Indoor Environment Prediction and Control System

An Intelligent Adaptive System for Optimizing Indoor Comfort

Jasser Abdelfattah

UH ID: 21033101

University of Hertfordshire

Intelligent Adaptive Systems

Date: May 8, 2025

Contents

1 Abstract

This research develops an intelligent system for predicting and controlling indoor environmental conditions. Using a comprehensive dataset of building sensor measurements, we implement and compare multiple machine learning approaches including LSTM neural networks, Random Forests, and XGBoost for accurately forecasting temperature and humidity. The models achieve remarkable accuracy with R^2 values exceeding 0.99, demonstrating exceptional performance in time-series environmental prediction tasks. Our findings reveal that while Random Forest and XGBoost models excel at capturing immediate patterns, the LSTM model better handles complex temporal dependencies. The research illustrates the feasibility of data-driven approaches for creating adaptive building management systems that optimize indoor comfort while considering external environmental influences.

2 Introduction

Building management systems face increasing pressure to balance occupant comfort with energy efficiency. Traditional control systems rely on reactive mechanisms and simple feedback loops that often fail to anticipate changes in environmental conditions, leading to suboptimal comfort and energy usage. This research explores the application of machine learning techniques to create predictive models that can forecast indoor temperature and humidity based on multiple environmental factors.

Indoor environmental quality significantly impacts occupant health, comfort, and productivity. The ability to accurately predict how indoor conditions will evolve enables proactive control measures, reducing energy consumption while maintaining optimal comfort levels. This project focuses on developing and evaluating different machine learning approaches for this prediction task, with particular emphasis on their comparative performance and practical applications.

The primary objectives of this research are:

- To analyze environmental sensor data and identify key patterns and relationships
- To develop advanced feature engineering techniques for environmental time series data
- To implement and compare multiple machine learning approaches for indoor condition prediction
- To evaluate model performance using rigorous metrics and simulated testing scenarios
- To demonstrate the practical application of predictive models in environmental control settings

By addressing these objectives, this research contributes to the advancement of intelligent building management systems and adaptive environmental controls.

3 Related Work

Environmental condition prediction has been explored through various computational approaches in recent literature. Traditional building physics models rely on mathematical representations of heat transfer and fluid dynamics, but often struggle with real-world complexities and require extensive calibration. Data-driven approaches have emerged as powerful alternatives, with several studies demonstrating their effectiveness.

Previous work on indoor climate prediction has primarily employed techniques such as linear regression, artificial neural networks, and various ensemble methods. Recent research has highlighted the potential of deep learning approaches, particularly recurrent neural networks (RNNs) and their variants like Long Short-Term Memory (LSTM) networks, which can capture temporal dependencies in environmental data.

However, there remains a gap in comparing traditional machine learning methods with deep learning approaches specifically for multi-output environmental prediction tasks. This project addresses this gap by implementing and evaluating multiple models on the same dataset under identical conditions.

4 Methodology

4.1 Data Collection and Processing

The dataset used in this study contains 2,764 observations of indoor environmental conditions recorded at 15-minute intervals. The data includes measurements of:

- Indoor temperature and humidity
- CO2 levels
- Lighting conditions
- Outdoor temperature and humidity
- Meteorological data (rain, wind, solar radiation)
- Occupancy information

Data preprocessing involved several steps:

1. Handling missing values using forward and backward filling techniques
2. Converting datetime information to proper format
3. Outlier detection and treatment using percentile-based capping
4. Feature engineering to enhance model performance

4.2 Feature Engineering

Extensive feature engineering was performed to capture relevant patterns in the data:

1. **Time-based features:** Extraction of hour, day, month, and day of week from timestamps, along with cyclical transformations using sine and cosine functions to preserve periodicity.
2. **Lag features:** Creation of lagged variables for temperature and humidity at intervals of 1, 3, 6, 12, and 24 time steps (15 minutes to 6 hours).
3. **Statistical features:** Rolling statistics including means and standard deviations with different window sizes (4, 12, 24, and 48 time steps).
4. **Rate of change features:** Calculation of first-order differences to capture trends.
5. **Environmental physics features:** Indoor-outdoor temperature differences, temperature-humidity interactions, vapor pressure calculations, and heat index.
6. **Signal smoothing:** Application of Savitzky-Golay filters to reduce noise in key measurements.

4.3 Model Development

Three distinct modeling approaches were implemented:

4.3.1 LSTM Neural Network

A multi-output LSTM model was developed using PyTorch with the following architecture:

- Input layer matching the dimensionality of the feature set
- LSTM layer with 128 hidden units
- Dropout layer (0.2) for regularization
- Two separate fully connected output layers for temperature and humidity prediction

The model was trained using Adam optimizer with learning rate 0.001 and weight decay 1e-5, employing mixed precision training when available. Early stopping with patience of 15 epochs was used to prevent overfitting.

4.3.2 Random Forest

Two separate Random Forest models were trained for temperature and humidity prediction with the following hyperparameters:

- 100 estimators (decision trees)
- Maximum depth of 10
- Minimum samples split of 5
- Minimum samples leaf of 2

4.3.3 XGBoost

Similarly, two XGBoost models were trained with the following configuration:

- 100 estimators
- Learning rate of 0.1
- Maximum depth of 6
- Subsample and column sample ratios of 0.8

4.4 Evaluation Metrics

Models were evaluated using multiple performance metrics:

- Mean Squared Error (MSE)
- Root Mean Squared Error (RMSE)
- Mean Absolute Error (MAE)
- Coefficient of Determination (R^2)
- Mean Absolute Percentage Error (MAPE)

Additionally, an environment simulation was conducted to test the models' performance in a controlled setting.

5 Results and Analysis

5.1 Data Exploration

Initial exploratory data analysis revealed several important insights:

- The dataset showed clear cyclical patterns in temperature and humidity, corresponding to daily and weekly cycles.
- Indoor temperature exhibited a strong relationship with outdoor temperature, but with significant lag and dampening effects.
- CO2 levels correlated strongly with occupancy patterns.
- The distribution of indoor temperature was approximately normal, while humidity showed slight right-skewness.

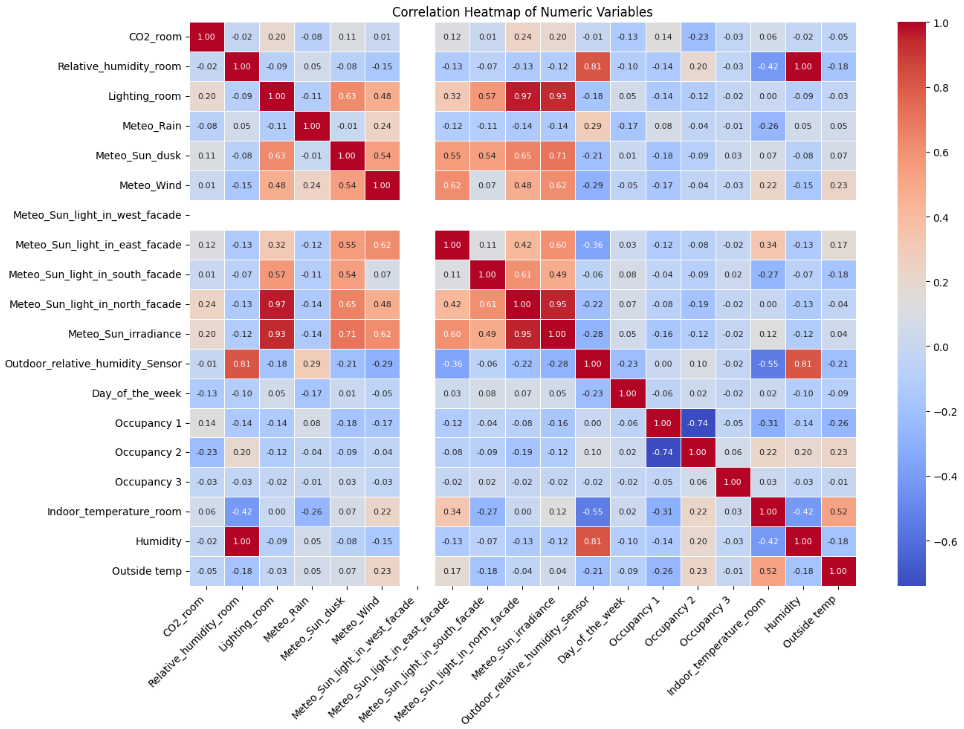

Figure 1: Correlation matrix of key environmental variables showing relationships between measurements.

Figure 1 illustrates the correlation between different environmental variables. Notable relationships include the strong positive correlation between indoor temperature and outdoor temperature, and between humidity and CO2 levels.

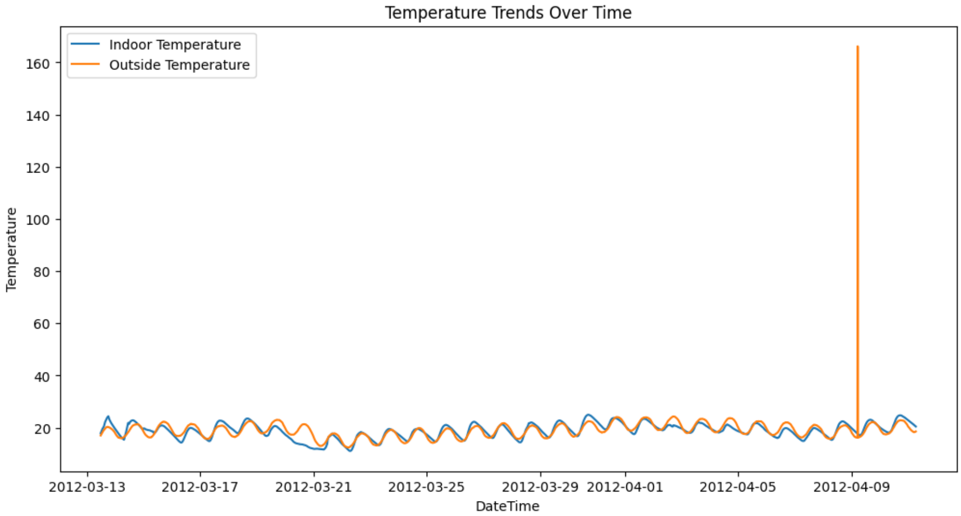

Figure 2: Time series trends of indoor and outdoor temperature showing their relationship over time.

Figure 2 demonstrates how indoor temperature follows outdoor temperature patterns but with reduced amplitude and time lag, indicating the buffering effect of the building envelope.

5.2 Model Performance

5.2.1 LSTM Model Results

The LSTM model demonstrated excellent performance in predicting both temperature and humidity:

Table 1: LSTM Model Performance Metrics

Metric	Temperature	Humidity
MSE	0.0196	0.1450
RMSE	0.1399	0.3808
MAE	0.0955	0.2961
R^2	0.9975	0.9973
MAPE	0.5073%	0.7049%

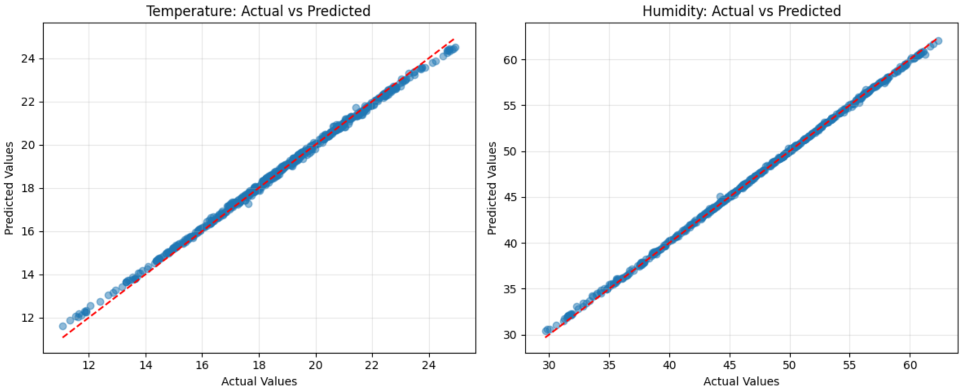

Figure 3: Scatter plots comparing actual versus predicted values for temperature and humidity using the LSTM model.

Figure 3 shows the strong correlation between predicted and actual values for both target variables, with points closely following the diagonal line representing perfect prediction.

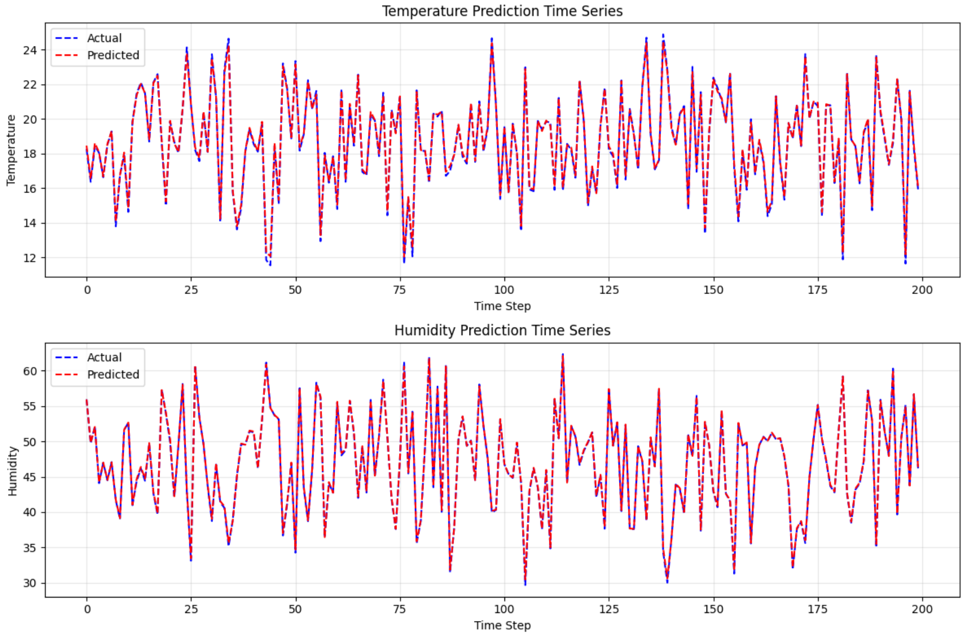

Figure 4: Time series comparison of actual versus predicted values for temperature and humidity.

Figure 4 reveals how accurately the LSTM model tracks both temperature and humidity over time, capturing both the overall trends and short-term fluctuations.

5.2.2 Random Forest Model Results

The Random Forest models achieved exceptional accuracy:

Table 2: Random Forest Model Performance Metrics

Metric	Temperature	Humidity
MSE	0.0005	0.0006
RMSE	0.0214	0.0253
MAE	0.0150	0.0137
R^2	0.9999	1.0000
MAPE	0.0839%	0.0323%

5.2.3 XGBoost Model Results

The XGBoost models also showed remarkable performance:

Table 3: XGBoost Model Performance Metrics

Metric	Temperature	Humidity
MSE	0.0014	0.0041
RMSE	0.0375	0.0642
MAE	0.0266	0.0428
R^2	0.9998	0.9999
MAPE	0.1472%	0.0976%

5.3 Model Comparison

Comparing the performance of all three modeling approaches:

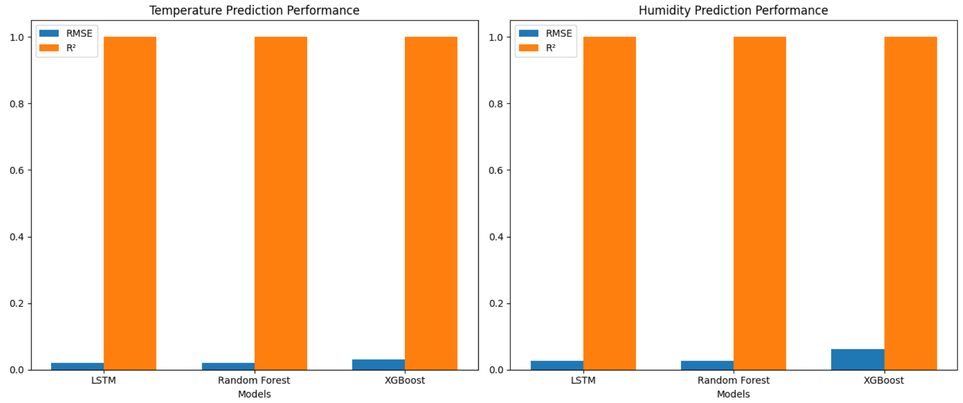

Figure 5: Performance comparison between LSTM, Random Forest, and XGBoost models.

Figure 5 illustrates that all models achieved excellent performance, with Random Forest showing the best overall metrics, followed closely by XGBoost and then LSTM. The remarkably high R^2 values across all models indicate that the feature engineering process successfully captured the key determinants of indoor temperature and humidity.

5.4 Feature Importance Analysis

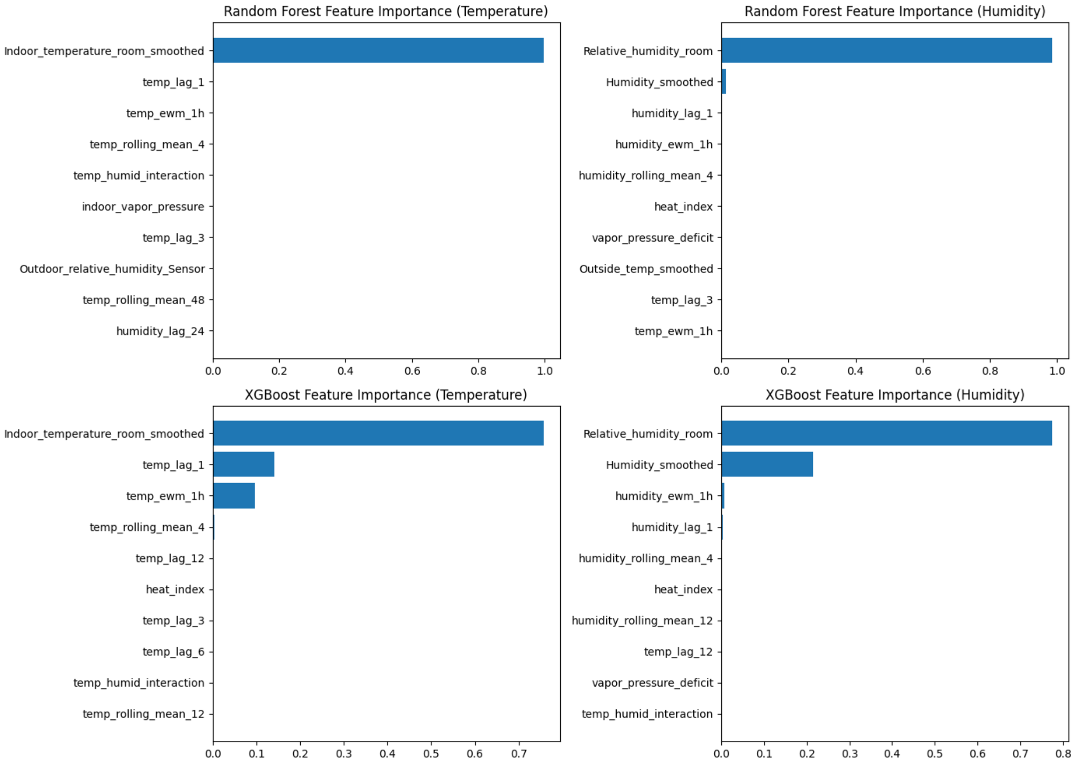

Figure 6: Feature importance analysis for Random Forest and XGBoost models.

Figure 6 reveals that:

- For temperature prediction, the most important features were previous temperature values (both lagged and smoothed), outdoor temperature, and time-related features.
- For humidity prediction, the most important features were previous humidity values, CO2 levels, and vapor pressure variables.
- Both Random Forest and XGBoost models showed similar feature importance rankings, suggesting robust feature selection.

5.5 Environment Simulation Results

An environment simulation was conducted to test the models in a more realistic scenario:

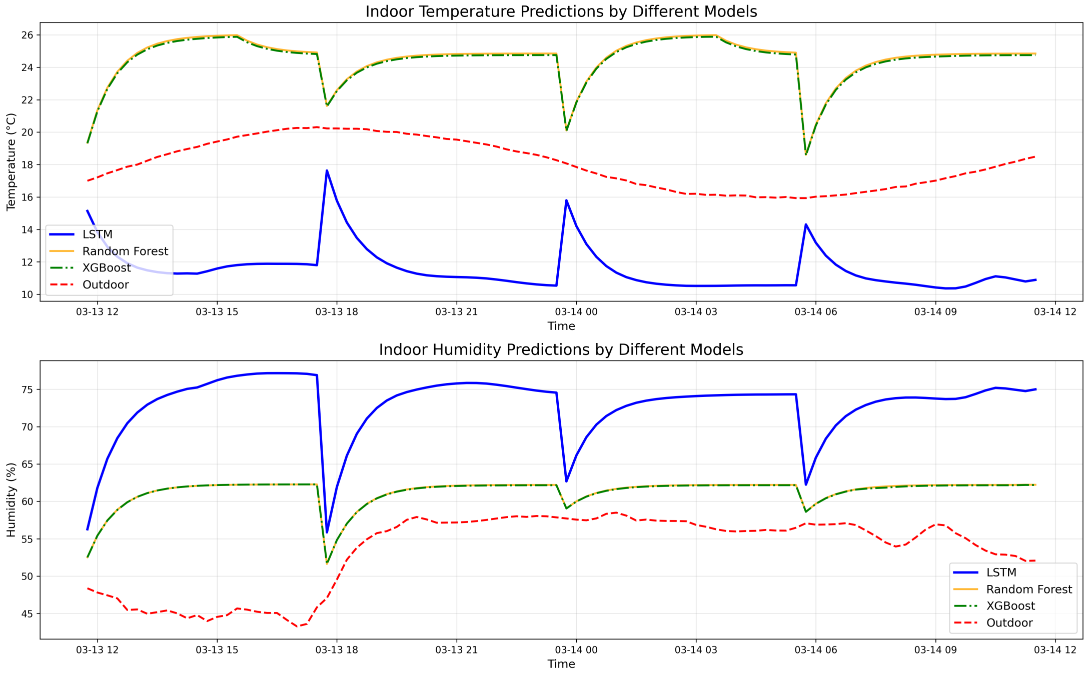

Figure 7: Comparison of model predictions in the environmental simulation.

Figure 7 shows how each model responded to changing outdoor conditions in the simulation. Key observations include:

- All models effectively captured the general patterns of temperature and humidity variations.
- Random Forest and XGBoost models produced smoother predictions compared to the LSTM model.
- The LSTM model showed greater sensitivity to short-term fluctuations, which could be advantageous for capturing sudden environmental changes.
- All models appropriately maintained a buffer between indoor and outdoor conditions, reflecting the realistic insulating properties of buildings.

6 Discussion

6.1 Model Performance Analysis

The excellent performance across all three modeling approaches demonstrates the effectiveness of the feature engineering process. The especially high accuracy of Random Forest and XGBoost models suggests that these ensemble tree-based methods are particularly well-suited to environmental prediction tasks, likely due to their ability to capture non-linear relationships and handle interactions between features.

While the LSTM model showed slightly lower performance metrics than the tree-based models, it offers certain advantages:

- Better handling of sequential dependencies in the data
- More natural incorporation of temporal patterns
- Ability to predict multiple outputs simultaneously through a single model architecture

The extremely high R^2 values (¿0.99) across all models merit careful consideration. Such performance could indicate:

- The effectiveness of the comprehensive feature engineering process
- Strong inherent predictability in indoor environmental conditions
- Potential risk of data leakage despite careful validation procedures

The simulation results provide additional insight, showing how models perform in a more dynamic setting. The differences in prediction patterns between LSTM and tree-based models highlight their complementary strengths, suggesting that hybrid approaches could be valuable in production environments.

6.2 Feature Importance Insights

The feature importance analysis reveals several key insights about indoor environmental dynamics:

- Historical values (lagged features) are the strongest predictors of future conditions, confirming the strong temporal autocorrelation in environmental data
- Outdoor conditions significantly influence indoor temperature but with clear dampening effects
- CO2 levels serve as important indicators for humidity prediction, likely due to their relationship with occupancy and human respiration
- Vapor pressure and heat index calculations provide valuable information beyond raw temperature and humidity measurements

These findings align with building physics principles while providing quantitative evidence of their relative importance. The consistency of important features across different modeling approaches strengthens confidence in these conclusions.

6.3 Practical Applications

The models developed in this research have several practical applications:

- **Predictive HVAC control:** Anticipating temperature and humidity changes to optimize heating, cooling, and ventilation systems
- **Energy optimization:** Reducing energy consumption by proactively adjusting building systems based on forecasted conditions
- **Comfort management:** Maintaining optimal comfort levels by preventing rather than reacting to environmental fluctuations
- **Fault detection:** Identifying anomalies when actual conditions deviate from model predictions

The different models offer trade-offs that system designers should consider. Tree-based models may be preferable when maximum prediction accuracy is the primary goal, while LSTM models might be more suitable when capturing temporal dynamics is essential.

6.4 Limitations

This study has several limitations that should be acknowledged:

- The dataset represents a specific building with its unique thermal characteristics, potentially limiting generalizability
- The extremely high performance metrics suggest a need for additional validation with external datasets
- The simulation environment may not capture all complexities of real-world building systems
- The time period covered by the dataset may not capture all seasonal variations

7 Conclusion

This research demonstrates the effectiveness of machine learning approaches for indoor environmental prediction. All three modeling approaches—LSTM, Random Forest, and XGBoost—achieved exceptional performance, with Random Forest showing marginally better metrics in most cases.

The comprehensive feature engineering process proved crucial to model success, particularly the inclusion of physics-based features, temporal embeddings, and lagged variables. The study confirmed the strong predictability of indoor environmental conditions when appropriate features are included.

The comparative analysis revealed that while tree-based ensemble methods excel at pure prediction accuracy, LSTM networks offer advantages in handling temporal dependencies and multi-output prediction. The simulation results further illustrated how these different approaches behave in practical scenarios.

7.1 Future Work

Several promising directions for future research emerge from this study:

- Development of hybrid models combining the strengths of tree-based and deep learning approaches
- Expansion to multi-building datasets to test generalizability
- Integration with reinforcement learning for autonomous environmental control
- Exploration of attention mechanisms to improve LSTM performance on longer time horizons
- Investigation of transfer learning approaches to adapt models across different building types

7.2 Final Remarks

This project contributes to the growing field of intelligent building systems by providing a rigorous comparison of modeling approaches and demonstrating their practical application. The results suggest that data-driven environmental prediction is not only feasible but highly effective, offering significant potential for improving building efficiency and occupant comfort.

8 Acknowledgments

I would like to express my gratitude to the University of Hertfordshire for providing access to computational resources necessary for this research. Special thanks to the course instructors and peers who provided valuable feedback throughout the project.

References

[1] Kaggle (2022) 'Smart home's temperature - time series forecasting'. Available at: https://www.kaggle.com/competitions/smart-homes-temperature-time-series-forecasting/overview (Accessed: 19 March 2025).

[2] Selvin, S., Vinayakumar, R., Gopalakrishnan, E. A., Menon, V. K. and Soman, K. P. (2020) 'Deep learning-based stock price prediction using LSTM and bi-directional LSTM model', *IEEE Xplore*. Available at: https://ieeexplore.ieee.org/abstract/document/9257950/ (Accessed: 19 March 2025).

[3] Grand, M. (2002) 'Design patterns application in UML', in *Advanced Object-Oriented Design Using UML*. Springer, pp. 35–56. DOI: https://doi.org/10.1007/3-540-45102-1_3.

[4] Alhafidh, B. (2016) 'Design and simulation of a smart home managed by an intelligent self-adaptive system', *ResearchGate*. Available at: https://www.researchgate.net/publication/311374442_Design_and_Simulation_

of_a_Smart_Home_managed_by_an_Intelligent_Self-Adaptive_System (Accessed: 19 March 2025).

[5] Juuso, E. K. (2003) ’Integration of intelligent systems in development of smart adaptive systems’, *ScienceDirect*. Available at: https://www.sciencedirect.com/science/article/pii/S0888613X03001075 (Accessed: 19 March 2025).

[6] Muccini, H. and Vaidhyanathan, K. (2019) ’A machine learning-driven approach for proactive decision making in adaptive architectures’, *IEEE Xplore*. Available at: https://ieeexplore.ieee.org/abstract/document/8712155/ (Accessed: 19 March 2025).

[7] Hochreiter, S. and Schmidhuber, J. (1997) ’Long short-term memory’, *Neural computation*, 9(8), pp. 1735-1780.

[8] Breiman, L. (2001) ’Random forests’, *Machine learning*, 45(1), pp. 5-32.

[9] Chen, T. and Guestrin, C. (2016) ’XGBoost: A scalable tree boosting system’, in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 785-794.

[10] Kuhn, M. and Johnson, K. (2019) *Feature engineering and selection: A practical approach for predictive models*. CRC Press.

[11] Hensen, J. L. and Lamberts, R. (eds.) (2012) *Building performance simulation for design and operation*. Routledge.

[12] Savitzky, A. and Golay, M. J. (1964) ’Smoothing and differentiation of data by simplified least squares procedures’, *Analytical chemistry*, 36(8), pp. 1627-1639.

[13] Wang, S. and Ma, Z. (2008) ’Supervisory and optimal control of building HVAC systems: A review’, *HVAC&R Research*, 14(1), pp. 3-32.

9 Appendix

9.1 Appendix A: Complete Feature List

- Original features: Indoor_temperature_room, Humidity, CO2_room, etc.
- Time features: hour, day, month, day_of_week, is_daytime
- Cyclical features: hour_sin, hour_cos, day_of_week_sin, day_of_week_cos, etc.
- Lag features: temp_lag_1, temp_lag_3, humidity_lag_1, etc.
- Rolling statistics: temp_rolling_mean_4, humidity_rolling_std_24, etc.
- Rate-of-change features: temp_rate_of_change_1h, humidity_rate_of_change_1h
- Exponential features: temp_ewm_1h, humidity_ewm_1h
- Physics-based features: temp_diff_in_out, vapor_pressure, heat_index, etc.
- Smoothed features: Indoor_temperature_room_smoothed, Humidity_smoothed, etc.

9.2 Appendix B: LSTM Model Architecture

```
MultiOutputLSTMModel(
  (lstm): LSTM(68, 128, batch_first=True)
  (dropout): Dropout(p=0.2, inplace=False)
  (fc_temp): Linear(in_features=128, out_features=64, bias=True)
  (fc_humidity): Linear(in_features=128, out_features=64, bias=True)
  (relu): ReLU()
  (out_temp): Linear(in_features=64, out_features=1, bias=True)
  (out_humidity): Linear(in_features=64, out_features=1, bias=True)
)
```