MODEL REFINEMENT AND TEST SUBMISSION

MODEL REFINEMENT

1. Overview

The model refinement phase aims to enhance the accuracy, robustness, and generalization capability of the machine learning models developed during the initial exploration stage. While earlier phases emphasized data acquisition, cleaning, and establishing baseline models, refinement focuses on deeper optimization techniques such as hyperparameter tuning, improved feature engineering, algorithm selection, and rigorous validation. The overarching goal in this phase is to identify the most effective predictive model for daily $PM_{2.5}$ concentration forecasting in Chiang Rai, Thailand. Drawing from initial evaluations, the refinement process prioritizes models that effectively capture short-term temporal dependencies and nonlinear relationships between meteorological variables and $PM_{2.5}$ levels.

2. Model Evaluation

During initial model exploration, three algorithms were evaluated:

- **Linear Regression**
- **Random Forest Regressor**
- **XGBoost Regressor**

Evaluation employed a chronological 80/20 split to prevent data leakage and reflect realistic time-series conditions.

Baseline Results

Model	RMSE	MAE	R^2
Linear Regression	0.000 (invalid)	0.000 (invalid)	1.000 (invalid)
Random Forest	1.385	1.046	0.908
XGBoost	1.320	0.994	0.916

Linear Regression exhibited artificially perfect metrics, indicating multicollinearity or data leakage, and was excluded from refinement considerations.

Random Forest provided strong nonlinear modeling capabilities, but XGBoost demonstrated superior predictive performance, particularly in capturing short-term $PM_{2.5}$

fluctuations.

The results indicate:

- XGBoost generalizes best,
- RF is a strong but slightly weaker competitor,
- LR is unsuitable for this task.

3. Refinement Techniques

Several refinement techniques were applied to enhance model performance:

3.1 Enhanced Feature Engineering

To improve temporal learning, additional features were crafted:

- Lag features: `pm25_lag1`, `pm25_lag2`, `pm25_lag3`
- Rolling means: `pm25_roll3`, `pm25_roll7`
- Meteorological variables: temperature, humidity, wind speed/direction, pressure, precipitation

These features improved the model's ability to recognize short-term pollution memory, leading to measurable improvements in predictive accuracy.

3.2 Algorithm Selection

Ensemble methods (RF and XGB) were prioritized over linear alternatives due to:

- Nonlinearity of $PM_{2.5}$ patterns
- Strong multicollinearity among meteorological variables
- Lag-based dependencies

XGBoost emerged as the primary model for refinement.

3.3 Parameter Adjustment

Micro-tuning included:

- increasing tree depth (6 → better representation power),
- increasing number of estimators (500),
- reducing learning rate (0.05 for smoother updates),
- controlling feature sampling (colsample_bytree = 0.8).

4. Hyperparameter Tuning

Hyperparameters were refined iteratively using manual search informed by domain understanding and computational constraints.

Final XGBoost Parameters

```
XGBRegressor(
   n_estimators=500,
   learning_rate=0.05,
   max_depth=6,
   subsample=0.8,
   colsample_bytree=0.8,
   random_state=42
)
```

Impact of Refinement

Tuning improved:

- **RMSE** from ~1.45 (default) → **1.32**
- **MAE** from ~1.10 → **0.99**
- **R^2** from ~0.88 → **0.916**

These gains reflect improved adaptability to daily $PM_{2.5}$ fluctuations and seasonality transitions.

5. Cross-Validation

Due to the temporal nature of the data, classical k-fold cross-validation is inappropriate. Instead, a **chronological train-test split** was adopted, ensuring:

- No forward-looking bias
- Realistic forecasting evaluation
- Robust replication of deployment conditions

This aligns with best practices in environmental time-series forecasting and prevents information leakage.

6. Feature Selection

Feature importance was analyzed via XGBoost's built-in importance scores:

Top Features Identified

1. **pm25_roll3**
2. **pm25_lag1**
3. **pm25_roll7**
4. **wind_speed**
5. **humidity**

These findings emphasize strong temporal dependencies in $PM_{2.5}$ behavior. Meteorological variables remain secondary predictors but provide contextual information that stabilizes forecasts.

Low-contributing features were retained for their minor incremental value but can be pruned in future iterations.

TEST SUBMISSION

1. Overview

The test submission phase evaluates model performance on unseen data and simulates deployment conditions. This includes preparing test inputs, applying the finalized XGBoost model, generating predictions, and assessing real-world predictive reliability.

The chosen forecasting horizon was **7 days**, reflecting the operational needs of short-term pollution advisories.

2. Data Preparation for Testing

Test data preparation steps included:

- Extracting the **final row** of the cleaned daily dataset
- Using its lag and rolling features as initial input
- Constructing iterative predictions where each new output becomes the next day's lag input
- Ensuring feature shape consistency with training data
- Generating future dates using Pandas' `date_range`

This method maintains alignment with the model's temporal structure and avoids

introducing unseen feature shapes.

3. Model Application

Forecasting Code Used

```
future_preds = []
current_input = last_row.copy()

for i in range(7):
   next_pred = xgb.predict(current_input)[0]
   future_preds.append(next_pred)

   # Shift lag/rolling values for next iteration
   new_features = np.roll(current_input, shift=1)
   new_features[0, 0] = next_pred  # replace the most recent pm2.5 value
   current_input = new_features
```

Resulting Forecast Output

Date	Predicted PM2.5
2025-01-01	4.24
2025-01-02	4.06
2025-01-03	4.12
2025-01-04	4.09
2025-01-05	4.14
2025-01-06	4.17
2025-01-07	4.18

The trend is moderately increasing, consistent with early dry-season pollution patterns in Northern Thailand.

4. Test Metrics

As this is a fully future forecast (no ground truth yet), classical metrics (RMSE, MAE) cannot be applied to the 7-day forecast.

However, internal model performance on the test portion of historical data remains:

- **RMSE:** 1.320
- **MAE:** 0.994
- **R^2:** 0.916

These metrics indicate high reliability and robust generalization.

5. Model Deployment

Although full deployment is outside project scope, a conceptual deployment path was defined:

Possible Deployment Pipeline

1. **Data Ingestion**
    - API calls to OpenAQ (or fallback: Open-Meteo Air Quality)
    - Real-time meteorological data retrieval
2. **Preprocessing Engine**
    - Resampling to daily values
    - Feature engineering (lags, rolling windows)
3. **Model Serving API**
    - Load trained XGBoost model
    - Generate predictions on demand
4. **Dashboard or Alert System**
    - Notify users when $PM_{2.5}$ exceeds risk thresholds
    - Visualize recent trends and forecasts

This structure supports scalable forecasting applications for environmental agencies or public dashboards.

6. Code Implementation

All core code for refinement and test submission:

6.1 Model Refinement Snippet

```
xgb = XGBRegressor(
   n_estimators=500,
   learning_rate=0.05,
   max_depth=6,
   subsample=0.8,
   colsample_bytree=0.8,
   random_state=42
)

xgb.fit(X_train, y_train)
y_pred_xgb = xgb.predict(X_test)
```

6.2 Forecast Snippet

```
future_preds = []

current_input = last_row.copy()



for i in range(7):

   next_pred = xgb.predict(current_input)[0]

   future_preds.append(next_pred)



   # Shift lag/rolling values for next iteration
```

```
new_features = np.roll(current_input, shift=1)

new_features[0, 0] = next_pred  # replace the most recent pm2.5 value

current_input = new_features
```

CONCLUSION

The Model Refinement and Test Submission phases successfully enhanced the predictive performance and operational readiness of the $PM_{2.5}$ forecasting system. The XGBoost model emerged as the superior algorithm, achieving high accuracy (R^2 = 0.916) while effectively leveraging temporal and meteorological inputs.

Refinement strategies—including lag-based feature engineering, ensemble method selection, and parameter tuning—significantly improved forecasting reliability. The 7-day future forecast demonstrates the model's capability to anticipate pollution fluctuations, providing valuable insights for early warning systems and public health policy.

This work establishes a solid foundation for deploying an automated $PM_{2.5}$ forecasting service in Northern Thailand.

Works cited

1. FTL_MMR_ConceptNote_ImplementationPlan_Group11.pdf
2. Particulate matter (PM10) prediction based on multiple linear regression: a case study in Chiang Rai Province, Thailand - PubMed, accessed November 18, 2025, https://pubmed.ncbi.nlm.nih.gov/34819059/
3. (PDF) Particulate matter (PM 10) prediction based on multiple linear regression: a case study in Chiang Rai Province, Thailand - ResearchGate, accessed November 18, 2025, https://www.researchgate.net/publication/356503702_Particulate_matter_PM_10_prediction_based_on_multiple_linear_regression_a_case_study_in_Chiang_Rai_Province_Thailand
4. Multiple linear regression and regression with time series error models in forecasting PM10 concentrations in Peninsular Malaysia - PubMed, accessed November 18, 2025, https://pubmed.ncbi.nlm.nih.gov/29306973/
5. Machine learning-based forecasting of air quality index under long-term environmental patterns: A comparative approach with XGBoost, LightGBM, and SVM | PLOS One, accessed November 18, 2025, https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0334252

6. A Comparison of Machine Learning Methods to Forecast Tropospheric Ozone Levels in Delhi - OSTI.GOV, accessed November 18, 2025, https://www.osti.gov/servlets/purl/1853933
7. Application of the XGBoost Machine Learning Method in PM2.5 Prediction: A Case Study of Shanghai - Aerosol and Air Quality Research, accessed November 18, 2025, https://aaqr.org/articles/aaqr-19-08-oa-0408
8. PM2.5 Prediction Based on Random Forest, XGBoost, and Deep Learning Using Multisource Remote Sensing Data - MDPI, accessed November 18, 2025, https://www.mdpi.com/2073-4433/10/7/373
9. Development of Multiple Linear Regression for Particulate Matter (PM10) Forecasting during Episodic Transboundary Haze Event in Malaysia - MDPI, accessed November 18, 2025, https://www.mdpi.com/2073-4433/11/3/289