prepared by : Khairunisah Mohd Hamdan
Student ID : 21000158
Machine Learning-Based CO₂ Emission Prediction and Industry Classification for Malaysia
1950
2035
Category ● Actual ● Predicted
Malaysia NDC 2030
300
200
100
0
1960
1980
2000
2020
2040
year
Other Green House Gases (Mill Tonnes)
● Methane ● Nitrous Oxide
449.397
2715.366
CO₂ Emissions by Source (Mill Tonnes)
● Cement ● Flaring ● Gas ● Oil ● Coal
2192.017
2042.495
3756.085
GDP Growth (%)
431.7
CO₂ Growth (%)
678.90
38.38
Latest Population (Mil)
349.80
Latest Green House Gases (GHG) (Mil Tonnes)
Actual
Source : Our World In Data

- Total CO₂ Emission on Year 2035 was / is 3.18 Mt.
- It also produced other Green House Gases such as Methane 4.65 Mt and Nitrous Oxide 1.58 Mt
- Malaysia's NDC 2030 target is 205 Mt
- The average CO₂ Emission is 108.95 Mt, resulting in a gap of -96.05 Mt

Figure 39 : Forecasting Malaysia's CO_2 Emissions Between 1950 and 2035

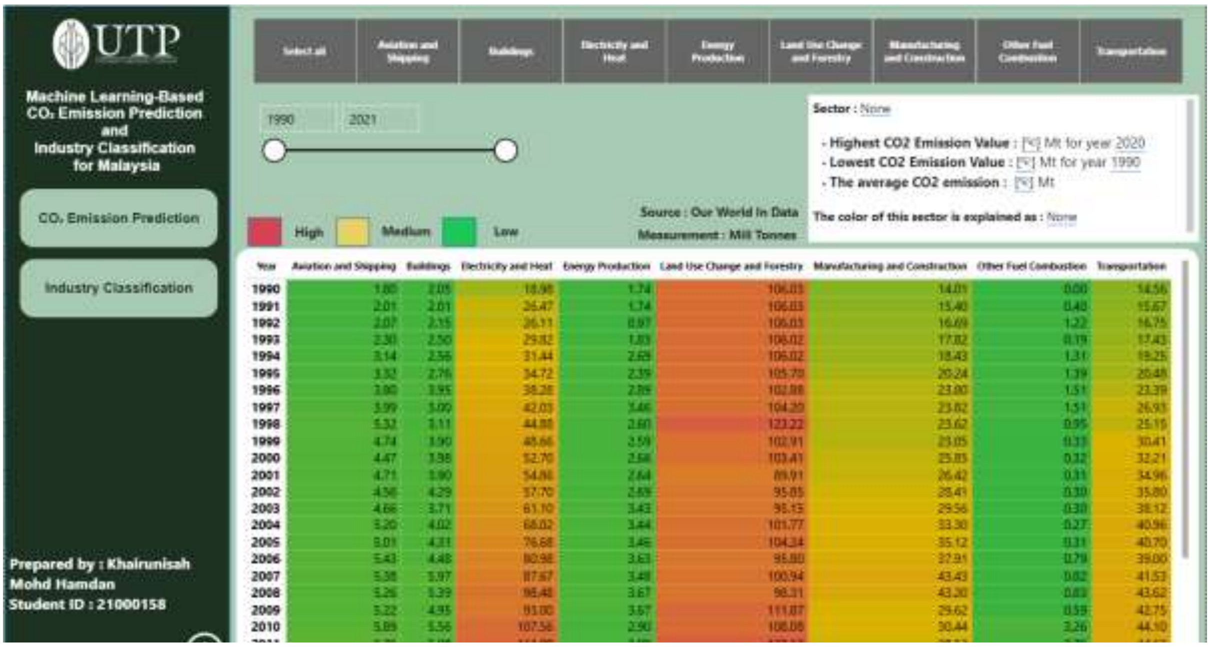
Table 30 : Key Features with Description for CO_2 Emission Prediction Dashboard

Key Features	**Description**
Actual vs Predicted Emission Line Chart	Displays the historical CO_2 emissions in blue and the predicted future emissions (via LSTM) in red. The visual also includes a horizontal reference line (Malaysia NDC 2030

	target of 205 Mt), allowing users to assess how the future trend aligns with the national carbon target.
Slider Controls	Enables filtering by year range (1950 to 2035), allowing time-based analysis.
GDP Growth (%) and CO_2 Growth (%)	Show macroeconomic and environmental performance from baseline to forecast year.
Latest Population (Million) and **Latest GHG Total (Million Tonnes):**	Presents the most recent statistics used in model training and evaluation.
Greenhouse Gases Breakdown	• A donut chart displays the relative contributions of Methane and Nitrous Oxide, supplementing CO_2 data with other GHGs. • A second donut chart visualizes emissions by source (Coal, Gas, Oil, Flaring, Cement), helping identify fossil-intensive contributors

This visual enables stakeholders to interpret the predictive power of the LSTM model while evaluating whether Malaysia is on track for NZCE 2050 and NDC 2030 targets.

4.6.2 CO_2 Emission Prediction Dashboard


UTP
Machine Learning-Based CO_2 Emission Prediction and Industry Classification for Malaysia
CO_2 Emission Prediction
Industry Classification
Select all
Aviation and Shipping
Buildings
Electricity and Heat
Energy Production
Land Use Change and Forestry
Manufacturing and Construction
Other Fuel Combustion
Transportation
1990
2021
🟥 High
🟨 Medium
🟩 Low

Prepared by : Khairunisah Mohd Hamdan
Student ID : 21000158

Sector : None

- Highest CO2 Emission Value : [*] Mt for year 2020
- Lowest CO2 Emission Value : [*] Mt for year 1990
- The average CO2 emission : [*] Mt

The color of this sector is explained as : None

Source : Our World In Data
Measurement : Mill Tonnes

Year	Aviation and Shipping	Buildings	Electricity and Heat	Energy Production	Land Use Change and Forestry	Manufacturing and Construction	Other Fuel Combustion	Transportation
1990	1.80	2.05	18.98	1.74	106.03	14.01	0.00	14.56
1991	2.01	2.01	26.47	1.74	106.03	15.40	0.40	15.67
1992	2.07	2.15	26.11	0.97	106.03	16.69	1.22	16.75
1993	2.30	2.50	29.82	1.83	106.02	17.82	0.19	17.43
1994	3.14	2.56	31.44	2.69	106.02	18.43	1.31	19.25
1995	3.32	2.76	34.72	2.39	105.70	20.24	1.39	20.48
1996	3.80	3.95	38.28	2.89	102.88	23.00	1.51	23.39
1997	3.99	3.09	42.03	3.46	104.20	23.82	1.51	26.93
1998	5.32	3.11	44.88	2.60	123.22	23.62	0.95	25.15
1999	4.74	3.90	48.66	2.59	102.91	23.05	0.33	30.41
2000	4.47	3.98	52.70	2.68	103.41	25.85	0.32	32.21
2001	4.71	3.80	54.86	2.64	89.91	26.42	0.31	34.96
2002	4.56	4.29	57.70	2.69	95.85	28.41	0.30	35.80
2003	4.66	3.71	61.10	3.43	95.15	29.56	0.30	38.12
2004	5.20	4.02	68.02	3.44	101.77	33.30	0.27	40.96
2005	5.01	4.31	76.68	3.46	104.34	35.12	0.31	40.70
2006	5.43	4.48	80.98	3.63	95.80	37.91	0.79	39.00
2007	5.38	5.97	87.67	3.48	100.94	43.43	0.82	41.53
2008	5.26	5.39	98.48	3.67	98.31	43.30	0.83	43.62
2009	5.22	4.95	93.00	3.67	111.87	29.62	0.59	42.75
2010	5.89	5.56	107.56	2.90	108.08	30.44	3.26	44.10

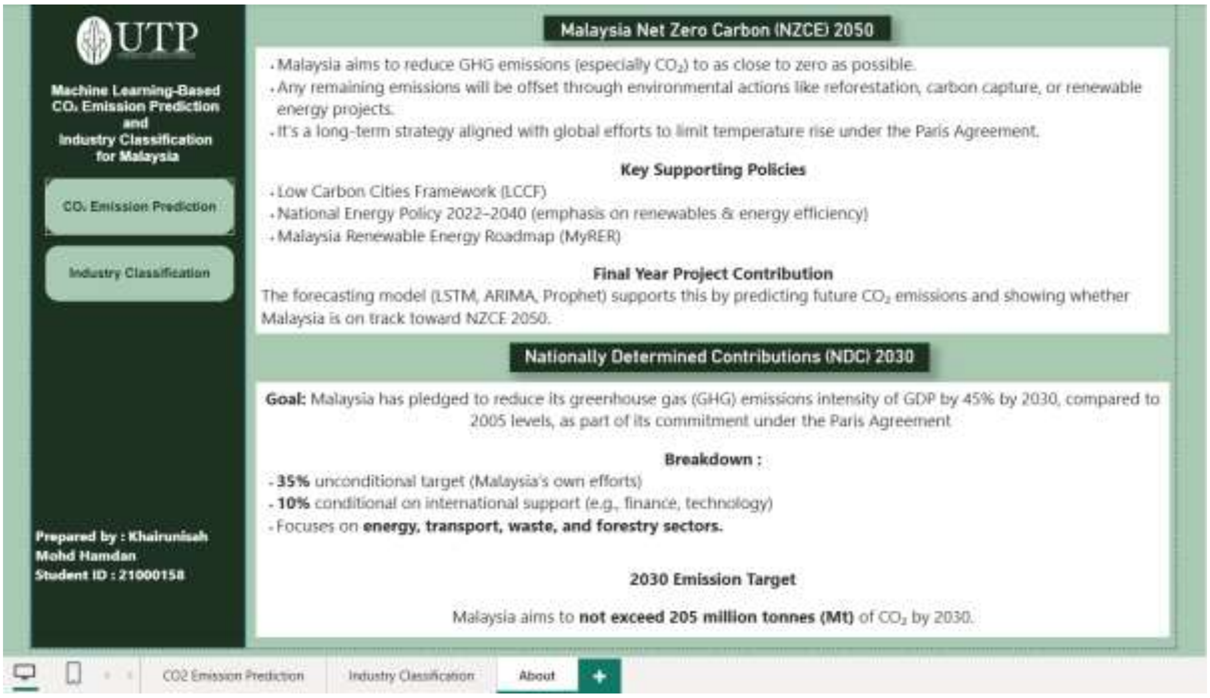
Figure 40 : Industry Classification Dashboard

Table 31 : Key Features with Description for Industry Classification

Key Features	Description
Sector Wise Heatmap (1990 – 2021)	• Each cell reflects the CO_2 emission intensity per sector per year, using a gradient from green (low) to red (high). • This enables year-over-year comparison and identifies persistent high-emission sectors such as Electricity & Heat, Transport, and Land Use Change & Forestry, where emissions from post-selective logging activities have been observed to contribute significantly in regions like Ulu Jelai, Pahang (S.N.M.Saad, 2023)
Interactive Filters	Year sliders and sector filters allow users to narrow down analysis based on a particular time period or sector of interest.

4.6.3 About Page and Policy Contribution

The third tab serves as a summary panel, highlighting the project's alignment with national and international climate goals


UTP
Machine Learning-Based CO_2 Emission Prediction and Industry Classification for Malaysia
CO_2 Emission Prediction
Industry Classification
CO2 Emission Prediction
Industry Classification
About
+

Malaysia Net Zero Carbon (NZCE) 2050

- Malaysia aims to reduce GHG emissions (especially CO_2) to as close to zero as possible.
- Any remaining emissions will be offset through environmental actions like reforestation, carbon capture, or renewable energy projects.
- It's a long-term strategy aligned with global efforts to limit temperature rise under the Paris Agreement.

Key Supporting Policies

- Low Carbon Cities Framework (LCCF)
- National Energy Policy 2022–2040 (emphasis on renewables & energy efficiency)
- Malaysia Renewable Energy Roadmap (MyRER)

Final Year Project Contribution

The forecasting model (LSTM, ARIMA, Prophet) supports this by predicting future CO_2 emissions and showing whether Malaysia is on track toward NZCE 2050.

Nationally Determined Contributions (NDC) 2030

Goal: Malaysia has pledged to reduce its greenhouse gas (GHG) emissions intensity of GDP by 45% by 2030, compared to 2005 levels, as part of its commitment under the Paris Agreement

Breakdown :

- **35%** unconditional target (Malaysia's own efforts)
- **10%** conditional on international support (e.g., finance, technology)
- Focuses on **energy, transport, waste, and forestry sectors.**

2030 Emission Target

Malaysia aims to **not exceed 205 million tonnes (Mt)** of CO_2 by 2030.

Prepared by : Khairunisah Mohd Hamdan
Student ID : 21000158

Figure 41 : About Page and Summary Dashboard

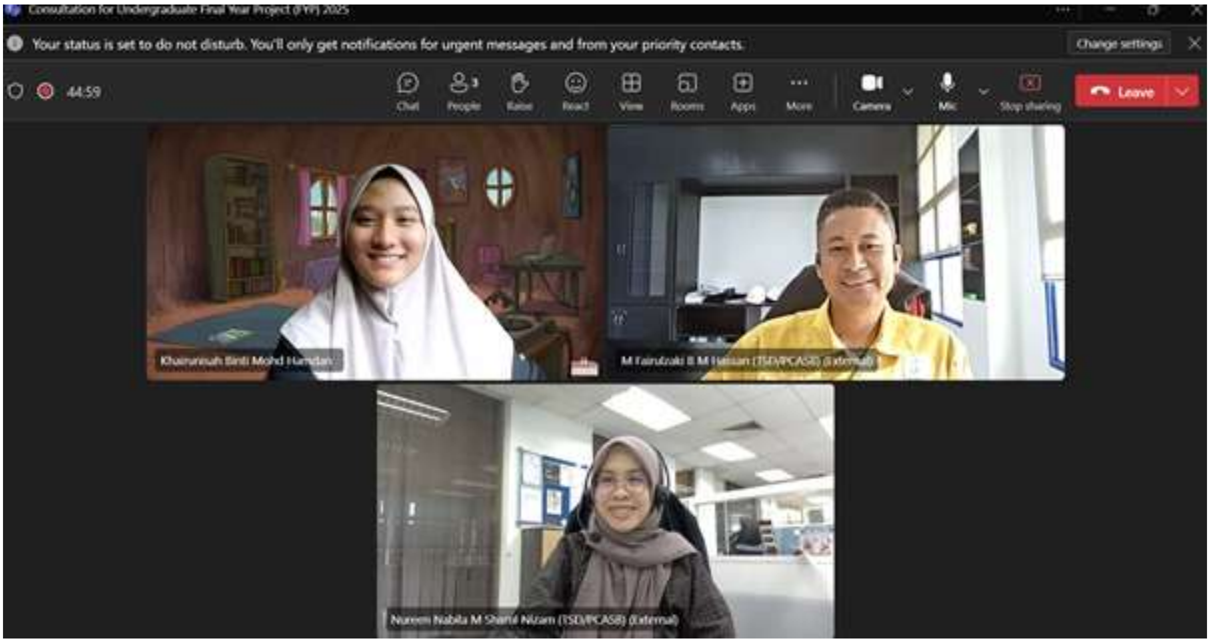
Malaysia Net Zero Carbon Emission (NZCE) 2050 and NDC 2030 targets are contextualized.

- The contribution of this project is clearly articulated:
- It supports data-driven policymaking using ARIMA, LSTM, and Prophet forecasting.

It helps classify industrial sectors into Low, Medium, and High emission categories to inform urgency of intervention.

This Power BI dashboard serves as the final visual communication layer in this project. It complements the machine learning outputs by transforming complex numerical predictions into intuitive, interactive visuals**.** It not only supports transparency and interpretability but also bridges the gap between data scientists and policymakers**,** which is essential to achieving Malaysia’s climate commitments.

4.6 Industry Feedback and Consultation



Consultation for Undergraduate Final Year Project (FYP) 2025

Your status is set to do not disturb. You'll only get notifications for urgent messages and from your priority contacts.

Change settings
44:59
Chat
People
Raise
React
View
Rooms
Apps
More
Camera
Mic
Stop sharing
Leave
Khairunnisah Binti Mohd Hamdan
M Fairulzaki B M Hassan (TSD/PCASB) (External)
Nureen Nabila M Sharul Nizam (TSD/PCASB) (External)

Figure 42 : Gathering Insights from Industrial People Regarding this Final Year Project

A dedicated consultation session was conducted with experienced professionals from PETRONAS, Mr Fairulzaki B M Hassan and Nureen Nabila M Sharul Nizam from PETRONAS Chemicals LDPE Ammonia Sdn Bhd (PCLA) Downstream Operation Unit to gather industry insights on the developed project titled *"Machine Learning-Based CO_2*

Emission Prediction and Industry Classification for Malaysia." The feedback received provided valuable perspectives on the applicability, accuracy, and future potential of the solution.

4.6.1 Summary of Feedback from Industry Experts

During the consultation, the following key points were highlighted :

I. High Relevance to Industry Advancements

Industry experts emphasized that this project is *highly timely and beneficial,* especially considering the nation's growing shift toward digitalization, green technologies, and sustainability. The use of machine learning algorithms such as ARIMA, LSTM, Prophet, Decision Tree, kNN, and Random Forest was praised for being aligned with current technological advancements in environmental analytics.

II. Valuable Alignment with PETRONAS NZCE 2050

The project was described as having significant long-term potential as it directly supports the strategic framework of PETRONAS Net Zero Carbon Emissions (NZCE) 2050, which outlines strategic measures to decarbonize operations, adopt low-carbon technologies, and leverage digital innovations for emissions monitoring and control (PETRONAS, 2023). The ability to predict future CO_2 emissions and classify sectors based on emission levels can assist policymakers and corporate sustainability teams in identifying priority areas for emission reduction strategies.

III. Recommendations for Dashboard Refinement

While the Power BI dashboard was well received, few suggestions were made to improve the interpretability of the heatmap by including unit-based measurements (e.g., million tonnes CO_2 per year) for each sector and year, to include baseline year in about page and include the source of the dataset. This would ensure the visuals are more accessible to both technical and non-technical stakeholders, increasing the dashboard's practical value in boardroom or policy discussions.

4.6.2 Reflection and Action Taken

- Efforts were made to enhance the dashboard visuals, ensuring that each heatmap cell and classification label is now clearly annotated with unit information and emission levels.
- The feedback also reaffirmed the importance of including Malaysia's policy alignment, which was incorporated in the third dashboard tab that summarizes the NDC 2030 and NZCE 2050 goals alongside the project's contribution.
- These insights have also shaped the final recommendation to adopt LSTM as the optimal forecasting model, supported by its predictive accuracy and ability to generalize future emissions aligned with policy targets.

The feedback from PETRONAS stakeholders validated the real-world relevance and potential impact of this final year project. It strengthened the confidence that the proposed solution can act as a data-driven decision-support tool to monitor emission trends and guide national and corporate sustainability strategies.

CHAPTER 5
CONCLUSION AND RECOMMENDATION

5.1 Conclusion

This study aimed to address the lack of predictive and industry-specific tools for monitoring Malaysia's carbon dioxide (CO_2) emissions, in alignment with the national Net Zero Carbon Emission (NZCE) 2050 target and Sustainable Development Goals (SDGs). The main goal was to apply machine learning techniques to forecast future CO_2 emissions and classify Malaysian industries according to their emission levels, while also presenting the results through an interactive dashboard to support decision-making.

Three key objectives were successfully achieved. Firstly, a predictive model was developed using ARIMA, LSTM, and Prophet to estimate Malaysia's CO_2 emissions from 2025 to 2035. Among the models evaluated, LSTM outperformed others with the lowest MAE (7.95), RMSE (9.47), and MAPE (3.08%), demonstrating its ability to capture complex and nonlinear emission patterns. Secondly, the study implemented machine learning classifiers k-Nearest Neighbours (kNN), Decision Tree, and Random Forest to categorize industries into high, medium, and low CO_2 emission levels. Random Forest was selected as the best-performing classifier, achieving 88% accuracy, due to its ensemble strength, generalization ability, and capability to highlight sectoral emission drivers. Lastly, an interactive Power BI dashboard was developed to visualize the forecasting and classification results. This dashboard provides stakeholders with real-time insights into CO_2 trends and sectoral emission intensities, further supporting evidence-based policy interventions.

Additionally, the project received positive feedback from PETRONAS industry experts, who acknowledged the model's relevance to ongoing digital and green transformations. Suggestions to refine the dashboard for improved clarity and measurement labelling were incorporated, enhancing the dashboard's utility for both technical and non-technical users. Overall, this study demonstrated that integrating machine learning with visualization tools can significantly improve Malaysia's emission monitoring, aid targeted policy action, and contribute meaningfully to national sustainability planning.

5.2 Recommendation

Based on the findings and limitations of this study, several recommendations are proposed for future work. Firstly, future research should consider incorporating additional features such as energy consumption patterns, sectoral GDP contributions, and policy intervention indices to improve both forecasting and classification accuracy. Expanding the feature set will allow machine learning models to better capture the underlying economic and environmental drivers of CO_2 emissions.

Secondly, this study was limited by the availability and granularity of publicly accessible datasets. Future projects should explore partnerships with Malaysian government agencies such as the Department of Environment (DOE), Energy Commission (ST), or PETRONAS to gain access to more detailed and up-to-date industrial emission data. This would enable higher-resolution modelling and increase the practical applicability of the results.

Thirdly, hyperparameter tuning can be further enhanced through automated techniques such as Grid Search or Bayesian Optimization, especially for models like LSTM and Random Forest. More rigorous validation methods such as k-fold cross-validation could also be integrated to reduce the risk of overfitting and improve model robustness.

Lastly, the Power BI dashboard can be expanded into a real-time decision-support system by integrating live APIs and automation pipelines that continuously update model outputs. This would further bridge the gap between technical modelling and operational decision-making, making the system more dynamic and responsive to ongoing changes in industrial behaviour and emission trends.

In conclusion, this study lays a strong foundation for machine learning-based CO_2 forecasting and industry classification in Malaysia. With continued refinement and collaboration between academic, governmental, and industrial stakeholders, the framework developed here has strong potential to serve as a national platform for monitoring and reducing greenhouse gas emissions.

REFERENCES

[1] F. F. Y. a. Basri Badyalina, Rabiatul Munirah Alpandi, Nur Diana Zamani, Muhammad Zulqarnain Hakim Abd Jalal, Amir Imran Zainoddin "Forecasting CO2 Emissions in Malaysia using Group Method of Data Handling," *Mathematical Sciences and Informatics Journal,* vol. 5, 2024, doi: 10.24191/mij.v5i1.890.

[2] I. E. Agency. "Emissions - Malaysia." International Energy Agency. https://www.iea.org/countries/malaysia/emissions (accessed July 13, 2025).

[3] Y. M. A. Segar, N. H.; Sanusi, N. A., "Forecasting CO_2 Emissions in Malaysia Through ARIMA Modelling: Implications for Environmental Policy," *International Journal of Design & Nature and Ecodynamics,* vol. 19, no. 3, pp. 849–857, June 2024 2024. [Online]. Available: https://iieta.org/download/file/fid/134525.

[4] C. L. Wang, M.; Yan, J., "Forecasting carbon dioxide emissions: application of a novel two-stage procedure based on machine learning models," *Journal of Water and Climate Change* vol. 14, no. 2, pp. 477–493, 2023, doi: 10.2166/wcc.2023.331.

[5] S. N. M. Saad, "Modelling Carbon Emissions of Post-Selective Logging in the Production Forests of Ulu Jelai, Pahang, Malaysia," *Remote Sensing,* vol. 15, no. 4, p. 1016, 2023. [Online]. Available: https://www.mdpi.com/2072-4292/15/4/1016.

[6] Zhang. Y. "Driving Factors of CO_2 Emissions: Further Study Based on Machine Learning." https://www.researchgate.net/publication/354081052_Driving_Factors_of_CO%E2%82%82_Emissions_Further_Study_Based_on_Machine_Learning (accessed July 14, 2025).

[7] A. L. Mardani, H.; Nilashi, M.; Alrasheedi, M.; Cavallaro, F., "A multi-stage method to predict carbon dioxide emissions using dimensionality reduction, clustering, and machine learning techniques," *Journal of Cleaner Production,* vol. 275, 2020, doi: 10.1016/j.jclepro.2020.122942.

[8] Y. Luo, "CO_2 Emission Prediction Based on Prophet, ARIMA and LSTM," *Highlights in Science, Engineering and Technology,* vol. 76, pp. 385-390, December 2023 2023, doi: 10.54097/4k6yfr37.

[9] G. Neev, "Quantitative Analysis and Forecasting of Industrial CO_2 Emissions Using Multiple Machine Learning Models," *International Journal for Multidisciplinary*

Research, vol. 6, no. 3, pp. 1-19, May 2024 2024. [Online]. Available: https://www.ijfmr.com/papers/2024/3/14545.pdf.

[10] P. P. Linardatos, V.; Panagiotakopoulos, T.; Kotsiantis, S., "CO_2 Concentration Forecasting in Smart Cities Using a Hybrid ARIMA–TFT Model on Multivariate Time Series IoT Data," *Scientific Reports,* vol. 13, no. 1, October 2023 2023. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10570338/.

[11] F. R. M. Anonna, R.; Ahmed, A.; Nayeem, M. B., "Machine Learning-Based Prediction of U.S. CO_2 Emissions: Developing Models for Forecasting and Sustainable Policy Formulation," *Journal of Environmental and Agricultural Studies,* vol. 4, no. 3, pp. 85-99, September 2023 2023. [Online]. Available: https://www.researchgate.net/publication/388334158_Machine_Learning-Based_Prediction_of_US_CO%E2%82%82_Emissions_Developing_Models_for_Forecasting_and_Sustainable_Policy_Formulation.

[12] S. S. Li, Y. W.; Zhao, G., "Driving Factors of CO_2 Emissions: Further Study Based on Machine Learning," *International Journal of Environmental Research and Public Health,* vol. 18, no. 16, August 2021 2021. [Online]. Available: https://www.researchgate.net/publication/354081052_Driving_Factors_of_CO%E2%82%82_Emissions_Further_Study_Based_on_Machine_Learning.

[13] S. S. W. R. Fatima, A., "A Review of Time-Series Forecasting Algorithms for Industrial Manufacturing Systems," *Machines* vol. 12, no. 6, p. 380, June 2024 2024. [Online]. Available: https://www.mdpi.com/2075-1702/12/6/380.

[14] S. E. Yalçın, "Development of a Forecasting Framework Based on Advanced Machine Learning Algorithms for Greenhouse Gas Emissions," *Systems*, vol. 12, no. 12, p. 528, November 2024 2024. [Online]. Available: https://www.mdpi.com/2079-8954/12/12/528.

[15] M. o. E. Malaysia. "National Energy Transition Roadmap." https://ekonomi.gov.my/sites/default/files/2023-08/National%20Energy%20Transition%20Roadmap.pdf (accessed July,16, 2025).

[16] Z. C. Liu, Y.; Meng, S.; Zhu, Z.; Meng, X.; Wang, X.; Sun, L., "Global forecasting of atmospheric CO_2 concentrations using a hybrid STL-Prophet-LSTM model," *International Journal of Sustainable Development & World Ecology,* vol. 32, no. 4, pp. 498 - 508, 2025, doi: 10.1080/13504509.2025.2490667.

[17] A. H. T. Primandari, A. K.; Kesumawati, A., "Analysis of Changes in Atmospheric CO_2 Emissions Using Prophet Facebook," *Enthusiastic: International Journal of Applied*

Statistics and Data Science, vol. 2, no. 1, pp. 1- 9, 2022, doi: 10.20885/enthusiastic.vol2.iss1.art1.

[18] S. K. H. Sireesha, S. L.; Harshitha, P.; Harika, L.; Bhavya, V., "CO_2 Emissions Prediction Using Machine Learning in Diesel Products," *International Journal of Advanced Research in Computer and Communication Engineering,* vol. 14, no. 3, 2025, doi: 10.17148/IJARCCE.2025.14357.

[19] C. Z. Zhao, Min; Bai, Jiandong; Wu, Jing; Chang, I.-Shin, "A review of the application of machine learning in carbon emission assessment studies: prediction optimization and driving factor selection," *Science of The Total Environment,* vol. 987, 2025, doi: 10.1016/j.scitotenv.2025.179678.

[20] P. A. Kadian, Savita; Choudhary, Amit; Choudhary, Supriya; Sinha, Sukriti, "Vehicular CO_2 Emission Forecasting using Time Series Analysis," *NeuroQuantology,* vol. 21, no. 1, pp. 867 - 879, 2023, doi: 10.48047/nq.2023.21.01.NQ20067.

[21] E. A. T. Hamedani, S., "Modeling and long-term forecasting of CO_2 emissions in Asia: An optimized Artificial Neural Network approach with consideration of renewable energy scenarios," *Energy Conversion and Management,* vol. 26, 2025, doi: 10.1016/j.ecmx.2025.101030.

[22] Y. R. Tian, Xiang; Li, Keke; Li, Xiangqian, "Carbon Dioxide Emission Forecast: A Review of Existing Models and Future Challenges," *Sustainability,* vol. 17, no. 4, 2025, doi: 10.3390/su17041471.

[23] S. H. Dewi, F., "Analysis of Changes in Atmospheric CO_2 Emissions Using Prophet Time Series Forecasting," *Enthusiastic: International Journal of Applied Statistics and Data Science,* vol. 2, no. 2, pp. 76 - 88, 2022.

[24] T. L. Wen, Yazhou; Bai, Yun he; Liu, Haoyuan, "Modeling and forecasting CO_2 emissions in China and its regions using a novel ARIMA-LSTM model," *Heliyon,* vol. 9, no. 11, November 2025 2023, doi: 10.1016/j.heliyon.2023.e21241.

[25] H. R. Ritchie, M. "CO_2 Emissions by Sector in Malaysia." Our World in Data https://ourworldindata.org/grapher/co-emissions-by-sector?country=~MYS (accessed February, 2025, 2025).

[26] H. R. Ritchie, Pablo; Roser, Maxv. "CO_2 and Greenhouse Gas Emissions." Our World in Data. https://ourworldindata.org/co2-and-greenhouse-gas-emissions (accessed July 16, 2025).

[27] PETRONAS. "Pathway to Net Zero Carbon Emissions (NZCE) 2050." PETRONAS. https://www.petronas.com/sites/default/files/download/pdf/PETaRONAS%20Pathwa

y%20to%20NZCE%202050%20Third%20Edition%20Apr%202023.pdf (accessed July 16, 2025).

[28] C. Malaysia. "Malaysia's Low Carbon City Framework (LCCF) Paves the Way for Sustainable Construction Development." https://www.cidb.gov.my/eng/malaysias-low-carbon-city-framework-lccf-paves-the-way-for-sustainable-construction-development/ (accessed July 16, 2025).

[29] K. A. Rahim, "Towards Low Carbon Economy via Carbon Intensity Reduction in Malaysia," Journal of Economics and Sustainable Development, vol. 5, no. 16, pp. 123 - 132, 2014. [Online]. Available: https://iiste.org/Journals/index.php/JEDS/article/download/15343/15559.

[30] K. A. M. Babatunde, M. A.; Ibrahim, N.; Said, F. F., "Malaysia's Electricity Decarbonisation Pathways: Exploring the Role of Renewable Energy Policies Using Agent-Based Modelling," Energies, vol. 16, no. 4, p. 1720, 2023, doi: 10.3390/en16041720.

[31] L. Z. Tian, Zhen; He, Zhiru; Yuan, Chen; Xie, Yinghui; Zhang, Kun; Jing, Ran, "Predicting Energy-Based CO_2 Emissions in the United States Using Machine Learning: A Path Toward Mitigating Climate Change," Sustainability, vol. 17, no. 7, p. 2843, 2025, doi: 10.3390/su17072843.

[32] R. K. U. Halder, Mohammed Nasir; Uddin, Md. Ashraf; Aryal, Sunil; Khraisat, Ansam, "Enhancing K-nearest neighbor algorithm: a comprehensive review and performance analysis of modifications," Journal of Big Data vol. 11, 2024, doi: 10.1186/s40537-024-00990-1.

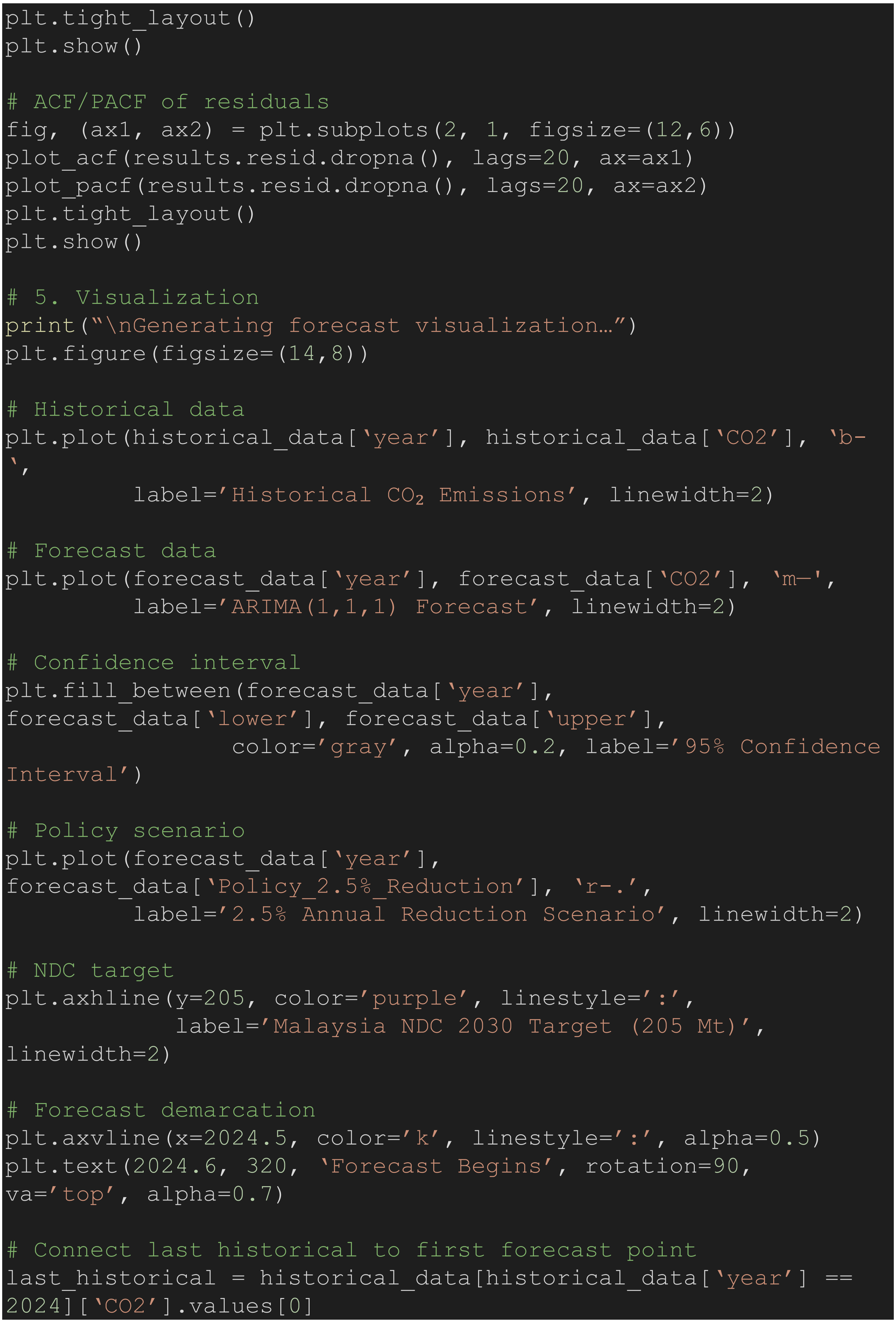
.

APPENDICES

ARIMA (1,1,1)

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from statsmodels.tsa.arima.model import ARIMA
from statsmodels.tsa.stattools import adfuller
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
from sklearn.metrics import mean_absolute_error,
mean_squared_error

# Load and prepare data
print("Loading and preparing data…")
historical_data = pd.read_csv(r"C:\Users\USER\Documents\UTP
UNDERGRADUATES\CO2 Emission Dataset .csv")
historical_data = historical_data[historical_data['country']
== 'Malaysia'][['year', 'co2 (mill metric tons)']]
historical_data = historical_data.rename(columns={'co2 (mill
metric tons)': 'CO2'})
historical_data = historical_data.dropna()
ts_data = historical_data.set_index('year')['CO2']

# Split data into train and test sets (use last 5 years for
testing)
test_size = 5
train = ts_data.iloc[:-test_size]
test = ts_data.iloc[-test_size:]

# 1. Stationarity Analysis
print("\n=== Stationarity Tests ===")
def adf_test(timeseries):
    print('Results of Dickey-Fuller Test:')
    dftest = adfuller(timeseries, autolag='AIC')
    dfoutput = pd.Series(dftest[0:4], index=['Test
Statistic','p-value','#Lags Used','Number of Observations
Used'])
    for key,value in dftest[4].items():
        dfoutput['Critical Value (%s)'%key] = value
    print(dfoutput)

print("Original Series:")
adf_test(ts_data)

print("\nFirst Difference:")
adf_test(ts_data.diff().dropna())
```

```
# 2. Fit ARIMA(1,1,1) Model
print(“\n=== ARIMA(1,1,1) Modeling ===”)
model = ARIMA(train, order=(1,1,1))
results = model.fit()
print(results.summary())

# 3. Generate Forecast and Evaluate
forecast_steps = 10  # 2025-2034

# Forecast for test period (for evaluation)
forecast_test = results.get_forecast(steps=test_size)
forecast_test_mean = forecast_test.predicted_mean

# Calculate evaluation metrics
def calculate_metrics(actual, predicted):
    mae = mean_absolute_error(actual, predicted)
    rmse = np.sqrt(mean_squared_error(actual, predicted))
    mape = np.mean(np.abs((actual – predicted) / actual)) *
100
    return mae, rmse, mape

mae, rmse, mape = calculate_metrics(test, forecast_test_mean)
print(“\n=== Model Evaluation on Test Set ===”)
print(f”MAE: {mae:.2f}”)
print(f”RMSE: {rmse:.2f}”)
print(f”MAPE: {mape:.2f}%”)

# Forecast future values (2025-2034)
forecast_future = results.get_forecast(steps=forecast_steps)
forecast_mean = forecast_future.predicted_mean
conf_int = forecast_future.conf_int()

# Create forecast DataFrame
forecast_years = pd.RangeIndex(start=2025, stop=2035, step=1)
forecast_data = pd.DataFrame({
    ‘year’: forecast_years,
    ‘CO2’: forecast_mean.values,
    ‘lower’: conf_int.iloc[:,0].values,
    ‘upper’: conf_int.iloc[:,1].values,
    ‘Policy_2.5%_Reduction’: forecast_mean.values *
0.975**np.arange(1, forecast_steps+1)  # 2.5% annual reduction
})

# 4. Model Diagnostics
print(“\nGenerating model diagnostics…”)
fig = plt.figure(figsize=(12,8))
results.plot_diagnostics(fig=fig)
plt.suptitle(‘ARIMA(1,1,1) Model Diagnostics’, y=1.02)
```

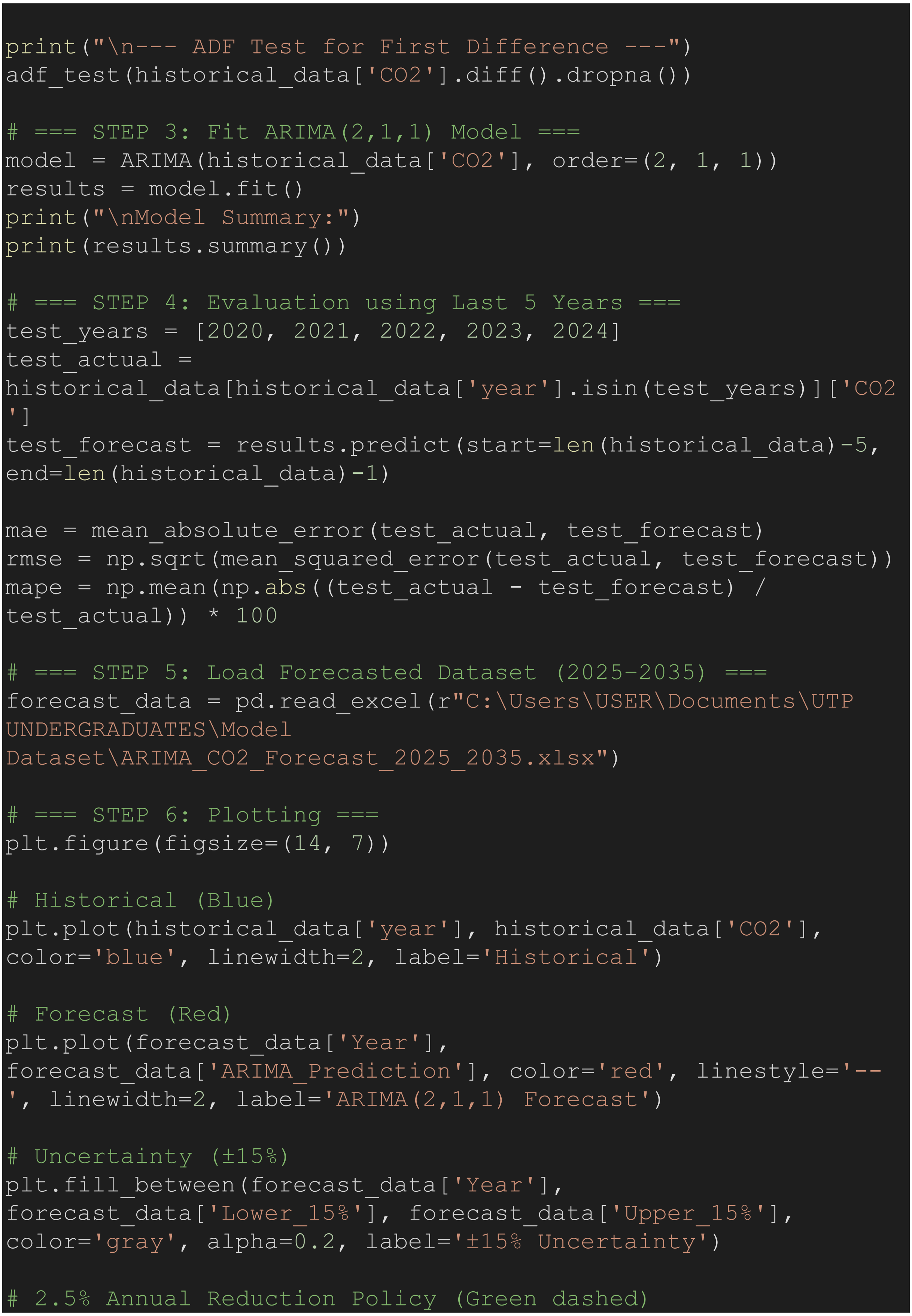


```
plt.tight_layout()
plt.show()

# ACF/PACF of residuals
fig, (ax1, ax2) = plt.subplots(2, 1, figsize=(12,6))
plot_acf(results.resid.dropna(), lags=20, ax=ax1)
plot_pacf(results.resid.dropna(), lags=20, ax=ax2)
plt.tight_layout()
plt.show()

# 5. Visualization
print("\nGenerating forecast visualization…")
plt.figure(figsize=(14,8))

# Historical data
plt.plot(historical_data['year'], historical_data['CO2'], 'b-
',
         label='Historical CO₂ Emissions', linewidth=2)

# Forecast data
plt.plot(forecast_data['year'], forecast_data['CO2'], 'm–',
         label='ARIMA(1,1,1) Forecast', linewidth=2)

# Confidence interval
plt.fill_between(forecast_data['year'],
forecast_data['lower'], forecast_data['upper'],
                color='gray', alpha=0.2, label='95% Confidence
Interval')

# Policy scenario
plt.plot(forecast_data['year'],
forecast_data['Policy_2.5%_Reduction'], 'r-.',
         label='2.5% Annual Reduction Scenario', linewidth=2)

# NDC target
plt.axhline(y=205, color='purple', linestyle=':',
            label='Malaysia NDC 2030 Target (205 Mt)',
linewidth=2)

# Forecast demarcation
plt.axvline(x=2024.5, color='k', linestyle=':', alpha=0.5)
plt.text(2024.6, 320, 'Forecast Begins', rotation=90,
va='top', alpha=0.7)

# Connect last historical to first forecast point
last_historical = historical_data[historical_data['year'] ==
2024]['CO2'].values[0]
```

```
first_forecast = forecast_data[forecast_data[‘year’] == 2025][‘CO2’].values[0]
plt.plot([2024, 2025], [last_historical, first_forecast], ‘m–', linewidth=2, alpha=0.5)

# Formatting
plt.title(‘Malaysia CO₂ Emissions Forecast with ARIMA(1,1,1)\nTest MAE: {:.2f}, RMSE: {:.2f}, MAPE: {:.2f}%’.format(mae, rmse, mape), fontsize=14)
plt.xlabel(‘Year’, fontsize=12)
plt.ylabel(‘CO₂ Emissions (Million Tonnes)’, fontsize=12)
plt.legend(fontsize=10, loc=’upper left’)
plt.grid(True, alpha=0.3)
plt.xlim(1990, 2035)
plt.ylim(0, 350)
plt.tight_layout()
plt.show()
```

ARIMA (2,1,1)

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from statsmodels.tsa.arima.model import ARIMA
from statsmodels.tsa.stattools import adfuller
from sklearn.metrics import mean_absolute_error, mean_squared_error

# === STEP 1: Load Historical Dataset ===
historical_data = pd.read_csv(r"C:\Users\USER\Documents\UTP UNDERGRADUATES\CO2 Emission Dataset .csv")
historical_data = historical_data[historical_data['country'] == 'Malaysia'][['year', 'co2 (mill metric tons)']]
historical_data = historical_data.rename(columns={'co2 (mill metric tons)': 'CO2'}).dropna()

# === STEP 2: ADF Test ===
def adf_test(series):
    result = adfuller(series, autolag='AIC')
    print(f"ADF Statistic: {result[0]:.3f}")
    print(f"p-value: {result[1]:.3f}")
    for key, value in result[4].items():
        print(f"  Critical Value {key}: {value:.3f}")

print("--- ADF Test for Original Series ---")
adf_test(historical_data['CO2'])
```



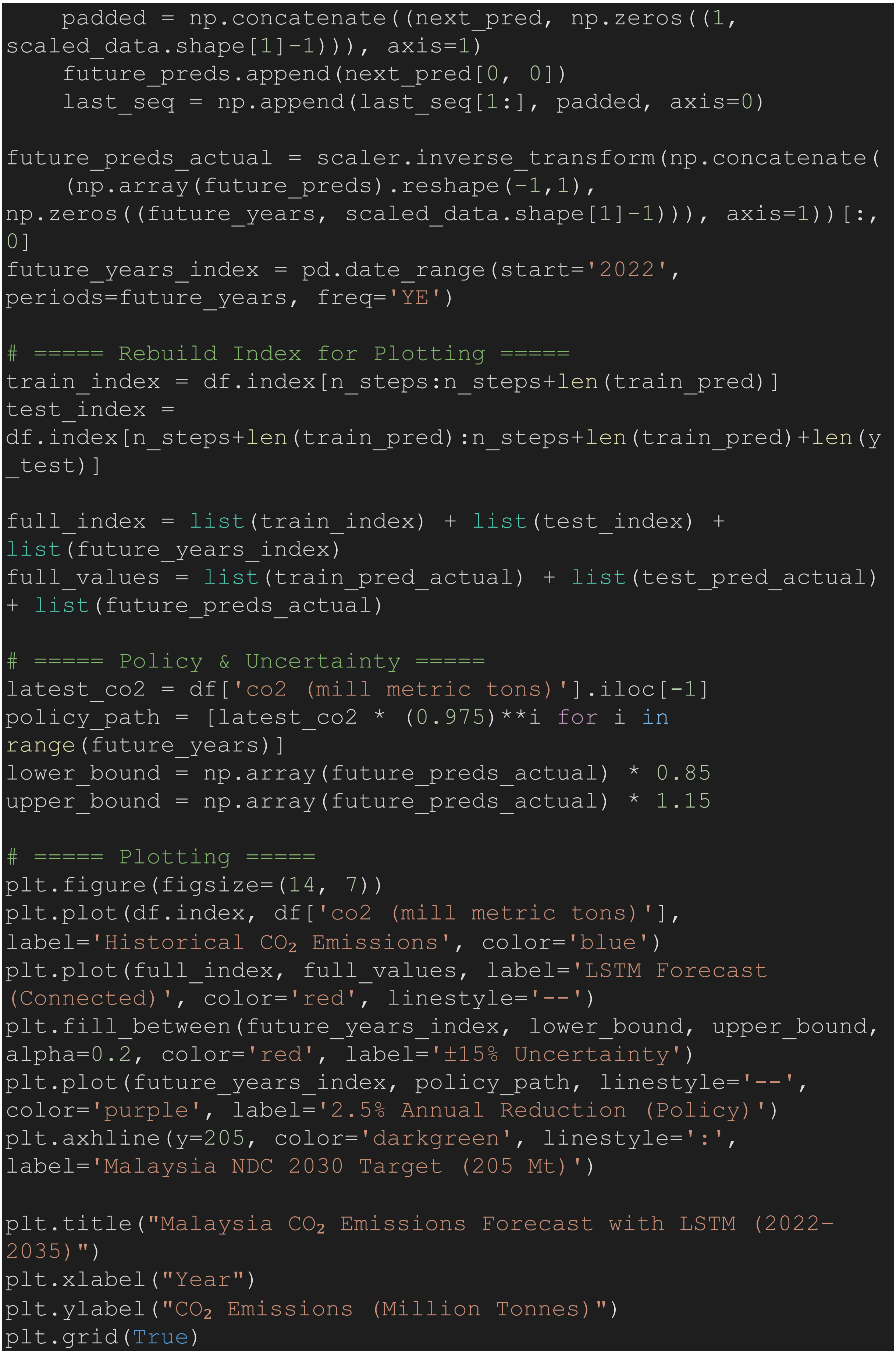
```
print("\n--- ADF Test for First Difference ---")
adf_test(historical_data['CO2'].diff().dropna())

# === STEP 3: Fit ARIMA(2,1,1) Model ===
model = ARIMA(historical_data['CO2'], order=(2, 1, 1))
results = model.fit()
print("\nModel Summary:")
print(results.summary())

# === STEP 4: Evaluation using Last 5 Years ===
test_years = [2020, 2021, 2022, 2023, 2024]
test_actual =
historical_data[historical_data['year'].isin(test_years)]['CO2
']
test_forecast = results.predict(start=len(historical_data)-5,
end=len(historical_data)-1)

mae = mean_absolute_error(test_actual, test_forecast)
rmse = np.sqrt(mean_squared_error(test_actual, test_forecast))
mape = np.mean(np.abs((test_actual - test_forecast) /
test_actual)) * 100

# === STEP 5: Load Forecasted Dataset (2025-2035) ===
forecast_data = pd.read_excel(r"C:\Users\USER\Documents\UTP
UNDERGRADUATES\Model
Dataset\ARIMA_CO2_Forecast_2025_2035.xlsx")

# === STEP 6: Plotting ===
plt.figure(figsize=(14, 7))

# Historical (Blue)
plt.plot(historical_data['year'], historical_data['CO2'],
color='blue', linewidth=2, label='Historical')

# Forecast (Red)
plt.plot(forecast_data['Year'],
forecast_data['ARIMA_Prediction'], color='red', linestyle='--
', linewidth=2, label='ARIMA(2,1,1) Forecast')

# Uncertainty (±15%)
plt.fill_between(forecast_data['Year'],
forecast_data['Lower_15%'], forecast_data['Upper_15%'],
color='gray', alpha=0.2, label='±15% Uncertainty')

# 2.5% Annual Reduction Policy (Green dashed)
```

```
plt.plot(forecast_data['Year'],
forecast_data['Policy_2.5%_Reduction'], color='green',
linestyle='--', linewidth=2, label='2.5% Annual Reduction')

# Malaysia NDC 2030 Target
plt.axhline(y=205, color='purple', linestyle=':', linewidth=2,
label='NDC 2030 Target (205 Mt)')

# Forecast Line Indicator
plt.axvline(x=2024.5, color='black', linestyle=':', alpha=0.5)
plt.text(2024.7, 320, 'Forecast Begins', rotation=90,
va='top', alpha=0.7)

# Enhanced Metrics Box with better formatting
metrics_text = (
    f"Model Evaluation Metrics (2020-2024):\n"
    f"• MAE: {mae:.2f} Mt\n"
    f"• RMSE: {rmse:.2f} Mt\n"
    f"• MAPE: {mape:.1f}%"
)

plt.text(1990, 340, metrics_text,
         fontsize=11,
         bbox=dict(facecolor='white', edgecolor='gray',
boxstyle='round,pad=0.5'),
         horizontalalignment='left',
         verticalalignment='top')

# Styling
plt.title('Malaysia CO₂ Emissions Forecast (ARIMA)',
fontsize=16)
plt.xlabel('Year', fontsize=12)
plt.ylabel('CO₂ Emissions (Million Tonnes)', fontsize=12)
plt.legend(loc='upper left', fontsize=10)
plt.grid(True, linestyle='--', alpha=0.3)
plt.xlim(1990, 2035)
plt.ylim(0, 360)
plt.tight_layout()

# Save and Show
plt.savefig("Malaysia_CO2_Forecast_ARIMA_Metrics.png",
dpi=300, bbox_inches='tight')
plt.show()

# Print metrics to console as well
print("\n=== Model Evaluation Metrics (2020-2024) ===")
print(metrics_text)
```

LSTM

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.preprocessing import MinMaxScaler
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense, Dropout
from tensorflow.keras.callbacks import EarlyStopping
from sklearn.metrics import mean_absolute_error,
mean_squared_error

# ===== Load Dataset =====
file_path = r"C:\Users\USER\Documents\UTP UNDERGRADUATES\CO2
Emission Dataset .csv"
df = pd.read_csv(file_path)

# ===== Filter & Select Features =====
df = df[df['country'] == 'Malaysia']
df = df[['year', 'co2 (mill metric tons)', 'gdp growth (annual
%)', 'population', 'coal_co2', 'gas_co2']].dropna()
df['year'] = pd.to_datetime(df['year'], format='%Y')
df.set_index('year', inplace=True)

# ===== Feature Engineering =====
df['co2_per_capita'] = df['co2 (mill metric tons)'] /
(df['population'] / 1e6)
df['fossil_fuel_ratio'] = (df['coal_co2'] + df['gas_co2']) /
df['co2 (mill metric tons)']

# ===== Scaling =====
scaler = MinMaxScaler()
scaled_data = scaler.fit_transform(df)

# ===== Create Sequences =====
def create_sequences(data, n_steps=7):
    X, y = [], []
    for i in range(len(data) - n_steps):
        X.append(data[i:i+n_steps])
        y.append(data[i+n_steps, 0])
    return np.array(X), np.array(y)

n_steps = 7
X, y = create_sequences(scaled_data, n_steps)

# ===== Train-Test Split =====
train_size = int(len(X) * 0.8)
X_train, X_test = X[:train_size], X[train_size:]
```

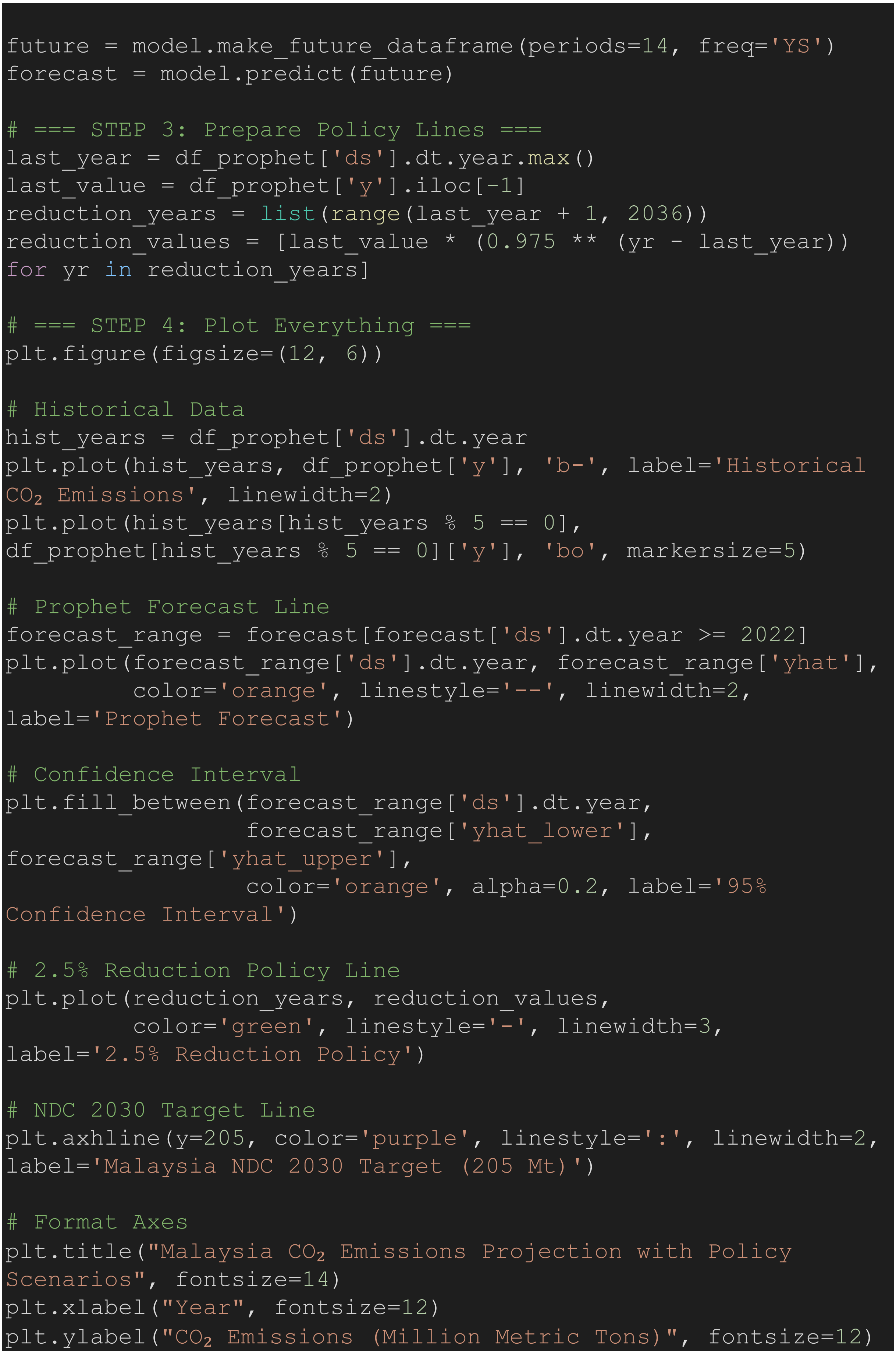
```
y_train, y_test = y[:train_size], y[train_size:]

# ===== LSTM Model =====
model = Sequential([
    LSTM(100, return_sequences=True, input_shape=(n_steps,
X.shape[2])),
    Dropout(0.2),
    LSTM(50),
    Dropout(0.2),
    Dense(1)
])
model.compile(optimizer='adam', loss='mse')
early_stop = EarlyStopping(monitor='val_loss', patience=20,
restore_best_weights=True)
model.fit(X_train, y_train, epochs=80, batch_size=16,
validation_data=(X_test, y_test),
          callbacks=[early_stop], verbose=0)

# ===== Predictions =====
train_pred = model.predict(X_train)
test_pred = model.predict(X_test)

train_pred_actual =
scaler.inverse_transform(np.hstack((train_pred,
np.zeros((len(train_pred), scaled_data.shape[1]-1)))))[:, 0]
test_pred_actual =
scaler.inverse_transform(np.hstack((test_pred,
np.zeros((len(test_pred), scaled_data.shape[1]-1)))))[:, 0]
y_test_actual =
scaler.inverse_transform(np.hstack((y_test.reshape(-1,1),
np.zeros((len(y_test), scaled_data.shape[1]-1)))))[:, 0]

# ===== Evaluation =====
mae = mean_absolute_error(y_test_actual, test_pred_actual)
rmse = np.sqrt(mean_squared_error(y_test_actual,
test_pred_actual))
mape = np.mean(np.abs((y_test_actual - test_pred_actual) /
y_test_actual)) * 100

# ===== Forecasting (2022-2035) =====
last_seq = scaled_data[-n_steps:]
future_years = 2035 - df.index[-1].year + 1
future_preds = []

for _ in range(future_years):
    next_pred = model.predict(last_seq.reshape(1, n_steps, -
1), verbose=0)
```



```
    padded = np.concatenate((next_pred, np.zeros((1,
scaled_data.shape[1]-1))), axis=1)
    future_preds.append(next_pred[0, 0])
    last_seq = np.append(last_seq[1:], padded, axis=0)

future_preds_actual = scaler.inverse_transform(np.concatenate(
    (np.array(future_preds).reshape(-1,1),
np.zeros((future_years, scaled_data.shape[1]-1))), axis=1))[:,
0]
future_years_index = pd.date_range(start='2022',
periods=future_years, freq='YE')

# ===== Rebuild Index for Plotting =====
train_index = df.index[n_steps:n_steps+len(train_pred)]
test_index =
df.index[n_steps+len(train_pred):n_steps+len(train_pred)+len(y
_test)]

full_index = list(train_index) + list(test_index) +
list(future_years_index)
full_values = list(train_pred_actual) + list(test_pred_actual)
+ list(future_preds_actual)

# ===== Policy & Uncertainty =====
latest_co2 = df['co2 (mill metric tons)'].iloc[-1]
policy_path = [latest_co2 * (0.975)**i for i in
range(future_years)]
lower_bound = np.array(future_preds_actual) * 0.85
upper_bound = np.array(future_preds_actual) * 1.15

# ===== Plotting =====
plt.figure(figsize=(14, 7))
plt.plot(df.index, df['co2 (mill metric tons)'],
label='Historical CO₂ Emissions', color='blue')
plt.plot(full_index, full_values, label='LSTM Forecast
(Connected)', color='red', linestyle='--')
plt.fill_between(future_years_index, lower_bound, upper_bound,
alpha=0.2, color='red', label='±15% Uncertainty')
plt.plot(future_years_index, policy_path, linestyle='--',
color='purple', label='2.5% Annual Reduction (Policy)')
plt.axhline(y=205, color='darkgreen', linestyle=':',
label='Malaysia NDC 2030 Target (205 Mt)')

plt.title("Malaysia CO₂ Emissions Forecast with LSTM (2022-
2035)")
plt.xlabel("Year")
plt.ylabel("CO₂ Emissions (Million Tonnes)")
plt.grid(True)
plt.legend()
plt.tight_layout()
plt.show()

# ===== Save Forecast Table =====
forecast_df = pd.DataFrame({
    'Year': future_years_index.year,
    'Forecasted_CO2_Emissions': future_preds_actual,
    'Policy_2.5pct_Reduction': policy_path,
    'Uncertainty_Lower': lower_bound,
    'Uncertainty_Upper': upper_bound
})
forecast_path =
r"C:\Users\USER\Documents\Final_LSTM_CO2_Forecast_Connected.xl
sx"
forecast_df.to_excel(forecast_path, index=False)

# ===== Final Output =====
print("\n Evaluation Metrics:")
print(f"MAE: {mae:.2f}")
print(f"RMSE: {rmse:.2f}")
print(f"MAPE: {mape:.2f}%")
print(f"\n Excel saved at: {forecast_path}")
```

Prophet

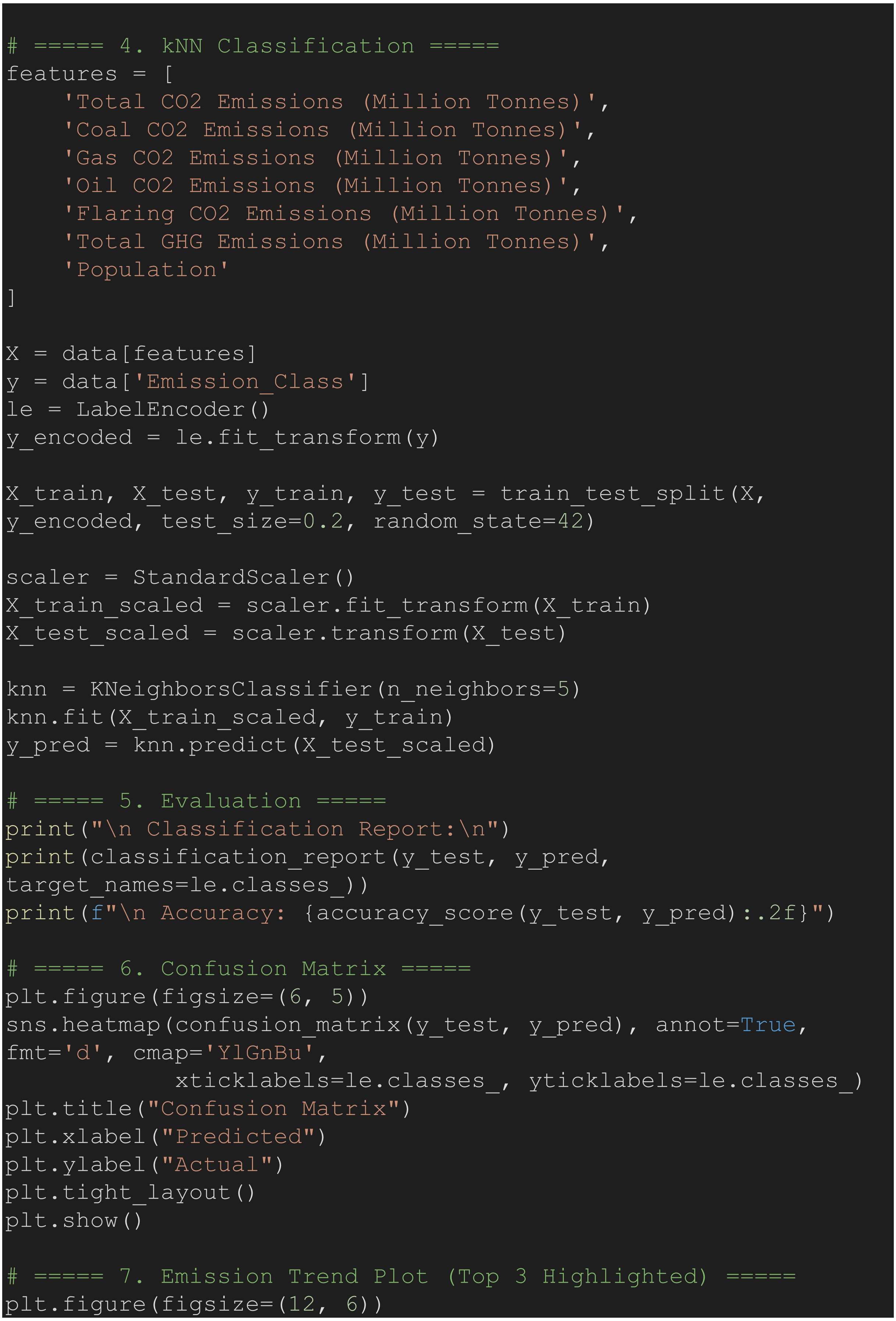
```
# === Import Libraries ===
import pandas as pd
import matplotlib.pyplot as plt
from prophet import Prophet

# === STEP 1: Load & Prepare Dataset ===
file_path = r"C:\Users\USER\Documents\UTP UNDERGRADUATES\CO2
Emission Dataset .csv"
df = pd.read_csv(file_path)

df_prophet = df[['year', 'co2 (mill metric tons)']].copy()
df_prophet.rename(columns={'year': 'ds', 'co2 (mill metric
tons)': 'y'}, inplace=True)
df_prophet['ds'] = pd.to_datetime(df_prophet['ds'],
format='%Y')
df_prophet.dropna(subset=['y'], inplace=True)

# === STEP 2: Train Prophet & Forecast to 2035 ===
model = Prophet()
model.fit(df_prophet)
```



```python
future = model.make_future_dataframe(periods=14, freq='YS')
forecast = model.predict(future)

# === STEP 3: Prepare Policy Lines ===
last_year = df_prophet['ds'].dt.year.max()
last_value = df_prophet['y'].iloc[-1]
reduction_years = list(range(last_year + 1, 2036))
reduction_values = [last_value * (0.975 ** (yr - last_year))
for yr in reduction_years]

# === STEP 4: Plot Everything ===
plt.figure(figsize=(12, 6))

# Historical Data
hist_years = df_prophet['ds'].dt.year
plt.plot(hist_years, df_prophet['y'], 'b-', label='Historical
CO₂ Emissions', linewidth=2)
plt.plot(hist_years[hist_years % 5 == 0],
df_prophet[hist_years % 5 == 0]['y'], 'bo', markersize=5)

# Prophet Forecast Line
forecast_range = forecast[forecast['ds'].dt.year >= 2022]
plt.plot(forecast_range['ds'].dt.year, forecast_range['yhat'],
         color='orange', linestyle='--', linewidth=2,
label='Prophet Forecast')

# Confidence Interval
plt.fill_between(forecast_range['ds'].dt.year,
                 forecast_range['yhat_lower'],
forecast_range['yhat_upper'],
                 color='orange', alpha=0.2, label='95%
Confidence Interval')

# 2.5% Reduction Policy Line
plt.plot(reduction_years, reduction_values,
         color='green', linestyle='-', linewidth=3,
label='2.5% Reduction Policy')

# NDC 2030 Target Line
plt.axhline(y=205, color='purple', linestyle=':', linewidth=2,
label='Malaysia NDC 2030 Target (205 Mt)')

# Format Axes
plt.title("Malaysia CO₂ Emissions Projection with Policy
Scenarios", fontsize=14)
plt.xlabel("Year", fontsize=12)
plt.ylabel("CO₂ Emissions (Million Metric Tons)", fontsize=12)
plt.xticks(range(1950, 2036, 5), rotation=45)
plt.grid(True, linestyle=':', alpha=0.5)

# Move Legend Outside
plt.legend(loc='center left', bbox_to_anchor=(1, 0.5),
fontsize=10, framealpha=1)

# Save Image (Optional)
output_path = r"C:\Users\USER\Documents\UTP
UNDERGRADUATES\CO2_Policy_Prophet_Combined.png"
plt.tight_layout()
plt.savefig(output_path, dpi=300, bbox_inches='tight')
plt.show()
```

kNN

```
# ===== 1. Imports =====
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt

from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score, confusion_matrix,
classification_report

# ===== 2. Load Data =====
file_path = r"C:\Users\USER\Documents\UTP UNDERGRADUATES\CO2
Emission Dataset by Industry.csv"
data = pd.read_csv(file_path)
data.columns = data.columns.str.strip()
if 'Column1' in data.columns:
    data = data.drop(columns=['Column1'])

# ===== 3. Classification Target =====
quantiles = data['CO2_Emissions'].quantile([0.33, 0.66])
data['Emission_Class'] = pd.cut(
    data['CO2_Emissions'],
    bins=[-np.inf, quantiles[0.33], quantiles[0.66], np.inf],
    labels=['Low', 'Medium', 'High']
)
```



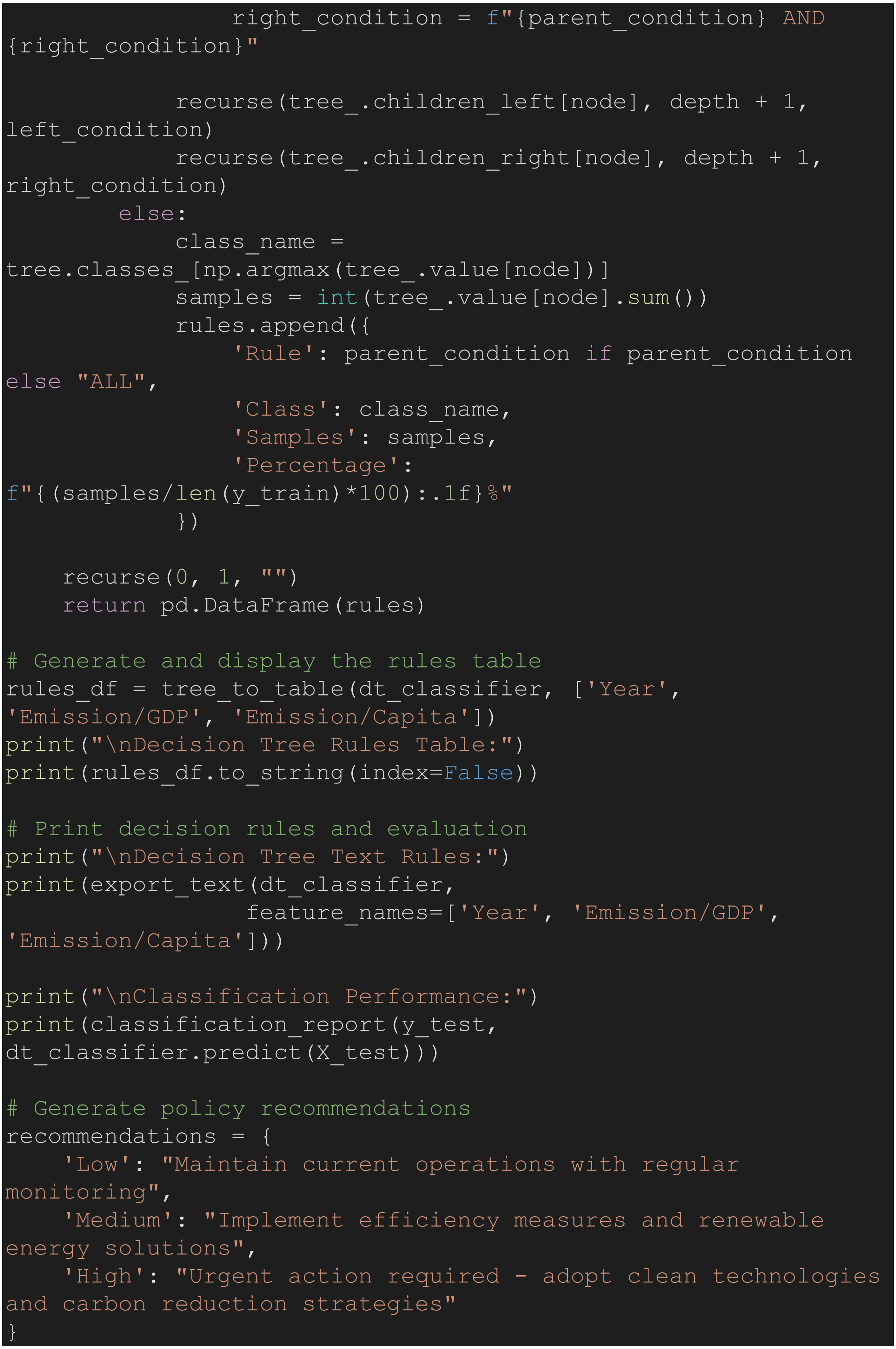
```python
# ===== 4. kNN Classification =====
features = [
    'Total CO2 Emissions (Million Tonnes)',
    'Coal CO2 Emissions (Million Tonnes)',
    'Gas CO2 Emissions (Million Tonnes)',
    'Oil CO2 Emissions (Million Tonnes)',
    'Flaring CO2 Emissions (Million Tonnes)',
    'Total GHG Emissions (Million Tonnes)',
    'Population'
]

X = data[features]
y = data['Emission_Class']
le = LabelEncoder()
y_encoded = le.fit_transform(y)

X_train, X_test, y_train, y_test = train_test_split(X,
y_encoded, test_size=0.2, random_state=42)

scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

knn = KNeighborsClassifier(n_neighbors=5)
knn.fit(X_train_scaled, y_train)
y_pred = knn.predict(X_test_scaled)

# ===== 5. Evaluation =====
print("\n Classification Report:\n")
print(classification_report(y_test, y_pred,
target_names=le.classes_))
print(f"\n Accuracy: {accuracy_score(y_test, y_pred):.2f}")

# ===== 6. Confusion Matrix =====
plt.figure(figsize=(6, 5))
sns.heatmap(confusion_matrix(y_test, y_pred), annot=True,
fmt='d', cmap='YlGnBu',
            xticklabels=le.classes_, yticklabels=le.classes_)
plt.title("Confusion Matrix")
plt.xlabel("Predicted")
plt.ylabel("Actual")
plt.tight_layout()
plt.show()

# ===== 7. Emission Trend Plot (Top 3 Highlighted) =====
plt.figure(figsize=(12, 6))
```

```
sector_totals =
data.groupby("Sector")["CO2_Emissions"].mean().sort_values(asc
ending=False)
top_sectors = sector_totals.head(3).index.tolist()

for sector in data["Sector"].unique():
    sector_data = data[data["Sector"] == sector]
    style = '-' if sector in top_sectors else '--'
    linewidth = 2.5 if sector in top_sectors else 1.2
    sns.lineplot(data=sector_data, x='Year',
y='CO2_Emissions', label=sector,
                 linestyle=style, linewidth=linewidth)

plt.title("CO₂ Emission Trends (Highlighting Top 3 Sectors)")
plt.ylabel("Emissions (Million Tonnes)")
plt.xlabel("Year")
plt.legend(bbox_to_anchor=(1.05, 1), loc='upper left')
plt.tight_layout()
plt.show()

# ===== 8. Heatmap: CO2 Intensity by Sector & Year =====
pivot = data.pivot_table(index='Sector', columns='Year',
values='CO2_Emissions', aggfunc='mean')
plt.figure(figsize=(14, 8))
sns.heatmap(pivot, cmap='YlOrRd', annot=True, fmt=".1f",
linewidths=0.5)
plt.title("CO₂ Emission Intensity by Sector and Year")
plt.xlabel("Year")
plt.ylabel("Sector")
plt.tight_layout()
plt.show()
```

Decision Tree

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.tree import DecisionTreeClassifier, export_text,
plot_tree, _tree
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report
from matplotlib.colors import ListedColormap

# Load the dataset
file_path = r"C:\Users\USER\Documents\UTP UNDERGRADUATES\CO2
Emission Dataset by Industry.csv"
```

```
df = pd.read_csv(file_path)

# Clean column names by stripping whitespace
df.columns = df.columns.str.strip()

# Prepare the data
df['GDP_USD'] = df['GDP (USD)'].str.replace('[$,]', '',
regex=True).astype(float)
df['Emission_per_GDP'] = df['CO2_Emissions'] / df['GDP_USD']
df['Emission_per_Capita'] = df['CO2_Emissions'] /
df['Population']

# Create the heatmap
plt.figure(figsize=(16, 9))
heatmap_data = df.pivot_table(values='CO2_Emissions',
                             index='Sector',
                             columns='Year',
                             aggfunc=np.mean)

sns.heatmap(heatmap_data,
            cmap='YlOrRd',
            annot=True,
            fmt=".1f",
            linewidths=.5,
            cbar_kws={'label': 'CO₂ Emissions (Million
Tonnes)'})
plt.title('Malaysia CO₂ Emissions by Sector and Year (1990-
2021)', fontsize=16, pad=20)
plt.xlabel('Year', fontsize=12)
plt.ylabel('Sector', fontsize=12)
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()

# Prepare data for classification
sector_yearly = df.groupby(['Sector', 'Year']).agg({
    'CO2_Emissions': 'mean',
    'Emission_per_GDP': 'mean',
    'Emission_per_Capita': 'mean'
}).reset_index()

# Create emission classes
percentiles = sector_yearly['CO2_Emissions'].quantile([0.33,
0.66]).values
sector_yearly['Emission_Class'] =
pd.cut(sector_yearly['CO2_Emissions'],
                                       bins=[-np.inf,
percentiles[0], percentiles[1], np.inf],
                                       labels=['Low',
'Medium', 'High'])

# Train Decision Tree classifier
X = sector_yearly[['Year', 'Emission_per_GDP',
'Emission_per_Capita']]
y = sector_yearly['Emission_Class']

X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)

dt_classifier = DecisionTreeClassifier(
    max_depth=4,
    min_samples_split=5,
    random_state=42
)
dt_classifier.fit(X_train, y_train)

# Add predictions to dataframe
sector_yearly['Predicted_Class'] = dt_classifier.predict(X)

# Create combined heatmap with classification overlay
plt.figure(figsize=(16, 9))
sns.heatmap(heatmap_data,
            cmap='YlOrRd',
            annot=True,
            fmt=".1f",
            linewidths=.5,
            cbar_kws={'label': 'CO₂ Emissions (Million
Tonnes)'},
            alpha=0.7)

# Overlay classification labels
class_data = sector_yearly.pivot(index='Sector',
                                columns='Year',
                                values='Predicted_Class')

for i, sector in enumerate(class_data.index):
    for j, year in enumerate(class_data.columns):
        if year in heatmap_data.columns:
            class_val = class_data.loc[sector, year]
            color = {'Low': 'green', 'Medium': 'orange',
'High': 'red'}[class_val]
            plt.text(list(heatmap_data.columns).index(year) +
0.5,
                    i + 0.5,
                    class_val[0],
                    ha='center',
                    va='center',
                    color=color,
                    fontsize=10,
                    fontweight='bold')

plt.title('Malaysia CO₂ Emissions with Sector Classification
(1990-2021)', fontsize=16, pad=20)
plt.xlabel('Year', fontsize=12)
plt.ylabel('Sector', fontsize=12)
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()

# Enhanced Decision Tree Visualization
plt.figure(figsize=(20,12))
plot_tree(dt_classifier,
          feature_names=['Year', 'Emission/GDP',
'Emission/Capita'],
          class_names=['Low', 'Medium', 'High'],
          filled=True,
          rounded=True,
          proportion=True,
          precision=2,
          fontsize=10)
plt.title("Decision Tree Structure for CO₂ Emission
Classification", fontsize=16)
plt.show()

# Create a readable rules table
def tree_to_table(tree, feature_names):
    tree_ = tree.tree_
    feature_name = [
        feature_names[i] if i != _tree.TREE_UNDEFINED else
"undefined!"
        for i in tree_.feature
    ]

    rules = []
    def recurse(node, depth, parent_condition):
        if tree_.feature[node] != _tree.TREE_UNDEFINED:
            name = feature_name[node]
            threshold = tree_.threshold[node]
            left_condition = f"{name} <= {threshold:.4f}"
            right_condition = f"{name} > {threshold:.4f}"

            if parent_condition:
                left_condition = f"{parent_condition} AND
{left_condition}"
```



```
                right_condition = f"{parent_condition} AND
{right_condition}"

            recurse(tree_.children_left[node], depth + 1,
left_condition)
            recurse(tree_.children_right[node], depth + 1,
right_condition)
        else:
            class_name =
tree.classes_[np.argmax(tree_.value[node])]
            samples = int(tree_.value[node].sum())
            rules.append({
                'Rule': parent_condition if parent_condition
else "ALL",
                'Class': class_name,
                'Samples': samples,
                'Percentage':
f"{(samples/len(y_train)*100):.1f}%"
            })

    recurse(0, 1, "")
    return pd.DataFrame(rules)

# Generate and display the rules table
rules_df = tree_to_table(dt_classifier, ['Year',
'Emission/GDP', 'Emission/Capita'])
print("\nDecision Tree Rules Table:")
print(rules_df.to_string(index=False))

# Print decision rules and evaluation
print("\nDecision Tree Text Rules:")
print(export_text(dt_classifier,
                 feature_names=['Year', 'Emission/GDP',
'Emission/Capita']))

print("\nClassification Performance:")
print(classification_report(y_test,
dt_classifier.predict(X_test)))

# Generate policy recommendations
recommendations = {
    'Low': "Maintain current operations with regular
monitoring",
    'Medium': "Implement efficiency measures and renewable
energy solutions",
    'High': "Urgent action required - adopt clean technologies
and carbon reduction strategies"
}
```

```
latest_year = df['Year'].max()
print(f"\nPolicy Recommendations for {latest_year}:")
latest_data = sector_yearly[sector_yearly['Year'] ==
latest_year][['Sector', 'Predicted_Class']]
latest_data['Recommendation'] =
latest_data['Predicted_Class'].map(recommendations)
print(latest_data.to_string(index=False))
```

Random Forest

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
from sklearn.model_selection import TimeSeriesSplit
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report,
accuracy_score
from sklearn.preprocessing import LabelEncoder
import os


# 1. DATA PREPARATION

# Use raw string for Windows paths or forward slashes
file_path = os.path.join('C:', os.sep, 'Users', 'USER',
'Documents', 'UTP UNDERGRADUATES', 'CO2 Emission Dataset by
Industry.csv')
data = pd.read_csv(file_path)

# Clean data
data.columns = data.columns.str.strip()
data = data.dropna(subset=['Year', 'Sector', 'Total CO2
Emissions (Million Tonnes)'])

# Create binary target
median_emission = data['Total CO2 Emissions (Million
Tonnes)'].median()
data['Emission_Level'] = np.where(
    data['Total CO2 Emissions (Million Tonnes)'] >
median_emission,
    'High', 'Low'
)


# 2. VISUALIZATIONS

# Set style properly
```

```
plt.style.use('seaborn-v0_8')  # Modern seaborn style
plt.rcParams['figure.dpi'] = 300

# Line Chart
plt.figure(figsize=(14, 7))
sns.lineplot(
    data=data,
    x='Year',
    y='Total CO2 Emissions (Million Tonnes)',
    hue='Sector',
    style='Sector',
    markers=True,
    dashes=False,
    linewidth=2,
    palette='tab20'
)
plt.title('Sector-wise CO₂ Emissions Over Time', fontsize=16)
plt.legend(bbox_to_anchor=(1.05, 1), loc='upper left')
plt.tight_layout()
plt.show()  # Display directly instead of saving

# Heatmap
plt.figure(figsize=(12, 8))
heatmap_data = data.pivot_table(
    index='Sector',
    columns='Year',
    values='Total CO2 Emissions (Million Tonnes)',
    aggfunc='mean'
)
sns.heatmap(
    heatmap_data,
    annot=True,
    fmt=".1f",
    cmap="YlOrRd",
    linewidths=0.5,
    cbar_kws={'label': 'CO₂ Emissions (Million Tonnes)'}
)
plt.title('Sector Emissions by Year', fontsize=16)
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()


# 3. MODEL EVALUATION

# Feature Engineering
data['Sector_Encoded'] =
LabelEncoder().fit_transform(data['Sector'])
```

```
data['Emission_Growth'] = data.groupby('Sector')['Total CO2
Emissions (Million Tonnes)'].pct_change()

# Prepare features
X = data[['Year', 'Sector_Encoded', 'Total CO2 Emissions
(Million Tonnes)', 'Emission_Growth']].fillna(0)
y = data['Emission_Level']

# Time-based split
data = data.sort_values('Year')
test_size = int(0.2 * len(data))
X_train, X_test = X[:-test_size], X[-test_size:]
y_train, y_test = y[:-test_size], y[-test_size:]

# Train Random Forest
rf = RandomForestClassifier(
    n_estimators=100,
    max_depth=3,
    min_samples_split=10,
    random_state=42
)
rf.fit(X_train, y_train)

# Evaluate
y_pred = rf.predict(X_test)
print("=== Random Forest Evaluation ===")
print(classification_report(y_test, y_pred))
print(f"\nAccuracy: {accuracy_score(y_test, y_pred):.2f}")

# Feature Importance
importance = pd.DataFrame({
    'Feature': X.columns,
    'Importance': rf.feature_importances_
}).sort_values('Importance', ascending=False)
print("\n=== Feature Importance ===")
print(importance)

# Learning Curve
from sklearn.model_selection import learning_curve
train_sizes, train_scores, test_scores = learning_curve(
    rf, X, y, cv=TimeSeriesSplit(n_splits=3),
    scoring='accuracy', n_jobs=-1
)

plt.figure(figsize=(10, 6))
plt.plot(train_sizes, np.mean(train_scores, axis=1),
label='Training')
```

```
plt.plot(train_sizes, np.mean(test_scores, axis=1),
label='Validation')
plt.title('Random Forest Learning Curve')
plt.xlabel('Training Examples')
plt.ylabel('Accuracy')
plt.legend()
plt.show()
```