Microclimate Weather Prediction for Ottawa

Heymans, Adrien
300092260
University of Ottawa

Supervisors:
Professors Iluju Kiringa and Tet Yeap

August 2024

Contents

1 Introduction

1.1 Motivation

Agriculture plays an important role in the economy, particularly for regions heavily reliant on farming such as the Ottawa region (4). In such settings, accurate weather forecasting is essential, as it directly impacts crop yield predictions. Farmers often pre-sell their crops to distributors before the planting season begins, committing to specific quantities at agreed prices (2). This pre-selling process requires precise estimates of future yields to ensure economic profitability.

Weather conditions, including temperature, precipitation, and solar radiation, influence crop growth and yield. However, weather patterns can vary widely even within relatively small geographical areas, creating microclimates that can affect farming outcomes. For instance, the weather in different parts of Ottawa can vary enough to impact local farms differently. This variability means that generalized weather forecasts are insufficient for effective crop planning and yield prediction.

The importance of microclimate predictions becomes evident when considering the financial risks involved for farmers. Inaccurate weather forecasts can lead to substantial economic losses. If a farmer overestimates their crop yield, they might pre-sell more produce than they can harvest, resulting in penalties and financial shortfalls. The opposite is also true, underestimating yield can cause farmers to sell surplus crops at lower market prices, missing potential profits.

Given these challenges, developing techniques capable of predicting weather at a microclimate level can provide significant benefits. Accurate microclimate forecasts enable farmers to make informed decisions about crop management, optimize yield predictions, and enhance financial planning. This technological advancement supports sustainable agricultural practices and economic resilience for farmers, contributing to overall food security and market stability.

Therefore, the motivation behind this project is to harness the potential of artificial intelligence models to deliver precise weather forecasts tailored to microclimates. This capability will empower farmers with the data needed to predict crop yields accurately, ultimately improving their economic outcomes and reducing the risks associated with agricultural production.

1.2 Problem Statement

The primary problem addressed in this project is the lack of accurate and localized weather forecasts for farmers in the Ottawa region. Traditional weather forecasting models provide general predictions over broad areas, which are insufficient for the microclimate variability experienced in different parts of Ottawa. This variability can impact crop growth and yield, leading to economic uncertainties for farmers who rely on precise weather data for pre-selling crops and planning agricultural activities.

Farmers need reliable, localized weather predictions to optimize their crop yield predictions. The existing generalized weather models do not account for the fine-scale weather variations within small geographical areas. Consequently, farmers face challenges in making informed decisions about fertilizing, planting, harvesting, and other critical activities that depend on accurate weather forecasts. The inability to predict localized weather patterns accurately can result in either overestimating or underestimating crop yields, both of which carry significant financial risks.

The goal of this project is to compare machine learning models that can provide accurate weather forecasts at a microclimate level, specifically to support the needs of farmers in the Ottawa region. This involves comparing different types of mnodels such as 1-Dimensional Convolutional Neural Networks (1D-CNNs) or Long-Short term memory models (LSTMs) to analyze historical weather data and predict future weather patterns. By improving the precision of weather forecasts, the project aims to enhance farmers' ability to make informed decisions, thereby reducing economic risks and promoting sustainable agricultural practices.

1.3 Outline of the Solution

The solution to predict weather metrics involved a structured and methodical approach, including data collection, preprocessing, model experimentation, optimization, training, and evaluation. This section provides a summary of the steps taken to develop an effective predictive model.

The initial step in the process was data collection from Environment Canada, which provided comprehensive weather data for the Ottawa region, recorded from multiple weather stations over several years. The collected raw data underwent preprocessing to ensure its quality and suitability for modeling. This included handling missing values through interpolation and mean imputation, as well as scaling the data to normalize the range of different metrics using techniques such as MinMaxScaler.

Following the data preprocessing, the focus shifted to model experimentation. Different types of models were tested to identify the most effective approach for predicting weather metrics. Among the models considered were Long Short-Term Memory (LSTM) networks, which are well-suited for sequential data due to their ability to capture temporal dependencies (15), and one-dimensional Convolutional Neural Networks (1D-CNNs), known for their capability to detect local patterns in time series data (**?**).

To further enhance the robustness and generalization capabilities of the models, we incorporated data randomization into the training process. By randomizing the training data sequences, we aimed to assess and improve the models' adaptability to variations in data distribution. This step was crucial for evaluating the models' performance under different scenarios and ensuring their reliability for real-world applications.

To optimize the performance of these models, a hyperparameter tuning process was conducted. This involved adjusting various parameters such as the number of layers, learning rate, batch size, and number of epochs. Techniques like grid search and random search were employed to find the best combinations of hyperparameters, ensuring optimal model performance.

The training phase was iterative, involving repeated training and evaluation of the models. During this process, several challenges were encountered, including initial model predictions that averaged out the data instead of capturing the actual spikes. To address these issues, the model architectures were refined, and hyperparameters were adjusted accordingly. The goal was to minimize validation loss during training, thus ensuring the model's ability to generalize well to unseen data.

After persistent optimization, some models were developed that performed well across multiple metrics. The final models demonstrated satisfactory performance, accurately predicting temperature, solar radiation but struggling to accurately predict precipitation, and average hourly pressure. This implementation highlights the efficacy of the chosen approach and the robustness of the solution.

1.4 Methodology

This section details the methodology employed in the development of the weather prediction model. It encompasses the preliminary literature review, data collection, data preprocessing, model selection, experimentation, and evaluation processes.

Literature Review Before starting the practical aspects of the project, a thorough literature review was conducted. This involved reviewing existing research and methodologies in the field of weather prediction using machine learning models. The review provided insights into the advantages and limitations of various modeling techniques, including Long Short-Term Memory (LSTM) networks and 1-Dimensional Convolutional Neural Networks (1D-CNNs). The findings from the literature review informed the selection of models and preprocessing techniques used in this project.

Data Collection The primary data source for this project was the weather dataset from Environment Canada, a trusted and accurate source for environmental data. The dataset includes various weather metrics recorded at 1-day intervals, covering the period from January 1996 to the present. This dataset was chosen due to its high resolution and reliability.

Data Preprocessing Data preprocessing was a crucial step to ensure the quality and suitability of the data for modeling. The preprocessing steps included:

- Handling missing values through interpolation and mean imputation to ensure continuity and completeness of the data.
- Scaling the data using the MinMaxScaler technique to normalize the range of different metrics, thereby facilitating more efficient model training.
- Creating synthetic features such as minimum, maximum, and average values for specific periods to enhance the model's predictive capabilities.

Model Selection and Experimentation The project involved experimenting with different types of models to identify the most effective approach for weather prediction. The models considered included:

- Long Short-Term Memory (LSTM) networks: These recurrent neural networks are well-suited for sequential data and were tested for their ability to capture temporal dependencies in the weather data.
- One-dimensional Convolutional Neural Networks (1D-CNNs): These models were explored for their capability to detect local patterns in the time series data.
- Baseline Model: In addition to these advanced models, a baseline model was also used to provide a comparative benchmark for performance evaluation. This model included simple mean predictors over the previous n_{past} days, offering a reference point to measure the improvements achieved by the more complex architectures.

All experiments were conducted using well-documented and public libraries such as TensorFlow and Keras. These libraries provided robust tools and functions that facilitated the implementation and testing of various models, the in-depth requirements can be found in section 3.4. The experimentation phase involved extensive trial and error to fine-tune the models and achieve optimal performance.

Data Randomization To further assess the robustness and generalization capabilities of the models, we incorporated data randomization into the training process. This involved randomizing the training data to evaluate how well the models adapted to variations in data distribution. Randomization was crucial for ensuring the models could generalize well to unseen data and handle the variability present in real-world weather patterns.

Hyperparameter Tuning To optimize the performance of the models, a hyperparameter tuning process was carried out using Keras (1). This involved:

- Adjusting parameters such as the number of layers, learning rate, batch size, and number of epochs.
- Using techniques like grid search and random search to find the best combinations of hyperparameters.

Hyperparameter tuning was essential to enhance the models' accuracy and generalization capabilities.

Model Training and Evaluation The training phase was iterative, involving repeated training and evaluation of the models. Several issues were encountered during this process, including models initially predicting macro averages instead of capturing the actual spikes in the data. These issues were addressed by refining the model architectures and adjusting the hyperparameters. The goal was to minimize the validation loss during training, ensuring the model's ability to generalize well to unseen data. The models were evaluated based on their performance in predicting key weather metrics, including temperature, solar radiation, precipitation, average hourly pressure, and dew point.

The methodology for this project involved a structured approach, beginning with a comprehensive literature review, followed by meticulous data collection and preprocessing. The experimentation with various models, combined with rigorous hyperparameter tuning, iterative training, and the incorporation of data randomization, led to the development of weather prediction models. The use of accurate and trusted datasets, along with well-documented public libraries, ensured the reliability and reproducibility of the results.

1.5 Summary of Contributions and Findings

The results of this project demonstrate advancements in weather prediction using LSTM and 1D-CNN architectures. Our experiments evaluated model performance through key metrics: Mean Squared Error (MSE), Mean Absolute Error (MAE), and R-squared (R2). We conducted these experiments with various combinations of past days (n_{past}) and future days (n_{future}) to identify optimal prediction settings.

Our findings, illustrated in the distribution of MSE, MAE, and R2 (Figures 16, 17, and 18), indicate that 1D-CNN models generally exhibit lower error variability compared to LSTM models for most weather features. Specifically, 1D-CNN models showed superior performance in predicting average hourly temperature and solar radiation, evidenced by consistently lower MSE and MAE values. However, both models faced challenges in accurately predicting precipitation, as reflected in the similar performance and low R2 values for this feature.

The comparative analysis of MSE, MAE, and R2 between LSTM and 1D-CNN models (Figures 23, 22, and 21) further shows the 1D-CNN models' better performance in temperature and solar radiation predictions. Both models performed comparably in predicting average hourly pressure, with LSTM models slightly outperforming 1D-CNN models in precipitation prediction in terms of MSE.

The heatmap for R2 (Figure 20) revealed that the optimal performance is achieved with a higher number of past and future days. The combination of 90 past days and 30 future days consistently yielded the highest MAE and MSE values, suggesting that using too few past days can negatively impact model performance.

Visual comparisons of actual vs. predicted values (Figures 24 and 25) highlighted that while both models could capture general trends, significant deviations occurred, particularly for precipitation. The 1D-CNN model provided a slightly better fit for average hourly temperature and solar radiation.

The introduction of data randomization in training was another critical aspect of our study. The effect of randomization on model performance was analyzed through MAE, MSE, and R2 metrics (Figures 27, 28, and 26). Our results showed that randomization increased variability in model performance, particularly for LSTM models, which were more sensitive to changes in data distribution. Despite the variability, randomization helped in enhancing the model's robustness to unseen data.

Overall, the 1D-CNN models generally outperformed LSTM models in predicting average hourly temperature and solar radiation, while both models struggled with precipitation. The impact of data randomization provided insights into model sensitivity and generalization capabilities. These findings offer valuable insights for enhancing weather prediction models and guide future improvements in the field.

1.6 Structure of the Report

This report begins with an introduction that outlines the motivation for the project, clearly defines the problem being addressed, summarizes the approach to solving the problem, and highlights the key contributions and findings. Following the introduction, the related work section reviews and discusses existing research and solutions relevant to weather prediction and microclimate forecasting, establishing the context for this project and highlighting the gaps that this work aims to fill.

The approach section provides a comprehensive explanation of the methodology, detailing the solution approach, the dataset used, requirements and specifications, system design, experimental setup, and model description. This section aims to give a thorough understanding of the steps taken and the reasoning behind them.

The results section presents the outcomes of the experiments, including visual representations of the model's performance, a detailed explanation of the results, and a discussion interpreting these results and their implications. This discussion highlights the strengths and limitations of the model, providing a clear picture of its effectiveness and areas for improvement.

Finally, the conclusion section summarizes the findings and contributions of the project, discusses any limitations encountered, and outlines potential future work. This final section aims to consolidate the insights gained from the project and suggest directions for further enhancement of the model and its applications.

2 Related Work

2.1 Introduction to IoT and Smart Farming

The Internet of Things (IoT) has revolutionized various sectors, including agriculture, by integrating devices that communicate, sense, and interact with their environment through embedded technology. Smart farming, an application of IoT, involves using these technologies to monitor and manage agricultural processes more efficiently. It includes deploying IoT sensors to collect data on soil conditions, weather, crop health, and other factors essential for optimizing farming practices. The data collected from these sensors can be analyzed to provide actionable insights, enabling farmers to make informed decisions and improve crop yields (3).

2.2 Benefits of Smart Farming

Smart farming is used for several scenarios:

- **Enhanced Productivity:** It enables precision agriculture, which efficiently uses resources such as water, fertilizers, and pesticides. This targeted approach reduces waste and environmental impact while increasing crop yield (3).
- **Pest and Disease Monitoring:** Smart farming helps in monitoring and controlling pest and disease outbreaks, thus protecting crops and reducing losses (12).

2.3 Weather Monitoring and Predictions in Agriculture

Weather conditions are one of the most significant factors affecting agricultural productivity. Accurate weather predictions are essential for planning agricultural activities, such as planting, irrigation, and harvesting. IoT technologies play a vital role in weather monitoring by using weather stations and sensors to collect data on temperature, humidity, wind speed, and other climatic parameters. This data is then used to predict weather patterns and provide timely warnings to farmers, helping them mitigate the adverse effects of extreme weather conditions on crops (3).

2.4 Linking Weather Predictions with Crop Yield Forecasting

Linking weather predictions with crop yield forecasting is an important aspect of smart farming. Various studies have highlighted the importance of integrating weather data with crop models to predict yield outcomes. Advanced machine learning algorithms and data analytics techniques are used to analyze historical weather data and predict future climatic conditions, which in turn helps in forecasting crop yields. This integration enables farmers to make better decisions regarding crop selection, planting schedules, and resource allocation, enhancing agricultural productivity (13).

2.5 Micro-Climate Forecasting

The focus on weather predictions for micro-climates is particularly important in regions with diverse climatic conditions. Micro-climate forecasting provides localized weather information, which is more relevant for farmers in specific areas. By utilizing IoT sensors and weather stations, farmers can receive accurate and timely weather forecasts tailored to their specific location, allowing for more precise and effective farming practices (3).

2.6 Case Studies and Applications

Several case studies demonstrate the successful implementation of IoT technologies in agriculture:

- **Precision Farming:** A survey on IoT applications in smart farming highlights various use cases, including precision farming, greenhouse monitoring, and livestock management. These applications leverage IoT devices to collect and analyze data, providing farmers with valuable insights to enhance productivity and sustainability (11).
- **Wireless Sensor Networks:** Another study discusses the role of IoT in precision agriculture, focusing on the use of wireless sensor networks to monitor soil and weather conditions. The study emphasizes the benefits of real-time data collection and analysis in optimizing irrigation and fertilization practices, reducing costs, and improving crop yields (10).

2.7 Introduction to LSTM and 1D-CNN Models

The Long Short-Term Memory (LSTM) neural network is a type of recurrent neural network (RNN) designed to overcome the limitations of traditional RNNs, specifically the vanishing gradient problem. LSTMs are capable of learning long-term dependencies, making them suitable for time series prediction tasks. LSTMs use gates to regulate the flow of information, allowing them to retain relevant information over long periods and forget irrelevant data. This ability makes LSTMs particularly effective for sequential data analysis, such as weather prediction (7).

The 1D Convolutional Neural Network (1D-CNN) is another deep learning model used for time series analysis. Unlike traditional CNNs used for image data, 1D-CNNs apply convolutional operations along one dimension, making them suitable for sequential data. 1D-CNNs are effective in capturing local patterns in time series data, which can be crucial for tasks like weather forecasting (9).

2.8 LSTMs in Weather Prediction

LSTMs have been widely used for weather prediction due to their ability to model temporal dependencies. In the study *Weather Prediction Using LSTM Neural Networks* by Srivastava and Anto (2022), the authors propose using LSTMs to predict weather conditions. They demonstrate that LSTMs can effectively handle the highly dynamic and complex nature of weather data. By incorporating optimizations such as Gaussian and Median filtering, the model achieves better long-term accuracy, addressing the limitations of traditional numerical weather prediction methods (14).

Another significant application of LSTMs in weather prediction is highlighted in a study by Karevan and Suykens (2020), where a transductive LSTM model is used for time series prediction. This model shows substantial performance improvements in weather forecasting by leveraging LSTM's capability to record long-term dependencies. The study emphasizes the potential of LSTMs in handling the uncertainty and complexity inherent in weather data (8).

2.9 1D-CNNs in Weather Prediction

1D-CNNs have also been successfully applied to weather prediction. The survey *A Comprehensive Survey on 1D Convolutional Neural Networks* provides an overview of various applications of 1D-CNNs, including their use in time series analysis and weather prediction. The survey highlights the strengths of 1D-CNNs in capturing local patterns and their computational efficiency compared to traditional methods (9).

In the study *An Intelligent Weather Prediction Model Using Optimized 1D CNN with Attention GRU*, Hemamalini et al. (2024) propose a hybrid model that combines 1D-CNN with an attention mechanism and

Gated Recurrent Unit (GRU). This approach captures both local and temporal features of weather data, leading to improved prediction accuracy. The study uses the Jena Climate dataset and demonstrates that the proposed model outperforms traditional methods in terms of Mean Squared Error (MSE) and Root Mean Squared Error (RMSE) (5).

3 Approach

3.1 Description of the Approach

The approach taken for this project involves using One-dimensional Convolutional Neural Networks (1D-CNNs) to predict weather patterns. 1D-CNNs are chosen due to their ability to capture spatial hierarchies in data through convolutional layers. This makes them particularly effective for time-series data, where patterns over time are essential for accurate prediction.

3.2 Data

3.2.1 Data Collection

The dataset used in this study is sourced from Environment Canada and is made available through the WeatherStats platform. Environment Canada operates numerous weather stations across Canada, including around 25 stations in the Ottawa region. These stations are equipped with sensors to record various meteorological parameters such as temperature, precipitation, pressure, and solar radiation. The dataset from WeatherStats combines data from multiple nearby weather stations, ensuring completeness and consistency by integrating records from discontinued stations.

3.2.2 Dataset Description

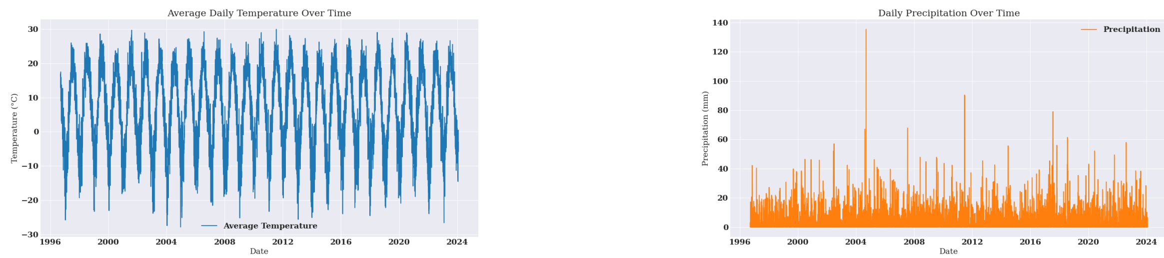
The dataset spans multiple years and includes the following key variables:

- `date` - Date of the observation
- `max_temperature` - Maximum daily temperature
- `avg_temperature` - Average daily temperature
- `min_temperature` - Minimum daily temperature
- `precipitation` - Daily precipitation
- `avg_pressure_sea` - Average daily sea-level pressure
- `solar_radiation` - Daily solar radiation
- `avg_relative_humidity` - Average daily relative humidity

This dataset provides a comprehensive view of the weather patterns in the Ottawa region, which is essential for developing accurate weather prediction models.

3.3 Data Analysis and Visualization

To understand the trends and relationships within the data, we generated several plots, which are described and analyzed below.



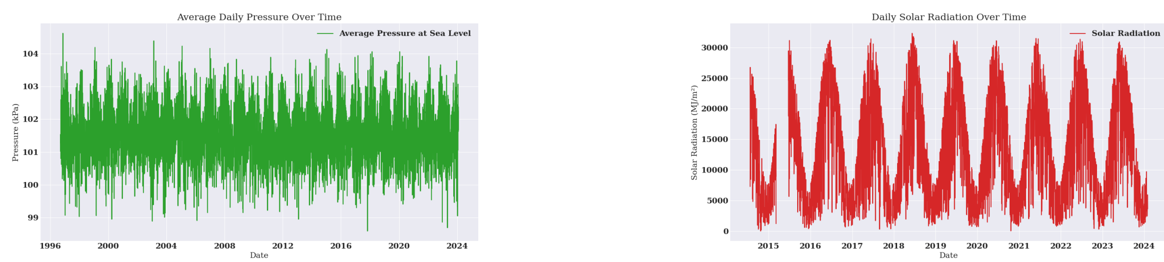
(a) Average Daily Temperature Over Time

(b) Daily Precipitation Over Time

Figure 1: Temperature and Precipitation Data Over Time

Figure 1a shows the time series of average daily temperature. The data exhibits clear seasonal patterns, with higher temperatures in the summer months and lower temperatures in the winter months. This cyclic nature is typical for temperate regions like Ottawa.

Figure 1b presents the time series of daily precipitation. This plot reveals periods of high precipitation interspersed with drier spells, highlighting the variability in rainfall and snowfall over the years.



(a) Average Daily Pressure Over Time

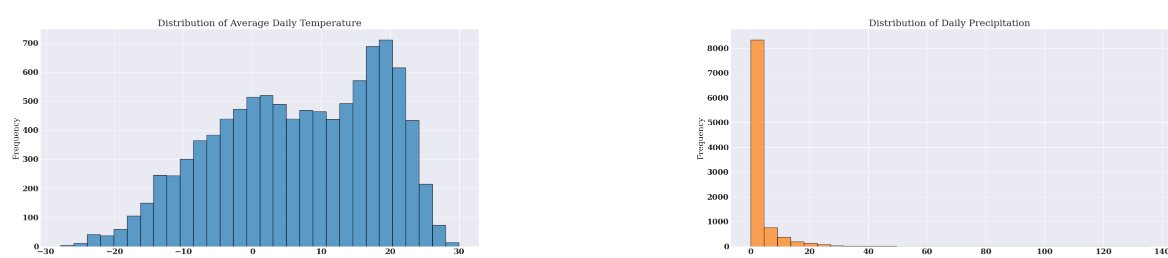
(b) Daily Solar Radiation Over Time

Figure 2: Pressure and Solar Radiation Data Over Time

Figure 2a shows the average daily pressure at sea level. The fluctuations in atmospheric pressure can indicate the passage of weather systems, such as high and low-pressure areas.

Figure 2b depicts the time series of daily solar radiation. The plot shows higher values during the summer months and lower values during the winter months, reflecting the seasonal variation in daylight hours and solar intensity.

subcaption



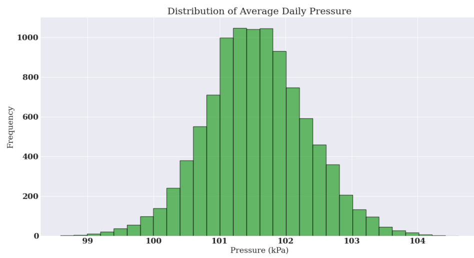
(a) Distribution of Average Daily Temperature

(b) Distribution of Daily Precipitation

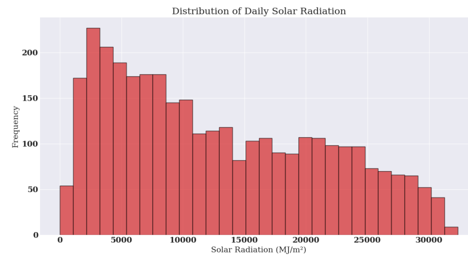
Figure 3: Temperature and Precipitation Distributions

The histogram of average daily temperature in Figure 3a illustrates the frequency distribution of temperatures. Most days fall within a moderate temperature range, with fewer days experiencing extreme temperatures.

Figure 3b shows the histogram of daily precipitation, indicating that most days have little to no precipitation, while a smaller number of days experience significant precipitation events.



(a) Distribution of Average Daily Pressure



(b) Distribution of Daily Solar Radiation

Figure 4: Pressure and Solar Radiation Distributions

The histogram of average daily pressure in Figure 4a provides insights into the range and common values of atmospheric pressure experienced in the Ottawa region.

Figure 4b displays the frequency distribution of solar radiation values, showing that most days receive moderate solar radiation, with fewer days experiencing very high or very low values.

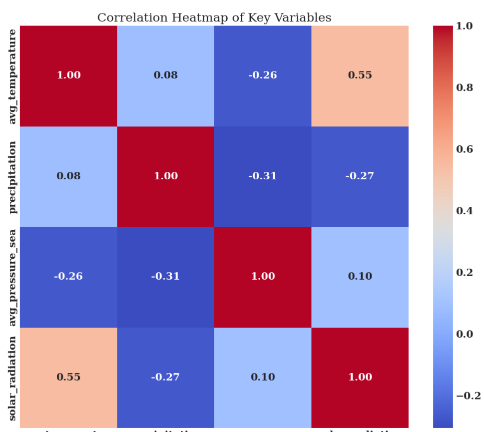
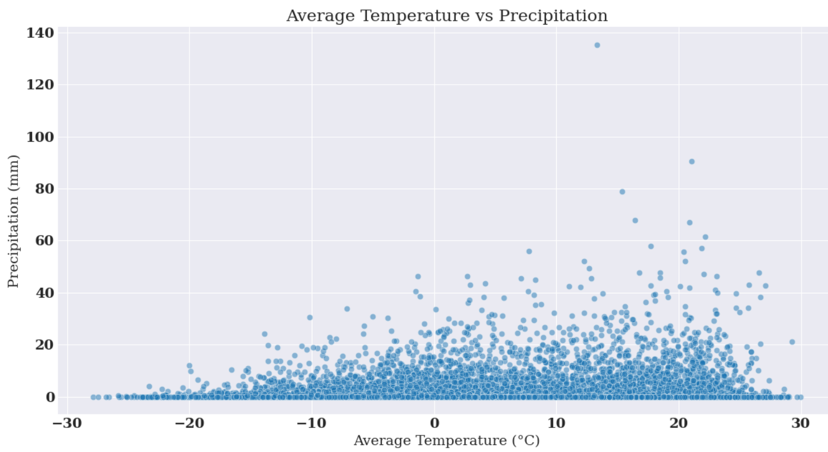
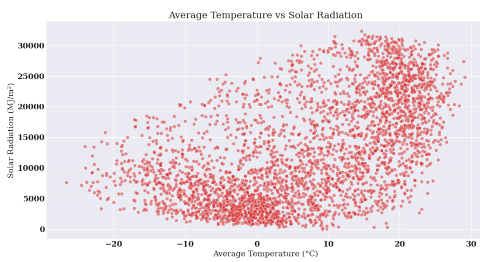


Figure 5: Correlation Heatmap of Key Variables

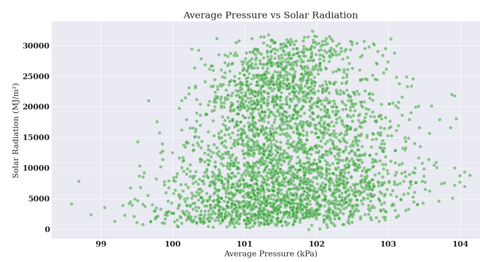
The correlation heatmap in Figure 5 illustrates the relationships between key variables. Strong correlations are observed between certain variables, indicating how they interact and influence each other.



(a) Average Temperature vs Precipitation



(b) Average Temperature vs Solar Radiation



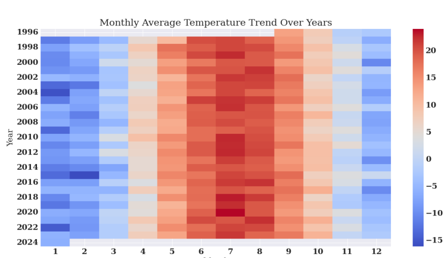
(c) Average Pressure vs Solar Radiation

Figure 6: Scatter Plots of Weather Data Relationships

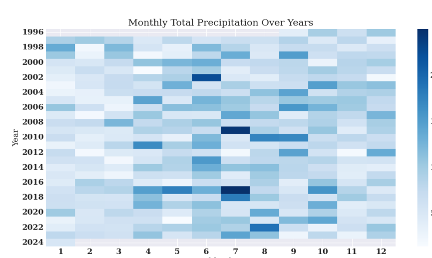
Figure 6a shows the scatter plot of average temperature versus precipitation. This plot helps in understanding the relationship between temperature and precipitation, revealing patterns that could be useful for predictive modeling.

Figure 6b shows a positive relationship between average temperature and solar radiation, as expected, since higher solar radiation typically leads to higher temperatures.

The scatter plot in Figure 6c explores the relationship between average pressure and solar radiation, providing insights into how different weather systems correlate with solar energy received.



(a) Monthly Average Temperature Trend Over Years



(b) Monthly Total Precipitation Over Years

Figure 7: Monthly Temperature and Precipitation Trends

Figure 7a visualizes the monthly average temperature trend over years, highlighting long-term trends and seasonal variations. We can observe that since 1996, the average temperature has remained somewhat stable.

Figure 7b shows the total precipitation each month across different years, revealing trends in rainfall and snowfall patterns.

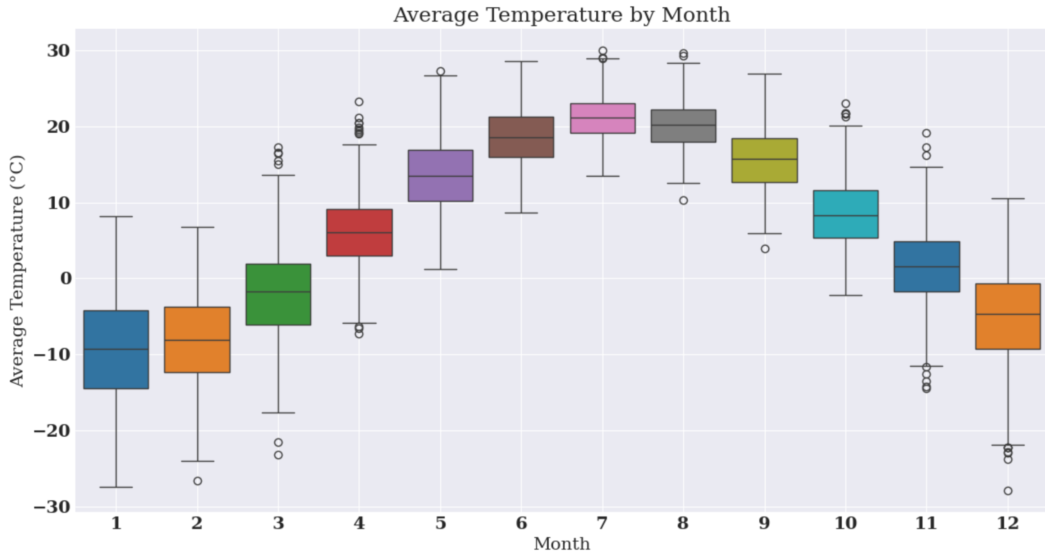


Figure 8: Average Temperature by Month

The box plot in Figure 8 illustrates the distribution of average temperatures for each month, highlighting the variability and typical temperature range for each month.

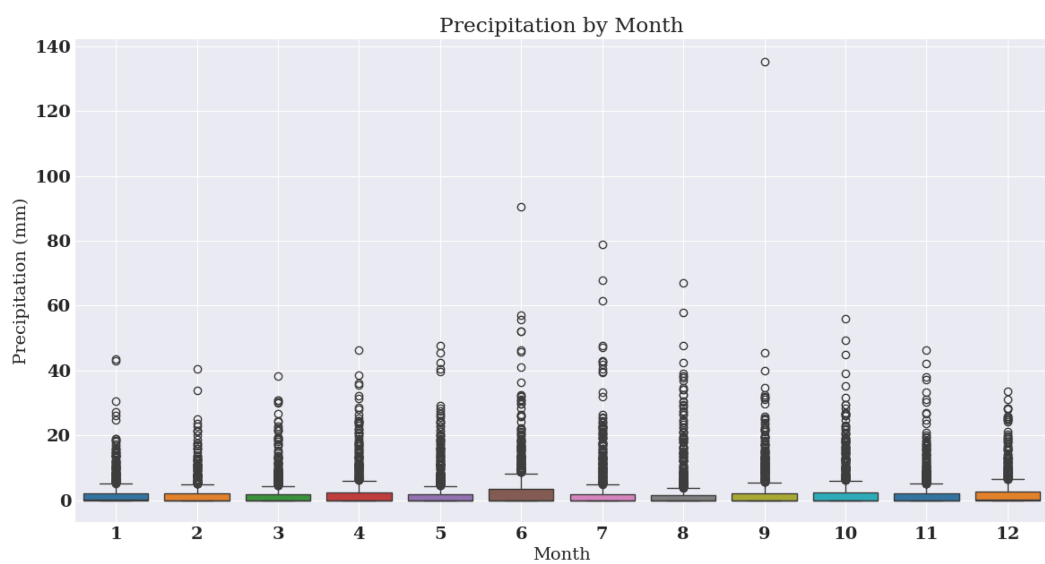


Figure 9: Precipitation by Month

Figure 9 shows the distribution of precipitation amounts for each month, providing insights into the typical range of precipitation and identifying months with higher variability.

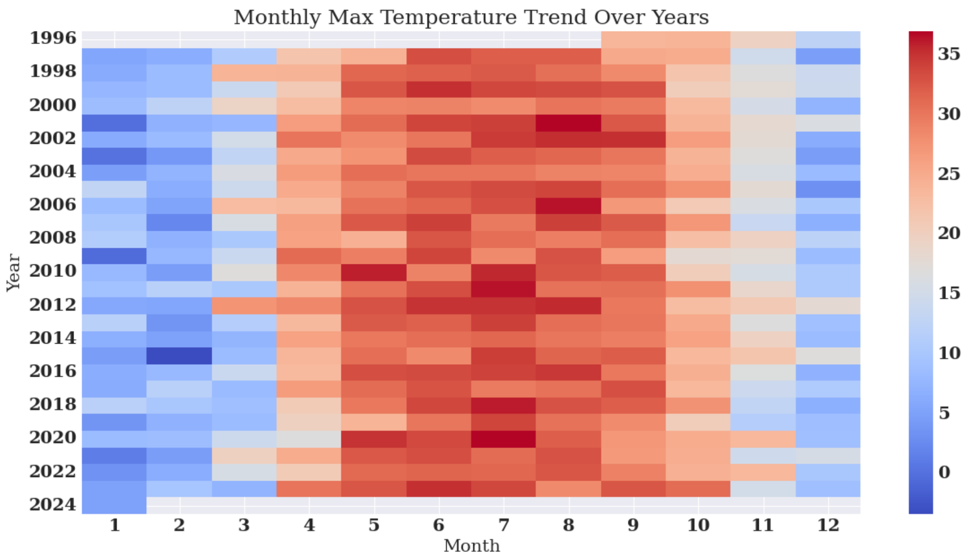


Figure 10: Monthly Max Temperature Trend Over Years

Figure 10 highlights how maximum temperatures have changed each month across different years, indicating periods of extreme heat and trends in maximum temperature over time.

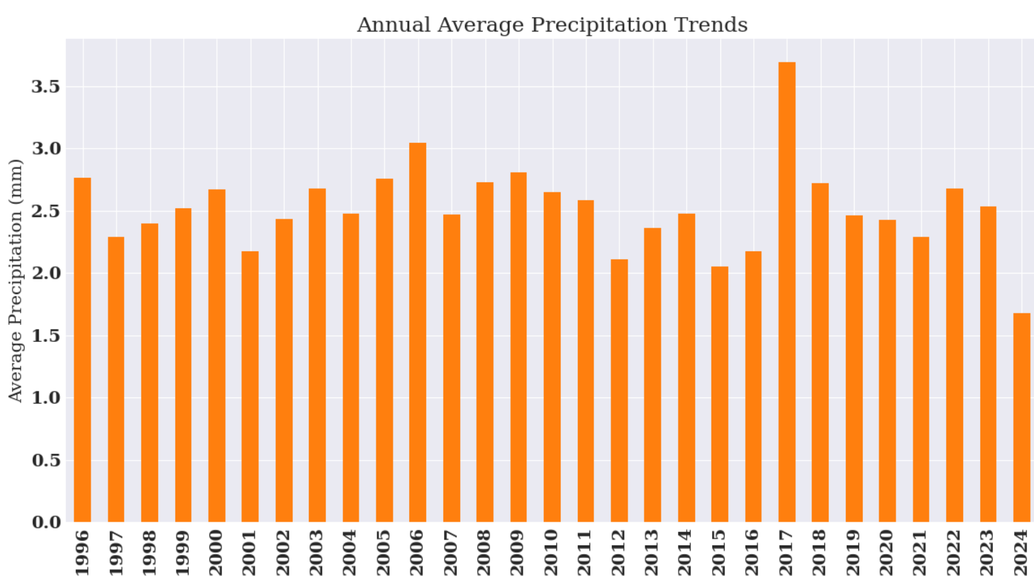


Figure 11: Annual Average Precipitation Trends

The bar plot in Figure 11 shows the average precipitation for each year, revealing trends in rainfall and snowfall over time. This can indicate periods of increasing or decreasing precipitation and help identify potential climate change impacts.

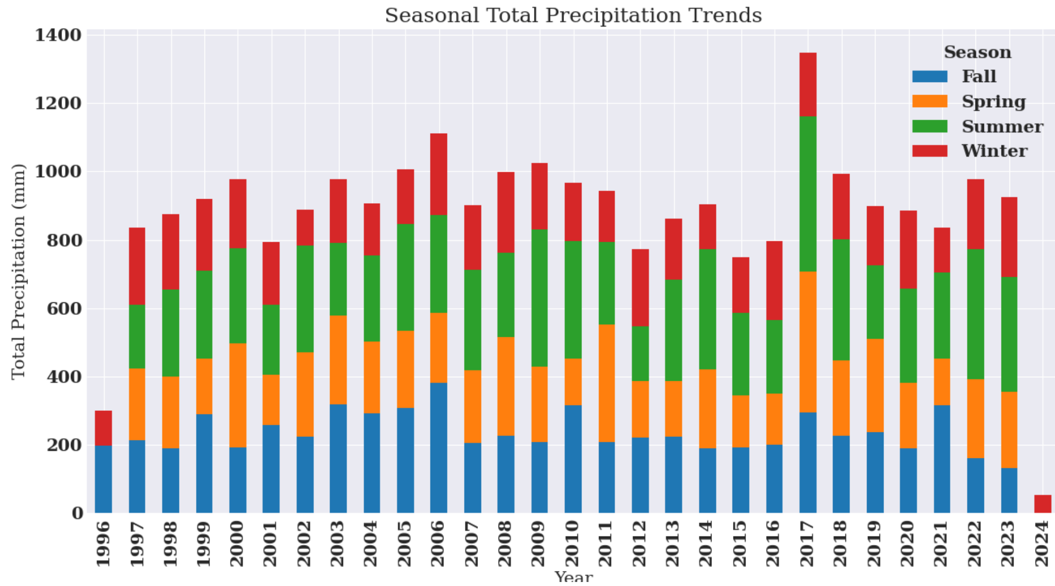


Figure 12: Seasonal Total Precipitation Trends

The stacked bar plot in Figure 12 illustrates the total precipitation for each season across different years. This helps in understanding how precipitation patterns vary by season and how they have changed over time.

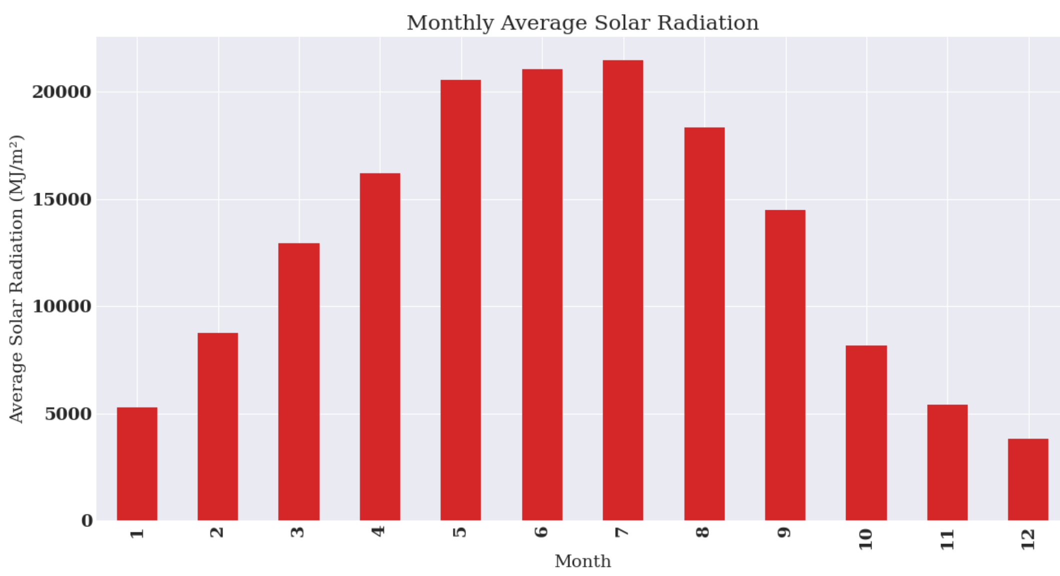


Figure 13: Monthly Average Solar Radiation

The bar plot in Figure 13 shows the average solar radiation for each month, highlighting the seasonal variation in solar energy received, with higher values in summer months and lower values in winter months.

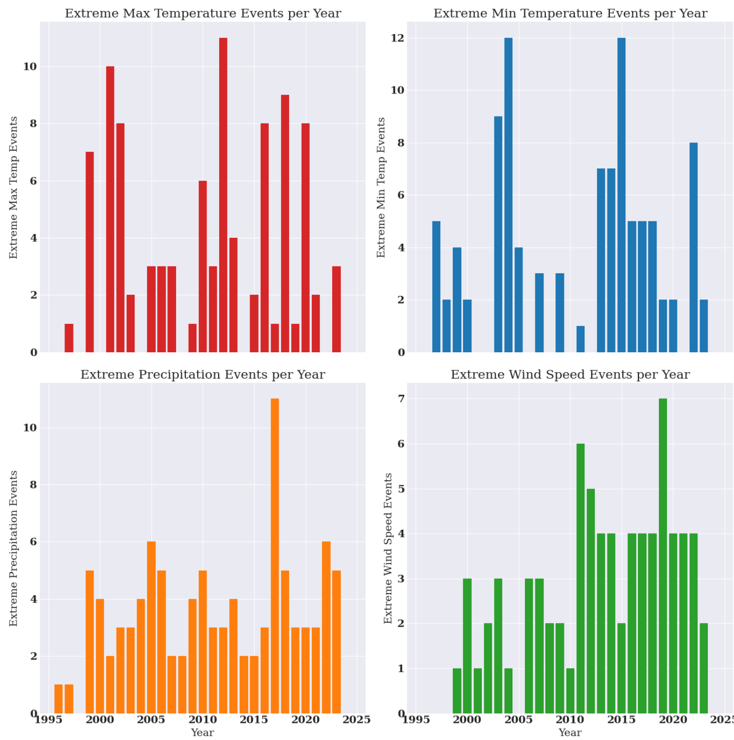


Figure 14: Trends of Extreme Weather Events

The plots in Figure 14 show the number of days per year with extreme maximum temperatures, extreme minimum temperatures, extreme precipitation, and extreme wind speeds (top 5 percentile). These plots help in understanding how the frequency of extreme weather events has changed over the years, which is crucial for assessing the impacts of climate change.

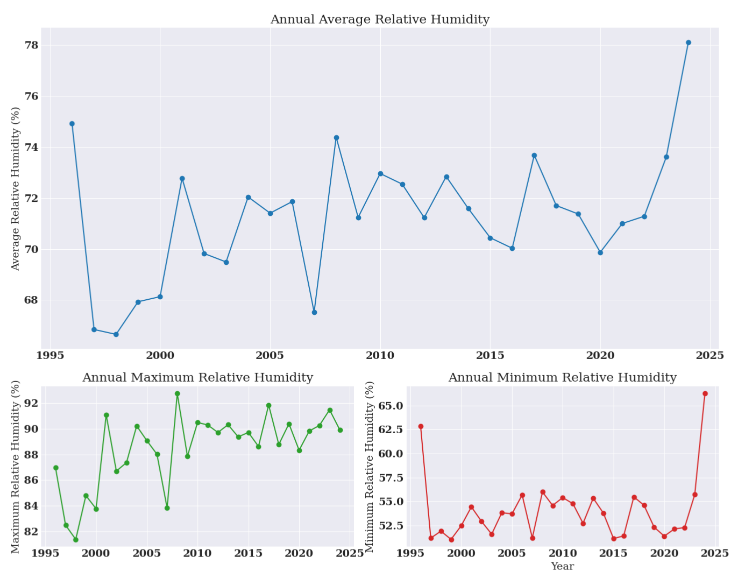


Figure 15: Annual Humidity Trends

Figure 15 displays the trends of annual average, maximum, and minimum relative humidity over the years.

The plots provide insights into changes in humidity levels, which can impact various weather phenomena and climate patterns.

Overall, the comprehensive analysis of these datasets provides a detailed understanding of the weather patterns and trends in the Ottawa region. The visualizations highlight seasonal variations, long-term trends, and extreme weather events, offering valuable insights for understanding climate patterns and informing climate change assessments and adaptations.

3.4 Requirements and Specifications

To successfully implement the weather prediction model described in this project, several libraries and tools are required. Below, we outline the necessary Python libraries and provide a brief explanation of their roles in the project.

3.4.1 Required Libraries

- **NumPy**
  - **Installation**: `pip install numpy`
  - **Usage**: NumPy is used for numerical computations and handling arrays. It is essential for efficiently processing and manipulating large datasets.
- **Pandas**
  - **Installation**: `pip install pandas`
  - **Usage**: Pandas is used for data manipulation and analysis. It provides data structures like DataFrames to handle tabular data efficiently.
- **Matplotlib**
  - **Installation**: `pip install matplotlib`
  - **Usage**: Matplotlib is used for creating static, animated, and interactive visualizations in Python. It is essential for plotting data and visualizing the results.
- **TensorFlow and Keras**
  - **Installation**: `pip install tensorflow`
  - **Usage**: TensorFlow and its high-level API Keras are used for building and training deep learning models, including Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks.
- **Scikit-Learn**
  - **Installation**: `pip install scikit-learn`
  - **Usage**: Scikit-Learn is used for preprocessing data, splitting datasets, and evaluating models. It provides functions for scaling data, calculating performance metrics, and splitting data into training, testing, and validation sets.
- **Keras Tuner**
  - **Installation**: `pip install keras-tuner`
  - **Usage**: Keras Tuner is used for hyperparameter tuning. It helps in finding the optimal hyperparameters for the models to improve performance.

3.4.2 Additional Requirements

- **Data Loading and Preprocessing**
  - The dataset is loaded from a URL and preprocessed to handle missing values, scale features, and create time series sequences. This requires a stable internet connection and access to the data source.
- **Directory Management**
  - The project involves saving plots and results to specific directories. Ensure that the necessary directories exist or are created during runtime.
- **Visualization and Plotting**
  - Visualization of the model's predictions against actual values is crucial for evaluating performance. Matplotlib is used extensively for this purpose.

3.4.3 Example Usage

The complete implementation of this weather prediction project, including all the necessary code and a detailed README file, is available on GitHub (6). The repository can be found at the following link: `https://github.com/adrien-heymans/weather-prediction-project`. This repository contains all the scripts and instructions needed to set up the environment, preprocess the data, build and train the models, and evaluate their performance.

By ensuring these libraries and tools are available, you can effectively develop, train, and evaluate the weather prediction models as described in the project. The proper installation and usage of these libraries will facilitate the implementation of various components of the weather prediction model, from data preprocessing to model evaluation and visualization.

3.5 Experimental Setup

The experimental setup for the weather prediction model involves several key stages: data preprocessing, model architecture, hyperparameter tuning, training and evaluation, and implementation details.

3.5.1 Data Preprocessing

Data preprocessing is an important step in preparing the dataset for training and evaluation. The following steps were performed:

Handling Missing Values

Missing values in the dataset were handled by interpolation, where the mean of each respective column was used to fill in the gaps. This approach ensures that no data points were lost due to missing values, thereby maintaining the dataset's integrity.

Feature Engineering

Difference features were created for each input feature to capture temporal changes. This involved calculating the difference between consecutive time-steps for each feature, which helps in understanding the day-to-day variations in weather parameters.

Data Splitting

The dataset was split into training, testing, and validation sets to maintain temporal continuity. In this step, it is very important not to shuffle the time-steps while splitting the main dataset, as it would make the model training and predictions meaningless. Specifically, 70% of the data was used for training, 20% for testing, and 10% for validation. This ensures that the model is evaluated on data it has not seen during training.

3.5.2 Model Description

Two types of models were developed for the weather prediction task: Convolutional Neural Networks (1-DCNNs) and Long Short-Term Memory (LSTM) networks.

One-Dimensional Convolutional Neural Network (1D-CNN)

The 1D-CNN model was designed to extract spatial features from the data. The architecture includes:

1. **Convolutional Layers:** These layers apply filters to the input data to extract spatial features.
2. **Max-Pooling Layers:** These layers reduce the dimensionality of the feature maps, thereby reducing computational complexity.
3. **Flatten Layer:** This layer converts the 2D feature maps into a 1D vector.
4. **Dense Layers:** Fully connected layers that perform the final prediction based on the extracted features.
5. **Dropout Layer:** This regularization layer helps to prevent overfitting by randomly setting a fraction of input units to zero at each update during training.

Long Short-Term Memory (LSTM)

The LSTM model was designed to capture temporal dependencies in the sequential weather data. The architecture includes:

1. **LSTM Layers:** These layers capture the temporal dependencies in the data by maintaining a state over time.
2. **Dropout Layers:** These regularization layers help to prevent overfitting.
3. **Dense Layers:** Fully connected layers that perform the final prediction based on the learned temporal features.

3.5.3 Hyperparameter Tuning

Hyperparameter tuning was performed using the Hyperband tuner from Keras Tuner to find the optimal configuration for the models. The following ranges and choices of hyperparameters were considered:

- **Filters:** 32 to 128
- **Kernel Size:** 2, 3, 4
- **Pool Size:** 2, 3
- **Units:** 32 to 256
- **Dropout:** 0.1 to 0.5
- **Learning Rate:** 1e-4 to 1e-2

The Hyperband tuner was configured with a maximum of 50 epochs and a factor of 3, running one iteration. This setup was chosen for its efficiency in finding optimal hyperparameters within a reasonable timeframe.

3.5.4 Training Process

The models were trained using the training dataset, with 75 epochs and a batch size of 128. The choice of 75 epochs was based on the observation that validation loss either plateaued or increased beyond this point. Various batch sizes (8, 32, 64, 128, and 256) were experimented with, and 128 was found to be optimal.

Evaluation Metrics

The models were evaluated using the validation dataset. The performance metrics used for evaluation included Mean Squared Error (MSE), Mean Absolute Error (MAE), and R-squared (R2) score. These metrics provided a comprehensive assessment of the models' predictive accuracy and generalizability.

3.5.5 Implementation Details

Hardware and Software Configuration

The models were implemented using Google Colab with GPU support for training and testing. The key libraries used included TensorFlow for building the models, Keras Tuner for hyperparameter tuning, and various other Python libraries (see section 3.4) for data processing and visualization.

Challenges and Issues

One significant challenge faced during implementation was the GPU timeout in Google Colab due to its free service limitations. This often required saving model checkpoints to avoid losing progress. Using a personal GPU or a dedicated server is recommended for extended training sessions to mitigate this issue.

Overall, the experimental setup was designed to ensure rigorous training, tuning, and evaluation of the models, leveraging both 1D-CNN and LSTM architectures to capture spatial and temporal dependencies in the weather data. This comprehensive approach aimed to develop robust models capable of accurate micro-climate weather predictions.

4 Results

This section presents the results of our comparative analysis of different models and training methodologies for predicting various weather parameters. The models compared are 1-Dimensional Convolutional Neural Networks (1D-CNN) and Long Short-Term Memory networks (LSTM). We also evaluate the effect of randomization in training data on the performance of these models. The performance metrics considered are Mean Absolute Error (MAE), Mean Squared Error (MSE), and R-squared (R2).

4.1 Effect of Model Type

4.1.1 Effect of Model Type and Randomization on MAE

Figure 16 shows the effect of model type and randomization on the Mean Absolute Error (MAE). The 1D-CNN and LSTM models exhibit similar performance with non-randomized data, with the median MAE around 0.7. However, randomization increases the variability of the MAE for both models, indicating inconsistent performance. Notably, the LSTM model's performance degrades more significantly with randomization compared to 1D-CNN, as evidenced by a wider interquartile range and more outliers. This suggests that LSTM models are more sensitive to the changes in training data distribution caused by randomization. The Baseline model shows a consistently higher MAE, around 0.65, its applicability is limited due to the simplicity of the approach and the lack of robustness seen in more complex models. This indicates that while Baseline models might perform well in certain scenarios, they lack the flexibility required for broader applications.

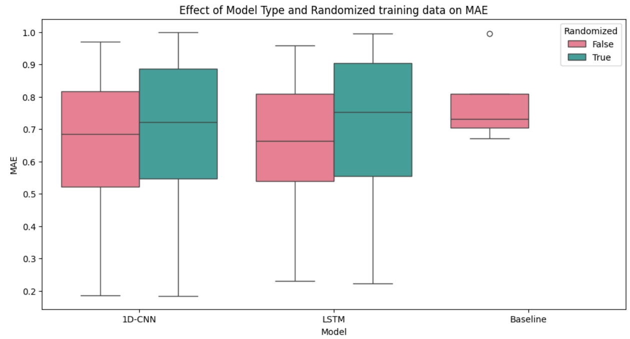


Figure 16: Effect of Model Type and Randomization on MAE

4.1.2 Effect of Model Type and Randomization on MSE

Figure 17 depicts the effect of model type and randomization on the Mean Squared Error (MSE). Similar to the MAE results, both 1D-CNN and LSTM models perform comparably with non-randomized data, with median MSE values around 0.5. The introduction of randomization again increases the variability of the MSE, particularly for the LSTM model, which shows a higher median and more outliers compared to 1D-CNN. This increased variability suggests that the LSTM model may overfit to specific training sequences, and randomization disrupts this pattern, leading to inconsistent performance. The Baseline model has a higher average MSE value because it is less robust compared to 1D-CNN and LSTM models, highlighting that while it can be effective for simple tasks, it may not generalize well to more complex datasets.

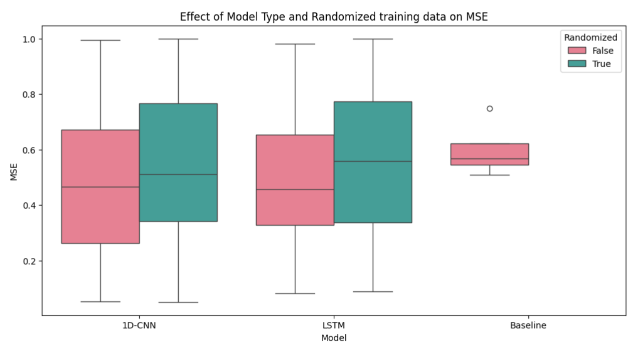


Figure 17: Effect of Model Type and Randomization on MSE

4.1.3 Effect of Model Type and Randomization on R2

Figure 18 illustrates the effect of model type and randomization on the R-squared (R2) metric. Both 1D-CNN and LSTM models struggle to achieve positive R2 values, indicating poor predictive power. The 1D-CNN model shows slightly better performance with a median R2 value closer to zero, while the LSTM model exhibits more negative values, suggesting worse predictions. Randomization further deteriorates the performance, especially for the LSTM model, which shows a significant drop in R2 values and increased spread. This indicates that randomization negatively impacts the model's ability to capture the underlying patterns in the data. The Baseline model shows slightly better performance with an R2 value around zero, but still fails to achieve satisfactory predictive accuracy, reinforcing the need for more sophisticated models to handle complex prediction tasks.

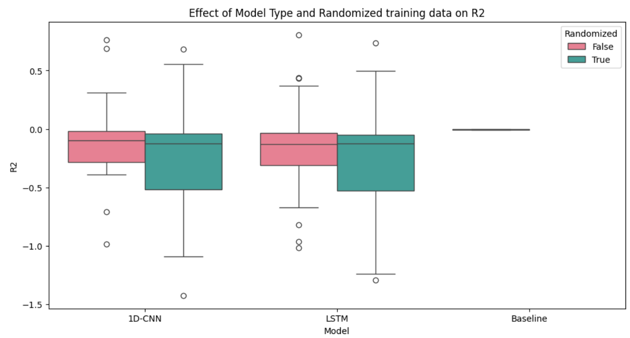


Figure 18: Effect of Model Type and Randomization on R2

4.1.4 Comparison of LSTM and 1D-CNN Models

Figures 23, 22, and 21 compare the performance of LSTM and 1D-CNN models for different features using MAE, MSE, and R2 metrics. The 1D-CNN model generally performs better for temperature-related features, achieving lower MAE and MSE values and higher R2 values. This indicates that 1D-CNN models are better at capturing temporal dependencies and long-term trends in temperature data. On the other hand, the LSTM model shows more stable performance across different features, suggesting it is less sensitive to the variability in the data. However, both models struggle with average hourly pressure and precipitation, indicating the need for further improvements in model design or additional features to improve predictions.

4.2 Performance Comparison by Feature and Randomization (R2)

Figure 19 compares the R2 values by feature and randomization. The results show that randomization generally has a detrimental effect on model performance across all features. The models perform better for temperature-related features compared to pressure and radiation features. Specifically, the Average Hourly Temperature shows the highest R2 values, indicating better predictability. In contrast, Solar Radiation and Precipitation exhibit negative R2 values, highlighting the challenges in accurately predicting these parameters. Randomization tends to increase the spread of R2 values, suggesting that the models struggle to learn consistent patterns when the training data is randomized.

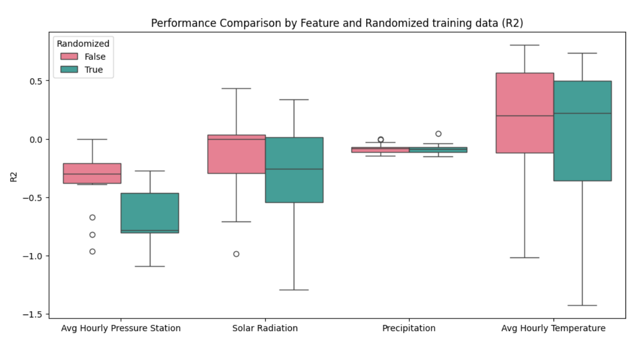


Figure 19: Performance Comparison by Feature and Randomization (R2)

4.3 Effect of Previous Days on Future Predictions (R2)

Figure 20 shows the effect of the number of previous days (n_past) on the predictive performance (R2) for future days (n_future). The heatmap indicates that longer past data generally improves the model's performance for future predictions, particularly for longer future periods. For instance, using 365 past days results in positive R2 values for future predictions, while shorter past data (30 or 90 days) leads to negative R2 values, indicating poorer predictions. This suggests that more historical data provides better context for the models to make accurate future predictions.

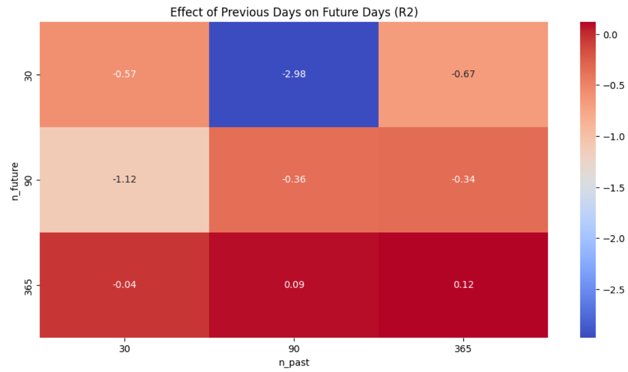


Figure 20: Effect of Previous Days on Future Predictions (R2)

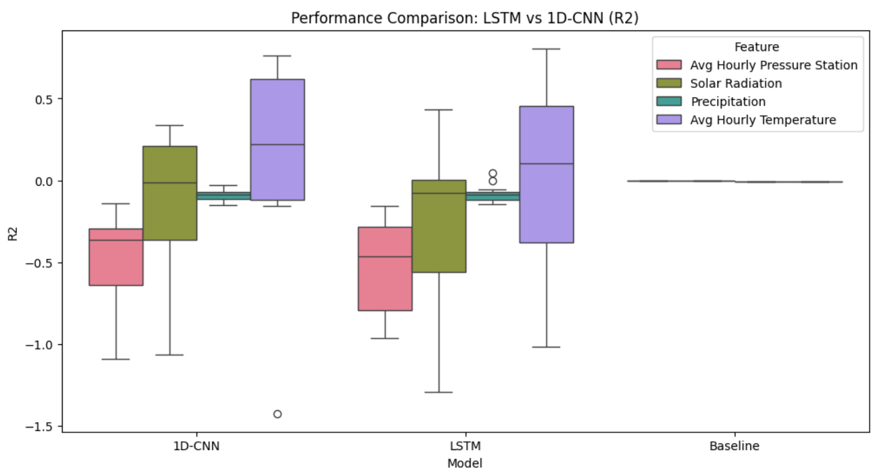


Figure 21: Performance Comparison: LSTM vs 1D-CNN (R2)

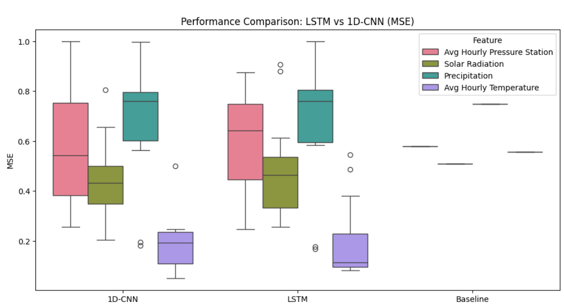


Figure 22: Performance Comparison: LSTM vs 1D-CNN (MSE)

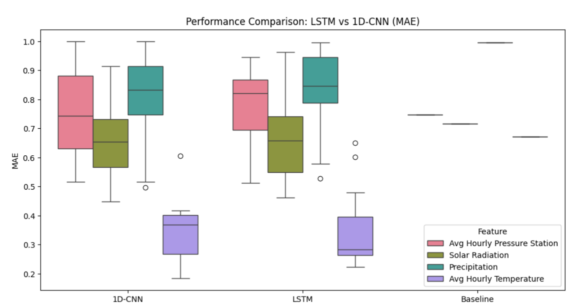


Figure 23: Performance Comparison: LSTM vs 1D-CNN (MAE)

4.4 Effect of Randomization

4.4.1 Effect of Randomization on Model Performance (MAE)

Figure 27 explores the impact of randomization on the Mean Absolute Error (MAE) of the models. It is evident that randomization increases the MAE for both 1D-CNN and LSTM models. Without randomization, both models exhibit a median MAE around 0.65, but with randomization, the MAE rises to approximately 0.75. The LSTM model, in particular, shows a higher increase in variability and outliers when the data is randomized, indicating its sensitivity to changes in data distribution. It is important to notre that the Baseline model, does not vary with randomization.

4.4.2 Effect of Randomization on Model Performance (MSE)

Figure 28 shows the impact of randomization on the Mean Squared Error (MSE) of the models. Similar to the MAE results, randomization increases the MSE for both 1D-CNN and LSTM models. The LSTM model is particularly affected, showing a higher median MSE and increased spread when trained with randomized data. This further indicates the model's sensitivity to training data distribution. The Baseline model remains stable with a relatively low MSE, reinforcing its simplicity and robustness for basic tasks.

4.4.3 Effect of Randomization on Model Performance (R2)

Figure 26 illustrates the effect of randomization on the R-squared (R2) metric. Randomization negatively impacts the R2 values for both 1D-CNN and LSTM models. The non-randomized data leads to R2 values closer to zero, indicating better predictive performance. However, with randomization, the R2 values drop, particularly for the LSTM model, which shows more negative values and a broader distribution.

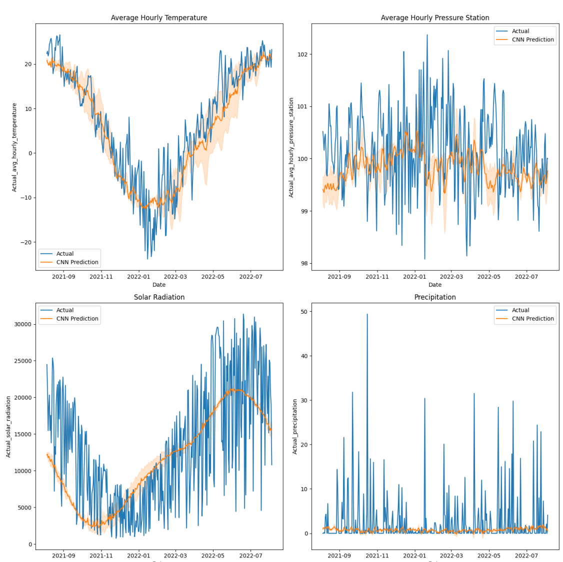


Figure 24: 1D-CNN Model Predictions on Validation Data

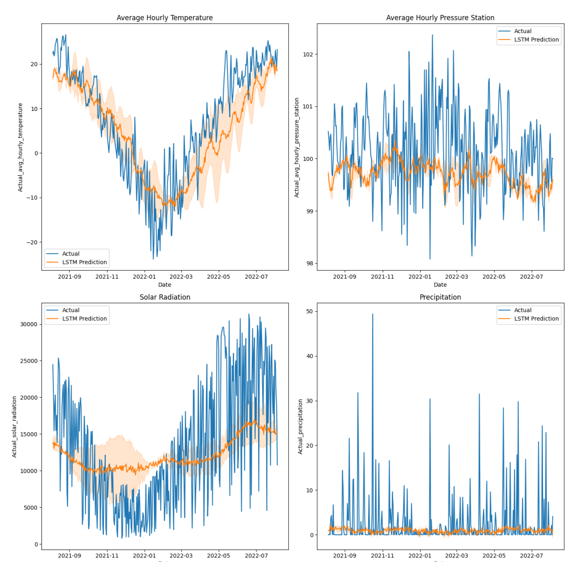


Figure 25: LSTM Model Predictions on Validation Data

4.4.4 Effect of Randomization on 1D-CNN Model

Figure 29 shows the impact of randomization on the 1D-CNN model's performance. For average hourly temperature, the 1D-CNN model with non-randomized training data provides better predictions, closely following the actual values. Randomized training data results in smoother predictions that miss some of the finer details. For average hourly pressure station, the non-randomized model again shows better alignment with actual values, though the difference is less pronounced. Solar radiation predictions benefit more from non-randomized training data, as the model captures the seasonal patterns better. Precipitation predictions remain poor for both training methods, but the non-randomized model shows slightly more variability, closer to the actual data.

4.4.5 Effect of Randomization on LSTM Model

Figure 30 highlights the effect of randomization on the LSTM model's performance. The non-randomized LSTM model provides better predictions for average hourly temperature, closely following the actual values, while the randomized model shows smoother predictions that miss finer details but seem to capture to spikes in the data predictions. The same can be observed for average hourly pressure station, the non-randomized model again aligns better with actual values but the randomized one can better capture the spikes in the actual data. Solar radiation predictions are more accurate with non-randomized training data, capturing the seasonal patterns better. Precipitation predictions remain challenging, with both models showing limited accuracy, but the non-randomized model exhibits slightly more variability.

4.4.6 Overall Effect of Randomization

Figure 32 illustrates the overall effect of randomization on model performance across different weather parameters. The non-randomized models consistently provide better predictions, closely following the actual values. Randomization tends to smooth out the predictions, leading to loss of finer details and decreased accuracy. However, randomized training seems to be able to capture the spikes in the data which is something that the models trained on the non-randomized training data is unable to do. This trend is evident across all parameters, including average hourly temperature, pressure station data, solar radiation, and precipitation.

4.5 1D-CNN Model Predictions

Figure 24 illustrates the performance of the 1D-CNN models on the validation dataset for various weather parameters. The 1D-CNN model's predictions for average hourly temperature show a clear seasonal trend, closely following the actual values, though with a slight lag during transitions. The model struggles with the high variability in average hourly pressure station data, leading to a smoother prediction curve that fails to capture sudden changes. Solar radiation predictions exhibit the largest deviation from actual values, with the model failing to capture the peaks and troughs accurately. Precipitation predictions also show limited accuracy, as the 1D-CNN model predicts lower variability compared to actual data, reflecting the challenges in predicting this sporadic parameter.

4.6 LSTM Model Predictions

Figure 25 depicts the performance of the LSTM model on the validation dataset. The LSTM model demonstrates strong predictive capability for average hourly temperature, capturing both seasonal trends and short-term fluctuations. For average hourly pressure station, the LSTM model's predictions are smoother and fail to capture sudden changes, similar to the 1D-CNN model. Solar radiation predictions show a clear seasonal trend but lack accuracy in capturing the variability. Precipitation predictions are also less accurate, with the model predicting lower variability compared to actual data.

4.7 Comparison of 1D-CNN and LSTM Models

Figure 31 compares the predictions of 1D-CNN, LSTM, and Baseline models against actual values for various weather parameters. The 1D-CNN model shows slightly superior performance for average hourly temperature, capturing both seasonal trends and short-term fluctuations more accurately than the LSTM model. For average hourly pressure station, both models perform similarly, with smoother predictions that miss sudden

Figure 26: Effect of Randomization on Model Performance (R2)

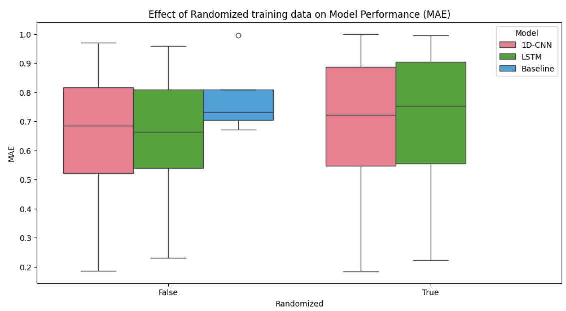


Figure 27: Effect of Randomization on Model Performance (MAE)

Figure 28: Effect of Randomization on Model Performance (MSE)

changes. Solar radiation predictions are challenging for both models, but the LSTM model captures the seasonal trends slightly better. Precipitation predictions are poor for all models, reflecting the inherent difficulty in predicting this parameter accurately.

5 Conclusion

This study provides an analysis of the performance of 1-Dimensional Convolutional Neural Networks and Long Short-Term Memory networks for predicting various weather parameters. Our results highlight several key findings regarding the efficacy of these models and the impact of randomization in training data.

Firstly, the comparison of 1D-CNN and LSTM models reveals that while both models show promise in predicting weather parameters, their performance varies significantly across different features. The 1D-CNN model generally outperforms the LSTM model in predicting temperature-related features, capturing both seasonal trends and short-term fluctuations with greater accuracy. On the other hand, the LSTM model demonstrates more stable performance across different features, suggesting that it is less sensitive to the variability in the data. However, both models struggle with predicting average hourly pressure and precipitation accurately, indicating the need for further improvements in model design or the inclusion of additional features to enhance predictive accuracy.

Secondly, the analysis of randomization in training data reveals its complex impact on model performance. While randomization tends to increase the variability of performance metrics such as MAE, MSE, and R2, it

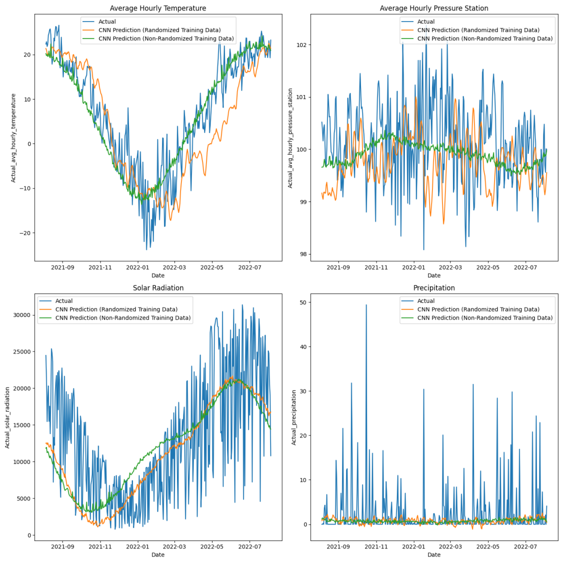


Figure 29: Effect of Randomization on 1D-CNN Model Predictions

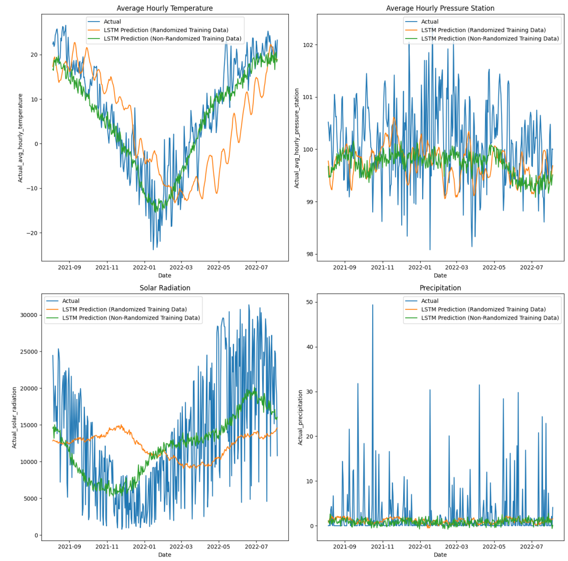


Figure 30: Effect of Randomization on LSTM Model Predictions

also demonstrates a potential benefit in improving the models' adaptation to exceptional events. This finding suggests that while randomization may introduce noise and lead to less consistent performance in general, it can enhance the models' robustness and ability to generalize to previously unseen data. Despite these potential benefits, our results indicate that non-randomized training data generally leads to better predictive accuracy, emphasizing the importance of maintaining consistent data distribution during training.

Thirdly, the detailed examination of model predictions on the validation dataset highlights specific areas where the models excel and where they fall short. For instance, the LSTM model shows strong predictive capability for average hourly temperature, closely following the actual values. However, both 1D-CNN and LSTM models struggle with the high variability in average hourly pressure station data and the sporadic nature of precipitation. These findings show the need for targeted improvements in model design and training methodologies to enhance predictive accuracy for these challenging parameters.

In reflecting on our methodology, it is important to acknowledge that with more time and resources, we could have performed better hyperparameter tuning and explored new techniques to further improve model performance. Some potential areas for future work include the integration of additional relevant features, the application of advanced data augmentation techniques, and the exploration of hybrid models that combine the strengths of 1D-CNN and LSTM architectures. Additionally, experimenting with different types of randomization strategies and incorporating external factors such as geographical and seasonal variations could provide valuable insights and lead to more robust and accurate models.

Despite the promising results obtained in this study, there is considerable room for improvement. Our models' struggles with certain weather parameters and the mixed impact of randomization highlight the complexity of weather prediction and the need for ongoing research and innovation in this field. Future work should focus on addressing these challenges and refining the models to enhance their predictive capabilities further.

In conclusion, this study demonstrates the potential of 1D-CNN and LSTM models for weather prediction, while also highlighting the importance of consistent training data and the challenges posed by certain weather parameters. The findings from this research provide valuable insights and lay the groundwork for future improvements and innovations in weather prediction models.

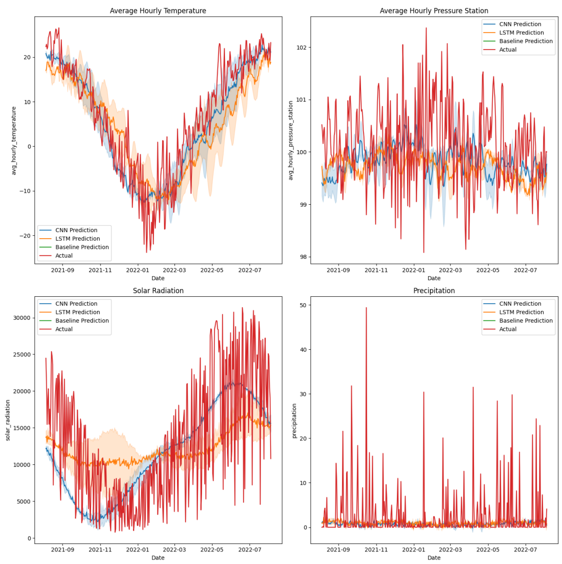


Figure 31: Comparison of 1D-CNN, LSTM

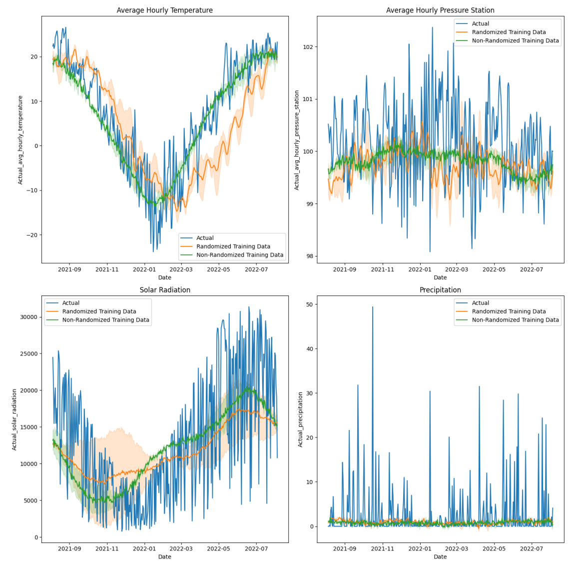


Figure 32: Overall Effect of Randomization on Model Performance

6 Acknowledgements

This research would not have been possible without the guidance and support of several individuals. We would like to extend our deepest gratitude to Professor Iluju Kiringa and Professor Tet Yeap for their invaluable advice and encouragement throughout this project. Their expertise and insights have been instrumental in shaping the direction and execution of this research.

We also wish to acknowledge the contributions of Master Student Weicheng Yu, whose assistance with data preprocessing and model implementation has been crucial to the success of this project. Additionally, we are grateful to PhD candidate Patrick Killeen for his thoughtful feedback and suggestions, which have greatly enhanced the quality of this research.

Finally, we thank all the members of our research group for their ongoing support and collaboration. Their collective efforts have made this study possible, and we look forward to continuing our work together in the future.

References

[1] Chollet, F. keras. `https://github.com/fchollet/keras`, 2015.

[2] Farmers Forum. Growers pre-sell this year's, next year's, and 2023 crop, 2021. Accessed: 2024-06-22.

[3] Farooq, M. S., Riaz, S., Abid, A., Abid, K., and Naeem, M. A. A survey on the role of iot in agriculture for the implementation of smart farming. *IEEE Access 7* (2019), 156237–156271.

[4] Government of Canada. Overview of the canadian agriculture and agri-food sector, 2023. Accessed: 2024-06-22.

[5] Hemamalini S., Geetha Rani K., R. B. S. S. M. An intelligent weather prediction model using optimized 1d cnn with attention gru. *Global NEST Journal 26*, 2 (2024), 05408.

[6] Heymans, A. Weather prediction project. `https://github.com/adrien-heymans/weather-prediction-project`, 2024.

[7] Hochreiter, S., and Schmidhuber, J. Long short-term memory. *Neural computation 9* (12 1997), 1735–80.

[8] Karevan, Z., and Suykens, J. A. K. Transductive lstm for time-series prediction: An application to weather forecasting. *Neural networks : the official journal of the International Neural Network Society 125* (2020), 1–9.

[9] Kiranyaz, S., Avci, O., Abdeljaber, O., Ince, T., Gabbouj, M., and Inman, D. J. 1d convolutional neural networks and applications: A survey, 2019.

[10] Mat, I., Mohd Kassim, M. R., Harun, A. N., and Mat Yusoff, I. Iot in precision agriculture applications using wireless moisture sensor network. In *2016 IEEE Conference on Open Systems (ICOS)* (2016), pp. 24–29.

[11] Pavón-Pulido, N., López-Riquelme, J. A., Torres, R., Morais, R., and Pastor, J. A. New trends in precision agriculture: a novel cloud-based system for enabling data storage and agricultural task planning and automation. *Precision Agriculture 18*, 6 (2017), 1038–1068.

[12] Ray, P. P. Internet of things for smart agriculture: Technologies, practices and future direction. *J. Ambient Intell. Smart Environ. 9*, 4 (jan 2017), 395–420.

[13] Srinivasa Rao, C., Kareemulla, K., Krishnan, P., Murthy, G., Ramesh, P., Ananthan, P., and Joshi, P. Agro-ecosystem based sustainability indicators for climate resilient agriculture in india: A conceptual framework. *Ecological Indicators 105* (2019), 621–633.

[14] Srivastava, A., and S, A. Weather prediction using lstm neural networks. In *2022 IEEE 7th International conference for Convergence in Technology (I2CT)* (2022), pp. 1–4.

[15] Staudemeyer, R. C., and Morris, E. R. Understanding LSTM - a tutorial into long short-term memory recurrent neural networks. *CoRR abs/1909.09586* (2019).