深 圳 大 学 实 验 报 告

课程编号：**2801000049**

课程名称：机器学习

实验项目名称：课程项目

学院：电子与信息工程学院

专业：电子信息工程

指导教师：麦晓春

报告人：廖祖颐 徐雷 姚宇铭 学号：2022110131 2022090123 2022280160 班级：文华班

实验时间：**2024/5/24——2024/6/28**

实验报告提交时间：**2024/6/28**

教务部制

基于 LTSM 的深圳天气预测

徐雷，姚宇铭,廖祖颐

Contributing percentage: 34% + 34% + 32%

Directory

Abstract ... 3
1. Introduction ... 4
    1.1 Task Background and Motivation ... 4
    1.2 Problem Statement ... 4
    1.3 Proposed Solution ... 4
    1.4 Contributions and Novelty ... 5
2. Method ... 6
    2.1 Data Preprocessing ... 6
        2.1.1 Handling Missing Values ... 6
        2.1.2 Data Conversion and Scaling ... 7
        2.1.3 Sequence Generation ... 7
    2.2 Model Definition ... 7
        2.2.2 Training Procedure ... 8
    2.3 Evaluation and Results ... 8
3. Experiments and Results ... 9
    3.1 Data Collection, Preprocessing and Analysis ... 9
        3.1.1Data Collection ... 9
        3.1.2Data Preprocessing ... 9
        3.1.3Analysis ... 10
    3.2 Evaluation Metrics ... 12
    3.3 Experimental Results and Analysis ... 14
        3.3.1 Temperature-related characteristics ... 14
        3.3.2 Barometric Pressure Related characteristics ... 16
        3.3.3 Humidity-related characteristics ... 18
        3.3.4 Rainfall related features and wind speed related characteristics ... 20
    4.1 The following result were obtained for the R24H prediction: ... 22
    4.2 About Precipitation Forecasting ... 23
    4.3 Why LSTM Models Struggle with Precipitation Prediction ... 24
    4.4 Enhancing Precipitation Prediction Models ... 24
References ... 26

Abstract

(250-300 words)

This study aims to predict various climate metrics using historical data through a Long Short-Term Memory (LSTM) model. The dataset includes features such as maximum and minimum temperatures, average daily temperature, pressure, humidity, precipitation, and wind direction. Initially, the data is preprocessed to handle missing values, convert it to numerical format, and interpolate linearly to ensure consistency. The data is then divided into training, validation, and test sets based on specific date ranges.

To prepare the data for the LSTM model, features are scaled using MinMaxScaler, and fixed-length sequences of seven days are created to capture temporal dependencies. These sequences are used to train, validate, and test the model. The LSTM model, with two hidden layers of 128 units each, processes the input sequences to predict the target climate metrics. During training, the Adam optimizer and Mean Squared Error (MSE) loss function are used, with early stopping to prevent overfitting.

Performance is evaluated using MSE, Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE). Visual comparisons between the predicted and true values are also made. The results show that the LSTM model effectively captures temporal patterns in the climate data, leading to accurate predictions of various metrics.

The study also explores prediction performance for different climate metrics over various time periods, validating the model's advantages in handling time series data. The LSTM model demonstrates high accuracy and robustness in dealing with complex climate variations and multidimensional data. Limitations and future improvement directions are discussed, including tuning model hyperparameters, increasing data volume, and integrating other machine learning methods.

This study highlights the potential of LSTM models in climate forecasting, providing a reliable approach to handling time series data and making precise predictions. The method's application can extend to other domains requiring accurate time series prediction.

1. Introduction

(Describe the task background, motivation, the problem you solve, the solution you propose to solve the problem, contributions and novelty)

1.1 Task Background and Motivation

Accurate weather prediction is crucial for a wide array of applications, ranging from agriculture and disaster management to daily activities such as planning outdoor events. Recent advancements in machine learning and deep learning have shown significant potential in improving the accuracy of weather forecasting models. Traditional statistical methods, while useful, often fail to capture the complex, non-linear relationships inherent in meteorological data. This limitation has highlighted the need for more sophisticated models capable of learning from historical data to make precise short-term predictions.

1.2 Problem Statement

This project addresses the challenge of short-term weather forecasting using deep learning techniques. Specifically, we focus on predicting various meteorological features, including maximum temperature (MAXT), minimum temperature (MINT), average temperature over 24 hours (AVGT24), and other related parameters. Given the past seven days of weather data, the goal is to predict the weather characteristics for the subsequent day. This task involves handling a time series dataset, dealing with missing values, scaling data appropriately, and selecting relevant features for the prediction model.

1.3 Proposed Solution

To tackle this problem, we propose the use of a Long Short-Term Memory (LSTM) network, a type of recurrent neural network (RNN) well-suited for sequential data. LSTMs are adept at capturing long-term dependencies and patterns in time series data, making them ideal for weather prediction tasks.

The solution involves the following key steps:

- Data Preprocessing: Handling missing values through interpolation, scaling features using MinMaxScaler, and splitting the dataset into training, validation, and testing sets based on time periods.
- Sequence Generation: Creating input sequences of seven days to predict the weather features for the next day.
- Model Definition: Designing an LSTM network with two layers, a hidden dimension of 128 units, and a fully connected output layer.
- Training and Evaluation: Training the model using the training set, validating its performance on the validation set, and employing early stopping to prevent overfitting. The best-performing model is then tested on the unseen test set.
- Performance Metrics: Evaluating the model's performance using Mean Squared Error (MSE), Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE).

1.4 Contributions and Novelty

The contributions of this project are manifold:

- Integration of Deep Learning in Weather Forecasting: This project demonstrates the effectiveness of LSTM networks in capturing the temporal dynamics of weather data, offering a more accurate alternative to traditional forecasting methods.
- Comprehensive Data Handling: By meticulously preprocessing the dataset, including handling missing values and scaling features, the project ensures that the model is trained on high-quality data, which is crucial for reliable predictions.
- Detailed Performance Evaluation: Through the use of multiple evaluation metrics and visualization of predictions against true values, the project offers a thorough assessment of the model's predictive capabilities.
- Application to Real-world Data: Utilizing meteorological data from Shenzhen, this project provides practical insights into the model's applicability and effectiveness in

real-world scenarios.

By addressing the complexities of short-term weather forecasting with advanced deep learning techniques, this project contributes to the growing body of research on machine learning applications in meteorology. The implementation showcases how modern AI methods can enhance predictive accuracy, ultimately benefiting various sectors that rely on precise weather information.

2. Method (Use at least two pages to illustrate the methodology)

2.1 Data Preprocessing

Data preprocessing is a critical step in preparing the dataset for training the LSTM model. The dataset used in this project comprises meteorological data from Shenzhen, which includes various weather parameters recorded over a period of time. The following steps were undertaken to preprocess the data:

2.1.1 Handling Missing Values

Meteorological datasets often contain missing values due to various reasons such as sensor malfunctions or data recording errors. In this project, missing values were initially replaced with NaNs using the replace function. Then, a linear interpolation method was employed to fill in these missing values. This method estimates the missing values by assuming a linear trend between known values, thus preserving the temporal continuity of the dataset.

2.1.2 Data Conversion and Scaling

To ensure all data points are numeric and suitable for the LSTM model, non-numeric entries were converted to numeric values using pd.to_numeric. Non-numeric entries that could not be converted were set to NaN and subsequently handled by the interpolation method.

Next, feature scaling was performed using the MinMaxScaler from scikit-learn. Scaling is essential to normalize the range of independent variables, thereby improving the training efficiency and convergence speed of the neural network. Each feature was scaled to a range between 0 and 1.

2.1.3 Sequence Generation

For the LSTM model to learn temporal dependencies, input sequences of a fixed length were created. In this project, a sequence length of 7 days was chosen to predict the weather features for the next day. The create_sequences function generates overlapping sequences from the data, where each sequence is used as an input sample for the LSTM.

2.2 Model Definition

The core of the project is the LSTM model, which is designed to capture and predict temporal patterns in the weather data. LSTMs are a special kind of RNN capable of learning long-term dependencies, making them suitable for time series forecasting[3].

2.2.1 LSTM Model Architecture

The LSTM model was implemented using PyTorch, a popular deep learning library. The model architecture consists of the following layers:

LSTM Layers: Two LSTM layers with 128 hidden units each. These layers process the input sequences and capture temporal dependencies.

Fully Connected Layer: A dense layer that maps the LSTM output to the target prediction.

The model architecture is encapsulated in the LSTMModel class.

2.2.2 Training Procedure

The training process involves the following steps:

Loss Function: Mean Squared Error (MSE) was chosen as the loss function to measure the model's prediction error.

Optimizer: The Adam optimizer was used to update the model parameters based on the computed gradients.

Early Stopping: To prevent overfitting, early stopping was implemented. If the validation loss did not improve for a specified number of epochs (patience), training was halted.

The model was trained for a maximum of 150 epochs with a batch size of 64. The best model, based on validation loss, was saved and used for final testing.

2.3 Evaluation and Results

The evaluation of the model's performance was conducted on the test dataset, which was not seen during training or validation. The following metrics were used to assess the model's predictive accuracy:

Mean Squared Error (MSE)

Mean Absolute Error (MAE)

Root Mean Squared Error (RMSE)

These metrics provide a comprehensive understanding of the model's error characteristics. Additionally, the true and predicted values were plotted to visualize the **model's performance.**

By meticulously following these steps, the project successfully demonstrated the feasibility and effectiveness of using LSTM networks for short-term weather forecasting, providing valuable insights

3. Experiments and Results

3.1 Data Collection, Preprocessing and Analysis

3.1.1Data Collection

The dataset used in this project is obtained from the Shenzhen Government Data Open Platform, specifically from the "Daily Data of Shenzhen Basic Stations." The dataset can be accessed through the following URL[1]:

3.1.2Data Preprocessing

Data preprocessing and analysis were performed using pandas and NumPy[2]. Missing values were handled using interpolation and any remaining missing values were dropped. The dataset was then split into training, validation, and test sets, followed by normalization using MinMaxScaler from scikit-learn[4]

Handling Missing Values:

Missing values in the dataset, represented by '-', are replaced with NaN.

Linear interpolation is used to fill in the missing values. Interpolation estimates missing values based on the values of neighboring data points, ensuring a continuous dataset.

After interpolation, any remaining missing values are dropped to ensure the dataset is complete and free from NaN values.

Dataset Splitting:

The dataset is split into three subsets: training, validation, and test sets.

The training set includes data from January 1, 2007, to December 31, 2018.

The validation set includes data from January 1, 2019, to December 31, 2021.

The test set includes data from January 1, 2022, to December 31, 2024.

Normalization:

The features and target variables are scaled using MinMaxScaler. This scaling method transforms the data to a range between 0 and 1.

Normalization is crucial for machine learning algorithms, especially those involving gradient descent, as it ensures that all features contribute equally to the model and prevents any single feature from dominating the learning process.

Sequence Creation:

Sequences of a fixed length (7 days) are created to capture temporal dependencies in the data. This involves creating overlapping sequences of input data and corresponding target values.

3.1.3Analysis

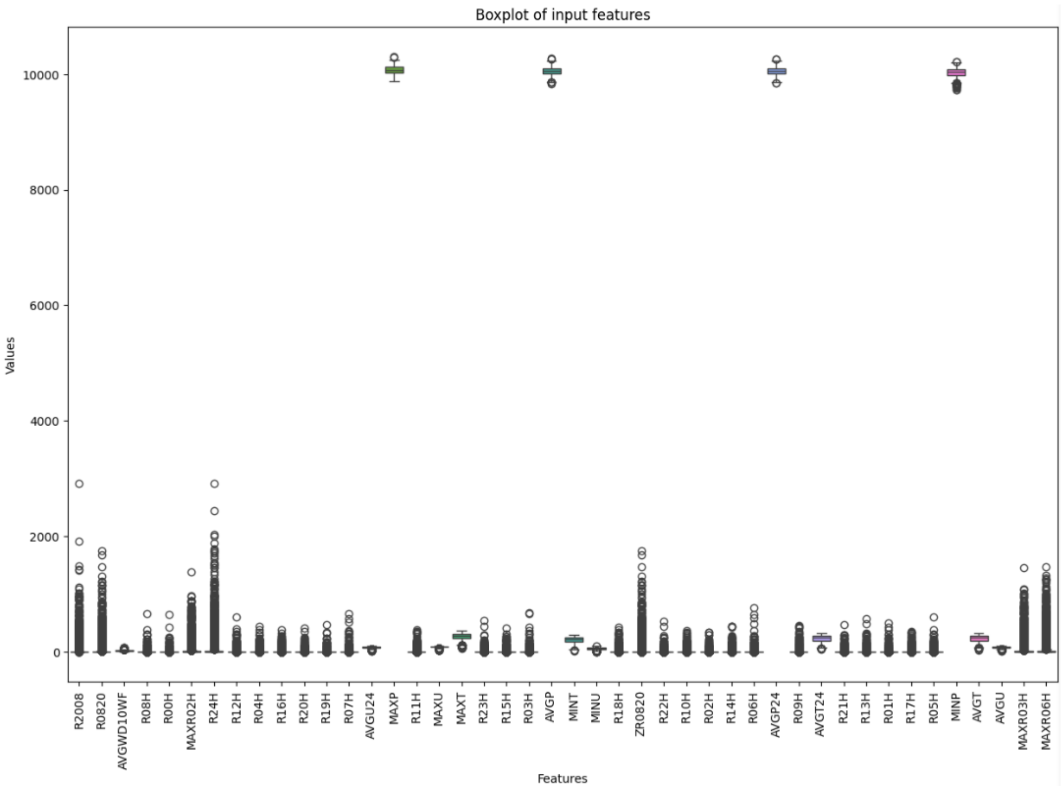

Fig1 Boxplot of input features

For example :

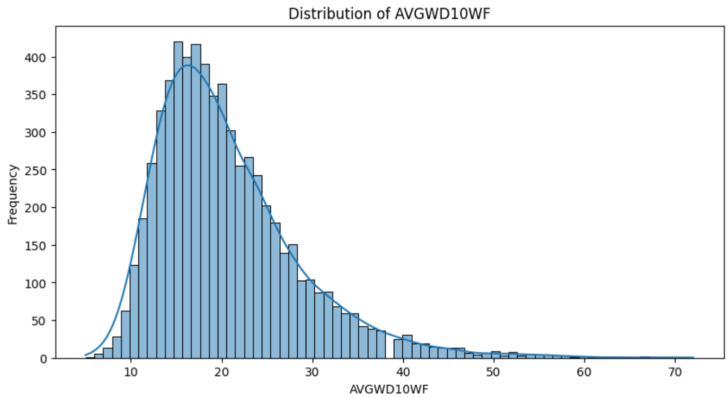

Fig2 Distribution of AVGWD1OWF

(1) Feature Distribution:

The boxplot indicates the spread and central tendency of each feature.

Most features have values clustered near the lower end of the scale, with some features showing wider spreads and higher values.

(2) Outliers:

Several features have significant outliers, indicated by the dots beyond the whiskers of the boxplot.

Features such as MAXR20H, R24H, ZR2080, and AVGWD10WF have numerous outliers, suggesting occasional extreme values in the data.

(3) Uniformity and Skewness:

Features like R24H, ZR2080, and MAXR20H show a high degree of variability, indicating a right-skewed distribution where most values are low, but there are occasional high values.

Features like AVGT24, MINT, and MAXT exhibit a more uniform distribution, with fewer extreme values.

(4) Feature Ranges:

Some features have values extending up to 10,000, such as AVGWD10WF, while others have much lower ranges.

This disparity in ranges suggests the necessity for normalization to ensure that each feature contributes equally to the model.

(5) Clustering of Values:

Most features exhibit clustering of values around the lower end, with median values typically close to the lower quartile.

The clustering indicates that the data has more frequent lower values with fewer higher values, common in environmental and meteorological datasets where extreme conditions are less frequent.

(6) Median and Quartiles:

The boxplot shows the median (central line in each box), which represents the middle value of the data.

The interquartile range (IQR), represented by the height of each box, shows the range within which the central 50% of values lie.

Features with smaller IQRs, such as MAXT and MINT, have more consistent values, while those with larger IQRs have more variability.

(7)Conclusion

The dataset contains significant variability and outliers, highlighting the importance of robust preprocessing steps such as normalization and handling of outliers.

Features exhibit different ranges and distributions, indicating that careful feature engineering and scaling are necessary to ensure effective model training.

3.2 Evaluation Metrics

In weather prediction models, evaluating the accuracy of predictions is crucial. Choosing the right metrics can help us understand the model's performance in real-world applications, optimize model parameters, and ultimately improve the reliability of predictions. Mean Squared Error (MSE), Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE) are three commonly used metrics, each with its own advantages and applicable scenarios, making them ideal choices for evaluating weather prediction models. Here comes the explanation.

- **Mean Squared Error (MSE):** MSE is the average of the squares of the prediction errors, expressed as $MSE = \frac{1}{n}\sum_{i=1}^{n} (y_i - \hat{y}_i)^2$ where y_i is the actual value, and $\hat{y}_i$ is the predicted value. The follow MAE and RMSE are the same. Because MSE penalizes larger errors more heavily (due to the square term), making the model focus more on reducing large prediction errors, which is very important for avoiding significant forecasting mistakes. So it is suitable for scenarios where large errors need to be strictly controlled, such as high-precision temperature forecasting.

- **Mean Absolute Error (MAE):** MAE is the average of the absolute values of the prediction errors, expressed as $MAE = \frac{1}{n}\sum_{i=1}^{n} |y_i - \hat{y}_i|$ Because MAE directly reflects the average size of errors, treating all types of errors equally, and is less sensitive to outliers than MSE. So it is suitable when the model needs to treat all types of errors equally, such as in daily weather forecasts where the user's perception of prediction errors is more intuitive.
- **Root Mean Squared Error (RMSE):** RMSE is the square root of MSE, expressed as $\sqrt{\frac{1}{n}\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}$ Because it is the square root of the error squared, RMSE increases sensitivity to larger errors while maintaining the same unit as the original data, making it easy to interpret and understand. So suitable for scenarios where precise control and reduction of the impact of large errors are necessary, such as the prediction of extreme weather events.

Summary

In weather forecasting, the importance of prediction accuracy is self-evident. Using MSE, MAE, and RMSE allows for a comprehensive evaluation of model performance, with each metric's characteristics enabling us to understand model accuracy and reliability from different angles. For example, in predicting extreme weather events like heatwaves or cold snaps, where large prediction errors can have serious consequences, the use of MSE and RMSE is more appropriate; while in everyday weather forecasts, where users may be more concerned with average performance, MAE provides a more intuitive measure of error.

Using these metrics collectively helps us identify which aspects of predictions are accurate and which need improvement, guiding the direction of model optimization and feature engineering, ultimately leading to more precise and reliable weather forecasts.

3.3 Experimental Results and Analysis

3.3.1 Temperature-related characteristics

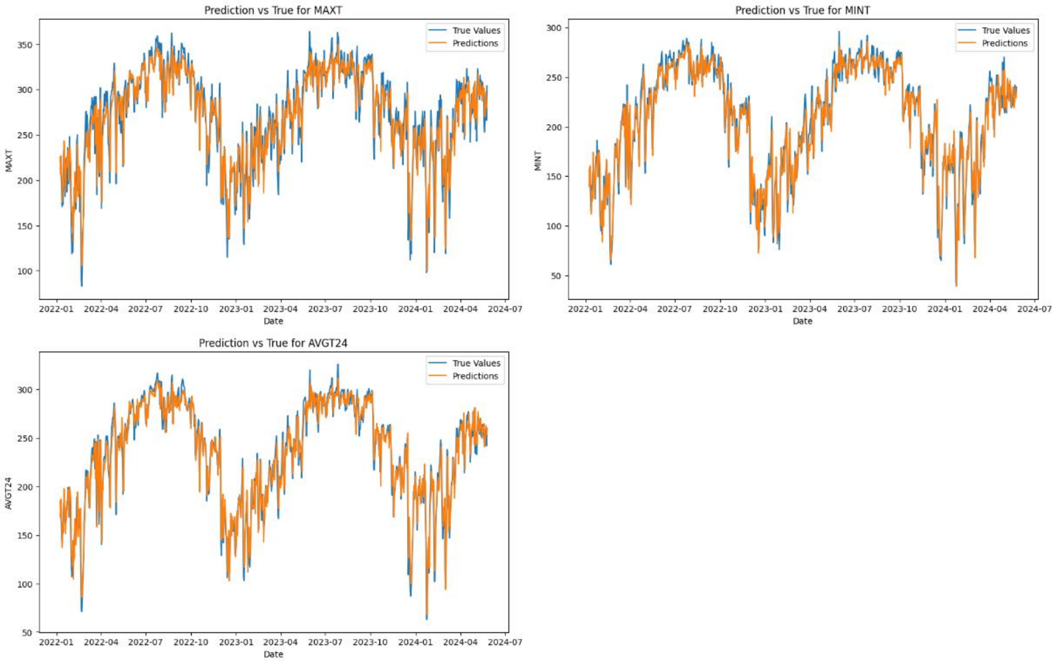

Fig 3 Model Prediction for Temperature

Table 1: Performance Metrics for Temperature Characteristics

Model	MSE	MAE	RMSE
MAXT	446.6089	16.4357	21.1331
MINT	232.8528	10.9805	15.2595
AVGT24	197.2475	10.2296	14.0445

Data visualization was performed using Seaborn to generate various plots for analyzing the dataset's distribution[5]

According to the charts, the prediction of maximum temperature closely matches the actual values, especially at peaks and troughs:

- Maximum Temperature (MAXT): MSE = 446.61
- Minimum Temperature (MINT): MSE = 232.85
- Average Temperature (24-hour periods) (AVGT24): MSE = 197.25

From the MSE values, we can observe that the prediction error is smallest for the average temperature and largest for the maximum temperature. This may indicate that predicting extremes (such as highest and lowest temperatures) is more challenging because these values might be influenced by extreme weather conditions, whereas averages tend to be more stable. Similar to the maximum temperature, the predicted values for the minimum temperature also follow the actual values' trends closely:

- Maximum Temperature (MAXT): MAE = 16.44
- Minimum Temperature (MINT): MAE = 10.98
- Average Temperature (24-hour periods) (AVGT24): MAE = 10.23

The MAE results show that the prediction accuracy for the average temperature is relatively high, while the error for the maximum temperature is relatively larger. This reinforces the results from the MSE, suggesting that the model might need further optimization when dealing with extreme values.

The predictions for average temperature generally align well with the true values, demonstrating the model's effectiveness:

- Maximum Temperature (MAXT): RMSE = 21.13
- Minimum Temperature (MINT): RMSE = 15.26
- Average Temperature (24-hour periods) (AVGT24): RMSE = 14.04

The RMSE results are consistent with the MAE and MSE, showing the largest prediction error for the maximum temperature and the smallest for the average temperature. This may be because maximum temperatures are more likely to be affected by extreme climatic changes, thus making them more difficult to predict.

Summary

In conclusion, the three metrics, MSE, MAE, and RMSE, consistently indicate that the prediction accuracy for average temperature is higher, while there is significant room for improvement in the prediction of maximum temperature. The analysis of these error metrics provides a crucial basis for future model adjustments and feature engineering, particularly in studying the factors affecting the prediction of extremes, especially under extreme weather conditions. Moreover, this also suggests that model training might need to incorporate more control variables or employ more complex model structures to enhance prediction accuracy and robustness.

3.3.2 Barometric Pressure Related characteristics

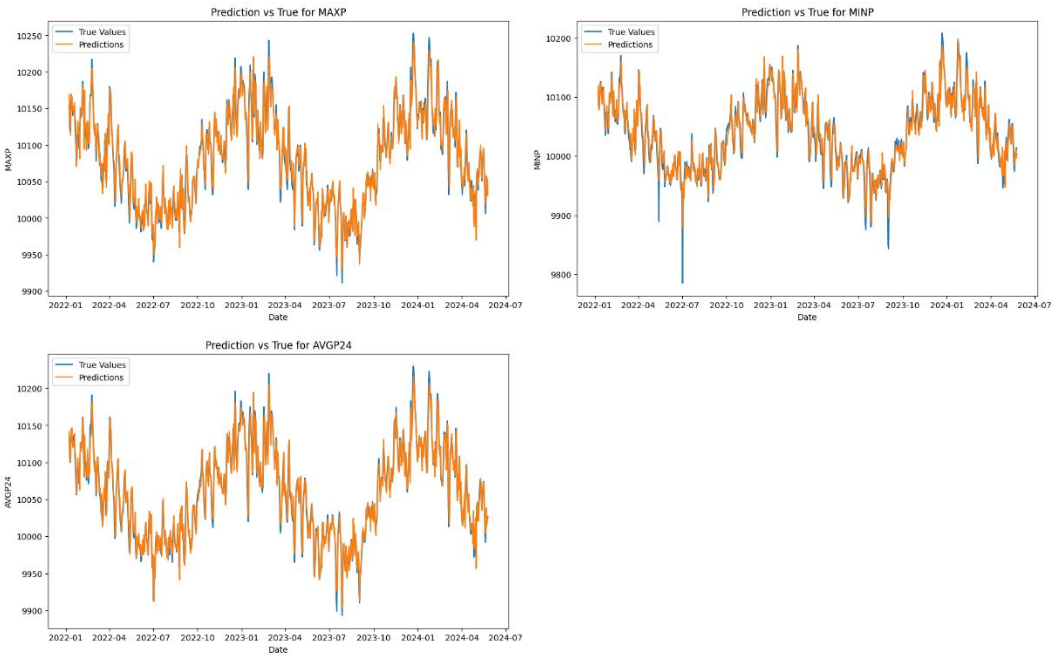

Fig 4 Model Prediction for Barometric Pressure

Table 2: Performance Metrics for Barometric Pressure Characteristics

Model	MSE	MAE	RMSE
MAXP	236.3396	11.5077	15.3733
MINP	336.2245	14.0370	18.3364
AVGP24	199.0201	10.6804	14.1074

Each graph displays a close match between the true values and the predictions, indicating the effectiveness of the model in capturing the trends and fluctuations in barometric pressure.

Maximum Barometric Pressure (MAXP)

- MSE: 236.34
- MAE: 11.51
- RMSE: 15.37

The MSE, MAE, and RMSE for MAXP suggest that the model is quite accurate, with an RMSE of 15.37 units. This implies that the model is capable of predicting maximum barometric pressure with a reasonable level of precision. The fluctuations in MAXP are closely followed by the model's predictions, which is evident from the overlay of the prediction and true values in the graph. The peaks and troughs are well-aligned, though minor deviations are observed, which account for the error metrics observed.

Minimum Barometric Pressure (MINP)

- MSE: 336.22
- MAE: 14.04
- RMSE: 18.34

The predictive performance on MINP shows higher error rates compared to MAXP. An RMSE of 18.34 indicates that the model experiences slightly more difficulty in accurately capturing the minimum pressure values. This could be due to more variability in the lower pressure values or less representation in the training data, necessitating potential model adjustments or additional training data focused on these lower ranges.

Average Barometric Pressure Over 24 Hours (AVGP24)

- MSE: 199.02
- MAE: 10.68
- RMSE: 14.11

The model performs best when predicting average barometric pressure over a 24-hour period, as shown by the lowest MSE and a fairly low RMSE compared to other metrics. The average values provide a smoothed overview of barometric pressure changes, which could be why the model predicts these values with higher accuracy. The close match between the predicted and actual graphs across the timeline suggests that the model effectively captures the overall trends without overfitting to outlier fluctuations.

Summary and Improvement ways

The analysis reveals that the model is generally effective in predicting barometric pressure with all metrics indicating good performance, though with some room for improvement in minimizing error rates, especially for minimum pressure predictions.

Considering refining the model or exploring more complex algorithms that might capture extreme values (highs and lows) more accurately. Reassess the input features and consider including additional variables that could influence barometric pressure changes, such as seasonal variations or geographical factors. Further analysis of where and why the largest residuals occur could provide insights into specific conditions under which the model's performance degrades.

3.3.3 Humidity-related characteristics

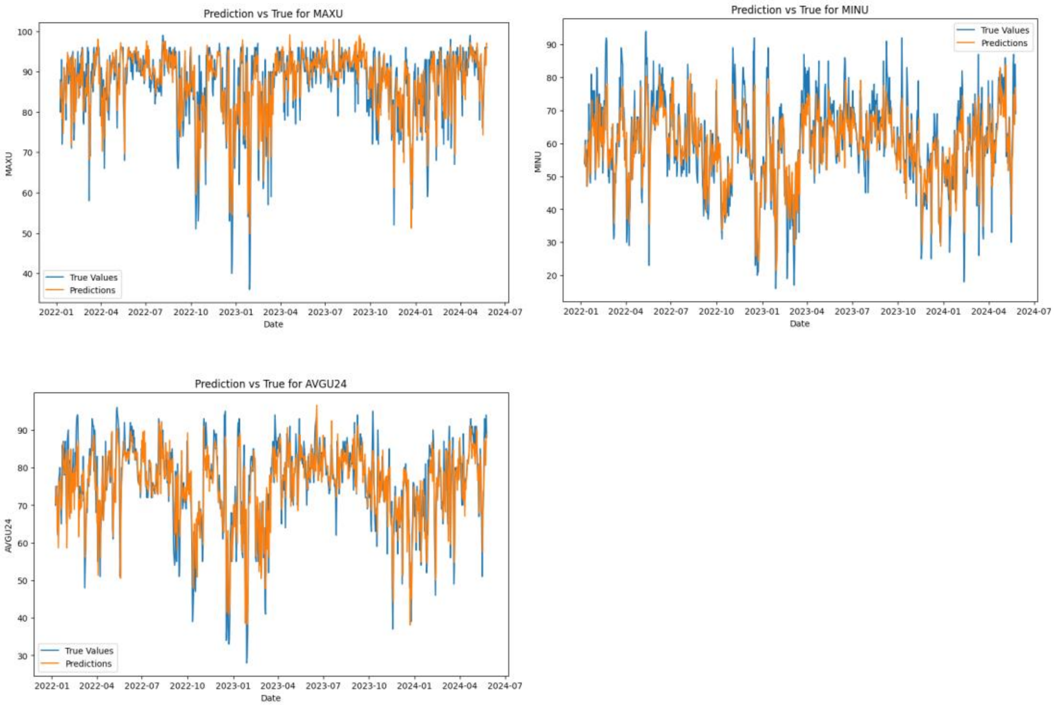

Fig 5 Model Prediction for Humidity

Table 3: Performance Metrics for Humidity Characteristics

Model	MSE	MAE	RMSE
MAXU	33.2020	4.0056	5.7621
MINU	87.5829	7.0632	9.3586
AVGU24	39.4699	4.7362	6.2825

Maximum Humidity (MAXU):

- MSE (Mean Squared Error): 33.20
- MAE (Mean Absolute Error): 4.01
- RMSE (Root Mean Squared Error): 5.76

The model closely mirrors the true values with minor deviations, indicating a strong ability to predict the peak humidity levels, which are often critical for weather-dependent activities.

Minimum Humidity (MINU):

- MSE: 87.58
- MAE: 7.06
- RMSE: 9.36

The predictions for minimum humidity show greater deviation compared to maximum humidity. The higher error metrics suggest that predicting lower humidity levels might be more challenging due to factors such as sudden atmospheric changes.

Average Humidity (24-Hour) (AVGU24):

- MSE: 39.47
- MAE: 4.74
- RMSE: 6.28

The average humidity predictions are moderately accurate, managing to follow the daily fluctuations effectively. The errors are lower than those for minimum humidity but slightly higher than for maximum humidity, highlighting a balanced performance across the day.

Graphical Analysis

The graphs for MAXU, MINU, and AVGU24 showcase the model's capability to track the actual humidity trends throughout the period. The model performance is notably adept at capturing the peaks in MAXU and the general trend in AVGU24, although there are sporadic mismatches in capturing the troughs in MINU, which could indicate potential areas for model tuning.

Summary

- The analysis of humidity predictions reveals that the model excels in capturing maximum humidity with the lowest errors, suggesting robustness in scenarios with high moisture levels.
- Improvements might be needed for minimum humidity predictions, where the error rates are higher, indicating challenges in capturing sudden drops in humidity which could be critical for precise weather forecasting.
- Overall, the model shows good potential in general humidity trend prediction but might benefit from enhancements in handling extreme values and sudden changes, possibly by incorporating more detailed local weather data or adjusting model sensitivity to rapid changes in humidity levels.

3.3.4 Rainfall related features and wind speed related characteristics

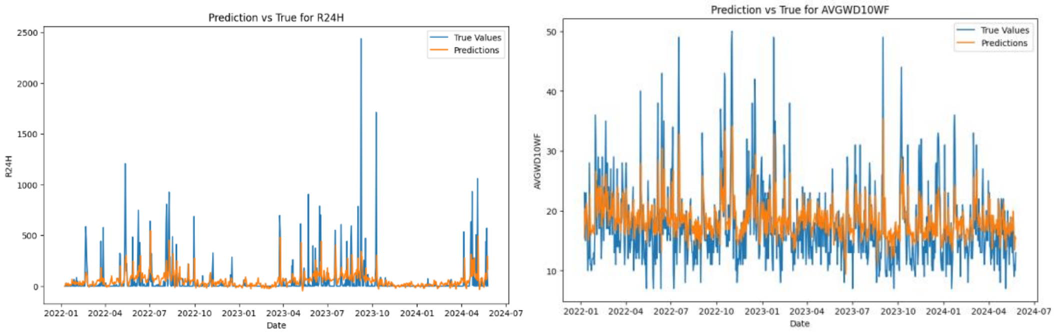

Fig 6 Model Prediction for Wind Speed and Rainfall

Table 4: Performance Metrics for Daily cumulative rainfall

Model	MSE	MAE	RMSE
R24H	25385.7305	72.2868	159.3290

Table 5: Performance Metrics for Ten-minute average wind speed

Model	MSE	MAE	RMSE
AVGWD10WF	33.1928	4.3890	5.7613

Daily Accumulated Rainfall (R24H)

- MSE (Mean Squared Error): 25385.73
- MAE (Mean Absolute Error): 72.29
- RMSE (Root Mean Squared Error): 159.33

The high values of MSE and RMSE indicate that the model struggles particularly with high intensity events, likely due to the skewed nature of rainfall data which includes many days with low or no rain and few very high values. The relatively high MAE further suggests that the average prediction error is substantial, which could be problematic for practical applications like flood forecasting.

Graph Analysis: The graph showcases the actual vs. predicted values for daily accumulated rainfall. The data points demonstrate extreme variability, with several spikes indicating episodes of heavy rainfall which are not accurately captured by the predictions. These discrepancies are especially noticeable during peak rainfall events where the model significantly underestimates the actual values.

Ten-Minute Average Wind Speed (AVGWD10WF)

- MSE: 33.19
- MAE: 4.39
- RMSE: 5.76

The errors are significantly lower than those observed for rainfall, suggesting that the model is better suited for predicting wind speed. The lower RMSE and MAE indicate a closer fit to the actual data, although the model still struggles with extreme values.

Graph Analysis: The prediction vs. true graph for AVGWD10WF shows a better alignment between predicted and actual values compared to rainfall, with predictions following the trends of actual measurements more closely. However, it still displays challenges in capturing the peaks accurately, which are critical in wind-related weather predictions.

Summary

- Model Performance: The model's performance on both parameters shows decent trend following but poor handling of extreme values. This could be due to the lack of enough representative data for extremes or the need for model adjustments specific to handling outliers.
- Data Quality and Model Training: Enhancing data quality, incorporating more diverse weather conditions, and using techniques like anomaly detection could improve model training. Additionally, using ensemble methods or more sophisticated time series models might capture extremes better.
- Practical Applications: For practical applications such as weather forecasting or climate study, improving the prediction accuracy of extreme events is crucial. Current models might be enhanced by integrating additional contextual data and reevaluating the feature engineering process.

4. Discussion

4.1 The following result were obtained for the R24H prediction:

Parameter	MSE	MAE	RMSE
R24H	25385.73046875	72.28683471679688	159.32899475097656

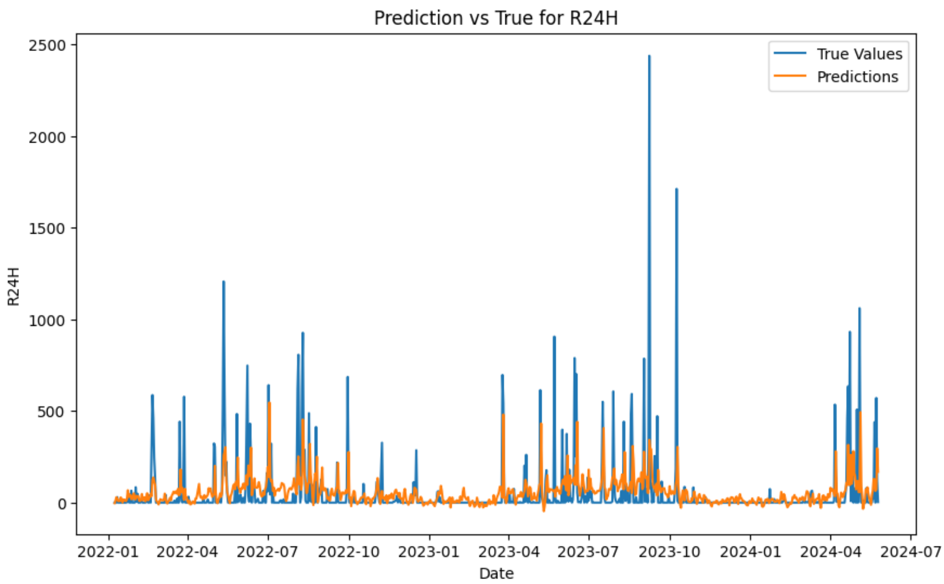

Fig7 Prediction vs True for R24H

Analysis

In this course, we predicted the average 24-hour precipitation, and finally found that the actual MAE and RMSE were too large (as shown in the table below). After predicting other features related to precipitation in the data set, it was found that the final result was the same as R24H, with poor fitting effect, and the amplitude of precipitation was difficult to predict accurately. After searching the relevant data, it is found that the precipitation itself is an unpredictable event with strong chaos and randomness, and the project is based on the data set of Shenzhen. The precipitation in Shenzhen has a large change, and the amplitude difference is large, which is also difficult to predict. Therefore, the prediction effect of the precipitation related features is not good.

4.2 About Precipitation Forecasting

Precipitation forecasting is inherently chaotic due to the complex and dynamic nature of the atmosphere. Edward Lorenz discovered in 1963 that even small errors in initial conditions, such as temperature and wind measurements, can amplify significantly, making it challenging to predict weather accurately beyond a certain timeframe. This sensitivity to initial conditions is a hallmark of chaotic systems, where tiny variations can lead to vastly different outcomes over time.

To address this, modern weather forecasting relies on numerical weather prediction (NWP) models, which use mathematical equations to simulate atmospheric dynamics. However, these models cannot perfectly predict weather conditions due to inherent limitations, such as insufficient grid resolution and model biases. To improve accuracy, statistical methods like Model Output Statistics (MOS) are used to correct forecasts by considering local effects and model errors.

Additionally, ensemble forecasting techniques are employed to handle the stochastic nature of weather processes. By running multiple simulations with slightly varied initial conditions or different physical parameterizations, ensemble forecasts can provide a range of possible outcomes, offering a probabilistic view of future weather. This approach helps to account for the uncertainty in initial conditions and extends the useful prediction window beyond what a single model run could achieve.

These methods underscore the inherent challenges in precipitation forecasting and the ongoing efforts to refine prediction models to better handle the chaotic nature of the atmosphere. However, the degree of fitting for precipitation amounts remains unsatisfactory due to the inherently unpredictable nature of precipitation itself. Precipitation is influenced by a multitude of factors, including temperature, humidity, wind patterns, and topography, which interact in complex and often unpredictable ways. Small-scale processes like convection and the formation of localized weather phenomena add further unpredictability. Consequently, despite advancements in forecasting techniques, accurately predicting the precise amount and location of precipitation remains an elusive goal due to its intrinsic variability and complexity.

4.3 Why LSTM Models Struggle with Precipitation Prediction

- **Complexity and Nonlinearity:** Meteorological processes, especially those involving precipitation, are highly complex and nonlinear. LSTM models may struggle to capture these intricate interactions fully.
- **Stochastic Nature:** Precipitation is highly stochastic, influenced by numerous unpredictable atmospheric variables. LSTM models, relying on historical data, often fail to generalize from these irregular patterns.
- **Data Noise:** Precipitation data can be noisy due to measurement errors and natural variability. This noise complicates the learning process for LSTM models.
- **Model Limitations**: Despite their ability to handle sequential data, LSTMs may not effectively capture long-term dependencies in highly variable data like precipitation. They also require significant hyperparameter tuning and extensive training data.

4.4 Enhancing Precipitation Prediction Models

- **Transformer Models:** These models, with their attention mechanisms, can focus on relevant parts of the input sequence, potentially capturing important dependencies more effectively than LSTMs.
- **Hybrid Approaches:** We can add some binary classifiers or four seasons classifiers to distinguish the difference in precipitation in different seasons, or combining LSTM with techniques like ensemble empirical mode decomposition (EEMD) and least squares support vector machines (LSSVM) can improve accuracy by decomposing complex time series and leveraging robust prediction methods.
- **Data Augmentation:** Enhancing the training set with synthetic data and extracting additional features can improve model performance.
- **Ensemble Methods:** Running multiple simulations with varied initial conditions can

provide a range of possible outcomes, offering a more probabilistic view of future weather.

5. Conclusions

This experiment involved building and training an LSTM model to predict various meteorological variables such as maximum temperature, minimum temperature, average temperature, maximum humidity, minimum humidity, average humidity, daily cumulative rainfall, average atmospheric pressure, highest atmospheric pressure, lowest atmospheric pressure, and ten-minute average wind speed. Using historical weather data, the model aimed to capture the dynamic changes in these variables over time and predict future values

Experimental Design: Initially, the data was cleaned, including parsing dates, converting data types, interpolating missing values, and deleting remaining missing entries. Then All columns except the time series were chosen as input features, and predictions were made for each meteorological variable. The dataset was divided into training, validation, and test sets to ensure the model could make accurate predictions on unseen data. A multi-layer LSTM network was employed to handle the complexity of sequence data, with a fully connected layer to output predictions. The model was trained over 150 epochs, using early stopping to prevent overfitting, and utilized the Adam optimizer and MSE loss function. Mean Squared Error, Mean Absolute Error, and Root Mean Squared Error were used to assess the accuracy of the model’s predictions.

Data analysis: The LSTM model was able to capture the trends of most meteorological variables effectively, especially in stable intervals. The model's prediction accuracy decreased for extreme values and rapid change intervals, particularly when predicting extreme weather events such as heavy rainfall and high wind speeds. The generated charts allowed for a visual comparison between the model’s predictions and the actual values, further validating the model's effectiveness.

This experiment demonstrated the effectiveness of LSTM in handling and predicting time series data, particularly in the application of meteorological data. However, for predicting extreme weather events, the model requires further optimization and adjustment, possibly by incorporating more features, using more complex network architectures, or employing ensemble models to improve prediction accuracy and robustness.

References

[1] Shenzhen Government Data Open Platform. "Daily Data of Shenzhen Basic Stations". https://opendata.sz.gov.cn/data/dataSet/toDataDetails/29200_00903511.

[2] Wes McKinney. "Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython". O'Reilly Media, 2017.

[3] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. "Deep Learning". MIT Press, 2016.

[4] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". Journal of Machine Learning Research, 12, 2825-2830, 2011.

[5] Michael Waskom. "Seaborn: Statistical Data Visualization". Journal of Open Source Software, 6(60), 3021, 2021.

指导教师批阅意见：   成绩评定：   指导教师签字： 年　　月　　日
备注：

注：1、报告内的项目或内容设置，可根据实际情况加以调整和补充。

2、教师批改学生实验报告时间应在学生提交实验报告时间后 10 日内。