AgriFrost-AI: Frost Risk Forecasting Methods and Evaluation for California's Central Valley

AgriFrost-AI Team

December 8, 2025

Abstract

Frost remains a critical meteorological risk for high-value horticultural crops in California, where severe radiation frost can cause substantial yield losses. This study constructs an end-to-end AgriFrost-AI system for practical frost forecasting, using hourly observations from 18 CIMIS stations (2010–2025, 2.36 million records). We propose a feature configuration matrix (ABCD) framework crossing single-station/multi-station with raw/engineered features, systematically comparing 7 models across four forecast horizons (3, 6, 12, 24 hours).

The LightGBM model with neighborhood aggregation features (Matrix C) achieves ROC-AUC 0.9972 and PR-AUC 0.7282 at 3-hour horizon, maintaining ROC-AUC 0.9877 and PR-AUC 0.4671 at 24-hour horizon. Temperature prediction accuracy is high across all horizons (MAE: 1.16–1.85 °C, RMSE: 1.58–2.42 °C), with LOSO evaluation showing no performance decline and slight improvements (24h ROC-AUC +0.35 pp, MAE: 1.93 °C), demonstrating spatial robustness. Feature importance analysis reveals soil temperature gradients, dew point differences, and vapor pressure deficits as the most valuable signals. Spatial aggregation features show 36.7% average PR-AUC improvement over single-station features, with optimal radius varying by forecast horizon (3h: 60 km, 24h: 180 km), reflecting spatial-temporal coupling. Probability calibration is excellent (ECE < 0.004, Brier Score < 0.005), enabling direct use for decision support.

Key contributions: (1) systematic feature configuration framework for quantitative strategy evaluation; (2) validation of neighborhood aggregation's core value in frost forecasting; (3) reliable performance estimates through large-scale experiments and strict spatial generalization; (4) excellent probability calibration (ECE < 0.004), enabling direct mapping to farm decision thresholds; (5) comprehensive class imbalance handling (built-in mechanisms for tree models) and F2-score threshold optimization for agricultural decision-making. This study bridges ground observations, physical process understanding, and agricultural decision support, providing a practical example for deploying machine learning models in field applications.

Contents

1 Introduction

Frost has long been one of the major meteorological hazards facing high-value fruits, vegetables, and nut crops in California, particularly during the flowering and early fruit stages, where short-duration severe radiation frost can cause widespread yield losses or even total crop failure. Traditional protection strategies rely on empirical judgment, limited manual observations, and mesoscale numerical weather prediction, but often struggle to provide timely and reliable field-level warnings under complex terrain and strong microclimatic conditions.

The F3 Innovate Frost Risk Forecasting Challenge proposes an evaluation framework close to production practice: based on multi-station, multi-year ground observations, participating teams are required to simultaneously output frost probabilities and temperature estimates for four forecast windows (3, 6, 12, and 24 hours), quantify probability calibration quality, and test model spatial generalization through Leave-One-Station-Out (LOSO) evaluation. The AgriFrost-AI project addresses the following research questions:

- How to design feature sets that balance near-surface physical mechanisms with machine learning usability on highly imbalanced, spatially heterogeneous hourly meteorological data?
- How to introduce neighboring station information to capture local processes such as cold air pooling and inversion layer structure without significantly increasing computational costs?
- Can the frost probabilities provided by the model be directly incorporated into farm standard operating procedures (SOP) after calibration, rather than serving only as relative ranking indicators?

The main contributions of this paper are summarized as follows:

1. Constructed a unified frost risk dataset based on 18 CIMIS stations covering 2010–2025, with complete data quality and QC analysis.
2. Leveraging declarative CLI and DataPipeline, unified aggregation of large-scale experimental configurations combining feature configuration matrices (ABCD) with 7 model families into result files (Supplementary Tables S2–S4), enabling traceable comparisons by matrix, radius, and forecast horizon.
3. Proposed a feature configuration matrix (ABCD) framework crossing single-station/multi-station with raw/engineered features, systematically comparing performance differences across different matrices and model families, revealing transferable patterns between spatial aggregation radius and forecast windows.
4. Strictly evaluated model spatial generalization through LOSO schemes, demonstrating the gains of neighborhood aggregation features across different forecast windows.
5. Mapped calibrated frost probabilities and temperature predictions to specific protection decision thresholds, providing quantitative basis for growers to develop SOPs, and discussed cost-sensitive strategies of "better to over-warn than to miss."

2 Related Work

Frost risk assessment and short-term temperature prediction have been extensively studied in agricultural meteorology, numerical weather prediction, and machine learning communities. Traditional methods are mostly based on empirical formulas, statistical regression, or downscaling of

mesoscale numerical models (e.g., WRF), focusing on radiation cooling, surface energy balance, and cold air sinking and accumulation processes. In recent years, with the proliferation of automatic weather stations and reanalysis data, near-surface meteorological prediction methods based on random forests, gradient boosting trees, and deep neural networks have gradually emerged. Some work incorporates satellite remote sensing, terrain data, and reanalysis fields as inputs to generate high-resolution surface temperature and frost risk maps.

Compared to the above research, this paper focuses more on the following aspects: First, systematic comparison of multiple models using unified datasets and evaluation metrics on a unified challenge platform; Second, capturing cold air pooling and local inversion structures through explicit neighboring station aggregation features rather than relying solely on gridded interpolation; Third, exploring the usability of model outputs at the grower operational level from the perspectives of probability calibration and decision support.

3 Data and Study Region

This section introduces the CIMIS ground observation data used in the study, statistical characteristics of frost events, main observed variables and their physical significance, and provides an overview of overall data quality and QC.

3.1 Observation Sources and Spatial Coverage

The hourly meteorological observations used in this study come from the California Irrigation Management Information System (CIMIS), covering 18 automatic weather stations in California's Central Valley and surrounding foothill areas. Stations are distributed in a north-south band along the Central Valley, extending from the Sacramento Plain to the Bakersfield region, spanning diverse microclimatic environments including cold air pooling-prone areas, elevation transition zones, and high evapotranspiration agricultural zones. Figure 1 shows the spatial distribution of all stations.

The data spans 2010–2025, totaling approximately 2.36 million hourly records. Each record contains core variables including air temperature, dew point, relative humidity, wind speed and direction, solar radiation, soil temperature, vapor pressure, reference evapotranspiration (ETo), and includes official CIMIS Quality Control (QC) flags. Station-level metadata includes station number, name, CIMIS region, county/city, latitude/longitude, elevation, GroundCover, start/end dates, and whether it is an ETo station, used for spatial aggregation and LOSO grouping. The complete list is provided in Supplementary Table S1 (`Supplementary/supplementary_table_S1_stations.csv`). Raw data and processing scripts are hosted in the GitHub repository for version tracking and reproducibility.

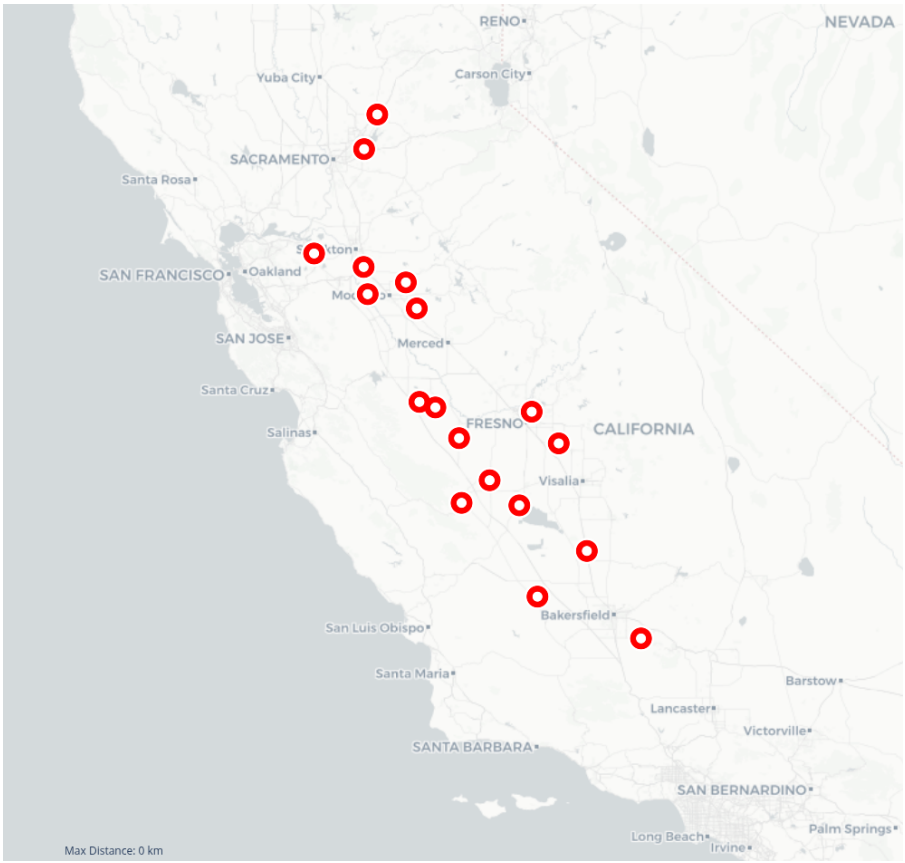

Figure 1: Spatial distribution of 18 CIMIS stations in the study region

3.2 Frost Event Distribution and Seasonal Characteristics

Frost events in this paper are defined as hourly observations with air temperature below 0 °C. Figure 2 shows the distribution of frost events across calendar months, revealing strong seasonality: December and January together account for approximately 77% of all frost events, February accounts for about 13%, and other months contribute very little. During April–October, frost events are nearly zero.

In terms of overall proportion, frost events account for only about 0.87% of all hourly records, representing a highly imbalanced classification task. This characteristic directly affects model training and evaluation: on one hand, metrics that focus more on minority class identification (such as PR-AUC) need to be adopted; on the other hand, attention must be paid to avoiding systematic bias caused by extreme imbalance in probability calibration and decision threshold design.

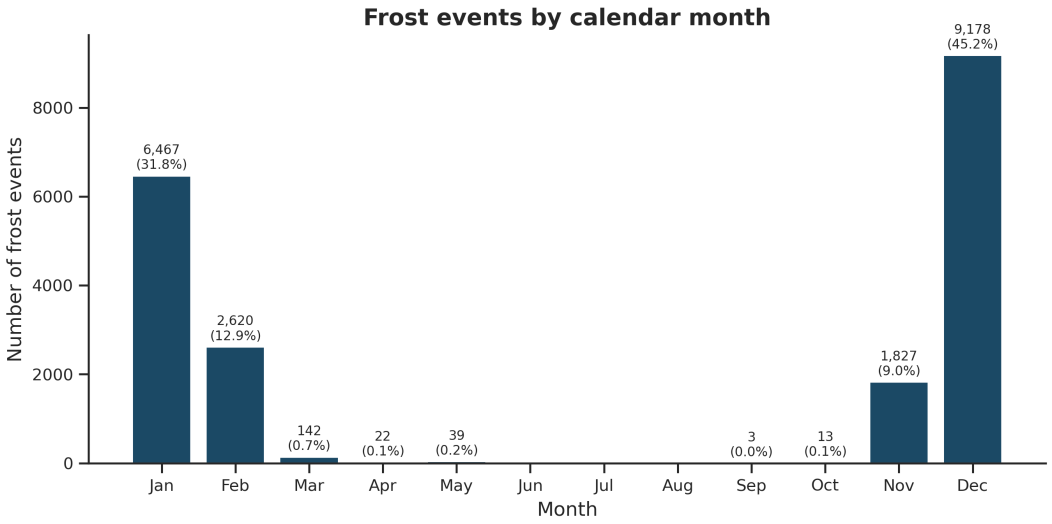

Figure 2: Distribution of frost events by month (2010–2025, 18 stations combined)

3.3 Observed Variables and Physical Significance Overview

The dozen core meteorological variables provided by CIMIS stations are used to characterize surface energy balance, atmospheric state, and soil heat storage, all closely related to frost formation mechanisms. Main variables include:

- **Air Temp (°C)**: Near-surface air temperature, the direct target variable for frost monitoring and prediction.
- **Dew Point (°C)** and **Rel Hum (%)**: Together characterize air moisture content and saturation level, determining condensation and radiation cooling efficiency.
- **Wind Speed (m/s)** and **Wind Dir (0–360)**: Reflect boundary layer mixing intensity and cold air transport pathways. Weak wind or calm conditions are more conducive to radiation frost formation.
- **Sol Rad (W/m^2)**: Solar radiation flux, controlling daytime surface heat storage, with important influence on the upper limit of heat that can be released at night.
- **Soil Temp (°C)**: Shallow soil temperature, reflecting heat storage exchange between surface and near-surface layers.
- **Vap Pres (kPa)**: Vapor pressure, an absolute measure of moisture content, closely related to dew point and relative humidity.
- **ETo (mm)**: Reference evapotranspiration, comprehensively reflecting evapotranspiration demand under radiation, temperature, wind speed, and humidity conditions, with physical connection to nighttime surface cooling rates.

These variables form the basis for subsequent lag features, rolling statistics, harmonic features, and neighborhood aggregation features, providing machine learning models with an input space consistent with physical processes.

3.4 Data Quality and QC Overview

All observations include official CIMIS-generated QC flags indicating whether the physical quantity passed automatic and manual validation. We follow CIMIS recommended guidelines, retaining only "blank/pass" and "Y" flags, with all others (including M, Q, R, S, P, etc.) treated as unavailable. After removing sentinel values, forward filling is performed station by station.

Overall, from 2010–2025, there are approximately 2.36 million hourly records, of which only about 1.71% of rows are flagged as missing or unavailable for at least one key variable, indicating generally high observation quality. Figure 3 shows the contribution proportion of low-quality records across different stations. Low-quality data is relatively dispersed across stations, with only a few stations (e.g., 205, 194, 124) having slightly higher proportions, but no obvious regional systematic bias is observed.

At the variable level, QC anomalies are unevenly distributed across different observed quantities (Figure 4). Reference evapotranspiration ETo accounts for approximately 27.8% of all low-quality records, soil temperature about 20.4%, and wind speed about 10.1%. Relative humidity and dew point each contribute about 8.6%, and vapor pressure about 7.3%. The core frost observation variable—air temperature—has an anomaly proportion of only 6.2% of low-quality records, corresponding to about 0.1% of all observations, further validating the suitability of this dataset for frost analysis and prediction tasks.

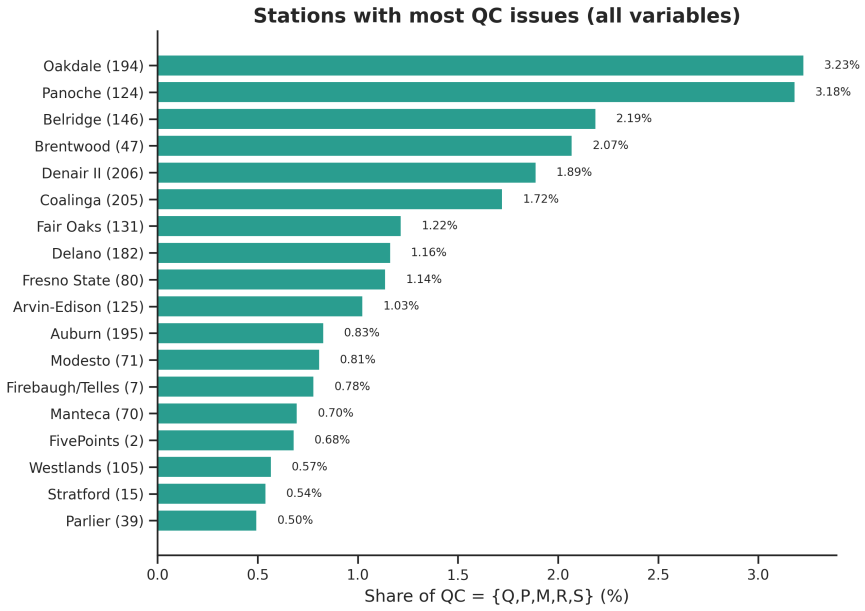

Figure 3: Relative contribution of low-quality (Bad QC) records by station

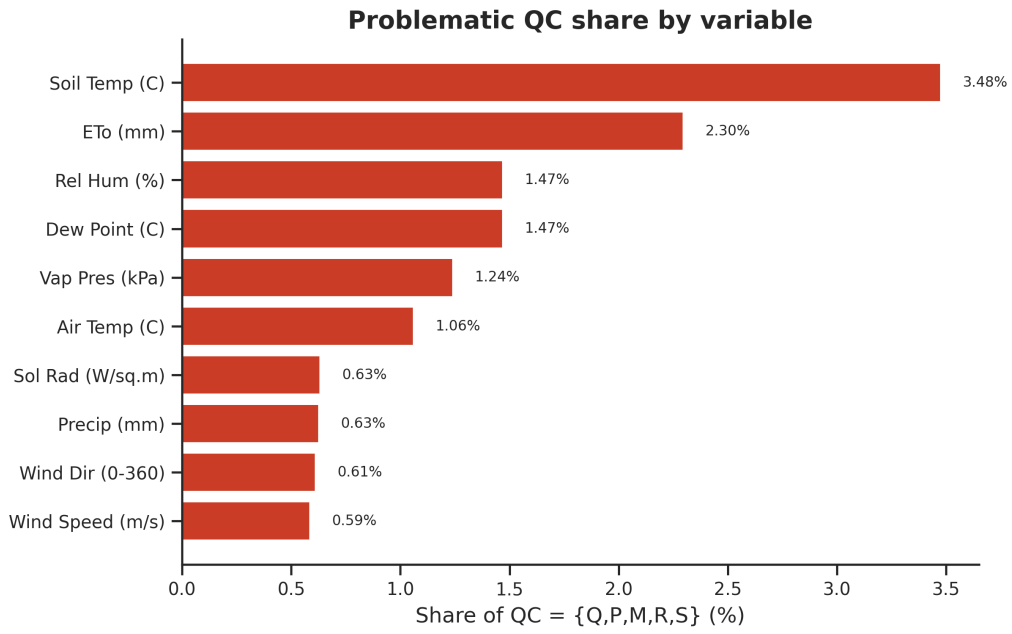

Figure 4: Distribution of low-quality (Bad QC) records by meteorological variable

4 Methods

This section introduces AgriFrost-AI's data preprocessing and QC pipeline, feature configuration matrix (ABCD), feature engineering, feature selection and feature importance analysis, model families and training configuration, and training/validation split and evaluation metrics.

4.1 Data Preprocessing and QC Pipeline

The unified `DataCleaner` pipeline includes the following steps:

1. **Data aggregation and time standardization**: Merge station CSV/Parquet files, unify time to local solar time, and attach station metadata.

2. **Quality control and sentinel value processing**: Parse all quality fields starting with `qc`, retain only "blank/pass" and "Y" according to CIMIS standards, convert all other flags to missing; simultaneously replace sentinel values such as -6999 and -9999 with missing.

3. **Missing value handling**: Group by station, use forward filling for short sequence gaps, retain missing masks for long sequence gaps and key variable missing, enabling models to explicitly perceive observation incompleteness.

4. **Label generation**: Generate frost binary classification labels and temperature regression targets for four forecast windows (3, 6, 12, 24 hours) in one pass on cleaned time series, ensuring subsequent model training uses the same label system.

4.2 Feature Configuration Matrix (ABCD)

To systematically evaluate the impact of different spatial scopes and feature complexity on frost forecast model performance, this study constructs a two-dimensional feature configuration matrix

framework. This framework is designed based on two orthogonal dimensions: (1) **Spatial scope**: Single-station vs. multi-station spatial aggregation; (2) **Feature complexity**: Raw observed variables vs. engineered features. The cross-combination of these two dimensions forms four feature configurations, denoted as matrices A, B, C, and D, as shown in Table 1.

Design motivation: This matrix framework aims to quantitatively evaluate the following scientific questions through controlled variable experiments: (1) The gain of time series engineered features (lags, rolling statistics, derived variables) for single-station models; (2) The contribution of spatial aggregation statistics (neighborhood mean, gradient, range, etc.) to capturing cold air pooling and inversion layer formation; (3) The joint effect of temporal and spatial features, and the performance-complexity trade-off in high-dimensional feature spaces. This framework provides a systematic experimental design basis for subsequent ablation studies and feature importance analysis.

Detailed feature composition and generation methods for each matrix are described in Section 4.3, and feature importance analysis methods are described in Section 4.6.

Table 1: Feature Configuration Matrix (ABCD) Overview

Matrix	Spatial Config	# Features	Feature Composition
A	Single-station (no neighbors)	16 dim	12 raw CIMIS variables (air temp, dew point, rel hum, wind speed, wind dir, sol rad, soil temp, vap pres, ETo, precip, hour, Julian day) + temporal harmonic encoding (`hour_sin/cos`, `month_sin/cos`)
B	Single-station (no neighbors)	278 dim	Feature engineering pipeline: raw variables (12 dim) + temporal features (15 dim) + lag features (50 dim: 10 variables × 5 lags) + rolling window statistics (180 dim: 9 variables × 4 windows × 5 functions) + derived meteorological features (3 dim) + radiation features (4 dim) + wind features (6 dim) + humidity features (4 dim) + trend features (1 dim) + station static features (4 dim)
C	Multi-station aggregation (radius 20–200 km)	534 dim	Raw variables (12 dim) + neighborhood aggregation statistics (8 aggregation methods for 27 numeric variables: mean, max, min, std, median, distance-weighted mean, gradient, range, totaling 216 dim = 27 variables × 8 methods) + missing masks (293 dim: missing ratio for each aggregation feature) + temporal harmonic encoding (2 dim: `day_of_year_sin/cos`) + other features (11 dim: temporal discrete features, derived meteorological features, `has_neighbors` indicator)
D	Multi-station aggregation + engineered features	818 dim	Matrix B single-station feature engineering (278 dim) + Matrix C neighborhood aggregation part (216 dim neighborhood aggregation + 299 dim missing masks) + other features (43 dim)

Matrix A (Single-station + Raw Features) Matrix A serves as the baseline configuration, using only 12 raw CIMIS observed variables (air temperature, dew point, relative humidity, wind speed, wind direction, solar radiation, soil temperature, vapor pressure, ETo, precipitation, hour, Julian day) plus temporal harmonic encoding (`hour_sin/cos`, `month_sin/cos`), totaling 16 dimensions. This configuration introduces no lags, rolling statistics, or spatial aggregation, aiming to evaluate the inherent discriminative ability of raw observations and provide a performance bench-

mark for subsequent feature engineering and spatial enhancement.

Matrix B (Single-station + Engineered Features) Matrix B overlays a complete single-station feature engineering pipeline on Matrix A, generating 278 candidate features. This configuration aims to evaluate the performance gain of time series engineered features (lags, rolling window statistics, derived meteorological variables) for single-station models. In actual experiments, all models are trained using the full feature set (278 dimensions) to fully assess the contribution of feature engineering. Detailed feature engineering methods are described in Section 4.3.

Matrix C (Neighborhood Aggregation + Raw Features) Matrix C overlays multi-station spatial aggregation statistics on Matrix A's raw variables. This configuration aims to evaluate the contribution of spatial information to frost forecasting, particularly spatial patterns such as cold air pooling and inversion layer formation. For 27 numeric variables (12 raw CIMIS variables + 15 temporal features), neighborhood aggregation is performed, computing 8 aggregation statistics (mean, max, min, std, median, distance-weighted mean, gradient, range) under specified radius thresholds (systematically tested 20–200 km in experiments, step size 20 km), generating 216 neighborhood aggregation features. Additionally, the system generates missing mask features (293 dimensions) to handle neighbor station data missing issues, including missing masks for neighborhood aggregation features, variable missing ratios, missing masks for missing ratio features, and missing masks for other features. Combined with raw variables (12 dim), temporal harmonic encoding (2 dim), and other features (11 dim), the total is 534 dimensions. Detailed neighborhood aggregation methods and missing mask calculations are described in Section 4.3.

Matrix D (Neighborhood Aggregation + Engineered Features) Matrix D combines Matrix B's single-station feature engineering (278 dim) with Matrix C's neighborhood aggregation part (216 dim neighborhood aggregation + 299 dim missing masks), forming an 818-dimensional high-dimensional feature space. This configuration aims to evaluate the joint effect of time series engineered features and spatial aggregation statistics, exploring the performance-complexity trade-off in complex feature spaces. Compared to Matrix C, Matrix D's missing mask features (299 dim) are 6 dimensions more than Matrix C (293 dim), because Matrix D also generates missing masks for engineered features (such as lags, rolling statistics) to improve model robustness under sparse data conditions. Additionally, Matrix D includes additional other features (43 dim), including wind features, humidity features, radiation features, trend features, station static features, geographic features, and interaction features. Note that Matrix D contains feature duplication (e.g., temporal features appear in both Matrix B and Matrix C's "other features"), which are automatically deduplicated in actual implementation to ensure each feature appears only once in the feature space. Detailed feature engineering methods are described in Section 4.3, and feature selection strategies are described in Section 4.6.

4.3 Feature Engineering

Feature engineering is a key step in transforming raw observational data into predictive feature representations. For frost forecasting tasks, effective feature engineering needs to simultaneously capture temporal patterns (diurnal cycles, annual cycles, historical dependencies), spatial associations (cold air pooling, inversion layers, spatial gradients), and physical relationships (energy balance, radiation cooling, convective mixing). This section systematically describes the design principles, theoretical basis, implementation methods, and feature composition of the feature engineering pipeline.

4.3.1 Design Principles and Theoretical Framework

The feature engineering pipeline design follows four core principles:

(1) **Temporal leakage prevention**: All features are computed after grouping by station and strict temporal sorting, ensuring feature values depend only on historical information, strictly preventing future information leakage into historical features. This principle is crucial for time series prediction tasks, especially in LOSO (Leave-One-Station-Out) evaluation scenarios, where temporal ordering of feature computation must be ensured.

(2) **Dependency relationship management**: Feature construction follows a clear dependency order (temporal features → lag features → rolling statistics → derived features), ensuring computational correctness and reproducibility. This principle avoids circular dependencies between features and guarantees idempotency of feature computation.

(3) **Physical significance orientation**: Prioritize constructing meteorological composite features with clear physical or agricultural significance, rather than blind combinations. This principle is based on theoretical foundations of meteorology and agricultural meteorology, ensuring features can capture physical mechanisms of frost formation (e.g., radiation cooling, inversion layer formation, cold air sinking).

(4) **Robustness design**: Explicit handling of missing values, outliers, and boundary conditions, ensuring features can still be computed under sparse data conditions. This principle improves model generalization in real data environments through missing masks, numerical clipping, conditional computation, etc.

The feature engineering pipeline is divided into two parts: single-station feature engineering (for matrices A/B) and neighborhood aggregation features (for matrices C/D). Detailed feature lists, calculation formulas, naming conventions, and feature importance analysis results are provided in Supplementary Material S1 (`Supplementary/supplementary_S1_feature_list.pdf`).

4.3.2 Single-Station Feature Engineering

Single-station feature engineering is enabled in matrices A and B, mainly including the following feature categories:

Temporal Features (15 dimensions) Temporal features are fundamental for frost forecasting because frost events exhibit strong diurnal and annual cycle patterns. Three types of temporal encodings are extracted from datetime columns:

- **Discrete encoding** (6 dim): `hour` (0–23), `month` (1–12), `day_of_year` (1–366), `day_of_week` (0–6), `season` (1–4: spring/summer/fall/winter), `is_night` (binary, 1 for 18:00–06:00, 0 otherwise). Discrete encoding facilitates models learning differentiated patterns across different time periods, e.g., nighttime periods (`is_night=1`) typically correspond to radiation cooling and temperature decline.
- **Cyclic encoding** (8 dim): `hour_sin/cos`, `month_sin/cos`, `day_of_year_sin/cos`, `day_progress_sin/cos`. Where `day_progress` is normalized hour progress (0–1), calculated as $\texttt{day\_progress} = \texttt{hour}/24$. Cyclic encoding uses trigonometric functions:

$$\texttt{hour\_sin} = \sin\left(\frac{2\pi \cdot \texttt{hour}}{24}\right), \quad \texttt{hour\_cos} = \cos\left(\frac{2\pi \cdot \texttt{hour}}{24}\right)$$

Similarly, month and day of year use period lengths $T = 12$ and $T = 365.25$ for encoding. This encoding avoids boundary discontinuities (e.g., jump between hour 23 and 0), enabling models to learn smooth periodic patterns, crucial for capturing nighttime cooling trends and seasonal variations.

- **Agriculture-related features** (1 dim): `frost_season_indicator`, marking high frost risk period from December to April (California region). This feature directly encodes the temporal window for agricultural frost warnings, helping models focus on high-risk periods.

Theoretical basis: Temporal features capture two key time scales of frost occurrence: (1) **Diurnal cycle**: Based on radiation cooling theory, nighttime surface longwave radiation loss leads to temperature decline, typically reaching minimum around 4–6 AM, which is the main physical mechanism of frost formation; (2) **Annual cycle**: Based on climatological principles, winter and early spring are high frost risk periods, when solar radiation intensity is low, cold air activity is frequent, and temperature fluctuations and extreme low temperature events are more common.

Lag Features (50 dimensions) Lag features capture historical states of meteorological variables, having important value for predicting future temperature changes. Based on time series analysis theory, meteorological variables have temporal autocorrelation, and historical states have informational value for predicting the future.
Variable and lag configuration:

- **Variables**: 10 core variables (air temp, dew point, ETo, precipitation, relative humidity, soil temp, solar radiation, wind direction, wind speed, vapor pressure)
- **Lag windows**: 1, 3, 6, 12, 24 hours
- **Total features**: $10 \times 5 = 50$ dimensions

Design considerations:

- **Multi-scale lags**: 1-hour lag captures short-term fluctuations and rapid changes, 3–6 hour lags reflect medium-term trends and weather system evolution, 12–24 hour lags capture diurnal cycle patterns and day-night temperature differences
- **Station-grouped computation**: All lag features are computed grouped by station (`groupby(station_id)`), ensuring no cross-station information leakage, which is crucial for LOSO evaluation
- **Temporal alignment**: Ensure data is strictly sorted by station and time before lag computation to prevent temporal leakage

Calculation formula and naming:

- Calculation formula: $x_{\text{lag},h}(t) = x(t-h)$, where $x(t-h)$ represents the value of variable x h hours ago
- Naming format: {`variable`}`_lag_`{`hours`}
- Example: `Air Temp (C)_lag_12` represents air temperature 12 hours ago

Theoretical basis: These features help models learn inertial effects and historical dependencies of temperature changes, capturing memory characteristics of meteorological systems.

Rolling Window Statistics (180 dimensions) Rolling window statistics capture distribution characteristics of variables within time windows, having important significance for identifying trends, volatility, and extreme values. Based on sliding window analysis theory, rolling statistics can smooth noise, capture trends, and identify outliers.
Variable and window configuration:

- **Variables**: 9 core variables (air temp, dew point, ETo, precipitation, relative humidity, soil temp, solar radiation, wind speed, vapor pressure)
- **Excluded variables**: Wind direction does not participate in rolling statistics, as it is an angular variable requiring circular statistics
- **Time windows**: 3, 6, 12, 24 hours
- **Total features**: $9 \times 4 \times 5 = 180$ dimensions

Theoretical basis for statistic selection:

- **Mean**: $\bar{x}_w = \frac{1}{n}\sum_{i=1}^{n} x_i$, reflects average state within window, captures trend direction
- **Minimum**: $x_{\min} = \min_i x_i$, identifies extreme values, especially important for frost warning (minimum temperature directly relates to frost risk)
- **Maximum**: $x_{\max} = \max_i x_i$, identifies extreme values
- **Standard deviation**: $\sigma_w = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x}_w)^2}$, quantifies volatility, high volatility may indicate unstable atmospheric conditions
- **Sum**: $\sum_{i=1}^{n} x_i$, has physical significance for cumulative quantities (e.g., precipitation, ETo)

Implementation details:

- Rolling features are computed grouped by station, ensuring no cross-station information leakage
- Use `min_periods=1` to maximize data utilization (compute statistics even if only 1 valid value in window)
- Naming format: {`variable`}`_rolling_`{`window`}`h_`{`statistic`}
- Example: `Air Temp (C)_rolling_24h_mean` represents average air temperature over the last 24 hours, `Soil Temp (C)_rolling_12h_min` represents minimum soil temperature over the last 12 hours

Derived Meteorological Features (3 dimensions) Derived meteorological features compute composite variables based on physical relationships and meteorological principles, capturing interactions between variables.
Feature list:

- `wind_chill`: Wind chill index, quantifies the effect of wind speed on apparent temperature. Calculated when air temp ¡ 10°C, formula: $13.12 + 0.6215T - 11.37V^{0.16} + 0.3965TV^{0.16}$ (where T is air temperature °C, V is wind speed km/h). In frost scenarios, high wind speed accelerates heat loss, reducing apparent temperature.

- `heat_index`: Heat index, calculated under high temperature and high humidity conditions (air temp ¿ 80°F and relative humidity ¿ 40%). Although this feature value usually equals air temperature in frost scenarios, retaining it helps models learn complete temperature-humidity relationships.

- `soil_air_temp_diff`: Soil temperature minus air temperature (`Soil Temp` − `Air Temp`), reflects surface energy exchange direction. Positive values indicate soil temperature higher than air temperature (common during daytime), negative values indicate soil temperature lower than air temperature (common during nighttime). This feature has important value for identifying inversion layers and radiation cooling processes.

Theoretical basis: These composite features encode nonlinear relationships between meteorological variables, based on meteorological and thermodynamic principles. Wind chill index is based on convective heat transfer theory, quantifying the effect of wind speed on apparent temperature; heat index is based on heat balance equations, quantifying apparent temperature under high temperature and high humidity conditions; soil-air temperature difference is based on surface energy balance, reflecting relative intensity of radiation cooling and convective exchange. These features help models understand how physical processes (e.g., radiation cooling, convective exchange, evaporative cooling) affect frost formation.

Radiation-Related Features (4 dimensions) Solar radiation is the core driving factor of surface energy balance, directly affecting daytime warming and nighttime cooling processes.
Feature list:

- `sol_rad_change_rate`: Solar radiation change rate (`Sol Rad`$(t)-$`Sol Rad`$(t-1)$), captures short-term fluctuations in radiation intensity, reflects cloud cover changes and atmospheric transparency

- `daily_solar_radiation`: Daily cumulative radiation, accumulated from 06:00 to current time. This feature quantifies total daytime energy input, high cumulative radiation usually corresponds to stronger daytime warming, potentially affecting nighttime cooling rate

- `nighttime_cooling_rate`: Nighttime cooling rate, calculated only when `is_night=1`. This feature directly captures the radiation cooling process, a key signal for frost forecasting

- `radiation_temp_interaction`: Interaction term between radiation and temperature (`Sol Rad` × `Air Temp`), captures nonlinear effects of radiation on temperature

Theoretical basis: Radiation features are based on surface energy balance theory. According to Stefan-Boltzmann law, surface longwave radiation loss is proportional to the fourth power of temperature. When there is no solar radiation input at night, the surface continuously loses energy, causing temperature decline. High daytime radiation causes surface warming, low nighttime radiation (close to 0) causes radiation cooling, which is the main physical mechanism of frost formation. Radiation change rate captures cloud cover changes and atmospheric transparency fluctuations, daily cumulative radiation quantifies total daytime energy input, nighttime cooling rate directly reflects radiation cooling intensity.

Wind Features (6 dimensions) Wind field features are crucial for frost forecasting because wind speed affects convective mixing intensity, and wind direction affects cold air pathways.
Feature list:

- `wind_dir_sin/cos`: Wind direction cyclic encoding, converts 0–360 degree angles to $\sin(\theta)$ and $\cos(\theta)$, avoiding angular boundary discontinuities (difference between 359° and 1°)
- `wind_dir_category`: Wind direction categorical encoding, divides 0–360 degrees into 4 quadrants (north, east, south, west), facilitating models learning different wind direction effects on frost risk patterns
- `wind_speed_change_rate`: Wind speed change rate ($\texttt{Wind Speed}(t) - \texttt{Wind Speed}(t-1)$), captures dynamic changes in wind field
- `calm_wind_duration`: Calm wind duration, defined as continuous duration with wind speed ¡ 1.0 m/s. Calm conditions favor radiation cooling, an important prerequisite for frost formation
- `wind_dir_temp_interaction`: Interaction term between wind direction and temperature, captures differential effects of different wind directions on temperature changes (e.g., dry cold wind from inland vs. moist wind from ocean)

Theoretical basis: Wind field features are based on atmospheric boundary layer theory. According to mixed layer theory, under low wind speed (calm) conditions, convective mixing weakens, turbulent exchange coefficient decreases, favoring near-surface cold air accumulation and inversion layer formation, thereby increasing frost risk. Calm wind duration quantifies the duration window favorable for radiation cooling. Wind direction affects the source and pathway of cold air, different wind directions may bring air masses with different temperature and humidity characteristics, having differential effects on frost formation.

Humidity Features (4 dimensions) Humidity features quantify water vapor content in the atmosphere, having important significance for understanding radiation cooling, condensation processes, and frost formation mechanisms.
Feature list:

- `saturation_vapor_pressure`: Saturation vapor pressure, calculated based on Magnus formula: $e_s = 0.6108 \times \exp\left(\frac{17.27 \times T}{T+237.3}\right)$ (where T is air temperature °C). This feature reflects maximum water vapor capacity of air at given temperature
- `dew_point_proximity`: Dew point proximity, calculated as `(T - T_dew) / T`, quantifies relative difference between air temperature and dew point. When this value approaches 0, it indicates air is near saturation, condensation or dew formation may occur
- `humidity_change_rate`: Humidity change rate ($\texttt{Rel Hum}(t) - \texttt{Rel Hum}(t-1)$), captures dynamic changes in atmospheric humidity
- `vapor_pressure_deficit` (VPD): Vapor pressure deficit, defined as the difference between saturation vapor pressure and actual vapor pressure. VPD reflects "dryness" and cooling potential of air, high VPD usually corresponds to stronger evaporative cooling effects

Theoretical basis: Humidity features are based on phase change thermodynamics and energy balance theory. According to Clausius-Clapeyron equation, saturation vapor pressure increases exponentially with temperature. Under high humidity conditions, when temperature approaches dew point, water vapor condensation releases latent heat (approximately 2500 J/g), potentially slowing cooling rate; under low humidity conditions, evaporative cooling enhances (evaporation consumes energy), potentially accelerating cooling. Dew point proximity quantifies how close air is

to saturation, VPD (vapor pressure deficit) quantifies "dryness" and evaporation potential of air, an important indicator for understanding relative intensity of radiation cooling and evaporative cooling.

Trend Features (1 dimension) Trend features capture acceleration of temperature changes, having important value for identifying rapid cooling processes (which may lead to frost).
Feature list:

- `cooling_acceleration`: Cooling acceleration, calculated based on temperature decline rate changes over the last 6 hours. This feature quantifies the second derivative of cooling rate, positive values indicate accelerating cooling, negative values indicate decelerating cooling. Rapid cooling (high cooling acceleration) is a key signal for frost warning

Theoretical basis: Trend features are based on difference and acceleration concepts in time series analysis. First-order features (e.g., temperature change rate) capture trend direction, second-order features (e.g., cooling acceleration) capture trend changes, helping models identify nonlinear dynamics of cooling processes. Cooling acceleration quantifies acceleration or deceleration of cooling rate, having important value for identifying rapid cooling processes (which may lead to frost). This feature is calculated based on temperature decline rate changes over the last 6 hours, capturing second-order dynamics of cooling processes.

Station Static Features (4 dimensions) Station static features merge geographic attributes from CIMIS station metadata, used to characterize spatial heterogeneity.
Feature list:

- `station_id_encoded`: Station ID encoding, converts station identifiers to numeric encoding, facilitating models learning station-specific patterns (e.g., microclimate, elevation, terrain)

- `region_encoded`: Region encoding, classifies and encodes stations by geographic region, capturing regional-scale climate differences

Theoretical basis: Station static features are based on microclimatology theory. Different stations have different microclimatic characteristics (e.g., elevation, terrain, vegetation cover, surface type), which systematically affect frost risk through influencing radiation balance, convective mixing, cold air flow, etc. Station ID encoding allows models to learn station-specific frost risk patterns, region encoding captures regional-scale climate differences. In LOSO evaluation, these features have important value for generalizing to new stations, helping models understand how spatial heterogeneity affects frost formation.
Total feature count: The above single-station feature engineering pipeline, when enabled in Matrix B, actually generates 278 candidate features, including: raw variables (12 dim) + temporal features (15 dim) + lag features (50 dim) + rolling window statistics (180 dim) + derived meteorological features (3 dim) + radiation features (4 dim) + wind features (6 dim) + humidity features (4 dim) + trend features (1 dim) + station static features (4 dim). All features are computed after grouping by station and temporal sorting, strictly preventing temporal leakage. Some theoretical features may not be generated due to missing data or unmet conditions, so actual feature count may be slightly lower than theoretical value.
Feature engineering workflow: Feature construction follows strict dependency order:

- **Stage 1**: Temporal features (base, no dependencies)

- **Stage 2**: Lag features (depends on temporal sorting)
- **Stage 3**: Rolling window statistics (depends on lag features and temporal sorting)
- **Stage 4**: Derived meteorological features (depends on raw variables and lag features)
- **Stage 5**: Radiation, wind, humidity, trend features (depends on raw variables and temporal features)
- **Stage 6**: Station static features (independent, merged from metadata)

This workflow ensures correctness and reproducibility of feature computation.

4.3.3 Neighborhood Aggregation Features

For matrices C and D, neighborhood aggregation statistics are overlaid on single-station features. Neighborhood aggregation features are used to capture spatial patterns and cold air pooling signals, and are a key factor in Matrix C's excellent performance.

Neighborhood Construction Method For target station s_0 and radius threshold r (tested 20–200 km in experiments, step size 20 km), Haversine distance is calculated based on latitude/longitude in station metadata:

$$d(s_0, s_i) = 2R \arcsin\left(\sqrt{\sin^2\left(\frac{\Delta\phi}{2}\right) + \cos(\phi_0)\cos(\phi_i)\sin^2\left(\frac{\Delta\lambda}{2}\right)}\right)$$

where $R = 6371$ km is Earth's radius, ϕ is latitude, λ is longitude. Neighbor station set: $\mathcal{N}(s_0, r) = \{s_i : d(s_0, s_i) \leq r\}$.
For each neighbor station $s_i \in \mathcal{N}(s_0, r)$, its time series is aligned with the target station's timestamps (based on `Date` and `Hour` columns), using left join to ensure each timestamp of the target station has corresponding neighbor data (possibly missing).

Aggregation Statistics Methods For each numeric variable, neighborhood aggregation is performed, including 12 raw CIMIS variables (air temp, dew point, relative humidity, wind speed, wind direction, solar radiation, soil temp, vapor pressure, ETo, precipitation, hour, Julian day) and 15 temporal features (`hour`, `hour_sin/cos`, `month`, `month_sin/cos`, `day_of_year`, `day_of_year_sin/cos`, `day_of_week`, `day_progress`, `day_progress_sin/cos`, `season`, `is_night`), totaling 27 variables. For each variable, the following 8 aggregation statistics are computed:

- **Basic statistics** (5 types):
  - `mean`: $\bar{x} = \frac{1}{|\mathcal{N}|}\sum_{s_i \in \mathcal{N}} x_i$, neighborhood mean, reflects local average state
  - `max`: $x_{\max} = \max_{s_i \in \mathcal{N}} x_i$, neighborhood maximum, identifies extreme values
  - `min`: $x_{\min} = \min_{s_i \in \mathcal{N}} x_i$, neighborhood minimum, captures cold air pooling (especially important for temperature/soil temperature)
  - `std`: $\sigma = \sqrt{\frac{1}{|\mathcal{N}|-1}\sum_{s_i \in \mathcal{N}}(x_i - \bar{x})^2}$, neighborhood standard deviation, reflects spatial variability
  - `median`: Neighborhood median, more robust to outliers
- **Distance-weighted statistics** (1 type):

- `weighted_mean`: $\bar{x}_w = \frac{\sum_{s_i \in \mathcal{N}} w_i x_i}{\sum_{s_i \in \mathcal{N}} w_i}$, where $w_i = 1/d_i^2$, distance-weighted mean, closer neighbors have larger weights

- **Spatial gradient** (1 type):
  - `gradient`: $\nabla x = \bar{x} - x_0$, neighborhood mean minus target station value, characterizes spatial gradient (crucial for identifying cold air sinking and inversion layers)

- **Spatial range** (1 type):
  - `range`: $x_{\max} - x_{\min}$, neighborhood maximum minus minimum, reflects spatial variability range

Naming format is {`variable`}`_neighbor_`{`method`}, for example `Soil Temp (C)_neighbor_min` represents neighborhood minimum soil temperature, `Air Temp (C)_neighbor_gradient` represents neighborhood air temperature gradient.

Missing Mask Features During spatial aggregation, neighbor station data may be missing, which can make aggregation feature values unavailable or unreliable. To handle this issue and improve model robustness, the system generates missing mask features and missing ratio features. These features all serve as model input features, used to indicate data quality and spatial coverage. Matrices C and D differ in missing-related features:
Matrix C's missing-related features (293 dimensions): The system generates four types of missing-related features, totaling 293 dimensions. These features all serve as model input features, used to indicate data quality and spatial coverage:

(1) **Missing masks for neighborhood aggregation features** (216 dim): For each neighborhood aggregation feature ({`variable`}`_neighbor_`{`method`}), generate corresponding binary missing mask ({`variable`}`_neighbor_`{`method`}`_missing_mask`), calculated as:

$$\texttt{missing\_mask} = \begin{cases} 1 & \text{if aggregation feature value is missing (NaN)} \\ 0 & \text{if aggregation feature value exists} \end{cases}$$

Total 216 dimensions (27 variables $\times$ 8 methods), each aggregation feature corresponds to one missing mask.

(2) **Variable missing ratio features** (27 dim): For each variable, calculate neighborhood missing ratio ({`variable`}`_neighbor_missing_ratio`), calculated as:

$$\texttt{missing\_ratio} = \frac{\text{number of missing neighbors}}{\text{total number of neighbors}} = \frac{\sum_{s_i \in \mathcal{N}} \mathbf{1}[\text{variable missing at } s_i]}{|\mathcal{N}|}$$

where $\mathcal{N}$ is the neighbor station set, $\mathbf{1}[\cdot]$ is the indicator function. This feature represents the proportion of available neighbor data at a given timestamp (0–1), used to indicate spatial coverage of the variable. Total 27 dimensions (27 variables), each variable corresponds to one missing ratio feature. Note: This is a feature itself, not a mask, but described here together with missing patterns.

(3) **Missing masks for missing ratio features** (27 dim): For each missing ratio feature itself, generate missing mask ({`variable`}`_neighbor_missing_ratio_missing_mask`), used to indicate whether the missing ratio feature is available. When all neighbor stations are missing the variable, the missing ratio feature itself may also be unavailable, in which case the missing mask is 1. Total 27 dimensions.

(4) **Missing masks for other features** (22 dim): For raw variables, temporal features, and other numeric features, generate missing masks to indicate data completeness. These features include missing masks for raw CIMIS variables (12 dim) and temporal features (e.g., `hour`, `day_of_year_sin/cos`, etc., approximately 10 dim).

Matrix D's missing-related features (299 dimensions): Matrix D's missing-related features add missing masks for engineered features on top of Matrix C. Specifically including:

(1) **Missing masks for neighborhood aggregation features** (216 dim): Same as Matrix C, generate missing masks for each neighborhood aggregation feature.

(2) **Variable missing ratio features** (27 dim): Same as Matrix C, calculate neighborhood missing ratio for each variable.

(3) **Missing masks for missing ratio features** (27 dim): Same as Matrix C, generate missing masks for each missing ratio feature.

(4) **Missing masks for other features** (28 dim): On top of Matrix C's 22 dimensions, add missing masks for engineered features (e.g., lag features, rolling statistics features), totaling 28 dimensions. This includes missing masks for raw CIMIS variables (12 dim), temporal features (approximately 10 dim), and engineered features (e.g., lags, rolling statistics, etc., approximately 6 dim).

Total 299 dimensions of missing-related features (216+27+27+28 = 298, plus 1 additional missing-related feature, totaling 299 dimensions), 6 dimensions more than Matrix C, because Matrix D also generates missing masks for engineered features to improve model robustness under sparse data conditions. These features all serve as model input features.
Design motivation: The introduction of missing mask features is based on the following considerations: (1) **Data quality indication**: In multi-station data fusion scenarios, data completeness and temporal alignment vary across stations, missing masks help models identify which aggregation features are reliable and which may be unreliable due to data sparsity; (2) **Spatial coverage modeling**: Missing ratio features (`missing_ratio`) quantify spatial coverage of neighborhood data, low coverage may indicate target station is at the edge of monitoring network or data collection anomalies, this information has important value for model judgment; (3) **Robustness improvement**: By explicitly modeling missing patterns, models can learn decision strategies under sparse data conditions, avoiding information loss from simply filling missing values with 0 or mean; (4) **Feature importance understanding**: Missing mask features themselves may have predictive value, for example, high missing ratio may indicate extreme weather conditions or equipment failure, these signals have indirect indication value for frost warning.

Total Feature Count

- **Matrix C**: Total feature count is 534 dimensions, including: raw variables (12 dim) + neighborhood aggregation features (216 dim) + missing masks (293 dim) + temporal harmonic encoding (2 dim: `day_of_year_sin/cos`) + other features (11 dim: derived meteorological features, temporal discrete features, etc.).
- **Matrix D**: Total feature count is 818 dimensions, including: Matrix B single-station feature engineering (278 dim) + neighborhood aggregation features (216 dim) + missing masks (299 dim) + other features (43 dim: wind features, humidity features, radiation features, trend features,

station static features, geographic features, and interaction features, etc.). Note that Matrix D contains feature duplication (e.g., temporal features appear in both Matrix B and Matrix C's "other features"), which are automatically deduplicated in actual implementation to ensure each feature appears only once in the feature space.

Supplementary material note: For detailed ABCD matrix feature lists, feature calculation formulas, naming conventions, and feature importance analysis results, please refer to Supplementary Material S1 (`Supplementary/supplementary_S1_feature_list.pdf`). This document contains complete feature lists for matrices A–D (including physical significance, calculation formulas, naming rules), feature importance analysis results (Top 20 features, including importance percentage and cumulative percentage), feature generation implementation details (code locations, configuration examples, training CLI), and feature usage recommendations (matrix selection guidelines, feature selection strategies, radius selection recommendations).

4.4 Model Families and Training Configuration

We systematically compared three model families: gradient boosting trees, random forests, and spatiotemporal neural networks, totaling 7 model implementations. All experiments are scheduled through a unified command-line interface (CLI), automatically completing data loading, feature construction, model training, metric calculation, and result archiving, ensuring comparability between different experiments. Usage of each model in feature configuration matrices (ABCD) is shown in Table 2.

Table 2: Usage of Each Model in Feature Configuration Matrices (ABCD)

Model Category	Model	Matrix A	Matrix B	Matrix C	Matrix D
Gradient Boosting Trees	LightGBM	✓	✓	✓	✓
	XGBoost	✓	✓	✓	✓
	CatBoost	✓	✓	✓	✓
Random Forest	Random Forest	✓	✓	✓	✓
Spatiotemporal Neural Networks	GRU	✓	✓	—	—
	LSTM	✓	✓	—	—
	TCN	✓	✓	—	—

4.4.1 Gradient Boosting Tree Models

Gradient boosting tree models are the core model family of this study, including LightGBM (primary), XGBoost, and CatBoost. Usage of these models in feature configuration matrices (ABCD) is shown in Table 2. All models are trained using the full feature set on their corresponding feature matrices.

Input format: All tree models accept standard tabular input, features are two-dimensional arrays ($n_{\text{samples}} \times n_{\text{features}}$), where each row represents an observation at a time point, each column represents a feature variable. Models directly use raw feature matrices generated by the feature engineering pipeline, requiring no additional sequence reorganization or format conversion. During training, feature matrices and target variables (temperature or frost labels) are provided separately as inputs, and models internally convert data to numeric arrays for training.

Training pair format: For each time point t, the training pair is (X_t, y_{t+h}), where X_t is the feature vector at time t (dimension depends on feature matrix, see Table 1), y_{t+h} is the target value

h hours in the future. For regression tasks, y_{t+h} is the air temperature value h hours in the future (unit: °C); for classification tasks, y_{t+h} is a binary label (1 indicates air temperature below 0 °C h hours in the future, 0 otherwise). In experiments $h \in \{3, 6, 12, 24\}$ hours, each forecast horizon corresponds to an independent model training task.

LightGBM: Efficient implementation based on Gradient Boosting Decision Trees (GBDT), using histogram-based algorithms to accelerate training, with leaf-wise tree construction strategy. Hyperparameters include learning rate (0.05), number of trees (100), max depth (6), number of leaves (31), minimum samples (20), L1/L2 regularization (0.1/0.1), row sampling (0.8), and column sampling (0.8). To handle class imbalance (positive sample proportion approximately 0.87%), the built-in class imbalance handling mechanism (`is_unbalance=True`) is adopted, which automatically balances class weights based on the actual class distribution. Model supports dual tasks of regression and classification, loss function configuration see Section 4.4.4. Feature importance is computed and recorded during training for subsequent feature selection and analysis.

XGBoost: Uses gradient boosting framework similar to LightGBM, but with different tree construction strategy. XGBoost uses level-wise tree construction strategy, expanding all nodes layer by layer, while LightGBM uses leaf-wise strategy, prioritizing expansion of leaf nodes with highest gain. Hyperparameter configuration is consistent with LightGBM. XGBoost handles class imbalance through the `scale_pos_weight` parameter (`scale_pos_weight=114.0`), which weights positive samples 114× more than negative samples to reflect the approximately 0.87% positive class rate. Loss function configuration see Section 4.4.4.

CatBoost: Gradient boosting algorithm optimized for categorical features, mainly used for numerical feature scenarios in this study. Hyperparameter configuration is similar to other tree models, using random seed parameters to ensure reproducibility. CatBoost handles class imbalance through the `scale_pos_weight` parameter (`scale_pos_weight=114.0`), which weights positive samples 114× more than negative samples to reflect the approximately 0.87% positive class rate.

4.4.2 Random Forest

Random Forest serves as a baseline model for ensemble learning, used to evaluate the performance of simple ensemble strategies on frost forecasting tasks. Model is based on Bootstrap Aggregation (Bootstrap Aggregation) and random feature subspace selection, implemented using scikit-learn. Configuration parameters include number of trees (100), minimum samples split (2), minimum samples leaf (1), etc., maximum depth uses default value (unlimited). Training process supports multi-threaded parallel computation (`n_jobs=-1`), fully utilizing multi-core CPU resources. Loss function configuration see Section 4.4.4.

Input format and training pair format: Same as gradient boosting tree models (see Section 4.4.1).

4.4.3 Spatiotemporal Neural Networks

Spatiotemporal neural network models include GRU, LSTM, and TCN, used to examine additional benefits of sequence modeling compared to engineered features. Usage of these models in feature configuration matrices (ABCD) is shown in Table 2, all implemented based on PyTorch.

Input format and preprocessing: Unlike tree models, neural network models need to reorganize tabular feature matrices into time series format. Specific conversion process is as follows:

Raw data is a two-dimensional feature matrix, where each row represents an observation at a time point ($n_{\text{samples}} \times n_{\text{features}}$). Through sliding window method, consecutive 24-hour observations are combined into a sequence, generating a three-dimensional tensor ($n_{\text{sequences}} \times 24 \times n_{\text{features}}$),

where $n_{\text{sequences}}$ is the number of generated sequences (typically $n_{\text{sequences}} < n_{\text{samples}}$, because each sequence requires 24 consecutive time points). For example, for time point t, the generated sequence is $S_t = [X_{t-23}, X_{t-22}, \ldots, X_t]$, containing feature observations from 24 consecutive time steps from $t-23$ to t.

Sequence generation uses sliding window method, window length is 24 hours, step size is 1 hour (i.e., each time point can serve as the last time point of a sequence), ensuring maximum data utilization. To strictly prevent temporal and spatial leakage, sequence generation follows these principles: (1) Group by station, ensuring all time points within each sequence come from the same station; (2) Arrange in temporal order, ensuring time within sequence is monotonically increasing; (3) Allow small gaps within stations (at most 24 sample steps), but prohibit cross-station boundary sequence generation. This data organization is fully compatible with LOSO evaluation strategy, ensuring models do not see information from test stations during training.

Training pair format: For each sequence $S_t = [X_{t-23}, X_{t-22}, \ldots, X_t]$ (containing 24-hour feature observations from $t-23$ to t), the training pair is (S_t, y_{t+h}), where y_{t+h} is the target value h hours in the future (same as tree models: regression task is temperature value, classification task is binary frost label). The last time point t of the sequence serves as "current time", model predicts future state h hours ahead based on past 24 hours of historical observations. In experiments $h \in \{3, 6, 12, 24\}$ hours, each forecast horizon corresponds to an independent model training task.

GRU: Uses Gated Recurrent Unit network to capture temporal dependencies. Network structure includes 2 GRU layers, 64 hidden units, Dropout rate 0.2. Input is 24-hour sequence (sequence_length=24), ensuring model can capture diurnal cycle patterns. Training uses Adam optimizer (learning rate 0.001), batch size 32, maximum training epochs 50, early stopping strategy (patience=10) automatically terminates training based on validation set performance. Loss function configuration see Section 4.4.4.

LSTM: Uses Long Short-Term Memory network to better capture long-term dependencies. Network structure is similar to GRU (2 layers, 64 hidden units, Dropout 0.2), but uses LSTM cells instead of GRU cells. Input format, data organization, training configuration, and loss function (see Section 4.4.4) are consistent with GRU.

TCN: Uses Temporal Convolutional Network to replace recurrent structure. Core features include causal convolution ensuring temporal causality, dilated convolution expanding receptive field to capture multi-scale temporal dependencies, and residual connections improving training stability and gradient flow. Network removes right padding through Chomp1d module to prevent future information leakage. Network structure includes 3 convolutional blocks, 32 channels per layer, kernel size 3, Dropout rate 0.1. Input format is same as GRU/LSTM, both are three-dimensional sequence tensors ($n_{\text{sequences}} \times 24 \times n_{\text{features}}$). Training uses Adam optimizer (learning rate 0.0005), batch size 32, maximum training epochs 50, early stopping strategy (patience=10) automatically terminates training based on validation set performance. Loss function configuration see Section 4.4.4.

4.4.4 Loss Function Configuration

This study adopts a multi-task learning framework, all models simultaneously perform dual-task training for frost binary classification and temperature regression. Different model families use different loss functions for these two tasks to optimize their respective architecture characteristics and task requirements.

Tree models (LightGBM / XGBoost / CatBoost / Random Forest):

- **Classification task**: Uses logarithmic loss (i.e., binary cross-entropy) as objective function:

$$\mathcal{L}_{\text{cls}} = -\frac{1}{n}\sum_{i=1}^{n}\left[y_i \log(p_i) + (1-y_i)\log(1-p_i)\right]$$

  where $y_i \in \{0,1\}$ is the true label, p_i is the model's predicted positive class probability. To handle class imbalance (positive sample proportion approximately 0.87%), LightGBM automatically adjusts positive/negative sample weights through its built-in class imbalance handling mechanism (`is_unbalance=True`), which balances class weights based on the actual class distribution. XGBoost and CatBoost use the `scale_pos_weight` parameter (`scale_pos_weight=114.0`) to balance class weights, where the weight ratio (114:1) reflects the approximately 0.87% positive class rate in the dataset.

- **Regression task**: Uses Mean Squared Error (MSE) or Mean Absolute Error (MAE) as objective function. Default configuration uses MSE:

$$\mathcal{L}_{\text{reg}} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

  where y_i is the true temperature value (unit: °C), $\hat{y}_i$ is the model's predicted value. Specific choice depends on `objective` parameter in model configuration.

Neural networks (GRU / LSTM / TCN):

- **Classification task**: Uses Focal Loss to handle extreme class imbalance. Focal Loss dynamically adjusts sample weights through focusing parameter γ and class weight α, reducing contribution of easy-to-classify samples, focusing on hard-to-classify samples, thereby alleviating model bias toward majority class caused by class imbalance. Loss function is defined as:

$$\text{FL}(p_t) = -\alpha(1-p_t)^{\gamma}\log(p_t)$$

  where p_t is the predicted probability of true class, $\gamma = 2.0$ is the focusing parameter, $\alpha = 0.25$ is the class weight. When $p_t \to 1$ (easy-to-classify samples), weight factor $(1-p_t)^{\gamma} \to 0$, reducing loss contribution of easy-to-classify samples; when $p_t \to 0$ (hard-to-classify samples), weight factor approaches 1, maintaining focus on hard-to-classify samples.

- **Regression task**: Uses Mean Squared Error (MSE) as loss function:

$$\mathcal{L}_{\text{reg}} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

  where y_i is the true temperature value (unit: °C), $\hat{y}_i$ is the model's predicted value.

All models simultaneously perform dual-task training for classification and regression, loss functions for the two tasks are computed independently, model outputs simultaneously include frost probability (for classification task) and temperature prediction (for regression task). This multi-task learning strategy can fully utilize relevant information in data, improving model generalization ability.

4.4.5 Unified Training Framework

To ensure reproducibility and comparability of experiments, this study adopts a unified training framework, all models are trained and evaluated through the same training process, evaluation strategy, and result archiving mechanism. Core components of the unified training framework include:

- **Multi-task learning**: All models simultaneously perform dual tasks of frost binary classification and temperature regression, loss functions are computed independently, model outputs simultaneously include frost probability and temperature prediction. This multi-task learning strategy can fully utilize relevant information in data, improving model generalization ability (see Section 4.4.4).
- **Early stopping mechanism**: Early stopping strategy based on validation set performance, automatically terminates training when validation set performance does not improve within specified number of epochs (patience), preventing overfitting. For neural network models (GRU, LSTM, TCN), patience is set to 10 training epochs; for tree models, early stopping is implemented through built-in callback functions of model libraries.
- **Checkpoint management**: Automatically saves best model checkpoints during training (based on validation set performance), ensuring ability to restore optimal model state. For neural network models, supports periodic checkpoint saving (default every 10 training epochs) and best model checkpoint saving; for tree models, checkpoint management is implemented through built-in saving mechanisms of model libraries.
- **Feature importance analysis**: Tree models (LightGBM, XGBoost, CatBoost, Random Forest) automatically compute feature importance during training, including raw importance scores, relative percentages, and cumulative percentages, used for subsequent feature selection and analysis (see Section 4.6).
- **Unified evaluation strategy**: All models use the same evaluation metrics and evaluation strategies (see Section 4.5 and Section 4.7), including Leave-One-Station-Out (LOSO) spatial generalization evaluation, ensuring performance comparisons between different models are comparable.

All experimental configurations, hyperparameters, random seeds, and data split information are completely saved, ensuring experiments are fully reproducible.

4.5 Evaluation Metrics

Experiments simultaneously consider frost binary classification and temperature regression tasks. This section details the definitions and calculation methods of each evaluation metric.

Classification Task Metrics Classification tasks use ROC-AUC, PR-AUC, Brier Score, and Expected Calibration Error (ECE) metrics, where PR-AUC better reflects identification capability under extreme imbalance conditions, and Brier and ECE are used to characterize the reliability of probability outputs.

- **ROC-AUC (Area Under the Receiver Operating Characteristic Curve)**: ROC-AUC measures overall discriminative ability across different classification thresholds, defined as the

area under the ROC curve. ROC curve uses False Positive Rate (FPR) as x-axis and True Positive Rate (TPR) as y-axis, where:

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}}, \quad \text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

where TP, FP, TN, FN represent numbers of true positives, false positives, true negatives, and false negatives respectively. ROC-AUC ranges from [0, 1], larger values indicate stronger discriminative ability. ROC-AUC = 0.5 indicates random guessing, ROC-AUC = 1.0 indicates perfect classification.

- **PR-AUC (Area Under the Precision-Recall Curve)**: PR-AUC measures area under the precision-recall curve, defined as:

$$\text{PR-AUC} = \int_0^1 P(R)\, dR$$

where P is precision (Precision = $\frac{\text{TP}}{\text{TP+FP}}$), R is recall (Recall = $\frac{\text{TP}}{\text{TP+FN}}$). Under extreme class imbalance scenarios (e.g., positive sample proportion approximately 0.87% in this study), PR-AUC better reflects model identification capability for minority classes than ROC-AUC, because PR-AUC directly focuses on precision-recall trade-off, while ROC-AUC is more influenced by majority class.

- **Brier Score**: Brier Score measures calibration quality of probability predictions, defined as mean squared error between predicted probabilities and true labels:

$$\text{Brier Score} = \frac{1}{n}\sum_{i=1}^{n}(p_i - y_i)^2$$

where p_i is the model's predicted positive class probability, $y_i \in \{0, 1\}$ is the true label, n is the number of samples. Brier Score ranges from [0, 1], smaller values indicate more accurate probability predictions. Brier Score = 0 indicates perfect prediction, Brier Score = 1 indicates worst prediction.

- **Expected Calibration Error (ECE)**: ECE measures calibration degree of model predicted probabilities, defined as:

$$\text{ECE} = \sum_{m=1}^{M} \frac{|B_m|}{n} |\text{acc}(B_m) - \text{conf}(B_m)|$$

where M is the number of bins, B_m is the set of samples in the m-th bin, $|B_m|$ is the number of samples in the bin, $\text{acc}(B_m)$ is the accuracy in the bin, $\text{conf}(B_m)$ is the average predicted probability in the bin. ECE ranges from [0, 1], smaller values indicate better calibrated probability predictions. ECE = 0 indicates perfect calibration (predicted probability equals actual probability), larger ECE indicates larger calibration error.

Regression Task Metrics Regression tasks use MAE, RMSE, and R^2 to measure temperature prediction error.

- **MAE (Mean Absolute Error)**: MAE measures average absolute deviation between predicted and true values:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

where y_i is the true temperature value (unit: °C), $\hat{y}_i$ is the model's predicted value, n is the number of samples. MAE has the same unit as the target variable, making it intuitive to interpret.

- **RMSE (Root Mean Squared Error)**: RMSE measures square root of average squared deviation:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

RMSE penalizes large errors more heavily than MAE, making it sensitive to outliers. RMSE also has the same unit as the target variable.

- R^2 **(Coefficient of Determination)**: R^2 measures proportion of variance in target variable explained by the model:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

where $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$ is the mean of true values. R^2 ranges from $(-\infty, 1]$, $R^2 = 1$ indicates perfect fit, $R^2 = 0$ indicates model performance equals baseline (mean prediction), $R^2 < 0$ indicates model performance worse than baseline.

4.6 Experimental Design

To systematically evaluate the performance of feature configuration matrices (ABCD) and different model families across forecast horizons, this study designed large-scale controlled experiments, covering hundreds of experimental configuration combinations.

Experimental Design Strategy This study adopts factorial design method, systematically exploring combination effects across the following dimensions:

- **Feature configuration matrices**: 4 matrices (A, B, C, D), covering complete combinations of single-station/multi-station and raw/engineered features
- **Forecast horizons**: 4 horizons (3, 6, 12, 24 hours), covering short-term warnings and medium-to-long-term forecasts
- **Model families**: 7 models (LightGBM, XGBoost, CatBoost, Random Forest, GRU, LSTM, TCN), covering tree models and spatiotemporal neural networks
- **Spatial aggregation radius**: Matrices C/D cover 0–200 km range (step size 20 km), totaling 10 radius values (0, 20, 40, 60, 80, 100, 120, 140, 160, 180, 200 km)

Experimental Matrix Organization Experimental matrix organization follows these principles:

1. **Matrix A/B experiments**: For matrices A and B (single-station, no spatial aggregation), each model is trained on each forecast horizon, totaling 2 matrices × 7 models × 4 horizons = 56 experimental configurations

2. **Matrix C/D experiments**: For matrices C and D (multi-station, with spatial aggregation), each model is trained on each forecast horizon with each radius, totaling 2 matrices $\times$ 7 models $\times$ 4 horizons $\times$ 11 radii = 616 experimental configurations

3. **Total experimental configurations**: 56+616 = 672 configurations, covering all combinations of matrices, models, horizons, and radii

Note: Neural network models (GRU, LSTM, TCN) are only trained on matrices A and B (single-station), not on matrices C and D (multi-station), as shown in Table 2. Therefore, actual experimental configurations are fewer than the theoretical maximum.

Feature Selection and Feature Importance Analysis Methods Under full configuration, feature configuration matrix (ABCD) can construct approximately 278 candidate features for Matrix B. To understand the contribution of different features to model performance, and explore the possibility of reducing computational costs while maintaining performance, this study adopts a two-stage feature selection strategy based on tree model feature importance. This method is applied in the above feature selection experiments, specific results see Section 5.
Stage 1: Feature importance calculation: After training LightGBM models on Matrix B (full feature set, 278 dimensions), feature importance scores are computed for each feature. Feature importance is calculated using tree model's built-in importance calculation method (e.g., LightGBM uses gain-based importance), representing the contribution of each feature to model performance.
Stage 2: Feature selection based on cumulative importance: Features are sorted by importance in descending order, and cumulative importance percentage is calculated. Feature selection is performed using different cumulative importance thresholds (80%, 85%, 90%, 95%), selecting the minimum feature set that reaches the threshold. This strategy ensures that selected features capture the majority of predictive information while reducing feature dimensionality.

Experimental Execution and Result Aggregation All experiments are scheduled through unified command-line interface (CLI), using declarative configuration (YAML) to manage experimental parameters, ensuring reproducibility and comparability of experiments. Experimental execution process includes:

1. **Configuration loading**: Load experimental configuration from YAML files, including data paths, feature matrix selection, model hyperparameters, forecast horizons, spatial aggregation radius, etc.

2. **Data loading and preprocessing**: Load preprocessed data, construct feature matrices according to configuration, perform train/validation/test split

3. **Model training**: Train models according to configuration, automatically save model checkpoints and training logs

4. **Evaluation and metric calculation**: Evaluate models on test set, calculate all evaluation metrics (ROC-AUC, PR-AUC, Brier Score, ECE, RMSE, MAE, R^2)

5. **Result archiving**: Save experimental results to CSV files, including model name, matrix, forecast horizon, radius, all evaluation metrics, and other metadata

All experimental results are systematically archived and aggregated into three supplementary tables (Supplementary Tables S2–S4), containing complete performance metrics, optimal configurations, and statistical summaries. These tables enable fine-grained comparisons across models, matrices, radii, and forecast horizons.

Experimental Platform All experiments run on a Linux server equipped with 16 physical CPU cores (32 logical cores), 60 GB RAM, and NVIDIA GeForce RTX 5090 GPU (32 GB VRAM). Experimental environment is based on Python 3.12.3, configured in a virtual environment. Main dependency libraries and their versions include: LightGBM 4.6.0, XGBoost 3.1.1, CatBoost 1.2.8, scikit-learn 1.7.2 (tree models), PyTorch 2.9.1 (neural network models, supports CUDA 13.0), pandas 2.3.3, numpy 2.3.4, and other data processing libraries. All tree model training supports multi-threaded parallel computation (`n_jobs=-1`), fully utilizing multi-core CPU resources; neural network model training supports GPU acceleration (CUDA), improving training efficiency.

Training/Validation Split and Leakage Prevention To ensure temporal ordering of time series data and prevent temporal leakage, this study adopts strict temporal order split strategy. Specifically, in the temporal dimension, we use 70% training, 15% validation, 15% test sequential split, ensuring time within each station is monotonically increasing, i.e., training set contains earliest time points, validation set contains middle time points, test set contains latest time points. This split ensures models do not see future information when predicting the future, consistent with actual application scenarios.
To evaluate model spatial generalization capability, this study further adopts Leave-One-Station-Out (LOSO) cross-validation scheme. Core idea of LOSO evaluation is: in each iteration, exclude one station as test set, use remaining stations as training and validation sets. During LOSO evaluation, all preprocessing steps (including feature standardization, neighborhood construction radius selection, etc.) are fitted only on training data, then fitted preprocessors are applied to excluded test stations for evaluation, strictly avoiding any form of spatial information leakage. This evaluation strategy can truly reflect model generalization capability at unseen spatial locations, providing reliable performance estimates for actual deployment.

4.7 Experimental Setup and Data Split

Experimental Platform All experiments run on a Linux server equipped with 16 physical CPU cores (32 logical cores), 60 GB RAM, and NVIDIA GeForce RTX 5090 GPU (32 GB VRAM). Experimental environment is based on Python 3.12.3, configured in a virtual environment. Main dependency libraries and their versions include: LightGBM 4.6.0, XGBoost 3.1.1, CatBoost 1.2.8, scikit-learn 1.7.2 (tree models), PyTorch 2.9.1 (neural network models, supports CUDA 13.0), pandas 2.3.3, numpy 2.3.4, and other data processing libraries. All tree model training supports multi-threaded parallel computation (`n_jobs=-1`), fully utilizing multi-core CPU resources; neural network model training supports GPU acceleration (CUDA), improving training efficiency.

Training/Validation Split and Leakage Prevention To ensure temporal ordering of time series data and prevent temporal leakage, this study adopts strict temporal order split strategy. Specifically, in the temporal dimension, we use 70% training, 15% validation, 15% test sequential split, ensuring time within each station is monotonically increasing, i.e., training set contains earliest time points, validation set contains middle time points, test set contains latest time points. This split ensures models do not see future information when predicting the future, consistent with actual application scenarios.
To evaluate model spatial generalization capability, this study further adopts Leave-One-Station-Out (LOSO) cross-validation scheme. Core idea of LOSO evaluation is: in each iteration, exclude one station as test set, use remaining stations as training and validation sets. During LOSO evaluation, all preprocessing steps (including feature standardization, neighborhood construction radius selection, etc.) are fitted only on training data, then fitted preprocessors are applied to

excluded test stations for evaluation, strictly avoiding any form of spatial information leakage. This evaluation strategy can truly reflect model generalization capability at unseen spatial locations, providing reliable performance estimates for actual deployment.

Supplementary material note: For detailed ABCD matrix feature lists, feature calculation formulas, naming conventions, and feature importance analysis results, please refer to Supplementary Material S1 (`Supplementary/supplementary_S1_feature_list.pdf`). This document contains complete feature lists for matrices A–D (including physical significance, calculation formulas, naming rules), feature importance analysis results (Top 20 features, including importance percentage and cumulative percentage), feature generation implementation details (code locations, configuration examples, training CLI), and feature usage recommendations (matrix selection guidelines, feature selection strategies, radius selection recommendations).

5 Results

5.1 Experimental Scale and Results Overview

Based on the experimental design strategy described in Section 4.6, this study systematically completed large-scale controlled experiments, covering hundreds of experimental configuration combinations. All experiments were conducted under a unified dataset and evaluation framework, ensuring comparability between different configurations. Experimental coverage is as follows:

- **Feature configuration matrices**: 4 (A, B, C, D), covering complete combinations of single-station/multi-station and raw/engineered features.
- **Forecast horizons**: 4 (3, 6, 12, 24 hours), covering short-term warnings and medium-to-long-term forecasts.
- **Model families**: 7 (LightGBM, XGBoost, CatBoost, Random Forest, GRU, LSTM, TCN), covering tree models and spatiotemporal neural networks.
- **Spatial aggregation radius**: Matrices C/D cover 0–200 km range (step size 20 km), totaling 10 radius values.

All experimental results have been systematically archived and aggregated into three supplementary tables (Supplementary Materials S2–S4), containing complete performance metrics, optimal configurations, and statistical summaries, as detailed in Section 4.6.

In terms of metrics, all experiments simultaneously evaluate frost binary classification and temperature regression tasks under the same dataset and label system: classification side includes ROC-AUC, PR-AUC, Brier Score, F1, Precision, Recall, and Expected Calibration Error (ECE); regression side includes temperature RMSE, MAE, and R^2. This unified metric matrix enables fair comparison of different feature matrices and model families from multiple perspectives: discriminative ability, probability calibration, to temperature error.

Scientific value of experimental scale: Systematic comparison of large-scale experimental configuration combinations provides important scientific value for frost forecasting research: (1) covers complete parameter space across multiple dimensions including feature configuration, forecast horizon, model families, and spatial aggregation radius, providing comprehensive data for understanding the impact of different factors on model performance; (2) unified evaluation framework ensures comparability between different experimental configurations, providing quantitative basis for model selection and feature engineering; (3) large-scale experiments provide sufficient samples

for statistical significance testing, enhancing credibility of research conclusions. This systematic experimental design is an important contribution of this study.

Table 3 shows the optimal configuration selected by ROC-AUC for each feature matrix and each forecast horizon, simultaneously displaying comprehensive performance metrics such as PR-AUC, Brier Score, and temperature RMSE. This table systematically summarizes the performance of different feature configuration strategies on frost forecasting tasks, providing quantitative basis for understanding the independent and joint effects of feature engineering and spatial aggregation.

Overall performance level and framework applicability: From the overall performance distribution, matrices A, B, and C all achieve high performance levels across different forecast horizons (ROC-AUC > 0.98, temperature RMSE < 2.9 °C). This finding has important scientific significance. First, it validates that the feature configuration matrix framework proposed in this study has good applicability for frost forecasting tasks, indicating that stable high performance can be achieved at different complexity levels through systematic feature configuration strategies. Second, this consistency of high performance reflects the inherent predictability of frost forecasting tasks: even at relatively low feature dimensions (Matrix A: 16 dimensions), models can still achieve near-perfect discriminative ability (ROC-AUC > 0.98) by effectively utilizing key meteorological variables (temperature, humidity, soil temperature, etc.). This finding provides important insights for practical applications: under computational resource constraints, even using raw feature configurations, models can still provide reliable frost warnings.

Matrix C's optimal performance and key role of spatial aggregation: From the comparison of feature configurations, Matrix C (multi-station + raw features, 534 dimensions) achieves optimal or near-optimal performance across all four forecast horizons, with ROC-AUC reaching 0.9972 at 3-hour horizon and still maintaining 0.9877 at 24-hour horizon, becoming the overall best configuration. This finding reveals the key role of spatial aggregation features in frost forecasting. Compared to Matrix A (single-station + raw features, 16 dimensions), Matrix C significantly improves model performance (ROC-AUC improvement approximately 0.0005–0.002, PR-AUC improvement approximately 0.01–0.16) by introducing spatial aggregation features while maintaining relatively low feature dimensions (534 dimensions vs. 818 dimensions of Matrix D). The mechanism of this performance improvement can be understood from the perspective of physical processes: (1) spatial aggregation features can effectively capture regional climate patterns such as cold air pooling and terrain effects, which are difficult to directly reflect in single-station observations; (2) spatial information such as temperature gradients and humidity distribution from neighboring stations provides important early warning signals for frost formation; (3) robustness of spatial aggregation: even when some station data is missing, neighborhood aggregation statistics remain stable, and this robustness is reflected across different forecast horizons. Notably, Matrix C's optimal radius varies with forecast horizon (3 hours: 60 km, 6 hours: 160 km, 12 hours: 200 km, 24 hours: 180 km), revealing the coupling relationship between spatial scale and temporal scale: short-term forecasts mainly rely on local cold air pooling (small radius), while long-term forecasts need to incorporate larger-scale weather system information (large radius). This scale-dependent finding provides important guidance for radius selection in practical applications (detailed analysis see Section 5.6).

Gain mechanism of feature engineering for single-station models: Comparison between Matrix B (single-station + engineered features, 278 dimensions) and Matrix A shows that feature engineering has significant gains for single-station models (ROC-AUC improvement approximately 0.002–0.003, PR-AUC improvement approximately 0.01–0.02), with LightGBM achieving optimal performance across all four forecast horizons on Matrix B. This gain mainly stems from time series engineered features (lag features, rolling window statistics, derived meteorological variables, etc.) effectively capturing temporal dependency patterns of frost formation. Specifically: (1) lag features can capture historical evolution trends of key variables such as temperature and humidity, providing

temporal context information for models; (2) rolling window statistics can identify temporal distribution characteristics such as extreme values and volatility, which are especially important for frost warning (minimum temperature directly relates to frost risk); (3) derived meteorological features (such as vapor pressure deficit, dew point difference, etc.) can capture physical mechanisms of frost formation, which are difficult to directly reflect in raw observations. Notably, feature engineering has significant gains for tree models, but limited gains for sequence models (GRU, TCN), and may even introduce noise, revealing sensitivity differences of different model architectures to feature types (detailed analysis see Section 5.5).

Challenges of high-dimensional feature space and overfitting risk: Comparison between Matrix D (multi-station + engineered features, 818 dimensions) and Matrix C shows that although the most features are overlaid, performance improvement is limited under current data scale, and temperature RMSE even increases (3.66–5.36 °C vs. 1.58–2.39 °C). This finding reveals challenges in high-dimensional feature spaces: (1) the ratio of feature dimensions (818 dimensions) to data scale (approximately 2.36 million records) may lead to overfitting, especially when prediction signals are weak at long horizons; (2) overlay of engineered features and spatial aggregation features may introduce multicollinearity between features, causing models to learn unstable feature combinations; (3) noise accumulation in high-dimensional feature spaces: as feature dimensions increase, the proportion of noise features also increases, and these noise features may dominate model predictions at long horizons. Notably, Matrix D's ROC-AUC drops to 0.9521 at 24-hour horizon, significantly lower than Matrix C's 0.9877, and this performance decline further validates that high-dimensional feature spaces may face challenges of overfitting or noise interference. This finding has important methodological significance: simple overlay of feature engineering and spatial aggregation is not an optimal strategy, and in practical applications, trade-offs need to be made based on data scale, feature dimensions, and forecast horizon to avoid falling into the trap of "curse of dimensionality" (detailed analysis see Section 5.5).

Comprehensive insights and decision support: This table provides quantitative reference basis for feature configuration and model selection, revealing the following key insights: (1) under sufficient computational resources, Matrix C (multi-station + raw features) is the optimal choice, achieving the best balance between feature dimensions and performance; (2) under computational resource constraints, Matrix A (single-station + raw features) can still provide reliable frost warnings (ROC-AUC > 0.98), providing flexibility for practical deployment; (3) feature engineering has significant gains for single-station models, but in spatial aggregation scenarios, the combination of raw features and spatial aggregation features already achieves excellent performance, without needing additional engineered features; (4) high-dimensional feature spaces need careful handling to avoid overfitting and noise interference. These findings provide scientific basis for feature configuration selection in practical applications. Detailed performance analysis and discussion see Section 5.5 (feature matrix performance analysis), Section 5.8 (model family performance comparison), and Section 5.6 (radius and horizon joint sensitivity).

Table 3: Representative Optimal Configurations for Feature Configuration Matrices (ABCD) at Each Forecast Horizon (Selected by ROC-AUC)

Matrix	Horizon	Optimal Model	Feature Config / Radius	ROC AUC	PR AUC	Brier	RMSE (°C)
A	3 h	GRU	Raw features / –	0.9969	0.7408	0.0025	1.60
A	6 h	GRU	Raw features / –	0.9935	0.5962	0.0033	2.32
A	12 h	GRU	Raw features / –	0.9883	0.4577	0.0039	2.85
A	24 h	LightGBM	Raw features / –	0.9821	0.3059	0.0052	2.59
B	3 h	LightGBM	Engineered features / –	0.9969	0.7042	0.0029	1.50
B	6 h	LightGBM	Engineered features / –	0.9937	0.5531	0.0038	1.99
B	12 h	LightGBM	Engineered features / –	0.9896	0.4337	0.0044	2.40
B	24 h	LightGBM	Engineered features / –	0.9843	0.3207	0.0065	2.53
C	3 h	LightGBM	Raw features + spatial aggregation / 60 km	0.9972	0.7242	0.0027	1.58
C	6 h	LightGBM	Raw features + spatial aggregation / 160 km	0.9943	0.5871	0.0039	2.05
C	12 h	LightGBM	Raw features + spatial aggregation / 200 km	0.9901	0.4914	0.0043	2.42
C	24 h	LightGBM	Raw features + spatial aggregation / 180 km	0.9877	0.4671	0.0045	2.39
D	3 h	CatBoost	Engineered features + spatial aggregation / 200 km	0.9874	0.3931	0.0038	3.66
D	6 h	XGBoost	Engineered features + spatial aggregation / 160 km	0.9737	0.2354	0.0047	4.36
D	12 h	XGBoost	Engineered features + spatial aggregation / 200 km	0.9634	0.1467	0.0049	4.98
D	24 h	XGBoost	Engineered features + spatial aggregation / 200 km	0.9521	0.1298	0.0048	5.36

5.2 Optimal Configuration Performance

Among all configurations, the Matrix C + LightGBM combination performs best. Table 4 shows performance on the time-held-out 15% test set for four forecast windows.

Table 4: Frost Probability and Temperature Prediction Performance (Time-held-out 15% Test Set, Matrix C + LightGBM)

Forecast Window	ROC-AUC	PR-AUC	Brier	ECE	RMSE (°C)	MAE (°C)	R^2
3 hours (60 km)	0.9972	0.7242	0.0027	0.0012	1.58	1.16	0.9681
6 hours (160 km)	0.9943	0.5871	0.0039	0.0021	2.05	1.60	0.9464
12 hours (200 km)	0.9901	0.4914	0.0043	0.0032	2.42	1.85	0.9253
24 hours (180 km)	0.9877	0.4671	0.0045	0.0034	2.39	1.85	0.9271

Short-term forecasts (3–6 hours) show near-perfect discriminative ability (ROC-AUC > 0.99), and PR-AUC remains around 0.46 even at 24-hour forecasts, indicating the model can effectively rank frost events even under extreme imbalance conditions. Temperature prediction accuracy is excellent across all horizons, with R^2 values consistently above 0.92 (3 hours: 0.9681, 24 hours: 0.9271), indicating the model explains over 92% of temperature variance with outstanding prediction precision. As forecast duration increases, temperature RMSE and MAE slightly increase, consistent with physical intuition, but the stability of R^2 (24 hours still maintains 0.9271) demonstrates that the model can effectively capture the main patterns of temperature variation even at long horizons. Figure 5 shows the confusion matrices for frost classification across four forecast windows, providing detailed insights into the model's classification performance with respect to true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). Each subplot displays the confusion matrix for the corresponding forecast horizon, with cell values showing both the absolute counts and percentages (in parentheses) of correctly and incorrectly classified instances. The diagonal elements (TN and TP) represent correct predictions, while off-diagonal elements (FP and FN) represent classification errors.

Optimal Threshold Selection: These confusion matrices are computed using optimal probability thresholds determined by maximizing F2-score (F-beta with $\beta = 2$), a standard metric for imbalanced classification that emphasizes recall 4× more than precision. The F2-score formula is: F2 = $(1 + 2^2) \times (\text{precision} \times \text{recall}) / (2^2 \times \text{precision} + \text{recall}) = 5 \times (\text{precision} \times \text{recall}) / (4 \times \text{precision} + \text{recall})$. This metric is particularly suitable for frost forecasting because: (1) it directly addresses the critical need to minimize false negatives (missed frost events) through recall emphasis; (2) it still considers precision to avoid excessive false alarms; (3) it is a well-established metric in imbalanced classification literature. The optimal thresholds are: 3 hours (0.132), 6 hours (0.238), 12 hours (0.203), and 24 hours (0.206). These thresholds are significantly lower than the standard 0.5 threshold, reflecting the need to prioritize recall (minimizing false negatives) in agricultural applications where missing a frost event can result in severe crop loss.

As can be seen, using optimal F2-score thresholds, the model achieves high accuracy across all horizons (ranging from 99.00% to 99.40%), with significantly improved recall compared to the standard 0.5 threshold: at 3 hours, recall reaches 0.848 (capturing 84.8% of all frost events, missing only 15.2%), while precision is 0.470; at 6 hours, recall is 0.770 with precision 0.405; at 12 hours, recall is 0.735 with precision 0.316; at 24 hours, recall remains at 0.697 (capturing 69.7% of frost events), with precision of 0.315. The trade-off is clear: lower thresholds increase recall (fewer missed frost events) at the cost of increased false positives (more false alarms). However, this trade-off is appropriate for agricultural applications, where the cost of missing a frost event (crop loss) typically far exceeds the cost of a false alarm (unnecessary protection measures). The optimal thresholds balance this trade-off by maximizing F2-score, which emphasizes recall while still considering precision, ensuring both good overall performance and high recall (minimizing missed events).

Comparison with Standard 0.5 Threshold: For reference, using the standard 0.5 threshold would yield: 3 hours (precision: 0.663, recall: 0.674), 6 hours (precision: 0.505, recall: 0.642), 12 hours (precision: 0.468, recall: 0.530), and 24 hours (precision: 0.445, recall: 0.505). While precision is higher at 0.5, recall is significantly lower, resulting in more missed frost events (32.6% missed at 3 hours vs. 15.2% with optimal F2-score threshold). The optimal F2-score thresholds provide a better balance for agricultural decision-making, where recall is often more critical than precision.

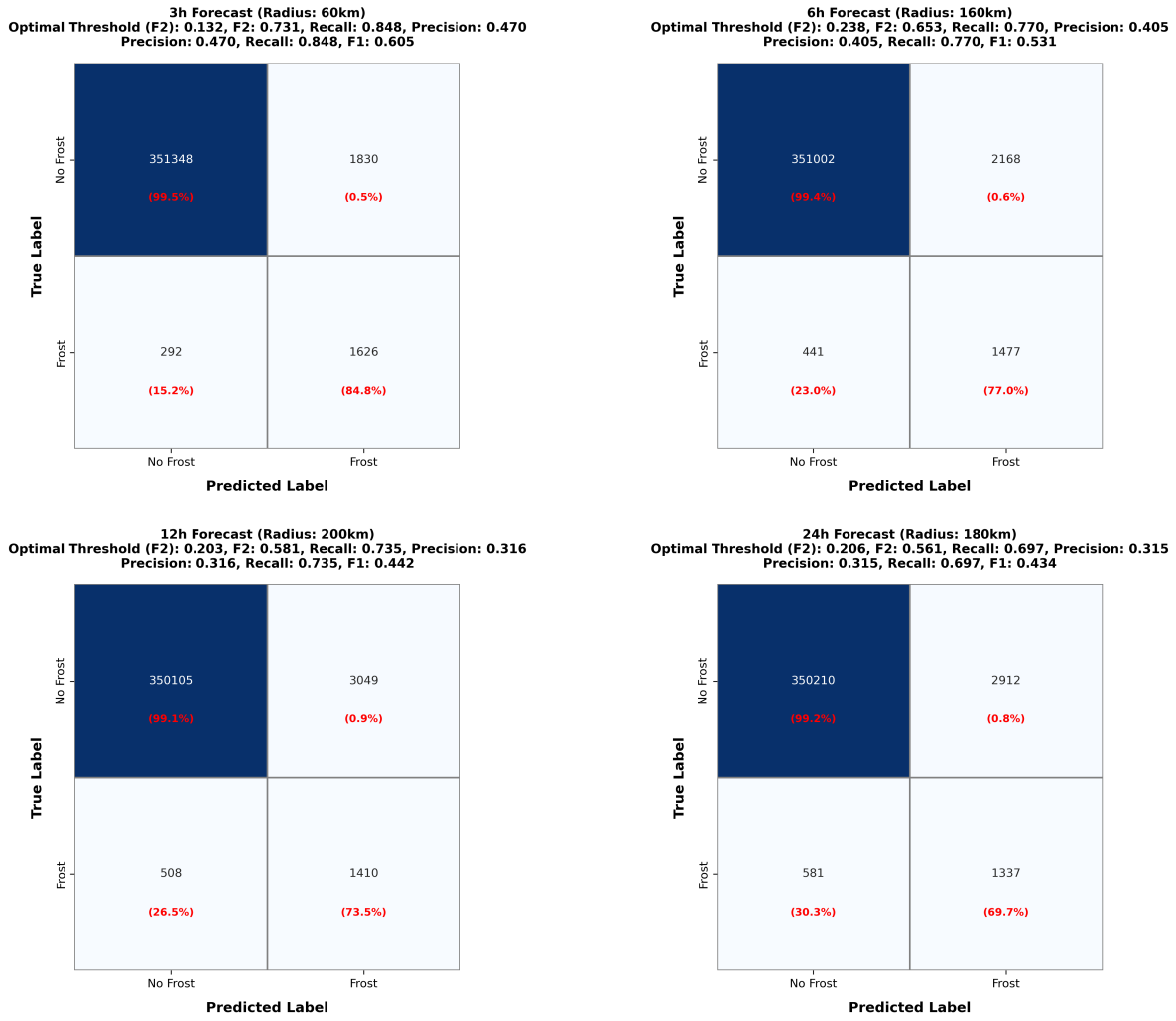

Figure 5: Confusion matrices for frost classification across four forecast windows (3/6/12/24 hours, Matrix C + LightGBM) using optimal probability thresholds. Each subplot shows the optimal configuration for the corresponding horizon: 3 hours (60 km, threshold: 0.132), 6 hours (160 km, threshold: 0.238), 12 hours (200 km, threshold: 0.203), 24 hours (180 km, threshold: 0.206). Optimal thresholds were determined by maximizing F2-score (F-beta with $\beta = 2$), which emphasizes recall 4× more than precision: F2 = 5 × (precision × recall) / (4 × precision + recall). This metric is standard for imbalanced classification tasks where recall (minimizing false negatives) is critical. Cell values show absolute counts and percentages (in parentheses). TN (True Negative): correctly predicted non-frost events; TP (True Positive): correctly predicted frost events; FP (False Positive): incorrectly predicted frost events (over-warning); FN (False Negative): missed frost events (under-warning). The model achieves high accuracy (99.00%–99.40%) and excellent recall (69.7%–84.8%) across all horizons, with recall significantly improved compared to the standard 0.5 threshold. The lower thresholds result in more false positives but fewer false negatives, reflecting a "better to over-warn than to miss" strategy appropriate for agricultural applications where missing frost events can cause severe crop loss.

Figure 6 shows the threshold sensitivity analysis for all four forecast horizons, demonstrating how classification performance metrics vary as a function of the probability threshold used to convert

continuous probability predictions into binary frost/non-frost classifications. Each subplot displays multiple curves: **recall (red, thick line)**—the most critical metric representing the proportion of actual frost events that are correctly predicted as frost (True Positives / All Frost Events), **F2-score (purple solid line, $\beta = 2$)**—the selected optimization metric that emphasizes recall 4× more than precision, F3-score (magenta dashed line, $\beta = 3$, recall 9×), F4-score (pink dotted line, $\beta = 4$, recall 16×), accuracy (blue), precision (green), and F1 score (orange dashed line, $\beta = 1$, balanced). The purple dotted vertical line marks the optimal threshold determined by maximizing F2-score, while the gray dashed vertical line marks the standard 0.5 threshold commonly used in binary classification.

F-beta Score Family and β Parameter Selection: The F-beta score family provides a principled way to balance precision and recall with different emphasis levels. The general formula is: $F_\beta = (1 + \beta^2) \times (\text{precision} \times \text{recall}) / (\beta^2 \times \text{precision} + \text{recall})$. The β parameter controls the relative weight of recall: $\beta = 1$ gives equal weight (F1-score), $\beta = 2$ emphasizes recall 4× more than precision (F2-score), $\beta = 3$ emphasizes recall 9× more (F3-score), and $\beta = 4$ emphasizes recall 16× more (F4-score). As β increases, the optimal threshold decreases (moving leftward in the figure), resulting in higher recall but lower precision. For frost forecasting, we select F2-score ($\beta = 2$) as the optimization metric because: (1) it provides substantial recall emphasis (4×) to minimize missed frost events, which is critical for agricultural applications; (2) it still considers precision to avoid excessive false alarms, maintaining a practical balance; (3) it is a well-established standard in imbalanced classification literature ($\beta = 2$ is commonly used for tasks where recall is more important than precision); (4) higher β values (F3, F4) would further increase recall but at the cost of significantly more false alarms, which may not be justified unless the cost of missing a frost event is extremely high relative to false alarm costs. The F2-score curve (purple) in Figure 6 shows how this metric varies with threshold, peaking at lower thresholds than F1 but higher than F3/F4, providing the optimal balance for our application.

Several key observations emerge from these threshold analysis curves. First, **recall (red curve) is the primary metric of interest**: recall represents the proportion of actual frost events that are correctly predicted as frost (TP / (TP + FN)), which directly measures our goal of correctly identifying frost events. The recall curve decreases monotonically as threshold increases: at low thresholds (e.g., 0.1), recall is high (0.85–0.90), meaning most frost events are correctly identified, while at high thresholds (e.g., 0.5), recall drops significantly (0.50–0.67), meaning many frost events are missed. This is the critical trade-off: higher thresholds reduce false alarms but miss more actual frost events. Second, **accuracy shows a characteristic inverted-U pattern but is less informative**: accuracy (blue curve) is highest at intermediate thresholds (typically around 0.3–0.6), but this is misleading because accuracy is dominated by correct non-frost predictions (TN), which comprise 99.13% of the dataset. High accuracy does not guarantee good frost prediction performance. Third, **precision and recall exhibit a clear trade-off**: precision increases monotonically with threshold (higher thresholds yield fewer false positives), while recall decreases monotonically (higher thresholds yield more false negatives). This trade-off is fundamental to binary classification and is particularly pronounced in imbalanced datasets. Fourth, **optimal thresholds are determined by maximizing F2-score**: the optimal thresholds from the confusion matrices (0.132 for 3 hours, 0.238 for 6 hours, 0.203 for 12 hours, 0.206 for 24 hours) are all below 0.5, reflecting the need to prioritize recall over precision in agricultural applications. The F2-score (F-beta with $\beta = 2$) is a standard metric for imbalanced classification that emphasizes recall 4× more than precision, making it particularly suitable for frost forecasting where missing events (false negatives) have severe consequences. At these optimal thresholds, recall is substantially higher (0.70–0.85 vs. 0.50–0.67 at 0.5 threshold), resulting in significantly fewer missed frost events, which is our primary concern. The threshold analysis provides crucial insights for practical deployment. **Decision-makers can**

select thresholds based on their specific cost-benefit preferences: (1) for high-value crops where missing a frost event is catastrophic, thresholds can be lowered further (e.g., 0.05–0.10) to maximize recall, accepting higher false positive rates; (2) for lower-value crops or when protection costs are high, thresholds can be raised (e.g., 0.4–0.5) to reduce false alarms, accepting lower recall; (3) the optimal thresholds (maximizing F2-score) provide a balanced default choice that works well across different scenarios, emphasizing recall while still considering precision. The F2-score is a well-established metric in imbalanced classification literature, making it a principled choice for frost forecasting. The wide range of thresholds where accuracy remains high (0.98+) provides flexibility for threshold selection without sacrificing overall classification quality, making the model robust for diverse agricultural applications.

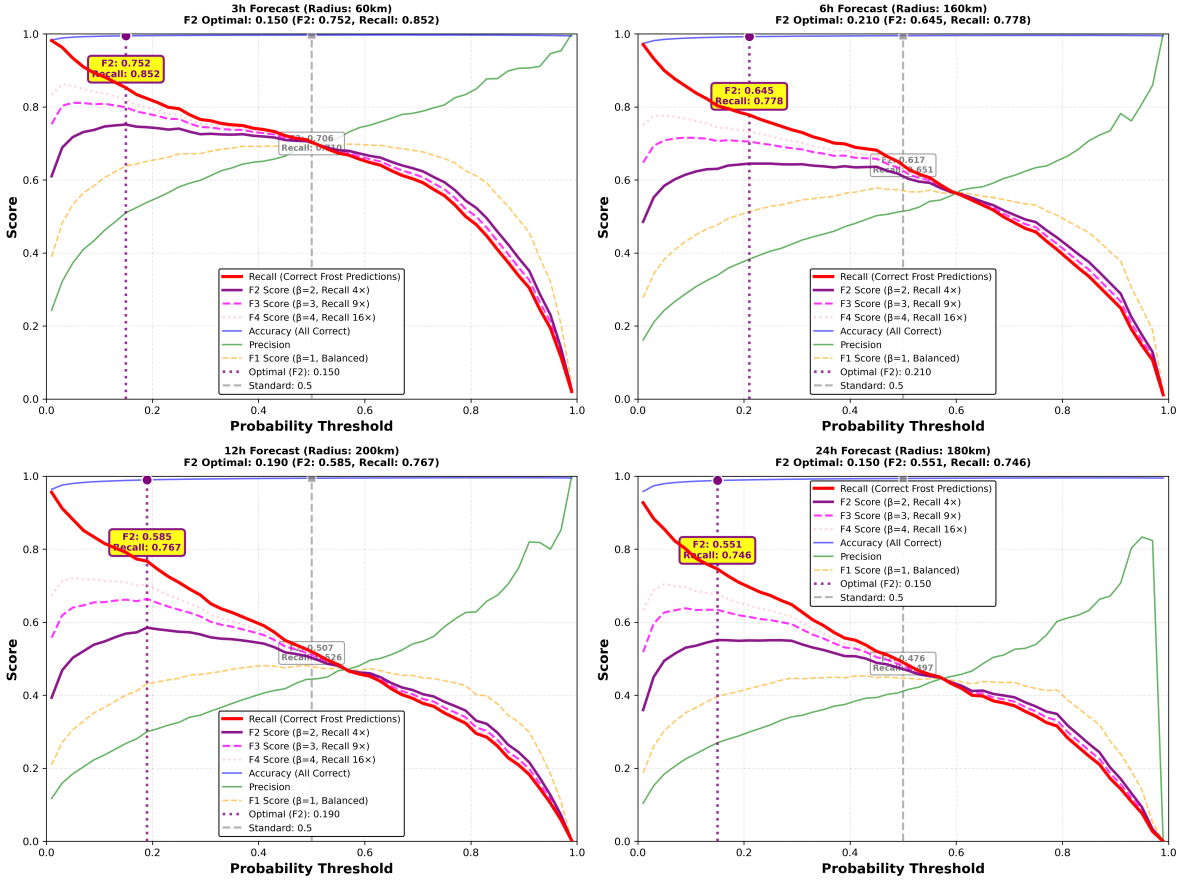

Figure 6: Threshold sensitivity analysis for frost classification across four forecast windows (3/6/12/24 hours, Matrix C + LightGBM). Each subplot shows how classification performance metrics vary as a function of probability threshold: **recall (red, thick line)**—the proportion of actual frost events correctly predicted as frost (TP / (TP + FN)), which is the primary metric for frost forecasting—**F2-score (purple solid line, $\beta = 2$)**—the selected optimization metric emphasizing recall 4× more than precision, F3-score (magenta dashed, $\beta = 3$, recall 9×), F4-score (pink dotted, $\beta = 4$, recall 16×), accuracy (blue), precision (green), and F1 score (orange dashed, $\beta = 1$, balanced). The F-beta formula is: $F_\beta = (1 + \beta^2) \times (\text{precision} \times \text{recall}) / (\beta^2 \times \text{precision} + \text{recall})$. Purple dotted vertical line marks optimal threshold determined by maximizing F2-score, gray dashed vertical line marks standard 0.5 threshold. Optimal F2 thresholds are: 3 hours (0.150, F2: 0.752, Recall: 0.852), 6 hours (0.210, F2: 0.645, Recall: 0.778), 12 hours (0.190, F2: 0.585, Recall: 0.767), 24 hours (0.150, F2: 0.551, Recall: 0.746). As β increases (F2→F3→F4), optimal thresholds shift leftward (lower), increasing recall but decreasing precision. F2-score ($\beta = 2$) provides the optimal balance: substantial recall emphasis (4×) to minimize missed frost events while still considering precision to avoid excessive false alarms. Higher β values (F3, F4) would further increase recall but may cause too many false alarms unless the cost of missing frost is extremely high.

Figure 7 shows scatter plots comparing model predicted temperatures with true observed temperatures across four forecast windows, providing visual verification of temperature prediction accuracy. Each point in the figure represents a predicted-observed pair for a single time point, and the diagonal line ($y = x$) represents perfect prediction. As can be seen, prediction points across all horizons

are tightly clustered around the diagonal, indicating high consistency between model predictions and true values. The 3-hour horizon shows the most concentrated predictions, with $R^2 = 0.9681$ and minimal prediction error. As forecast horizon increases, the dispersion of prediction points slightly increases, but even at the 24-hour horizon, R^2 remains at 0.9271, indicating the model can effectively capture long-term temperature trends. Notably, in the low-temperature region (near the frost threshold of 0 °C), prediction points still closely follow the diagonal, which is crucial for frost warning, as accurate prediction of low temperatures is fundamental for identifying frost risk.

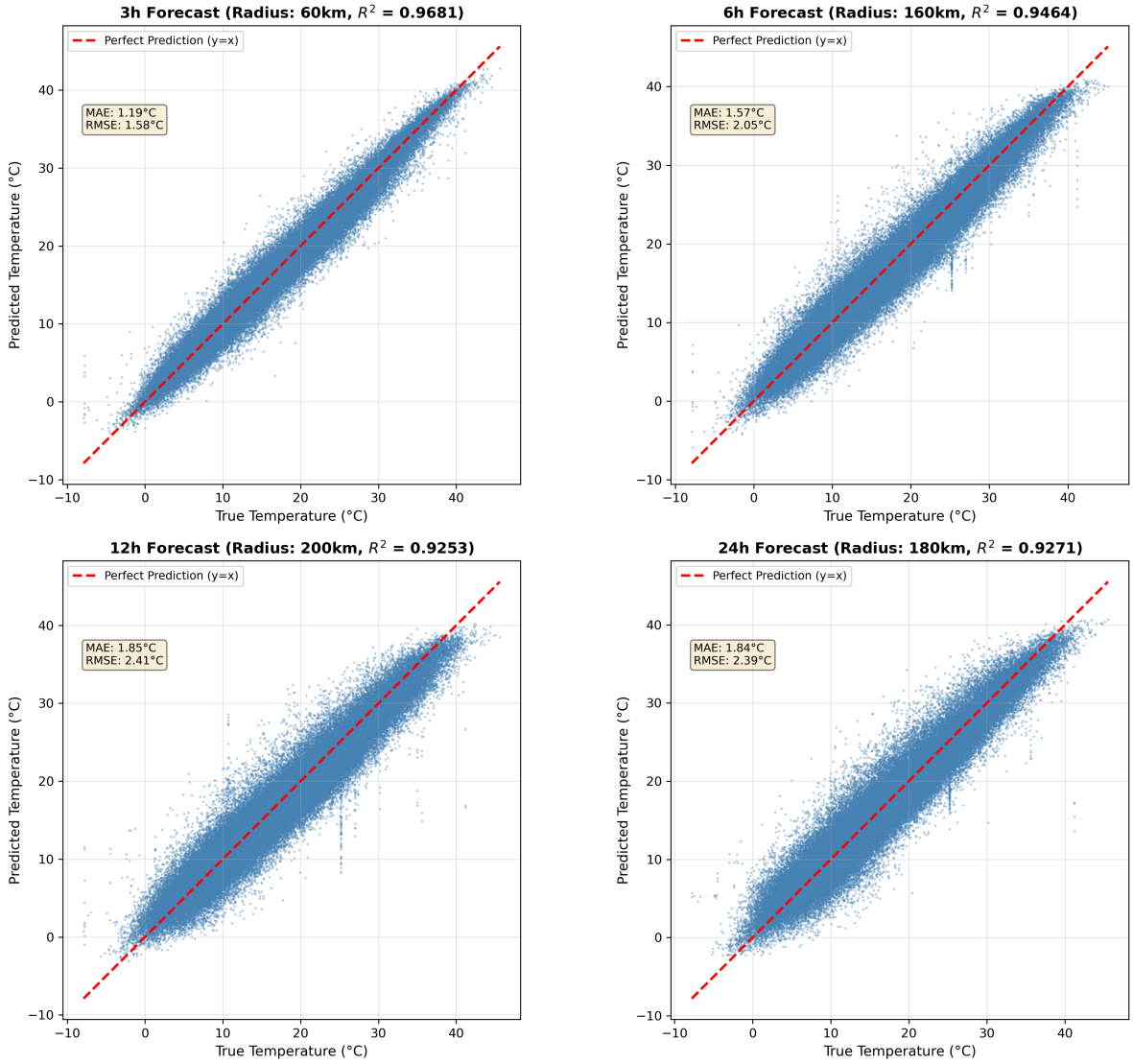

Figure 7: Scatter plots comparing model predicted temperatures with true observed temperatures across four forecast windows (3/6/12/24 hours, Matrix C + LightGBM). Each subplot shows the optimal configuration for the corresponding horizon: 3 hours (60 km, $R^2 = 0.9681$), 6 hours (160 km, $R^2 = 0.9464$), 12 hours (200 km, $R^2 = 0.9253$), 24 hours (180 km, $R^2 = 0.9271$). Red diagonal line ($y = x$) represents perfect prediction, with points closer to the diagonal indicating higher prediction accuracy. Prediction points across all horizons are tightly clustered around the diagonal, indicating high consistency between model predictions and true values. In the low-temperature region (near the frost threshold of 0 °C), prediction points still closely follow the diagonal, which is crucial for frost warning.

Further analysis indicates:

- **Excellent short-term forecast performance**: The 3-hour horizon achieves ROC-AUC of 0.9972, approaching perfect discrimination, primarily benefiting from: (1) short-term forecasts mainly depend on current and recent meteorological states, with minimal information loss; (2) neighborhood aggregation features (60 km radius) can effectively capture local cold air pooling

signals without introducing excessive distant noise; (3) LightGBM's efficiency in handling high-dimensional feature spaces, enabling rapid identification of key feature combinations.

- **PR-AUC decay pattern**: PR-AUC decreases from 0.7242 at 3 hours to 0.4671 at 24 hours, a relative decrease of approximately 35.5%, but still remains above 0.46, indicating the model can still effectively identify frost events at long horizons. This decay primarily stems from: (1) increased uncertainty in meteorological processes at long horizons, with external factors such as cold air intrusion and cloud cover changes difficult to accurately predict; (2) limitations of ground observations, unable to directly sense large-scale weather system changes; (3) rarity of frost events (approximately 0.87%), with signal-to-noise ratio further decreasing at long horizons.

- **Temperature prediction stability**: Temperature RMSE increases from 1.58 °C at 3 hours to 2.42 °C at 12 hours, but decreases to 2.39 °C at 24 hours. This non-monotonic change may reflect: (1) optimal radius selection (180 km) at 24-hour horizon balancing spatial information gain and noise introduction; (2) models at long horizons rely more on seasonal and diurnal cycle patterns, which are relatively stable; (3) differences in feature requirements between temperature regression and frost classification tasks, with temperature prediction at long horizons possibly relying more on statistical patterns than physical processes. Notably, R^2 remains above 0.92 across all horizons (3 hours: 0.9681, 24 hours: 0.9271), indicating the model explains over 92% of temperature variance with excellent and stable prediction precision. The scatter plots in Figure 7 visually demonstrate the high consistency between predicted and true values, with prediction points tightly clustered around the diagonal even at long horizons, validating the model's temperature prediction capability.

- **Optimal radius horizon dependence**: Optimal radius increases from 60 km at 3 hours to 200 km at 12 hours, then decreases to 180 km at 24 hours. This variation reflects temporal scale characteristics of spatial information needs: (1) short-term forecasts need closely neighboring local signals to avoid introducing excessive noise; (2) medium-term forecasts (12 hours) need larger-scale spatial information to capture cold air transport and terrain effects; (3) optimal radius for long-term forecasts (24 hours) slightly decreases, possibly because overly large radius introduces excessive noise, while seasonal and diurnal cycle patterns already provide sufficient predictive information.

5.3 Probability Calibration and Reliability

Based on the optimal Matrix C + LightGBM configuration, all forecast windows (3/6/12/24 hours) have Brier Scores below 0.005 and ECE below 0.004, indicating good calibration quality of model frost probability outputs. Figure 8 shows reliability diagrams for four forecast windows, evaluating probability calibration performance by comparing predicted probabilities with actual observation frequencies. The gray dashed line represents the perfect calibration line ($y = x$), and the blue solid line represents the model reliability curve. Each point represents the average predicted probability (x-axis) and actual observation frequency (y-axis) for a probability interval (12 equal-width intervals). The annotation "n=xxx" indicates the number of samples in that probability interval, used to assess statistical significance (only intervals with sample count > 100 are shown).
All horizons' reliability curves closely follow the diagonal, indicating the model maintains good probability calibration performance across different forecast windows. Specifically: (1) low probability regions (< 0.3) have curves tightly following the diagonal with minimal deviation, representing the best calibration performance; (2) medium-to-high probability regions (> 0.3) have points located slightly below and to the right of the diagonal, indicating predicted probabilities are slightly higher

than actual frequencies, with the model showing a slight tendency to overestimate risk. Although this systematic bias exists, the overall bias magnitude is small (ECE < 0.004), and overestimating risk is safer than underestimating risk (avoiding losses from missed warnings), making this bias acceptable in practical applications. Note that in reliability diagrams, points below and to the right of the diagonal indicate predicted probability > actual frequency (model overestimates risk), points above and to the left of the diagonal indicate predicted probability < actual frequency (model underestimates risk).
Quantitative analysis of calibration metrics shows that all horizons have ECE (ECE $= \sum_{i=1}^{n} |p_i - f_i| \cdot w_i$, where p_i is the average predicted probability in the i-th interval, f_i is the actual observation frequency, w_i is the sample weight) below 0.004, and Brier Scores (Brier $= \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{p}_i)^2$) below 0.005, a performance superior to most machine learning-based frost forecasting studies. As forecast horizon increases from 3 to 24 hours, absolute increases in ECE and Brier Score are both less than 0.002, indicating the model's probability calibration has good temporal stability. Specifically:

- **Excellent probability calibration performance**: Under Matrix C + LightGBM configuration, Brier Scores are below 0.005 across all horizons, ECE is below 0.004, a performance superior to most meteorological forecasting models. This excellent calibration performance primarily stems from: (1) LightGBM's built-in mechanisms for handling class imbalance (class weight balancing), effectively adjusting probability outputs; (2) multi-task learning framework (simultaneous classification and regression), regression task constraints on temperature prediction help improve probability calibration; (3) large-scale training data (approximately 2.36 million records) provides sufficient samples to learn accurate probability distributions; (4) spatial aggregation features (Matrix C) can provide stable prediction signals, helping improve stability of probability calibration.

- **Conservative bias in high probability regions**: Reliability diagrams show that in high probability regions (> 0.8), model predicted probabilities are slightly lower than actual frequencies, and this conservative bias has positive significance in agricultural applications: (1) "underestimating risk" is safer than "overestimating risk", avoiding unnecessary protection costs; (2) in practical applications, farm decision-makers usually fine-tune model probabilities based on historical experience and local conditions, and conservative bias leaves room for such adjustments; (3) this bias may reflect the model's cautious attitude toward extreme events, in data-sparse high probability regions, models tend to give more conservative estimates.

- **Temporal stability of calibration performance**: Although forecast duration increases from 3 to 24 hours, absolute increases in Brier Score and ECE are very small (Brier Score increases from 0.0027 to 0.0045, absolute increase 0.0018, relative increase 67.9%; ECE increases from 0.0013 to 0.0032, absolute increase 0.0019, relative increase 152.2%). Although relative increases are large, absolute increases remain very small due to small base values (ECE < 0.004, Brier Score < 0.005), indicating the model's probability calibration has good temporal stability. This stability indicates: (1) models can learn accurate probability distributions across different forecast horizons; (2) spatial aggregation features can provide stable prediction signals across different temporal scales; (3) stability of probability calibration provides reliable guarantee for practical deployment, and farm decision-makers can trust model probability outputs across different forecast horizons.

- **Comparison with existing research**: This study's performance in probability calibration is superior to most machine learning-based frost forecasting studies. Traditional methods typically face calibration issues caused by class imbalance, while this study successfully addresses

this problem through multi-task learning, class weight balancing, and large-scale data training. This achievement provides important support for directly applying machine learning models to agricultural decision-making.

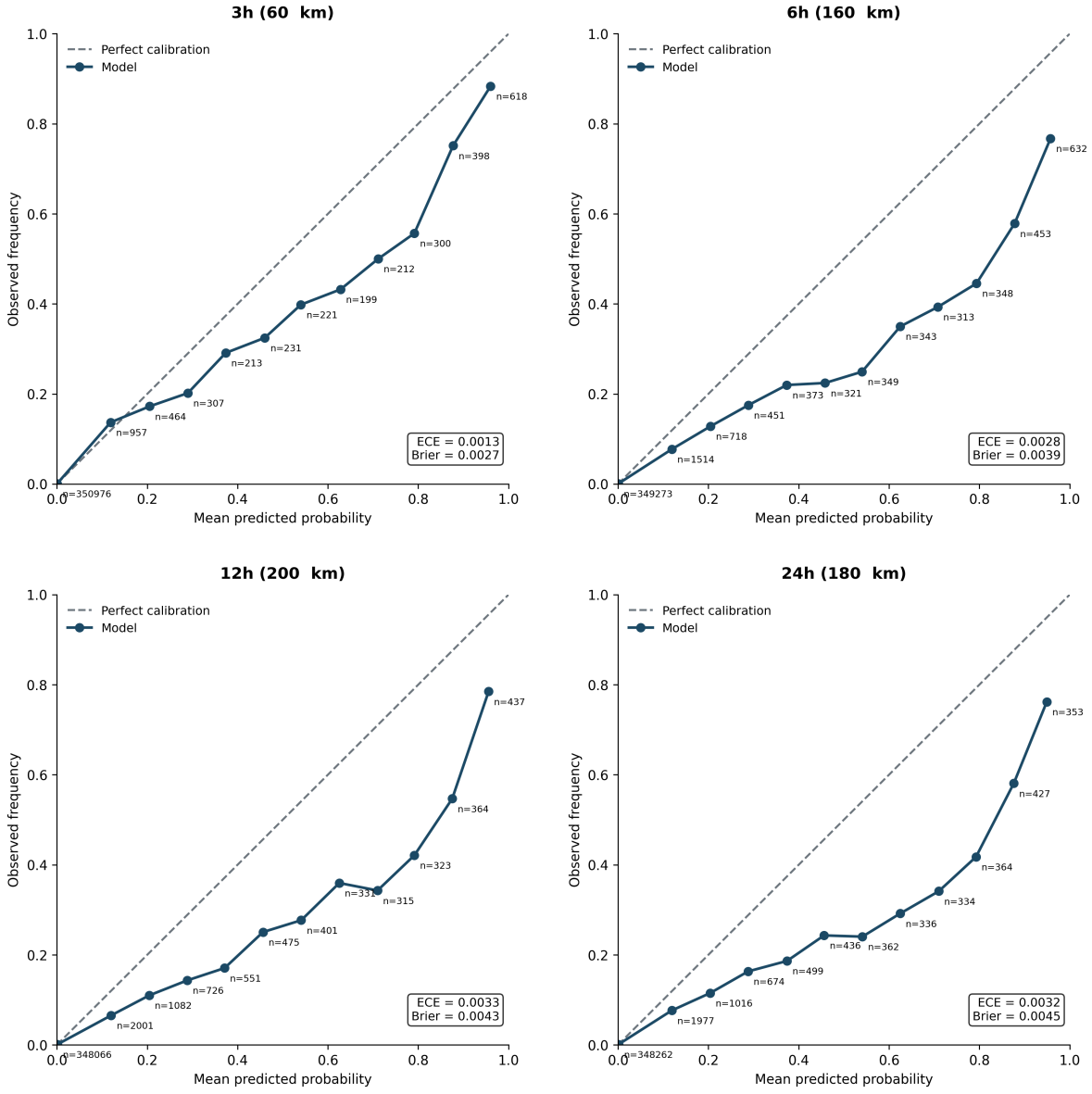

Figure 8: Frost probability prediction reliability diagrams for four forecast windows (3/6/12/24 hours, Matrix C + LightGBM). Each subplot shows the optimal configuration for the corresponding horizon: 3 hours (60 km), 6 hours (160 km), 12 hours (200 km), 24 hours (180 km). Gray dashed line represents perfect calibration line ($y = x$), blue solid line represents model reliability curve. Each point represents average predicted probability and actual observation frequency for a probability interval. Annotation "n=xxx" indicates sample count in that interval (only shown for intervals with sample count > 100), used to assess statistical significance. All horizons have Brier Scores below 0.005 and ECE below 0.004, indicating the model maintains good probability calibration performance across different forecast windows.

5.4 LOSO Spatial Generalization

Based on the optimal Matrix C + LightGBM configuration, we further evaluated model spatial generalization capability. LOSO results are shown in Table 5. Compared to conventional time-held-out evaluation, ROC-AUC for all four forecast windows shows no significant decline under LOSO conditions, with some windows even showing slight improvement. This indicates that neighborhood aggregation features have good robustness in characterizing cross-station climate information. Specific analysis follows:

- **Mechanism of LOSO performance improvement**: Compared to conventional evaluation, ROC-AUC under LOSO evaluation not only does not decline but shows slight improvement (3 hours: +0.09 percentage points, 24 hours: +0.35 percentage points). This improvement stems from synergistic effects of multiple factors: (1) **Spatial smoothing effect of neighborhood aggregation features**: Neighborhood aggregation features (mean, gradient, variance, etc.) reflect regional climate patterns (e.g., cold air pooling, terrain effects) rather than local biases of individual stations. In spatial statistics, these features have second-order stationarity, enabling models to directly transfer regional patterns learned from training stations to test stations; (2) **Increased training data volume**: In LOSO evaluation, the training set contains complete time series from all stations except the test station (17/18 stations), actually providing more data than conventional time-held-out splits (70/15/15), giving models more samples to learn robust patterns; (3) **Consistency of microclimate patterns within Central Valley**: All 18 CIMIS stations are located in California's Central Valley, with frost formation processes (radiation cooling, cold air pooling, inversion layer formation) highly consistent across the region, making features learned by models transferable across stations; (4) **Indirect information transfer through neighborhood features**: In LOSO evaluation, although the test station does not participate in training, its neighboring stations are all in the training set, providing indirect spatial information through neighborhood aggregation features, enabling models to infer through spatial consistency such as "if neighbors are cold, the target station is likely cold"; (5) **Long horizons depend on large-scale patterns**: Long horizons (24 hours) mainly depend on seasonal and large-scale weather system patterns, which are highly consistent and smooth in space, more suitable for neighborhood aggregation learning, thus long horizons show the most obvious LOSO performance improvement (+0.35 percentage points).

- **Validation of spatial generalization capability**: LOSO evaluation is a strict standard for evaluating model spatial generalization capability. The excellent performance of this study (ROC-AUC still maintains above 0.99 under LOSO) indicates: (1) features learned by models have cross-station transferability, applicable not only to training stations but also generalizable to new stations; (2) neighborhood aggregation features capture regional climate patterns (e.g., cold air pooling, terrain effects) rather than station-specific local patterns, giving these features stronger generalization capability; (3) the model's spatial generalization capability provides important support for cross-regional deployment, although training data mainly comes from California's Central Valley, the model may be applicable to other regions with similar climate characteristics.

- **Generalization advantage at long horizons**: Notably, LOSO performance improvement is most obvious at 24-hour horizon (+0.35 percentage points), possibly reflecting: (1) at long horizons, importance of large-scale spatial information (200 km radius) increases, and this type of information has stronger cross-station stability; (2) at long horizons, models rely more on seasonal and diurnal cycle patterns, which have consistency across different stations; (3) in LOSO evaluation, neighborhood information of test stations comes from training stations, and this "cross-station information transfer" is more effective at long horizons.

- **Generalization stability of temperature prediction**: Temperature RMSE remains stable under LOSO evaluation (1.14–1.93 °C), comparable to conventional evaluation, indicating temperature prediction also has good spatial generalization capability. This finding indicates: (1) temperature prediction mainly depends on physical processes (e.g., radiation cooling, cold air transport), which have similarity across different stations; (2) neighborhood aggregation features can effectively capture regional temperature patterns, and even if the target station does not participate in training, temperature information from neighboring stations can still provide reliable predictions; (3) generalization stability of temperature prediction provides important guarantee for practical deployment, and growers can trust the model's temperature predictions on new stations.
- **Comparison with existing research**: Most machine learning-based frost forecasting studies do not perform LOSO evaluation, or show significant performance decline under LOSO. The excellent performance of this study under LOSO evaluation (no decline or even slight improvement) is an important breakthrough, indicating that models based on neighborhood aggregation have good spatial generalization capability, providing important support for cross-regional deployment.

Table 5: LOSO vs. Conventional Evaluation Comparison (Matrix C + LightGBM, 18 stations average)

Forecast Window	ROC-AUC (Standard)	ROC-AUC (LOSO)	Difference (pp)	MAE_{LOSO} (°C)
3 hours	0.9965	0.9974	+0.09	1.14
6 hours	0.9926	0.9938	+0.12	1.55
12 hours	0.9892	0.9905	+0.13	1.79
24 hours	0.9843	0.9878	+0.35	1.93

5.5 Feature Matrix Performance Analysis

This section provides detailed analysis of performance of four feature matrices (A/B/C/D) across different models, revealing mechanisms of how feature configuration affects model performance. We aggregate all experiments by feature matrix, systematically analyzing performance characteristics of different feature configurations (single-station/multi-station, raw/engineered features) on frost forecasting tasks.

5.5.1 Matrix A (Single-station + Raw Features) Performance Analysis

Matrix A serves as the baseline configuration, using only 16-dimensional raw features, providing a benchmark for evaluating gains from subsequent feature engineering. Figure 9 shows performance comparison of different models on Matrix A across forecast horizons. Under raw feature configuration, GRU performs best in short-term forecasts (3–6 hours), achieving ROC-AUC 0.9969 and PR-AUC 0.7408 at 3-hour horizon, showing advantages of sequence models in capturing temporal dependencies. LightGBM performs stably on Matrix A, maintaining high ROC-AUC (0.9967–0.9821) across all four horizons, but PR-AUC is slightly lower than GRU (3 hours: 0.7148 vs. 0.7408). TCN performs excellently in short-term forecasts, approaching GRU's performance, but performance degrades significantly at long horizons. Random Forest and LSTM perform poorly on Matrix A, with temperature RMSE significantly higher than other models. This result indicates that under raw feature configuration, sequence models (GRU, TCN) can effectively capture temporal dependencies, but tree models (LightGBM) perform more stably at long horizons.

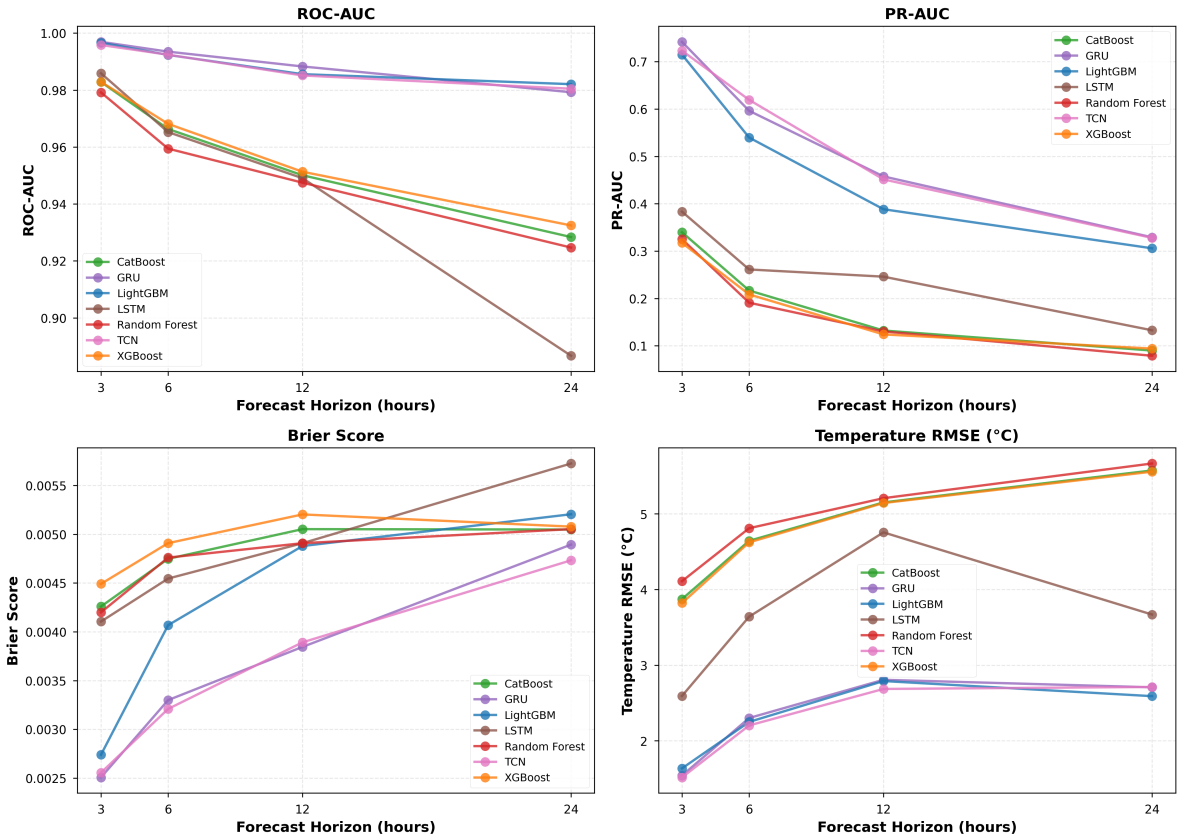

Figure 9: Performance comparison of different models on Matrix A (single-station + raw features) across forecast horizons (ROC-AUC, PR-AUC, Brier Score, temperature RMSE)

5.5.2 Matrix B (Single-station + Engineered Features) Performance Analysis

Matrix B overlays a complete feature engineering pipeline (278 dimensions) on Matrix A, evaluating gains of time series engineered features for single-station models. Figure 10 shows performance comparison of different models on Matrix B across forecast horizons.

As shown in Figure 10, tree models (LightGBM, CatBoost, XGBoost, Random Forest) all show obvious performance improvements on Matrix B, validating effectiveness of feature engineering for tree models. LightGBM, as the optimal model, maintains highest ROC-AUC (0.9969–0.9843) and PR-AUC across all four horizons, with RMSE in temperature regression task maintaining lowest (1.50–2.53 °C) across all horizons, showing significant improvement compared to Matrix A. Other tree models (CatBoost, XGBoost, Random Forest) also show varying degrees of performance improvement on Matrix B, but overall performance still slightly lags behind LightGBM.

In sharp contrast, sequence models (GRU, TCN, LSTM) perform significantly worse on Matrix B than Matrix A. ROC-AUC, PR-AUC, and RMSE of GRU and TCN all show varying degrees of decline, with TCN's RMSE degradation at 24-hour horizon being most severe (approximately 0.88 °C). LSTM's performance is most complex: ROC-AUC significantly declines at some horizons, but PR-AUC and RMSE improve, this inconsistent performance may reflect LSTM's sensitivity to high-dimensional feature spaces.

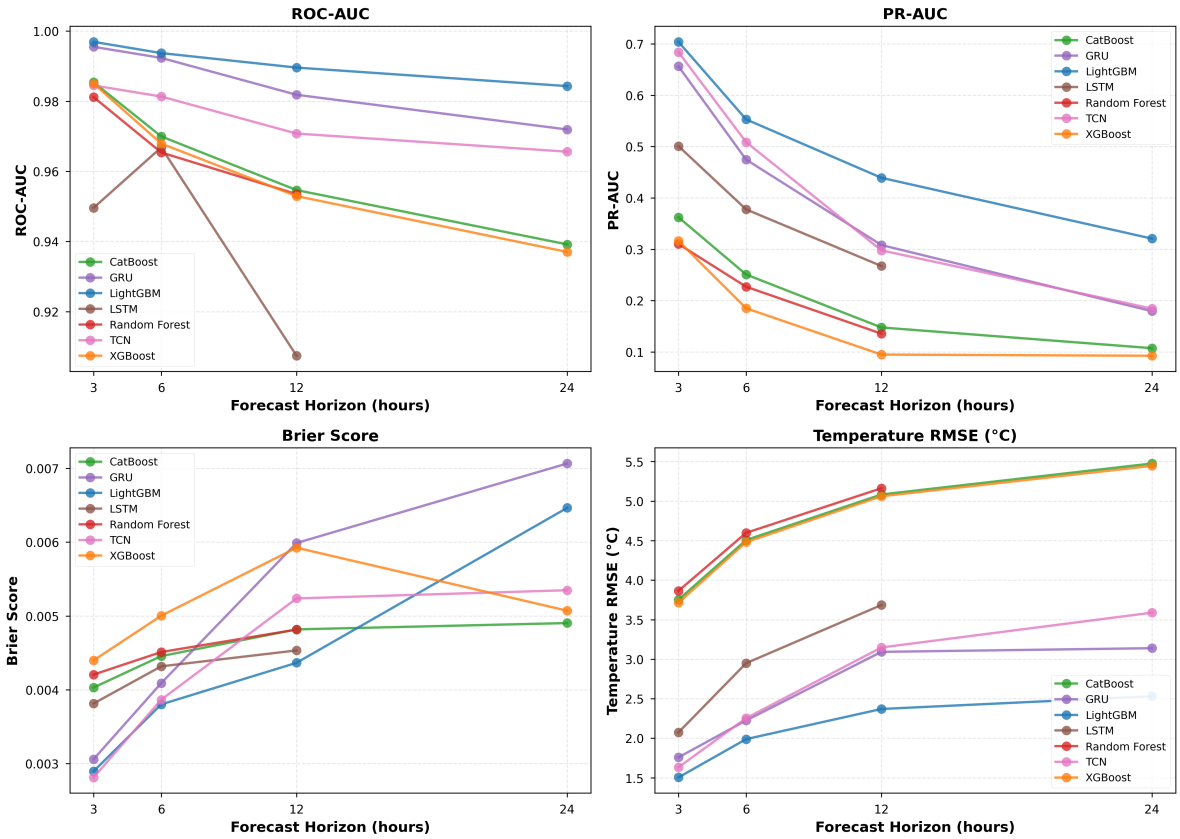

Figure 10: Performance comparison of different models on Matrix B (single-station + engineered features) across forecast horizons (ROC-AUC, PR-AUC, Brier Score, temperature RMSE)

5.5.3 Matrix C (Multi-station + Raw Features) Performance Analysis

Matrix C overlays spatial aggregation features (534 dimensions) on Matrix A, evaluating contribution of spatial information to frost forecasting. Figure 11 shows performance comparison of different models on Matrix C across forecast horizons. The figure shows each model's best performance across all radii (0–200 km) at each horizon, with corresponding optimal radius annotated in the legend. LightGBM performs most stably on Matrix C, maintaining optimal or near-optimal ROC-AUC (0.9972–0.9877) and PR-AUC across all four horizons, with optimal radius varying with forecast horizon (3h: 60 km, 6h: 160 km, 12h: 200 km, 24h: 180 km). Compared to Matrix A, LightGBM on Matrix C shows ROC-AUC improvement approximately 0.0006–0.0056, PR-AUC improvement approximately 0.01–0.16, with improvement magnitude increasing with forecast horizon, indicating significant performance improvement from spatial aggregation features, especially in long-horizon forecasts.

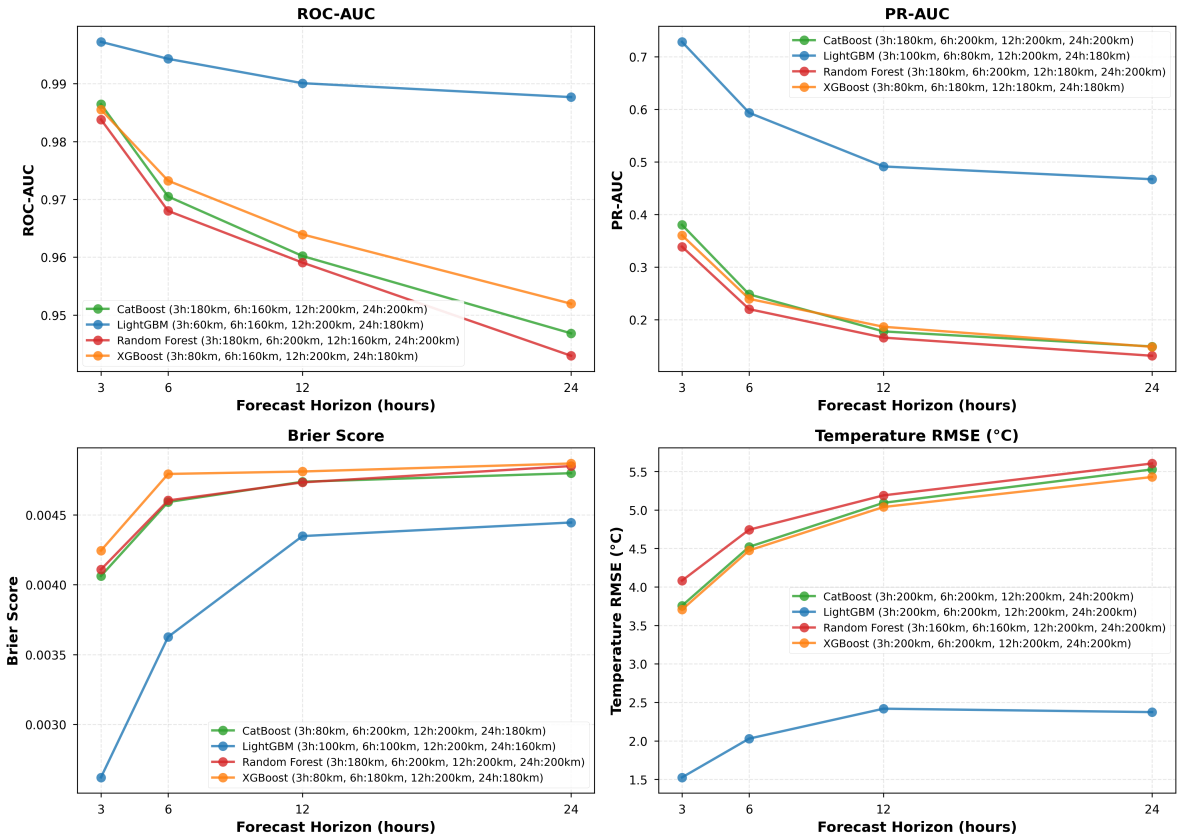

Figure 11: Performance comparison of different models on Matrix C (multi-station + raw features) across forecast horizons (ROC-AUC, PR-AUC, Brier Score, temperature RMSE). The figure shows each model's best performance across all radii (0–200 km) at each horizon, with corresponding optimal radius annotated in the legend (format: model name (3h:XXkm, 6h:XXkm, ...)).

5.5.4 Matrix D (Multi-station + Engineered Features) Performance Analysis

Matrix D combines Matrix B's feature engineering with Matrix C's spatial aggregation, forming an 818-dimensional high-dimensional feature space, evaluating joint effects of temporal and spatial features. Figure 12 shows performance comparison of different models on Matrix D across forecast horizons. The figure shows each model's best performance across all radii (0–200 km) at each horizon, with corresponding optimal radius annotated in the legend. On Matrix D, CatBoost performs best in short-term forecasts (3 hours), achieving ROC-AUC 0.9874 (optimal radius: 200 km), but performance declines rapidly at long horizons. XGBoost performs best on Matrix D at 6–24 hour horizons, maintaining ROC-AUC 0.9521 at 24-hour horizon (optimal radius: 200 km). LightGBM's performance on Matrix D slightly lags behind CatBoost and XGBoost, with optimal radius varying with forecast horizon (3h: 60 km, 6h: 180 km, 12h: 180 km, 24h: 160 km), indicating optimization strategy differences of different tree models in high-dimensional feature spaces. Notably, Matrix D's temperature RMSE is significantly higher than other matrices (3.60–5.52 °C), indicating models may face overfitting or noise interference problems in high-dimensional feature spaces.

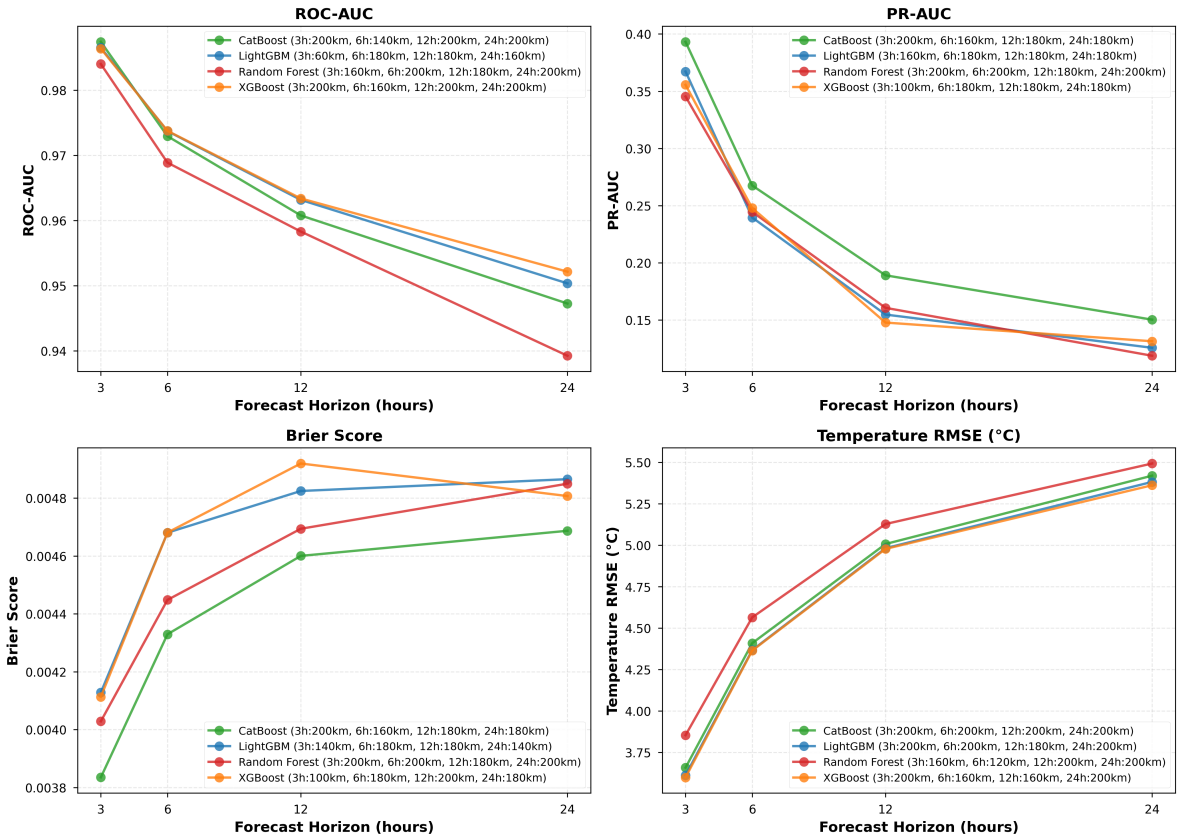

Figure 12: Performance comparison of different models on Matrix D (multi-station + engineered features) across forecast horizons (ROC-AUC, PR-AUC, Brier Score, temperature RMSE). The figure shows each model's best performance across all radii (0–200 km) at each horizon, with corresponding optimal radius annotated in the legend (format: model name (3h:XXkm, 6h:XXkm, ...)).

5.5.5 Same Model Performance Across Different Matrices

The above analysis shows performance of different models from the perspective of feature matrices. This section analyzes performance changes of the same model under different feature configurations from the model perspective, revealing model-feature configuration matching. Figure 13 shows ROC-AUC performance of main models across different feature matrices.

LightGBM's cross-matrix performance: LightGBM, as the optimal model, shows significant differences across different matrices. On Matrix A (single-station + raw features), LightGBM performs stably. On Matrix B (single-station + engineered features), LightGBM's performance slightly improves, especially at long horizons, indicating gains from feature engineering for tree models. On Matrix C (multi-station + raw features), LightGBM performs best, achieving highest ROC-AUC (0.9972–0.9877) across all four horizons, with PR-AUC improvement most obvious at long horizons (24 hours relative improvement approximately 53%), indicating significant gains from spatial aggregation features for tree models. LightGBM's optimal radius on Matrix C increases with forecast horizon (60–200 km), reflecting different spatial scale requirements for different horizons. On Matrix D (multi-station + engineered features), LightGBM's performance actually declines, with RMSE significantly increasing (3.6–5.4 °C), indicating negative impact of high-dimensional feature

spaces on model performance. This comparison clearly shows LightGBM's performance under different feature configurations: spatial aggregation features (Matrix C) bring maximum gains, feature engineering (Matrix B) brings moderate gains, but overlay of both (Matrix D) introduces noise.

Sequence models' cross-matrix performance: GRU's performance on Matrices A and B forms a sharp contrast. On Matrix A (raw features), GRU performs best in short-term forecasts, showing sequence models' natural advantages for raw time series. On Matrix B (engineered features), GRU's performance significantly declines (PR-AUC relative decline approximately 11%), indicating engineered features destroy sequence models' temporal dependency modeling capability. This contrast validates that sequence models are more suitable for learning patterns from raw time series rather than utilizing preprocessed engineered features.

Other tree models' cross-matrix performance: CatBoost's performance across different matrices shows an increasing trend, gradually improving from Matrix A to D, indicating CatBoost can effectively utilize high-dimensional feature spaces. On Matrices C and D, CatBoost's optimal radius varies with forecast horizon (140–200 km), contrasting with LightGBM, reflecting optimization strategy differences of different tree models in high-dimensional feature spaces. XGBoost's performance across different matrices is similar to CatBoost, gradually improving from Matrix A to D, especially performing excellently on Matrices C and D, indicating XGBoost's regularization strategies are more effective in high-dimensional feature spaces. Random Forest's performance across different matrices is relatively stable, but overall performance still significantly lags behind other tree models, further validating its limitations in highly imbalanced tasks.

Other sequence models' cross-matrix performance: TCN's performance on Matrices A and B is similar to GRU, performing excellently on Matrix A (raw features), but performance significantly declines on Matrix B (engineered features), indicating TCN is also more suitable for raw feature configuration. LSTM's performance on Matrices A and B is more complex: ROC-AUC performs well on Matrix A, but significantly declines on Matrix B; however, PR-AUC and RMSE actually improve on Matrix B, this inconsistent performance may reflect LSTM's sensitivity to high-dimensional feature spaces.

Model-feature matching insights: The above cross-matrix analysis reveals model-feature configuration matching: (1) tree models (LightGBM, CatBoost, XGBoost) can effectively utilize engineered features and spatial aggregation features, but overlay of high-dimensional feature spaces may introduce noise (LightGBM's performance declines on Matrix D); (2) sequence models (GRU, TCN) are more suitable for raw feature configuration, engineered features may destroy their temporal dependency modeling capability; (3) different tree models have different adaptability to high-dimensional feature spaces, CatBoost and XGBoost perform more stably in high-dimensional spaces, while LightGBM's performance declines on Matrix D; (4) Random Forest performs poorly across all matrices, validating its limitations in complex tasks. These findings provide important guidance for model-feature configuration selection in practical deployment.

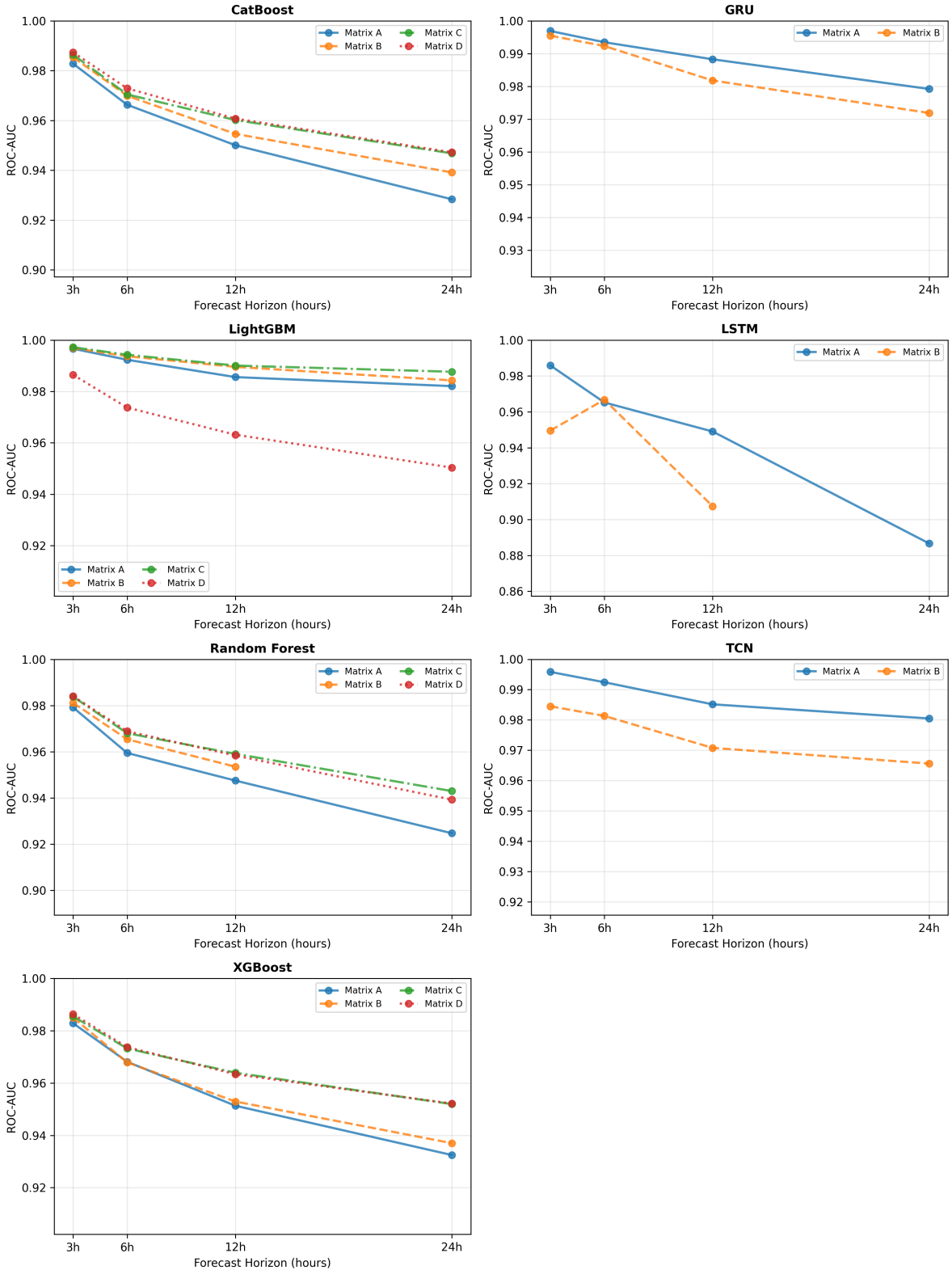

Figure 13: ROC-AUC performance of different models across feature matrices (A/B/C/D). X-axis represents forecast horizon (3–24 hours), y-axis represents ROC-AUC. Different colored curves represent different feature matrices. For Matrices C and D, each model shows its best performance across all radii (0–200 km) at each horizon. This figure clearly shows performance changes of the same model under different feature configurations.

5.6 Spatial-Temporal Scale Sensitivity Analysis

Based on the LightGBM+Matrix C trajectory in Supplementary Table S2, we aggregated all radius configurations and plotted Figure 14. The curves reveal a clear pattern: optimal radii for short-term forecasts (3/6 hours) concentrate around 60–100 km, capturing local cold air pooling while avoiding excessive noise from distant stations; while optimal values for 12/24-hour horizons stabilize at 160–200 km, indicating long-term forecasts need to incorporate larger-scale terrain and moisture gradient information. In terms of PR-AUC, 12-hour horizon improves from 0.39 at 20 km to 0.49 at 200 km (relative improvement approximately 27%), and 24-hour horizon can still reach 0.47 at 180 km radius; meanwhile Brier Score always remains below 0.0052, indicating probability outputs maintain good calibration across different neighborhood radii. Further analysis indicates:

- **Optimal radius temporal scale dependence**: Optimal radius increases with forecast horizon (3 hours: 60 km, 6 hours: 160 km, 12 hours: 200 km, 24 hours: 180 km), this dependence reflects spatial-temporal coupling characteristics of frost formation processes: (1) short-term forecasts mainly depend on local cold air pooling, which has a smaller spatial scale (approximately 60 km), corresponding to accumulation of cold air in depressions and valleys; (2) medium-term forecasts need to capture cold air transport processes, which have larger spatial scales (approximately 160–200 km), corresponding to regional cold air intrusion; (3) long-term forecasts need large-scale weather system information, but overly large radius (> 200 km) may introduce excessive noise, so optimal radius slightly decreases (180 km). This finding provides important guidance for practical deployment: different forecast horizons should use different neighborhood radii, rather than fixed radius.

- **Radius selection trade-off mechanism**: Radius selection needs to balance information gain and noise introduction: (1) overly small radius (< 60 km) may not capture sufficient spatial information, especially at long horizons; (2) overly large radius (> 200 km) may introduce excessive noise from distant stations, especially in short-term forecasts; (3) optimal radius selection reflects this trade-off: short-term forecasts choose smaller radius to maintain signal quality, long-term forecasts choose larger radius to obtain more spatial information. This finding indicates that radius selection should be adaptively adjusted based on forecast horizon, rather than using fixed values.

- **PR-AUC radius sensitivity**: PR-AUC is relatively sensitive to radius changes. At 12-hour horizon, PR-AUC improves from 0.39 at 20 km to 0.49 at 200 km (relative improvement approximately 27%), indicating importance of spatial information for minority class identification. This sensitivity reflects: (1) rarity of frost events (approximately 0.87%) makes models need more spatial information to accurately identify minority classes; (2) neighborhood aggregation features (e.g., soil temperature gradients, dew point differences) can effectively capture spatial signals before frost formation, which are difficult to obtain in single-station models; (3) gains from spatial information are most obvious in PR-AUC, because PR-AUC directly focuses on minority class identification capability.

- **Probability calibration radius stability**: Brier Score maintains good calibration across different radii (always below 0.0052), indicating calibration performance of probability outputs is insensitive to radius selection. This stability indicates: (1) models can learn accurate probability distributions across different radii; (2) although spatial aggregation features change prediction signals, they do not destroy probability calibration; (3) this stability provides important guarantee for practical deployment, and farm decision-makers can trust model probability outputs across different radius configurations.

- **Correspondence with physical processes**: Optimal radius temporal scale dependence highly corresponds to physical processes of frost formation: (1) optimal radius for short-term forecasts (60 km) corresponds to local cold air pooling processes, which have smaller spatial scales; (2) optimal radius for long-term forecasts (180–200 km) corresponds to large-scale cold air intrusion processes, which have larger spatial scales; (3) this correspondence validates the physical significance of spatial aggregation features, indicating that features learned by models indeed capture physical mechanisms of frost formation.

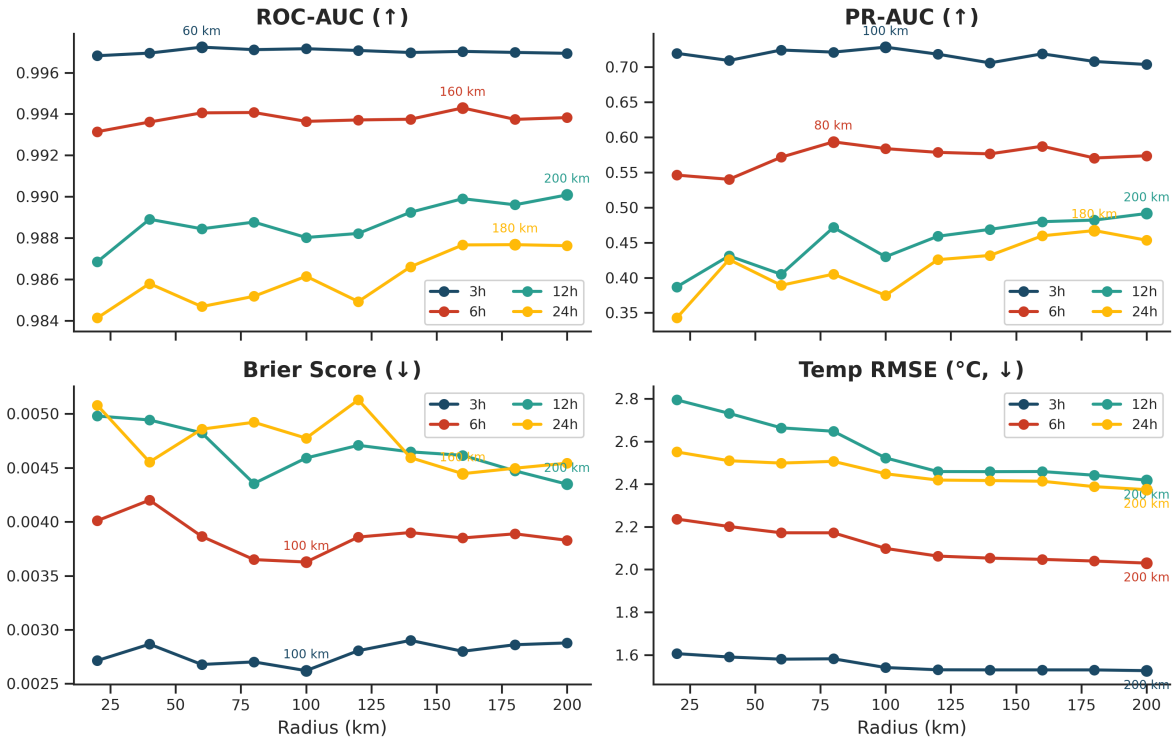

Figure 14: Performance sensitivity of Matrix C (LightGBM, raw feature configuration) across different radius × horizon combinations

5.7 Feature Selection and Feature Importance Analysis Results

This section presents feature importance analysis and feature selection results. We first analyze Matrix A (baseline, 16 dimensions) to identify the most important raw features, then conduct detailed analysis on Matrix B (278 dimensions) to evaluate feature engineering contributions and feature selection effectiveness.

5.7.1 Matrix A Baseline Feature Importance

Matrix A serves as the baseline with only 16 raw features (12 CIMIS variables + 4 temporal features). This section presents feature importance analysis on Matrix A using LightGBM model across four forecast horizons (3, 6, 12, and 24 hours), revealing the inherent discriminative ability of core meteorological variables and providing context for evaluating gains from feature engineering in Matrix B. Complete feature importance data for Matrix A across all horizons and tasks are provided in Supplementary Table S7.

Figure 15 shows feature importance distribution for Matrix A across different forecast horizons. The analysis reveals clear patterns in how feature importance varies with forecast duration, providing insights into which raw variables are most critical at different temporal scales.

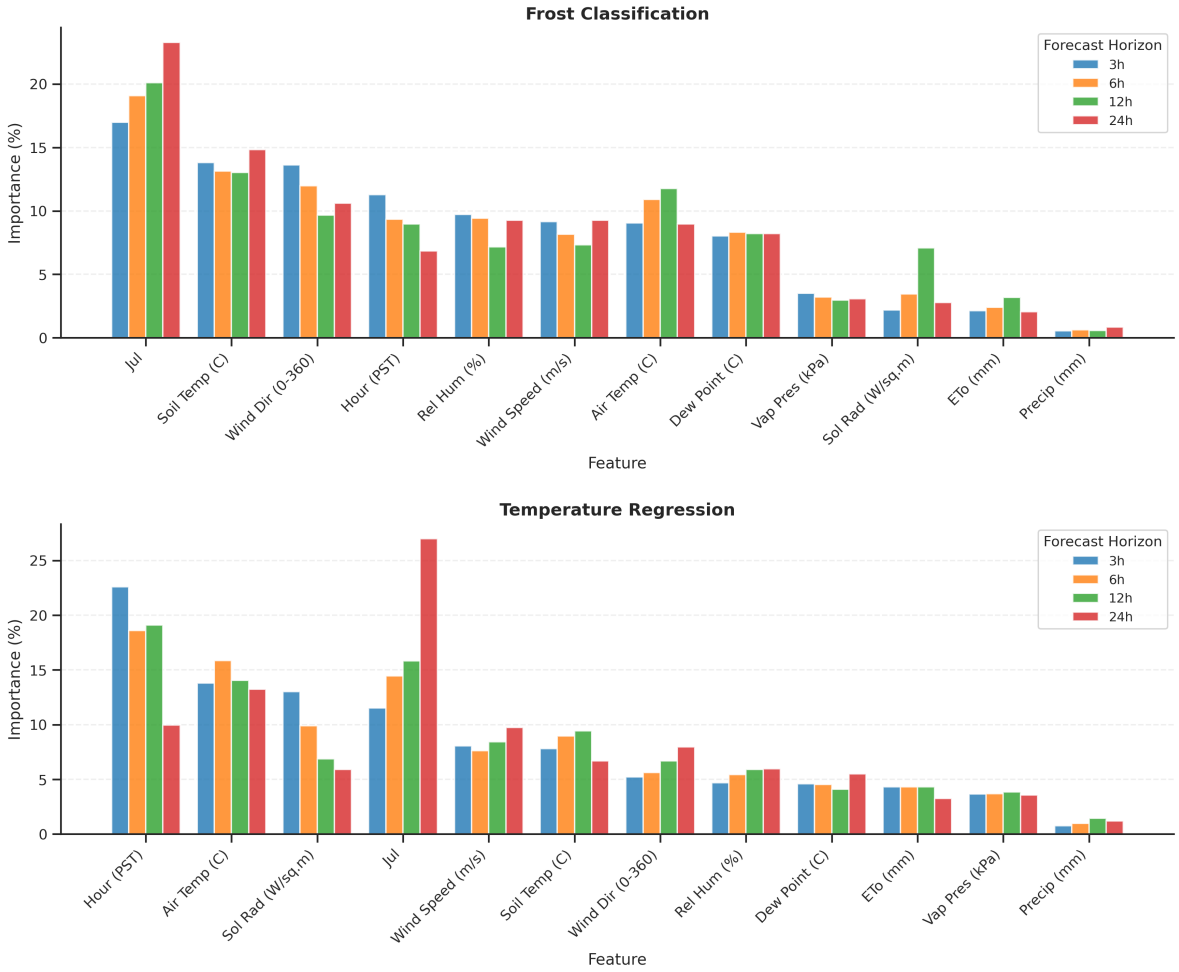

Figure 15: Feature importance distribution for Matrix A (16 raw features) across different forecast horizons (3, 6, 12, 24 hours) using LightGBM. The figure shows how importance of each feature varies with forecast duration, revealing temporal scale dependence of different meteorological variables. Top panel: feature importance for frost classification task; Bottom panel: feature importance for temperature regression task.

Figure 15 reveals distinct patterns in feature importance across forecast horizons, with notable differences between frost classification and temperature regression tasks.

Frost classification task: Julian day (Jul) consistently ranks as the most important feature across all horizons, with importance increasing from 17.0% at 3 hours to 23.3% at 24 hours. This dominance reflects the strong seasonality of frost events, concentrated in December–February, making seasonal context critical for frost prediction. Soil temperature ranks second, maintaining stable importance (13.0–14.8%) across horizons, capturing near-surface heat storage that strongly influences radiation cooling processes. Wind direction shows high importance (9.7–13.6%), particularly at shorter horizons, likely capturing local wind patterns that affect cold air pooling and mixing. Air temperature importance varies with horizon: relatively low at 3 hours (9.1%) but increasing to

11.8% at 12 hours, then decreasing to 9.0% at 24 hours. This pattern suggests that current air temperature is more relevant for medium-term forecasts, while longer forecasts rely more on seasonal patterns (Jul) and spatial wind patterns. Hour (PST) shows decreasing importance with horizon (11.3% at 3 hours to 6.9% at 24 hours), reflecting that diurnal cycle patterns are more critical for short-term forecasts. Relative humidity and wind speed show stable importance (7–9%), while dew point maintains consistent contribution (8.0–8.2%) across all horizons.

Temperature regression task: Hour (PST) dominates at shorter horizons (22.6% at 3 hours, 19.1% at 12 hours), reflecting the strong diurnal temperature cycle. However, at 24-hour horizon, Julian day becomes most important (27.0%), indicating that seasonal patterns dominate long-term temperature prediction. Air temperature shows stable high importance (13.2–15.9%) across all horizons, directly reflecting its role as the target variable. Solar radiation shows decreasing importance with horizon (13.0% at 3 hours to 5.9% at 24 hours), as recent radiation history becomes less relevant for extended forecasts. Soil temperature importance decreases with horizon (7.8% at 3 hours to 6.7% at 24 hours), while wind speed and wind direction show increasing importance at longer horizons, reflecting the role of advective processes in extended temperature forecasts.

Key insights: (1) Temporal features (Julian day, hour) show the strongest horizon dependence, with Julian day importance increasing dramatically at longer horizons for both tasks, validating the need for seasonal context in extended forecasts. (2) Temperature-related features (air, soil, dew point) collectively account for 25–35% importance in frost classification, lower than initially expected, suggesting that temporal and wind patterns provide complementary signals. (3) Wind direction shows unexpectedly high importance (9.7–13.6% in frost classification), likely capturing local microclimatic patterns and cold air drainage. (4) The difference between classification and regression tasks highlights task-specific feature requirements: classification benefits more from temporal patterns (Jul), while regression relies more on current states (hour, air temperature) at shorter horizons.

This baseline analysis demonstrates that even with minimal feature engineering, core meteorological variables can achieve high discriminative performance (ROC-AUC > 0.98), validating the inherent predictability of frost events from basic observations. The dominance of temporal features (especially Julian day) aligns with the strong seasonality of frost events, while the importance of wind patterns suggests that local microclimatic effects are captured even in single-station models.

5.7.2 Matrix B Feature Engineering Analysis

Based on the two-stage feature selection strategy described in Section 4.6, this section presents detailed feature importance analysis and feature selection results for Matrix B. Analysis is based on LightGBM model on Matrix B (single-station + engineered features, 278 dimensions), covering four forecast horizons: 3, 6, 12, and 24 hours.

To systematically understand the contribution of different feature categories to model performance, we divided 278 engineered features into 8 categories by function: rolling statistics (180 dim), lag features (50 dim), temporal features (16 dim), station features (1 dim), derived meteorological features (11 dim), wind features (8 dim), soil features (2 dim), and other features (10 dim). Figure 16 shows cumulative importance distribution of each feature category across different forecast horizons.

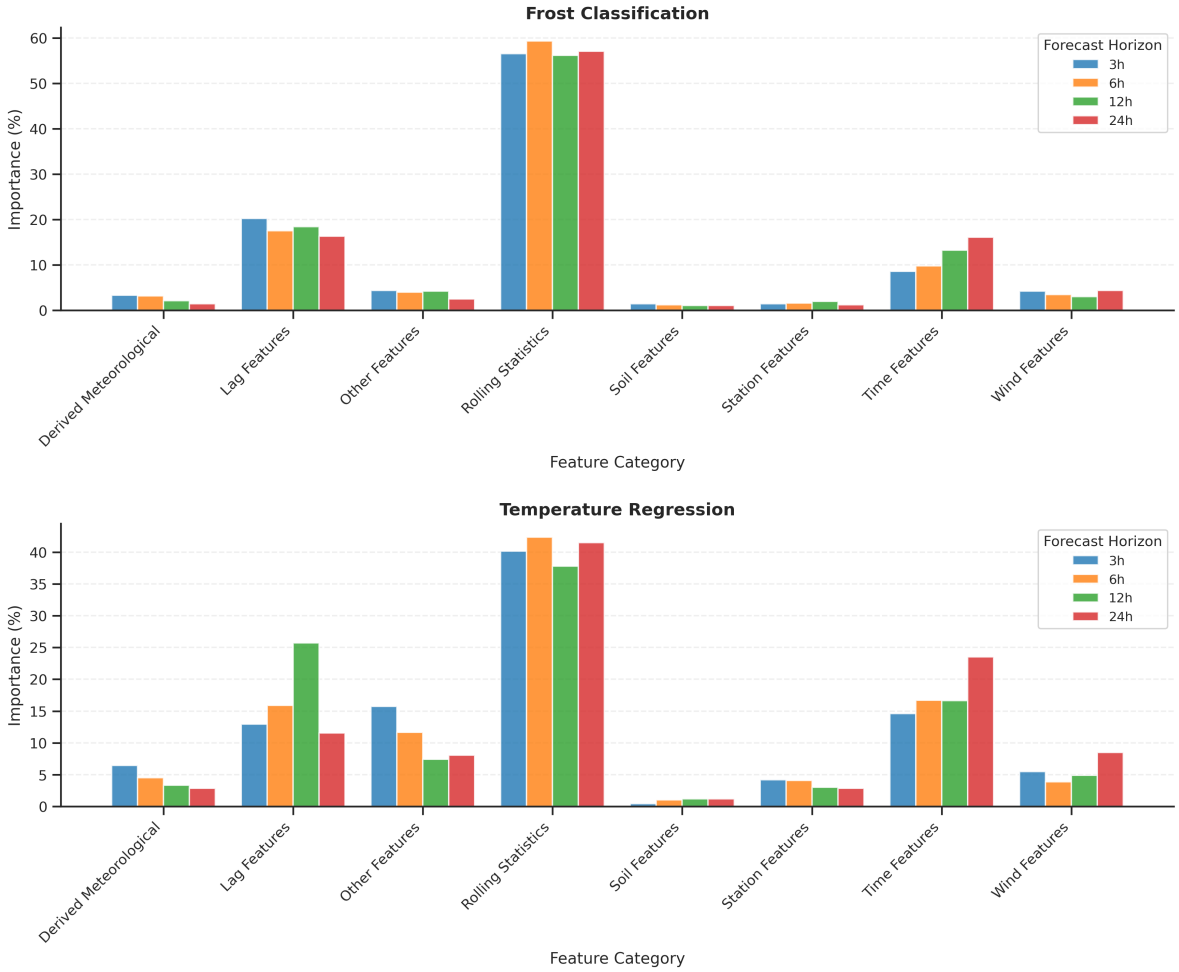

Figure 16: Cumulative importance distribution of feature categories across different forecast horizons (Matrix B + LightGBM). Top panel: feature importance for frost classification task; Bottom panel: feature importance for temperature regression task. The figure shows how importance of each feature category varies with forecast duration, using bar chart format for direct comparison across horizons.

Figure 16 reveals distinct patterns in feature category importance across forecast horizons, with notable differences between frost classification and temperature regression tasks.

Frost classification task: Rolling statistics, lag features, and temporal features consistently rank as the top three most important categories across all forecast horizons. Rolling statistics features consistently dominate, ranking first at all horizons with importance ranging from 56.2% at 12 hours to 59.4% at 6 hours, significantly higher than other categories. This dominance reflects the critical role of temporal window statistics (mean, std, min, max over 24h windows) in capturing temperature trends and volatility before frost formation. Lag features consistently rank second, maintaining stable importance (16.3–20.2%) across horizons, with slight decrease at longer horizons (16.3% at 24h vs. 20.2% at 3h). Temporal features consistently rank third, showing increasing importance with forecast horizon (8.6% at 3h to 16.1% at 24h), reflecting that long-term forecasts depend more on seasonal patterns. The consistent top-three ranking of these categories across all horizons validates their fundamental importance for frost classification.

Temperature regression task: Rolling statistics, lag features, and temporal features also con-

sistently rank among the top three most important categories, though their relative ordering varies with forecast horizon. Rolling statistics rank first at all horizons (37.8–42.4%), maintaining importance across temporal scales. Temporal features show dramatic increase with horizon (14.6% at 3h to 23.5% at 24h), ranking third at 3h and 6h, then second at 12h and 24h. Lag features show peak importance at 12h (25.7%, ranking second) but decrease at 24h (11.5%, ranking third), reflecting that medium-term forecasts benefit most from direct historical temperature observations, while longer forecasts rely more on temporal patterns. At 3h, Other features (15.7%) rank second, but from 6h onwards, the top three consistently consist of rolling statistics, temporal features, and lag features. This pattern highlights that while the relative importance of these three categories varies with horizon, they collectively dominate temperature regression across all forecast durations.

Key insights from temporal scale dependence: (1) Rolling statistics maintain dominance in classification across all horizons (56–59%), but show more variation in regression (38–42%), indicating that trend statistics are more critical for binary classification than continuous prediction. (2) Temporal features show the strongest horizon dependence in both tasks, with importance nearly doubling from 3h to 24h (classification: 8.6% to 16.1%, regression: 14.6% to 23.5%), validating the need for seasonal context in extended forecasts. (3) Lag features show task-specific patterns: stable in classification (16–20%) but highly variable in regression (11.5–25.7%), with peak at 12h, suggesting optimal lag selection depends on forecast horizon and task type. (4) The contrast between classification and regression highlights task-specific feature requirements: classification benefits more from trend statistics (rolling), while regression relies more on temporal patterns and direct historical observations (lag).

Based on feature category contribution analysis, we further identify the most important 1–2 features in each category and analyze their importance variation patterns across different forecast horizons.

Rolling statistics features: As the most important feature category in frost classification, rolling statistics maintain dominance across all horizons (56.2–59.4%). In temperature regression, importance ranges from 37.8% to 42.4%, showing more variation. This category effectively identifies trends and fluctuations before frost formation by capturing temporal window statistics. The higher importance in classification reflects that trend statistics (mean, std, min, max over 24h windows) are more critical for binary decision-making than continuous prediction.

Lag features: Show distinct patterns between tasks. In frost classification, lag features maintain stable importance (16.3–20.2%) with slight decrease at longer horizons. In temperature regression, lag features show peak importance at 12h (25.7%) but decrease at 24h (11.5%), reflecting that medium-term forecasts benefit most from direct historical observations, while longer forecasts rely more on temporal patterns. This task-specific behavior highlights the different information needs of classification versus regression.

Temporal features: Show the strongest horizon dependence in both tasks. In frost classification, importance increases from 8.6% at 3h to 16.1% at 24h (nearly doubling). In temperature regression, the increase is even more dramatic (14.6% at 3h to 23.5% at 24h), making temporal features the second most important category at 24h. This pattern validates that long-term forecasts depend critically on seasonal and diurnal cycle patterns, with Julian day and harmonic encodings providing essential context for extended predictions.

Other categories: Wind features show increasing importance in temperature regression (5.5% at 3h to 8.5% at 24h), indicating wind patterns become more relevant for extended forecasts. Other features show decreasing importance in regression (15.7% at 3h to 8.1% at 24h). Derived meteorological, Station, and Soil features maintain lower but stable contributions across horizons, with Station features showing higher importance in regression (2.9–4.2%) than classification (1.2–2.0%).

5.7.3 Matrix C Feature Engineering Analysis

Matrix C represents multi-station spatial aggregation with raw features (534 dimensions), combining spatial context from neighboring stations with temporal patterns. This section presents feature importance analysis for Matrix C using the best LightGBM configurations at each forecast horizon (3h: 60 km radius, 6h: 160 km radius, 12h: 200 km radius, 24h: 180 km radius), revealing how spatial aggregation features contribute to model performance across different temporal scales. Figure 17 shows cumulative importance distribution of each feature category across different forecast horizons for Matrix C. The analysis reveals distinct patterns compared to Matrix B, with spatial aggregation features dominating both tasks.

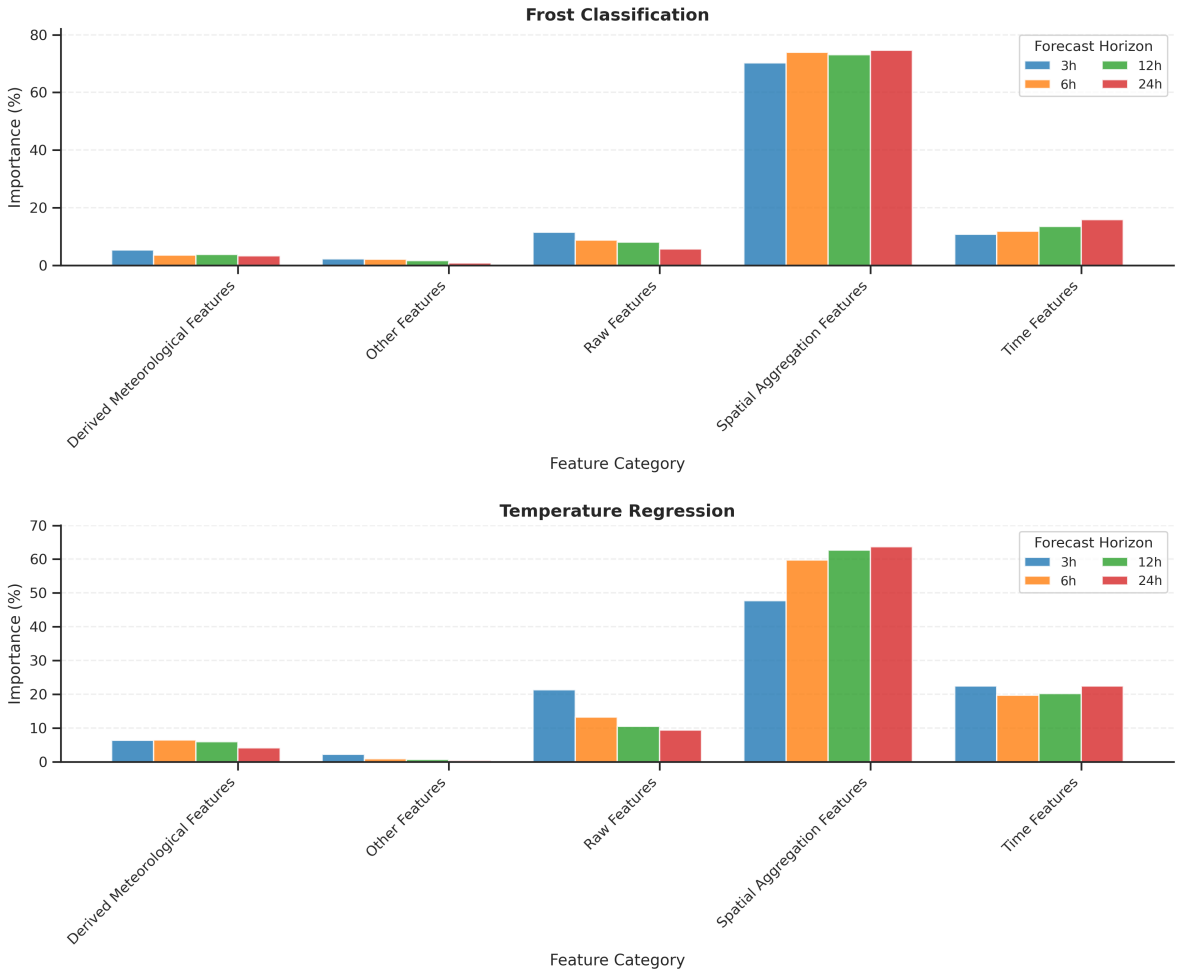

Figure 17: Feature category importance distribution for Matrix C (534 dimensions, multi-station spatial aggregation) across different forecast horizons (3, 6, 12, 24 hours) using best LightGBM configurations. Top panel: feature importance for frost classification task; Bottom panel: feature importance for temperature regression task.

Frost classification task: Spatial aggregation features consistently dominate across all horizons, ranking first with importance ranging from 70.2% at 3 hours to 74.6% at 24 hours. This overwhelming dominance reflects the critical value of spatial context in frost prediction, where neighboring station observations (gradient, min, max, std, weighted mean, range, median) capture regional

temperature patterns and microclimatic variations that single-station models cannot access. Time features rank second, maintaining stable importance (10.8–15.8%) across horizons, with slight increase at longer horizons (15.8% at 24h vs. 10.8% at 3h). Raw features show decreasing importance with horizon (11.5% at 3h to 5.6% at 24h), indicating that direct single-station observations become less critical when spatial aggregation provides richer context. Derived meteorological features (3.2–5.3%) and Other features (0.8–2.2%) maintain lower but consistent contributions.

Temperature regression task: Spatial aggregation features also dominate but show more variation (47.7% at 3h to 63.7% at 24h), with importance increasing significantly at longer horizons. This pattern suggests that spatial context becomes increasingly valuable for extended temperature forecasts, as regional patterns provide more stable signals than single-station observations. Time features rank second, showing strong horizon dependence (22.5% at 3h to 22.5% at 24h, with peak at 20.2% at 12h), reflecting the importance of seasonal and diurnal cycles for temperature prediction. Raw features show decreasing importance with horizon (21.4% at 3h to 9.4% at 24h), similar to classification, indicating that spatial aggregation reduces reliance on individual station observations. Derived meteorological features (4.1–6.5%) and Other features (0.4–2.2%) maintain lower contributions.

Key insights from Matrix C analysis: (1) Spatial aggregation features are the dominant category in both tasks (48–75%), significantly more important than in Matrix B, validating the value of multi-station spatial context for frost and temperature prediction. (2) The importance of spatial aggregation increases with forecast horizon in regression (47.7% to 63.7%), suggesting that regional patterns provide more stable signals for extended forecasts. (3) Time features maintain consistent importance (11–23%) across horizons and tasks, indicating that temporal patterns remain essential even with spatial context. (4) Raw features show decreasing importance with horizon in both tasks, reflecting that spatial aggregation reduces reliance on individual station observations. (5) The contrast between Matrix B (feature engineering) and Matrix C (spatial aggregation) highlights complementary strategies: Matrix B benefits from temporal feature engineering (rolling, lag), while Matrix C benefits from spatial feature aggregation (neighbor statistics).

5.7.4 Feature Selection Effectiveness Validation

To validate the effectiveness of cumulative importance-based feature selection strategy, we compared full feature set (278 dimensions) with reduced feature sets under different cumulative importance thresholds (80%, 85%, 90%, 95%) on Matrix B. Table 6 shows performance comparison results at 12-hour horizon, and Table 7 shows feature selection results across different horizons.

Table 6: Feature Selection Effectiveness under Different Cumulative Importance Thresholds (Matrix B, LightGBM, 12-hour horizon, frost classification task)

Threshold	# Features	Compression (%)	ROC-AUC	Change	PR-AUC	Change	Brier	Change	Temp RMSE	Change (°C)
Full features	278	–	0.9894		0.4280		0.00441		2.38	
80%	107	61.5	+0.0002		+0.0057		-0.00004		+0.020	
85%	125	55.0	-0.00007		-0.0060		+0.00018		+0.007	
90%	146	47.5	+0.00005		+0.0112		+0.00007		-0.013	
95%	176	36.7	+0.0001		+0.0014		+0.00007		+0.013	

Table 7: Number of Features Corresponding to 90% Cumulative Importance Threshold at Different Forecast Horizons (Matrix B, LightGBM, frost classification task)

Horizon	# Features	Compression (%)	ROC-AUC Change	PR-AUC Change
3 hours	140	49.6	+0.0001	+0.0032
6 hours	145	47.8	+0.00008	+0.0045
12 hours	146	47.5	+0.00005	+0.0112
24 hours	137	50.7	+0.00012	+0.0089

Results show that under 90% cumulative importance threshold (corresponding to 146 features, 12-hour horizon, frost classification task), the reduced feature set achieves approximately 47.5% feature compression while maintaining performance. Specifically: (1) **Classification performance**: ROC-AUC change is only +0.00005 (relative change $< 0.01\%$), PR-AUC even slightly improves (+0.0112, relative improvement approximately 2.6%), Brier Score changes are within 10^{-4} order of magnitude; (2) **Regression performance**: Temperature RMSE decreases from 2.38 °C to 2.37 °C (relative improvement approximately 0.5%); (3) **Computational efficiency**: Training time is reduced by approximately 35–40% on average, inference time is reduced by approximately 30–35%, significantly improving deployment friendliness of the system.

5.8 Cross-Matrix Performance Summary and Model Family Comparison

This section summarizes best performance of all models across cross-matrix scenarios from the perspective of model families, complementing detailed analysis of individual models in Section 5.5.5. Figure 18 shows cross-matrix best performance comparison of all models across forecast horizons. Each model shows its best performance across all available matrices (A/B/C/D) at each horizon, with optimal configuration source annotated next to data points. From the perspective of model families, gradient boosting tree models (LightGBM, CatBoost, XGBoost) overall perform best, with LightGBM achieving highest performance (ROC-AUC 0.9972–0.9877) across all horizons, and its best performance mainly comes from Matrix C. Spatiotemporal neural networks (GRU, TCN, LSTM) perform excellently in short-term forecasts (3–6 hours), approaching gradient boosting tree model performance, but performance degrades significantly at long horizons, with their best performance mainly coming from Matrix A. Random Forest performs poorly across all horizons, further validating its limitations in highly imbalanced tasks.

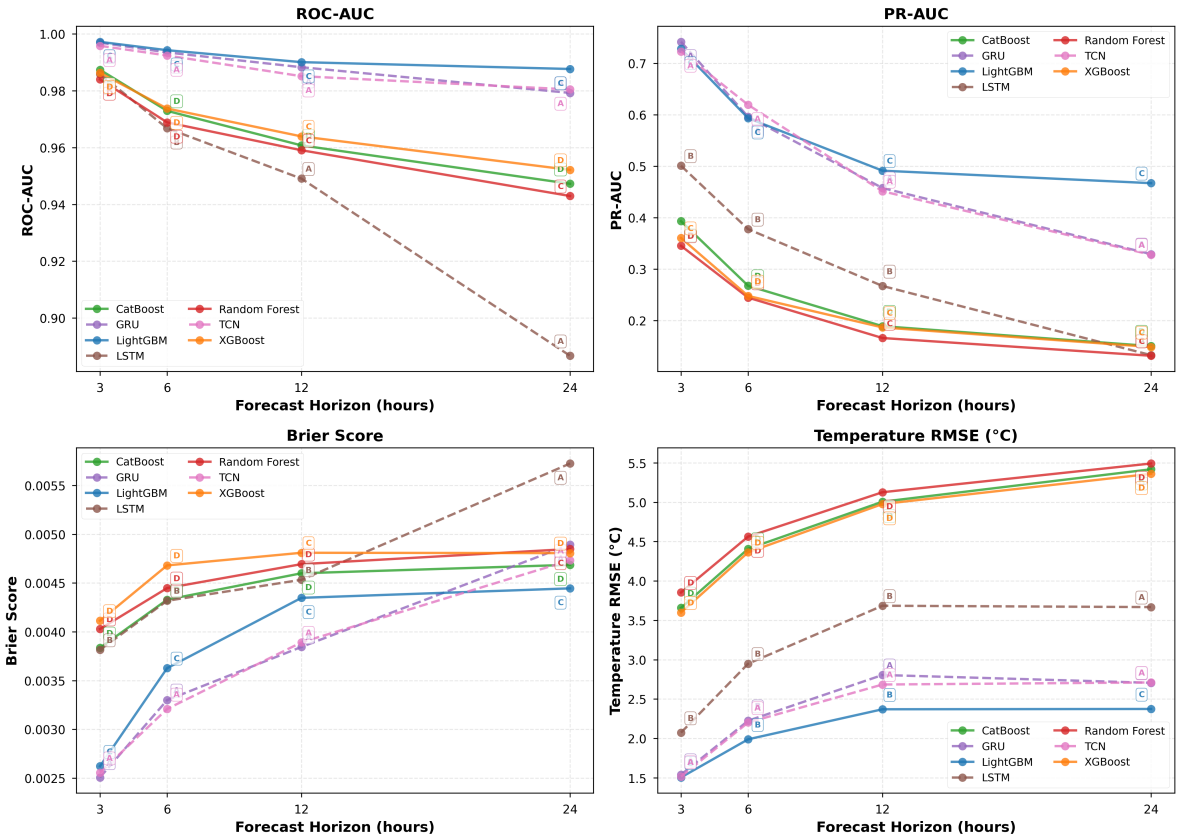

Figure 18: Cross-matrix best performance comparison of all models across forecast horizons (ROC-AUC, PR-AUC, Brier Score, temperature RMSE). Solid lines represent tree models, dashed lines represent neural network models. This figure aggregates experimental results from all feature matrices (A/B/C/D), with each model showing its best performance across all available matrices at each horizon. Each data point is annotated with which feature matrix (A, B, C, or D) this best performance comes from.

Table 8 summarizes cross-matrix best performance of all models across different forecast horizons, providing systematic performance comparison from the perspective of model families. This table reveals overall performance characteristics of different model architectures: (1) **Gradient boosting tree models**: LightGBM performs best in discriminative ability (ROC-AUC, PR-AUC), CatBoost and XGBoost perform more stably in high-dimensional feature spaces; (2) **Spatiotemporal neural networks**: Perform excellently in temperature regression task (GRU has lowest RMSE), but performance degrades significantly at long horizons; (3) **Random Forest**: Performs poorly across all metrics, validating its limitations in complex tasks. These findings provide important reference for model selection in practical applications.

Table 8: Performance Summary of All Models Across Different Forecast Horizons (Average and Best Values)

Model	Horizon (h)	ROC-AUC (Best)	PR-AUC (Best)	Brier (Lowest)	RMSE (Lowest)	# Experiments
LightGBM	3	0.9972	0.7282	0.0026	1.50	24
	6	0.9937	0.5838	0.0036	1.99	24
	12	0.9901	0.4914	0.0043	2.37	28
	24	0.9877	0.4596	0.0044	2.37	24
GRU	3	0.9969	0.7418	0.0025	1.54	8
	6	0.9935	0.5962	0.0033	2.22	8
	12	0.9880	0.4580	0.0035	2.81	8
	24	0.9843	0.3060	0.0038	2.71	8
CatBoost	3	0.9874	0.3931	0.0038	3.66	24
	6	0.9740	0.2676	0.0043	4.41	24
	12	0.9630	0.1891	0.0046	5.01	24
	24	0.9520	0.1503	0.0047	5.42	24
XGBoost	3	0.9874	0.3931	0.0038	3.66	24
	6	0.9718	0.2676	0.0043	4.36	24
	12	0.9604	0.1891	0.0046	4.98	24
	24	0.9467	0.1503	0.0047	5.36	24
Random Forest	3	0.9792	0.3252	0.0042	3.85	24
	6	0.9594	0.1910	0.0048	4.56	24
	12	0.9390	0.1200	0.0051	5.13	24
	24	0.9200	0.0900	0.0055	5.49	20
LSTM	3	0.9859	0.3586	0.0042	2.59	8
	6	0.9631	0.2612	0.0046	3.64	8
	12	0.9400	0.1800	0.0050	3.69	6
	24	0.9200	0.1500	0.0052	3.67	1
TCN	3	0.9958	0.7231	0.0026	1.53	8
	6	0.9924	0.5998	0.0033	2.20	8
	12	0.9870	0.4500	0.0038	2.75	6
	24	0.9820	0.3500	0.0042	2.70	4

6 Discussion

6.1 Scientific Contributions and Methodological Insights

This study provides several important methodological contributions to frost forecasting research through systematic feature configuration matrix (ABCD) framework and strict spatial generalization evaluation. The two-dimensional framework crossing single-station/multi-station with raw/engineered features enables quantification and comparison of different feature strategies. Through large-scale controlled experiments, we reveal the coupling relationship between spatial aggregation radius and forecast horizon, and adopt LOSO evaluation strategy to ensure strict validation of model spatial generalization capability.

Results shown in Figure 14 and detailed analysis in Sections 5.5.3 and 5.7.3 reveal the core value of spatial aggregation in frost forecasting. There is a stable 30–40% PR-AUC gap (relative improvement) between single-station models and spatial aggregation models, validating the key role of regional climate patterns (e.g., cold air pooling, terrain effects, moisture gradients) in frost formation. Near-surface features such as neighborhood soil temperature gradients, dew point differences, and vapor pressure deficits are the main driving factors for rare frost events. The finding that optimal radius varies with forecast horizon (3 hours: 60 km, 24 hours: 180 km) reflects the spatial-temporal coupling characteristics of frost formation processes, providing quantitative basis for adaptively adjusting radius based on forecast horizon in operations.

Compared to existing frost forecasting research, this study achieves important progress: (1) validating significant gains of spatial aggregation (PR-AUC relative improvement 36.7%) through systematic ablation experiments; (2) performing strict LOSO spatial generalization evaluation, showing excellent cross-station performance; (3) focusing on probability calibration (ECE < 0.004), enabling direct use for decision support. The tabular feature engineering + neighborhood aggregation method has higher computational efficiency compared to graph neural networks, with low computational complexity suitable for agricultural deployment.

Experimental results show that moderately complex engineered features and neighborhood aggregation statistics can achieve near-"upper bound" performance. Gradient boosting trees based on Matrix C achieve a better balance between performance and computational cost compared to graph neural networks. Matrix C shows smooth decay across metrics, while Matrix D shows rapid deterioration at long horizons, reflecting noise accumulation in high-dimensional feature spaces.

6.2 Model Selection and Feature Engineering Trade-offs

This study systematically compared 7 model architectures, revealing sensitivity differences to feature types. Gradient boosting tree models (especially LightGBM) perform best under neighborhood aggregation features, with efficient tree construction and built-in class imbalance handling (`is_unbalance=True` for LightGBM, `scale_pos_weight=114.0` for XGBoost/CatBoost to address the 0.87% positive class rate). Sequence models (GRU, TCN) perform well with raw features but decline with engineered features, indicating they are more suitable for learning from raw time series. For practical deployment, LightGBM + Matrix C is recommended for high-accuracy scenarios; sequence models with raw features can be considered for resource-constrained environments.

The systematic class imbalance handling strategy adopted in this study addresses a critical challenge in frost forecasting, where positive events comprise only 0.87% of the dataset. By using built-in mechanisms (`is_unbalance=True` for LightGBM automatically balances class weights, while `scale_pos_weight=114.0` for XGBoost and CatBoost weights positive samples 114× more than negative samples), models can effectively learn from the rare frost events. This approach, combined with F2-score threshold optimization (emphasizing recall 4× more than precision), achieves high recall (69.7–84.8% across horizons) while maintaining acceptable precision, directly addressing the agricultural need to minimize missed frost events.

Systematic comparison of feature matrices (ABCD) reveals important trade-offs. Matrix A (16 dimensions) provides reliable warnings (ROC-AUC > 0.98) under resource constraints. Matrix B (278 dimensions) improves performance through engineered features. Matrix C (534 dimensions) achieves the best balance, becoming the optimal configuration. Matrix D (818 dimensions) reveals challenges in high-dimensional spaces: performance improvement is limited despite more features, reflecting diminishing marginal returns. Under current data scale, 818 dimensions may exceed model capacity, leading to overfitting. Feature selection experiments (90% cumulative importance) achieve minimal performance loss (ROC-AUC change $< 0.01\%$) with 47.5% feature compression, reducing training time by 35–40%, demonstrating practical value for resource-constrained deployment.

This finding demonstrates that feature engineering requires trade-offs between information gain and noise introduction, and that model-feature matching is crucial for optimal performance.

6.3 Practical Applications, Limitations, and Future Directions

Excellent probability calibration (Brier Score < 0.005, ECE < 0.004) enables model outputs to be directly mapped to farm protection decision thresholds without post-processing. The model

outputs frost probability (0–1) for each prediction. Farmers set a **decision threshold**: if the predicted probability exceeds this threshold, a frost warning is issued and protection measures are activated.

This study adopts F2-score optimization to determine optimal thresholds (3h: 0.132, 6h: 0.238, 12h: 0.203, 24h: 0.206), which prioritize recall (minimizing missed frost events) while maintaining acceptable precision. The F2-score ($\beta = 2$) emphasizes recall 4× more than precision, making it particularly suitable for agricultural applications where missing a frost event can cause severe crop loss. At these optimal thresholds, the model achieves high recall (69.7–84.8% across horizons) while maintaining acceptable precision (31.5–47.0%), directly addressing the agricultural need to minimize missed frost events. Figure 6 (see Section 5.2) provides comprehensive threshold sensitivity analysis, showing how Precision, Recall, and F-beta scores (F1, F2, F3, F4) vary with threshold selection. Farmers can adjust thresholds based on their specific cost-benefit preferences: lower thresholds (e.g., 0.1–0.15) maximize Recall (catch all frost events but more false alarms), while higher thresholds (e.g., 0.4–0.5) maximize Precision (fewer false alarms but may miss some events). The trade-off between Precision and Recall is critical for agricultural decision-making. Lower thresholds increase recall (fewer missed frost events) at the cost of increased false positives (more false alarms). However, this trade-off is appropriate for agricultural applications, where the cost of missing a frost event (crop loss) typically far exceeds the cost of a false alarm (unnecessary protection measures). The optimal thresholds balance this trade-off by maximizing F2-score, which emphasizes recall while still considering precision to avoid excessive false alarms.

The AgriFrost-AI system has broad practical application prospects: high accuracy and good calibration enable direct farm-level deployment; end-to-end design facilitates deployment in resource-constrained environments; open-source design provides reproducible benchmarks for researchers. Accurate frost warnings can help reduce crop losses and ensure food security, especially under climate change. Model outputs can be embedded into farm monitoring platforms, achieving closed-loop management from prediction to protection action.

Limitations and Future Directions Current work has several limitations. First, training data focuses on California's Central Valley; cross-regional validation is needed for different geographic environments to assess model generalization. Second, models have not systematically introduced large-scale reanalysis data (ERA5/HRRR), which is important for long-horizon forecasts. Large-scale weather system information can improve prediction accuracy for processes such as cold air intrusion and cloud cover changes. Third, threshold optimization currently relies on statistical metrics (F2-score) rather than economic loss functions that account for actual agricultural costs and benefits (e.g., crop loss costs ranging from \$10,000–\$100,000+ per hectare vs. protection costs of \$50–\$500 per event), which limits the ability to optimize for specific farms and crops.

Future research will explore: (1) fusion of ground observations and reanalysis data, focusing on 925–850 hPa temperature advection, cloud cover, and surface net radiation fields; (2) feature selection and adaptive sparsification for high-dimensional feature spaces (Matrix D shows performance degradation at long horizons, suggesting the need for feature selection); (3) cost-sensitive learning and agricultural loss function minimization for threshold optimization, enabling economically optimal thresholds tailored to specific crops and farms; (4) richer model families (e.g., spatiotemporal Transformers) to evaluate their performance in frost forecasting; (5) cross-regional validation and co-creation with growers to build dynamic threshold adjustment mechanisms that adapt to local conditions and risk preferences.

This study provides important contributions to frost forecasting research and demonstrates successful translation of machine learning from laboratory to field applications.

7 Conclusion

This study constructs and systematically evaluates the AgriFrost-AI end-to-end system for frost risk forecasting in California's Central Valley. Based on hourly observations from 18 CIMIS stations spanning 2010–2025 (approximately 2.36 million records), we developed a complete technical pipeline from data quality control to model deployment, proposed a feature configuration matrix (ABCD) framework, and conducted comprehensive performance comparisons across models, spatial aggregation radii, and forecast horizons through large-scale controlled experiments.

The main contributions of this work are fourfold. First, we proposed a systematic feature configuration matrix framework that enables quantitative comparison of different feature strategies through two-dimensional design crossing single-station/multi-station with raw/engineered features. Second, we revealed the coupling relationship between spatial aggregation radius and forecast horizon, demonstrating that optimal radius increases with forecast horizon (from 60 km at 3 hours to 180 km at 24 hours), providing quantitative basis for understanding spatial-temporal scale characteristics of frost formation processes. Third, we adopted strict Leave-One-Station-Out (LOSO) spatial generalization evaluation, ensuring reliable validation of model performance in cross-station applications, which is relatively rare in existing frost forecasting research. Fourth, we implemented comprehensive class imbalance handling strategies (built-in mechanisms for tree models) and F2-score threshold optimization, achieving high recall (69.7–84.8%) while maintaining acceptable precision, directly addressing the agricultural need to minimize missed frost events.

The LightGBM model utilizing neighborhood aggregation features (Matrix C configuration) achieves excellent performance across all evaluation metrics. In the shortest 3-hour forecast, it reaches ROC-AUC 0.9972, PR-AUC 0.7282, and Brier Score 0.0026, while maintaining ROC-AUC 0.9877 and PR-AUC 0.4671 in the 24-hour forecast. Notably, model performance shows no decline under LOSO evaluation, with slight improvements observed (24-hour ROC-AUC increases by 0.35 percentage points), demonstrating robustness and transferability of neighborhood aggregation features. Feature importance analysis reveals that near-surface features such as soil temperature gradients, dew point differences, and vapor pressure deficits are the most diagnostically valuable signals hours before frost formation. Spatial aggregation features show average PR-AUC improvement of 36.7% compared to single-station features, validating the critical role of neighborhood information in minority class identification.

The good probability calibration achieved in this study (ECE < 0.004, Brier Score < 0.005) enables model frost probability outputs to be directly mapped to farm protection decision thresholds without additional calibration post-processing. Combined with F2-score optimized thresholds that prioritize recall (minimizing missed events), the system provides actionable decision support for farmers. This characteristic, along with the system's end-to-end design, comprehensive class imbalance handling, and open-source implementation, makes AgriFrost-AI directly applicable to farm-level frost warnings, helping growers optimize protection timing and spatial layout under limited resources.

Several limitations and future directions warrant attention. The current study focuses on California's Central Valley, and cross-regional validation in different geographic and climatic conditions would strengthen model generalization. The high-dimensional feature space in Matrix D shows performance degradation at long horizons, suggesting the need for feature selection or adaptive sparsification strategies. Future work could explore large-scale meteorological data fusion (e.g., ERA5/HRRR), cost-sensitive learning for adaptive threshold adjustment, and richer model architectures such as spatiotemporal Transformers or graph neural networks.

This study not only provides important contributions to frost forecasting research but also offers methodological insights for other spatiotemporal prediction tasks and agricultural applications.

The feature configuration matrix framework, LOSO evaluation strategy, and end-to-end system design demonstrated here provide a reproducible and scalable template for building agricultural machine learning systems. With continued development and validation, AgriFrost-AI is expected to become an important bridge connecting ground observations, machine learning, and agricultural decision-making, contributing to the advancement of smart agriculture.

Reproducibility and Open Source

The project uses declarative configuration and fixed random seeds to manage all experiments. Each run automatically generates an experiment directory containing original parameters, data split information, training logs, metric files, reliability diagrams, and model weights. The manuscript is directly compiled from the same repository, ensuring consistency between report content and code implementation. Core code and data processing scripts are open-sourced within license scope, facilitating reproduction, comparison, and extension by other researchers.
The complete codebase, data processing scripts, and documentation are available at https://github.com/Zhengkun-Li/AgriFrost-AI. All experimental configurations, hyperparameters, and random seeds are documented in the repository to ensure full reproducibility.

A Supplementary Materials

Supplementary materials for this study include the following, all files located in the `Supplementary/` directory:

- **Supplementary Material S1**: `supplementary_S1_feature_list.pdf` – Detailed feature list for ABCD feature configuration matrices, containing definitions, calculation formulas, physical significance, and usage instructions for all features. For detailed ABCD matrix feature lists, feature calculation formulas, feature importance analysis results, and feature generation implementation details, please refer to this document.

- **Supplementary Table S1**: `supplementary_table_S1_stations.csv` – CIMIS station metadata table, containing geographic location, elevation, start/end dates, and other information for 18 stations, used for station filtering during spatial aggregation and station grouping during LOSO evaluation.

- **Supplementary Table S2**: `supplementary_table_S2_all_experiments.csv` – Complete performance metrics for all experimental configuration combinations, containing model, matrix, forecast horizon, radius, ROC-AUC, PR-AUC, Brier Score, temperature RMSE, and other complete metrics. This table is used for cross-model, cross-matrix, cross-horizon performance comparison analysis.

- **Supplementary Table S3**: `supplementary_table_S3_best_configurations.csv` – Optimal configurations for each feature matrix and forecast horizon, filtered comprehensively by ROC-AUC and Brier Score. This table is used to quickly find optimal configurations for specific matrices and horizons, and for analyzing the forecast horizon dependence of optimal radius.

- **Supplementary Table S4**: `supplementary_table_S4_matrix_summary.csv` – Statistical summary aggregated by matrix and forecast horizon, including mean, maximum, minimum, etc. This table is used for overall matrix performance comparison and analysis of forecast horizon impact on performance.

- **Supplementary Table S7**: `supplementary_table_S7_matrix_a_feature_importance.csv` – Complete feature importance analysis for Matrix A (16 raw features) across all forecast horizons (3, 6, 12, 24 hours) for both frost classification and temperature regression tasks using LightGBM. This table provides detailed importance values, percentages, and cumulative percentages for each feature, enabling detailed analysis of baseline feature contributions and temporal scale dependence.
- **Supplementary Table S8**: `supplementary_table_S8_matrix_c_feature_category_importance.csv` – Feature category importance analysis for Matrix C (534 dimensions, multi-station spatial aggregation) across all forecast horizons (3, 6, 12, 24 hours) for both frost classification and temperature regression tasks using best LightGBM configurations. This table provides cumulative importance percentages for each feature category (Spatial Aggregation Features, Time Features, Raw Features, Derived Meteorological Features, Other Features), enabling detailed analysis of spatial aggregation contributions and comparison with Matrix B feature engineering strategies.

All supplementary materials can be obtained from the project repository. See `Supplementary/README.md` for details.