

WEATHER-DRIVEN DEEP LEARNING FOR AIRPORT RUNWAY DIRECTION FORECASTING

Hyewook Kim[1], Yeonju Eun[1] & Daekeun Jeon[1]

[1]Air Traffic Management Research Team, Korea Aerospace Research Institute, Daejeon, Republic of Korea

Abstract

The runway operating direction at an airport shapes arrival and departure operations, including the available capacity and the routes aircraft fly on approach and departure. Forecasting this direction is therefore important for airport surface operations and traffic flow coordination. This study presents a weather-driven deep learning framework for hourly runway direction forecasting over a 24-hour horizon using standardized meteorological observations and numerical weather prediction forecasts. Because runway direction changes only a few times a day, a standard per-hour classification loss is dominated by abundant non-transition hours and provides limited training signals for transition events. To address this limitation, we propose a transition-sensitive objective that combines per-hour classification with auxiliary terms for transition timing and transition frequency. The framework is evaluated using five years of data from Incheon International Airport. The proposed model achieves 88.34% prediction accuracy and 0.85 Macro-F1 averaged across the 24-hour horizon. Ablation results show that the timing and frequency terms provide complementary benefits. The timing term improves the localization of predicted transitions by encouraging them to occur near true transition hours, while the frequency term reduces transition-count errors by penalizing excessive or missing transitions over the forecast horizon. Their combination yields the most balanced improvement across transition detection, timing accuracy, and transition-count consistency.

Keywords: Runway direction forecasting, Runway configuration, Time-series classification, Deep learning

1. Introduction

A runway configuration refers to the set of runways active at a given time, the direction in which they are operated, and the way arrival and departure traffic is assigned to them. It is a central element of airport operations: it governs runway utilization, the approach and departure routes aircraft fly, and the capacity available for sequencing and traffic-flow coordination. Accurate runway configuration forecasts are therefore important for strategic traffic management and operational planning.

A runway configuration can be separated into two components: the *operating direction* and the *runway operation mode*. The operating direction specifies the orientation in which active runways are used. Since a runway can normally be operated in either of two opposite directions, the selected direction is primarily governed by wind conditions: aircraft generally take off and land into the wind, and controllers may initiate a direction change when wind conditions cross operational thresholds. The runway operation mode then specifies how arrivals and departures are assigned to individual runways within the selected direction, such as arrival-only, departure-only, or mixed-use operation.

This distinction is important because the forecasting problem depends on how much discretion remains after the operating direction is selected. At airports where the operation mode is determined by published runway operating procedures, predicting the operating direction is the central problem in runway configuration forecasting: once the direction is known, the prescribed procedure can be applied to infer the runway configuration. At airports where the operation mode remains subject

to controller discretion, by contrast, a complete configuration forecast requires predicting both the operating direction and the operation mode.
This study focuses on Incheon International Airport (RKSI), where the runway operation mode is fixed by a published procedure that assigns specific runways to arrivals and departures within each operating direction [12]. Predicting the operating direction at RKSI therefore yields a complete runway configuration forecast. Accordingly, this paper addresses hourly forecasting of the binary northbound/southbound operating direction at RKSI over a 24-hour horizon, with "transition" denoting a direction change between consecutive forecast hours.

1.1 Prior Research

Prior research on runway configuration can be grouped into optimization-based and forecasting-based approaches. Optimization models compute configuration schedules that minimize delay or congestion costs subject to weather, capacity, and queueing constraints [1, 2]. Forecasting models, by contrast, aim to predict the runway configuration that controllers are likely to implement over a future planning horizon.
Early forecasting studies modeled configuration choice as a probabilistic behavioral decision, often using multinomial logit models that account for controller preference to retain the incumbent configuration [3, 4]. More recent studies have used machine learning models with meteorological and operational inputs, including supervised classifiers using Meteorological Aerodrome Reports (METAR), Localized Aviation Model Output Statistics Program (LAMP) forecasts, and Aviation System Performance Metrics (ASPM) features [5]; encoder–decoder models using configuration history, observed and forecast weather, and traffic demand [6]; convolutional neural networks over gridded numerical weather prediction fields [7]; and hybrid temporal–attention architectures [8]. Probabilistic forecasts over longer horizons have also been studied [9]. A related line of research directly predicts configuration transition timing using ensemble regression methods [10].
These studies demonstrate that runway configuration can be predicted from weather and operational data. However, most models are trained and evaluated primarily through average per-hour configuration accuracy. This objective is not fully aligned with the operational importance of rare direction transitions. A direction transition affects routing, sequencing, and capacity planning even when it occupies only a small fraction of the forecast horizon. This motivates a forecasting objective that explicitly accounts for both transition timing and transition frequency.

1.2 Research Gap and Contribution

A runway direction forecast should satisfy two requirements. First, it should accurately predict the direction at each forecast hour. Second, it should represent transition behavior realistically, including both the timing and the number of predicted transitions. Standard per-hour classification losses directly address the first requirement but do not explicitly control the second. Class-weighted losses or oversampling can increase the penalty on minority direction labels, but they still evaluate each hour independently and do not directly penalize misplaced or excessive transitions. An operationally useful forecaster should therefore retain high per-hour direction accuracy while also predicting when transitions occur and how often they occur.
This study addresses this gap through a transition-sensitive learning framework for 24-hour runway direction forecasting. The main contributions are as follows.

1. **Transition-sensitive training objective.** The proposed objective augments a per-hour classification loss with two auxiliary terms: one for transition timing and one for transition frequency. The timing term encourages predicted transitions to occur near true transition hours, while the frequency term penalizes unrealistic transition counts over the forecast horizon. Ablation experiments verify the complementary roles of these two terms.

2. **Weather-driven input design.** The proposed predictor forecasts the operating direction using only standardized observations from the Automated Meteorological Observation System (AMOS) and numerical weather prediction forecasts from the Local Data Assimilation and Prediction System (LDAPS). This setting is operationally relevant because wind drives runway

direction, and the required 24-hour forecasts are issued routinely by national meteorological agencies in standardized form, so the predictor runs on readily available data with little preparation. Integrating additional drivers such as arrival and departure traffic demand is left to future work.

The framework is implemented as an inverted Transformer with horizon-aware pooling, an architecture chosen to capture cross-feature interactions between observed and forecast meteorological variables. It is evaluated at RKSI using five years of operational and meteorological data.

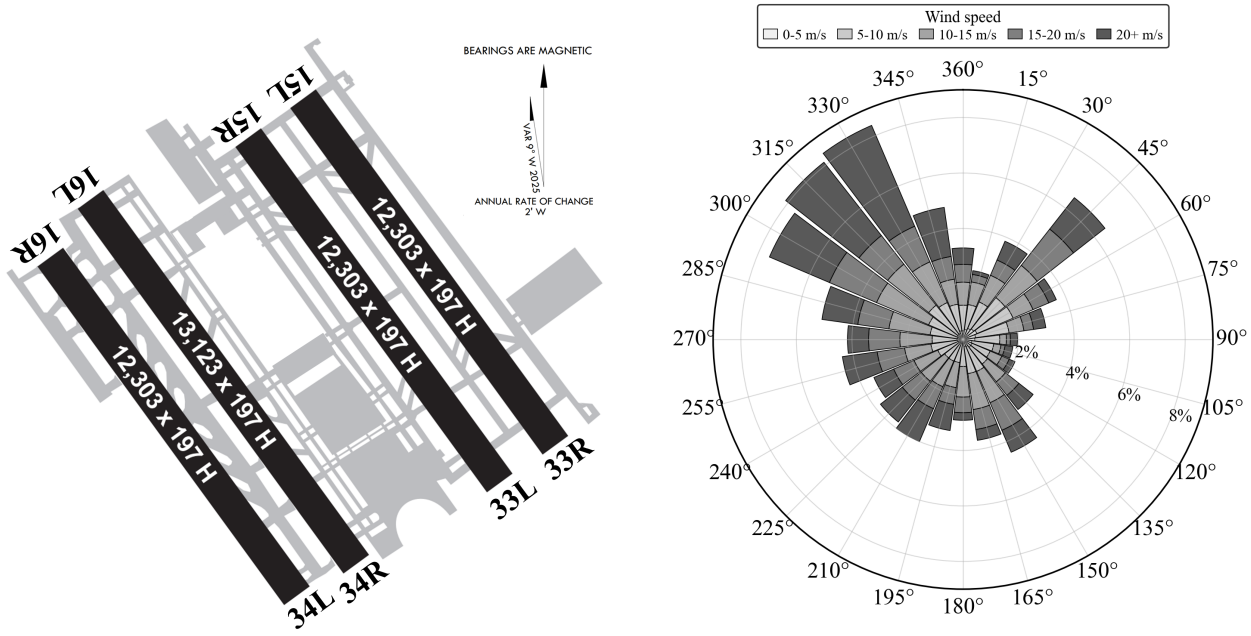
2. Case Study Airport: Incheon International Airport

RKSI operates four parallel runways oriented approximately north–south (Fig. 1a). In northbound operation the runways are designated 33L/R and 34L/R; in southbound operation the same four physical strips are designated 15R/L and 16R/L. The operation mode within each direction is fixed by published procedure [12]: in northbound operation, RWY 33L and 34R serve departures while RWY 33R and 34L serve arrivals; in southbound operation, RWY 15R and 16L serve departures while RWY 15L and 16R serve arrivals.

The direction itself is selected primarily by wind. The Preferential Runway System (PRS) recommends northbound operations up to 8 kt of tailwind and southbound operations beyond that [11], formalizing the wind-driven nature of the direction decision and providing a practical threshold at which controllers may initiate a direction change. Because prevailing winds at RKSI favor northbound operations (Fig. 1b), historical runway direction usage is highly asymmetric.

Operational records from 2020 show that northbound operations account for approximately 74% of all operating hours. The average persistence is 32.4 ± 29.5 h for the northbound direction and 11.1 ± 8.5 h for the southbound direction. Direction transitions are infrequent, averaging 0.85 ± 1.02 per day; 51.4% of days have no transition, 19.9% have exactly one, and 28.7% have two or more.

These characteristics—a strong directional imbalance, long persistence, and rare changes—make RKSI a useful testbed for evaluating transition-sensitive runway direction forecasting. A model can obtain high per-hour accuracy by exploiting directional persistence, but operational value depends on whether it can also detect rare transitions.



(a) RKSI aerodrome chart [13].

(b) Wind rose from 2020 RKSI observations.

Figure 1 – Incheon International Airport (RKSI) runway layout and historical wind characteristics.

3. Problem Formulation

We formulate runway direction forecasting as a binary time-series prediction problem. Let $y_{t+h} \in \{0, 1\}$ denote the runway operating direction at hour $t + h$, where 1 indicates northbound operation and 0

indicates southbound operation. The objective is to predict the sequence $y_{t+1:t+H}$ over the next $H = 24$ hours.

At decision time t, the model input $X_t \in \mathbb{R}^{(L+H)\times N}$ stacks the past $L = 12$ hours of AMOS observations and the next $H = 24$ hours of LDAPS forecasts. Each row $x_\tau \in \mathbb{R}^N$ is the feature vector at hour τ, comprising the N meteorological and temporal features listed in Table 1. The input sequence is written as

$$X_t = \big[\underbrace{x_{t-L+1}, \ldots, x_t}_{\substack{\text{AMOS}\\ \text{Observation}}}, \underbrace{x_{t+1}, \ldots, x_{t+H}}_{\substack{\text{LDAPS}\\ \text{Forecast}}}\big]^\top \in \mathbb{R}^{(L+H)\times N}. \tag{1}$$

The observation segment $x_{t-L+1:t}$ provides recent context on the weather and runway direction state sustaining the current operation. The forecast segment $x_{t+1:t+H}$ provides forward-looking meteorological signals that may indicate future direction changes. A deep learning model f_θ takes X_t as input and outputs direction probabilities

$$p_{t+h} = P(y_{t+h} = 1 \mid X_t), \quad h = 1, \ldots, H. \tag{2}$$

Each direction probability p_{t+h} is then converted into a binary prediction $\hat{y}_{t+h}$ by thresholding at 0.5: outputs at or above 0.5 map to 1 (northbound) and outputs below 0.5 map to 0 (southbound):

$$\hat{y}_{t+h} = \mathbb{I}(p_{t+h} \geq 0.5). \tag{3}$$

Table 1 – Input features used in the runway direction forecasting model.

Feature	Description
s_d, c_d	Sine / cosine of annual cycle, $\sin/\cos(2\pi d/365)$
s_h, c_h	Sine / cosine of diurnal cycle, $\sin/\cos(2\pi h/24)$
I_{wknd}	Weekend indicator (1 = weekend, 0 = weekday)
I_{fcst}	Segment indicator (1 = forecast hour, 0 = observed history)
y	Active operating direction (1 = northbound, 0 = southbound; 0.5 for forecast hours)
τ_{since}	Hours since the last direction change (0 for forecast hours)
TW, CW	Tailwind / crosswind component (kt)
$\Delta\text{TW}, \Delta\text{CW}$	Hourly change in tailwind / crosswind (kt)
$\overline{\text{TW}}, \overline{\text{CW}}$	3-hour moving-average tailwind / crosswind (kt)
$T_{2\text{m}}, T_{\text{dew}}$	2-m air temperature (°C); dew point (K)
P_{precip}, P_{sfc}	precipitation (mm); surface pressure (Pa)

Notes: d and h denote day-of-year and hour-of-day. Wind components are computed relative to the northbound runway heading. The recent direction state y and elapsed time τ_{since} are derived from the active runway record logged in AMOS. The model therefore uses only AMOS and LDAPS feeds.

4. Model Architecture

The forecasting task requires integrating heterogeneous meteorological variables across both the observation-history and forecast-horizon segments. We use an encoder-only inverted Transformer [14] as the backbone architecture. Unlike standard time-series Transformers that tokenize the input along the time axis and produce one token per hour, the inverted formulation tokenizes along the feature axis and produces one token per input variable. Attention is therefore computed across features rather than across hours.

This design is motivated by the structure of the prediction problem. The forecast horizon is fixed and relatively short at 24 hours, while the predictive signal depends strongly on interactions among meteorological variables, including tailwind, crosswind, wind changes, persistence indicators, and forecast-segment information. The inverted formulation provides a natural way to emphasize these cross-feature interactions. Architectural comparisons with conventional time-step tokenization are left for future work.

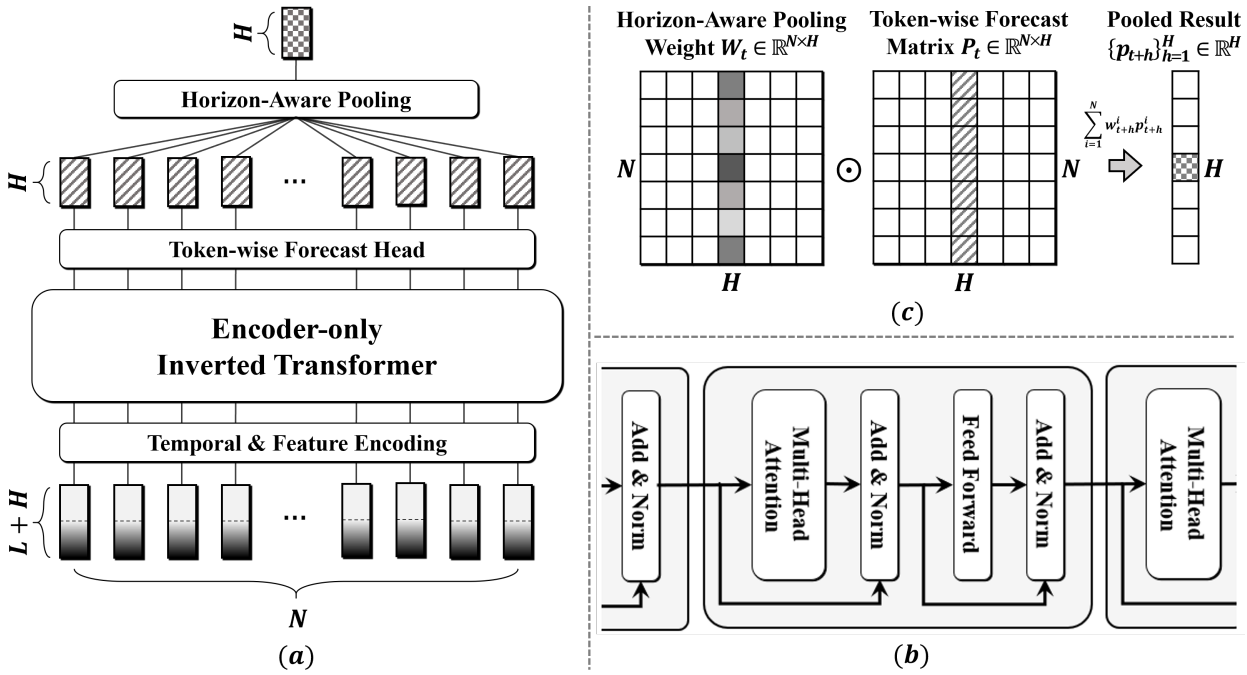


Figure 2 – Proposed model architecture for runway direction forecasting: (a) feature-tokenized inverted Transformer, (b) encoder block, and (c) horizon-aware pooling over token-wise forecasts.

4.1 Encoder-Only Inverted Transformer

Under the inverted formulation, the model input $X_t \in \mathbb{R}^{(L+H)\times N}$ is tokenized into N tokens of length $L+H$. The i-th token, $x_{i,t} = [x_{i,t-L+1}, \ldots, x_{i,t+H}]$, contains the temporal evolution of the i-th feature across the observation–forecast window.

Each token is embedded into the model dimension d via two additive components. A shared temporal projection $\mathrm{TempEnc} : \mathbb{R}^{L+H} \to \mathbb{R}^d$ compresses every token's time series through a single weight matrix common to all features. Because the projection is shared, features with similar temporal patterns over the window—for example, tailwind and its 3-hour moving average during steady winds—collapse to nearly identical latent vectors, leaving downstream attention unable to distinguish physically distinct inputs that carry different predictive signal. A learnable feature embedding $\mathrm{FeatEnc} : \{1, \ldots, N\} \to \mathbb{R}^d$ resolves this by adding a feature-specific identity vector to the temporal output. The combined embedding is

$$z_i = \mathrm{TempEnc}(x_{i,t}) + \mathrm{FeatEnc}(i) \in \mathbb{R}^d, \tag{4}$$

followed by dropout. Stacking across features yields the token matrix $Z \in \mathbb{R}^{N\times d}$, which is processed by stacked multi-head self-attention (Fig. 2b) and position-wise feed-forward layers. Self-attention operates as

$$\mathrm{Attention}(Q,K,V) = \mathrm{Softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V, \tag{5}$$

where Q, K, and V are linear projections of Z. Because the tokens correspond to features rather than hours, the attention weights quantify pairwise dependencies among features—the cross-feature interactions that the embedding scheme was designed to preserve.

4.2 Token-Wise Forecasting with Horizon-Aware Pooling

Each encoder token is mapped to a 24-hour probability sequence through a linear forecasting head, yielding token-wise probabilities $p_{i,t+h}$ for $i = 1, \ldots, N$ and $h = 1, \ldots, H$, which together form the matrix P_t (Fig. 2c). Forecasting at the token level preserves feature-specific predictive signals and provides a structured basis for aggregation.

Let $w_{i,h}$ denote the horizon-aware aggregation weights, computed from the encoder representations

and normalized across features such that $\sum_{i=1}^{N} w_{i,h} = 1$. The pooled probability at horizon h is

$$p_{t+h} = \sum_{i=1}^{N} w_{i,h}\, p_{i,t+h}, \qquad h = 1, \ldots, H. \tag{6}$$

Because $w_{i,h}$ varies across both features and forecast horizons, the model can adapt feature weights as predictive relevance changes with lead time. At short horizons, persistence cues such as the active direction y and the time since the last change τ_{since} typically carry more weight; at longer horizons, forecast wind components such as TW and CW become more influential. The pooled sequence is converted to binary predictions via $\hat{y}_{t+h} = \mathbb{I}(p_{t+h} \geq 0.5)$.

5. Transition-Sensitive Training Objective

Per-hour classification losses such as binary cross-entropy minimize misclassified forecast hours but do not explicitly control transition timing or transition frequency. Because transitions occupy only a small fraction of the 24-hour horizon, a model can achieve high per-hour accuracy while still exhibiting two operationally important failure modes: misplaced transition timing and unrealistic transition counts.

Figure 3 illustrates these failure modes. The ground truth remains in the northbound direction (N) from $T1$ to $T4$, changes to southbound (S) at $T5$, and persists through $T8$. Forecast #1 comes closest, predicting the transition one hour early at $T4$. Forecasts #2 and #3 misplace the transition timing and introduce extra transitions, yielding three predicted transitions against one true transition. All three forecasts produce the same number of misclassified hours, so a per-hour loss cannot distinguish them, even though only #1 reflects the true transition behavior.

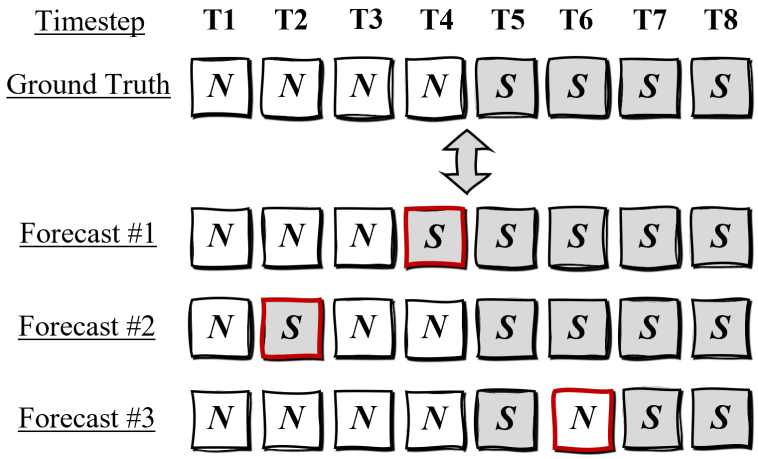


Figure 3 – Two failure modes of per-hour losses. The red border indicates mispredicted hours.

We therefore introduce a transition-sensitive composite objective L that supplements a per-hour loss term L_{state} with two additive loss terms targeting transition timing L_{sw} and transition frequency L_{rate}:

$$L = \lambda_{\text{state}} L_{\text{state}} + \lambda_{\text{sw}} L_{\text{sw}} + \lambda_{\text{rate}} L_{\text{rate}}, \tag{7}$$

where λ_{state}, λ_{sw}, and λ_{rate} are nonnegative coefficients. The two auxiliary terms are designed to address the two failure modes identified above: L_{sw} targets mislocalized timing, while L_{rate} targets unrealistic counts.

This formulation differs from class-weighted BCE or oversampling. Those approaches can increase the penalty associated with minority direction (southbound, which accounts for only 26% of operating hours), but they still treat each forecast hour independently. The proposed auxiliary losses operate on changes between consecutive predictions, directly targeting transition timing and transition count.

5.1 Per-Hour State Loss

L_{state} is the Bernoulli negative log-likelihood between the predicted probability p_{t+h} and the ground-truth label y_{t+h} at each forecast hour:

$$L_{\text{state}} = \frac{1}{H}\sum_{h=1}^{H}\left[-y_{t+h}\log p_{t+h} - (1-y_{t+h})\log(1-p_{t+h})\right]. \tag{8}$$

Because L_{state} evaluates each forecast hour independently, it provides stable hour-by-hour direction supervision but cannot represent the transition failure modes identified above. The next two subsections introduce auxiliary terms that target these aspects directly.

5.2 Transition Timing Loss

We need a differentiable surrogate for the probability that a transition occurs between hours $t+h-1$ and $t+h$. Such a transition is equivalent to the two consecutive directions differing, which can be expressed in terms of the predicted direction probabilities p_{t+h-1} and p_{t+h} as

$$c_{t+h} = p_{t+h-1}(1-p_{t+h}) + (1-p_{t+h-1})p_{t+h}, \qquad h = 1,\ldots,H. \tag{9}$$

This expression takes a value in $[0,1]$: it approaches 0 when the predicted direction persists across the two hours and approaches 1 when the predicted direction changes, providing a smooth, differentiable proxy for the discrete transition event.

The ground-truth transition indicator is $z_{t+h} = \mathbb{I}(y_{t+h} \neq y_{t+h-1})$, equal to one at hours where a direction change actually occurred and zero elsewhere. Because transitions are sparse and discretized at hourly resolution, training directly against z_{t+h} provides a highly localized, discontinuous gradient. To stabilize training and tolerate small timing errors, we soften the target by placing a Gaussian bump around each true-transition hour k (those satisfying $z_k = 1$):

$$\tilde{z}_{t+h} = \max_{k:\,z_k=1} \exp\left(-\frac{(t+h-k)^2}{2\sigma^2}\right). \tag{10}$$

Because transition hours are vastly outnumbered by non-transition hours, L_{sw} applies focal-weighted binary cross-entropy [15] between c_{t+h} and $\tilde{z}_{t+h}$ to focus learning on the rare transition hours:

$$L_{\text{sw}} = \frac{1}{H}\sum_{h=1}^{H}\alpha_{t+h}(1-q_{t+h})^{\gamma}\,\text{BCE}(c_{t+h},\tilde{z}_{t+h}), \tag{11}$$

with $q_{t+h} = \tilde{z}_{t+h}c_{t+h} + (1-\tilde{z}_{t+h})(1-c_{t+h})$ and $\alpha_{t+h} = \alpha\tilde{z}_{t+h} + (1-\alpha)(1-\tilde{z}_{t+h})$. The factors α and γ follow the standard focal-loss formulation [15].

5.3 Transition Frequency Loss

L_{sw} concentrates training pressure at the true-transition hours through its focal weighting, but this leaves false transition predictions at other hours relatively under-penalized. Such spurious predictions accumulate across the horizon and cause the forecast to overshoot the true transition count. L_{rate} remedies this by penalizing the gap between the model's expected transition count over the horizon, $\sum_{h=1}^{H} c_{t+h}$, and the observed count, $\sum_{h=1}^{H} z_{t+h}$, via the Huber loss:

$$L_{\text{rate}} = \rho\left(\sum_{h=1}^{H} c_{t+h} - \sum_{h=1}^{H} z_{t+h}\right), \qquad \rho(u) = \begin{cases} \frac{1}{2}u^2, & |u| \le \delta, \\ \delta\left(|u| - \frac{1}{2}\delta\right), & |u| > \delta. \end{cases} \tag{12}$$

6. Experiments and Results

6.1 Experimental Settings

The model was evaluated on five years (2017–2021) of runway direction and weather data from RKSI under the day-ahead forecasting setting defined in Section 3., split randomly into training, validation, and test sets in a 70/10/20 ratio.

The encoder uses model dimension $d_{\text{model}} = 80$, 3 encoder layers, 4 attention heads, feed-forward dimension 320, and dropout 0.1, with input window $L = 12$ and forecast horizon $H = 24$. Training used Adam with batch size 4096, learning rate 10^{-3}, and weight decay 10^{-5}, with early stopping when Switch F1 did not improve for 10 consecutive epochs (defined in Section 6.2). The composite loss coefficients were $\lambda_{\text{state}} = 1.0$, $\lambda_{\text{sw}} = 0.7$, $\lambda_{\text{rate}} = 0.01$, with $\sigma = 0.8$, $\alpha = 0.75$, $\gamma = 1.0$, selected by grid search. Each training run used five random seeds. Reported metrics are mean $\pm$ standard deviation across seeds.

6.2 Evaluation Metrics

Performance is evaluated from two complementary perspectives. The first is per-hour direction prediction: how accurately the model predicts the operating direction at each forecast hour. The second is transition-event detection: how well the model identifies transitions in the predicted sequence and aligns them with the true transitions.

Direction prediction is evaluated at the hour level. Accuracy is the fraction of hours correctly predicted over the 24-hour horizon. Macro-F1 is the unweighted mean of the two direction-wise F1 scores (northbound and southbound); we also report direction-wise Precision, Recall, and F1 separately for both operating directions.

Transition-event prediction is evaluated by one-to-one matching between predicted and true transitions within a temporal tolerance window of ± 1 h ('tight') or ± 2 h ('mid'). Three properties of the resulting matches are reported. 'Detection quality' comprises Switch Precision (fraction of predicted transitions matched to a true one), Switch Recall (fraction of true transitions matched to a predicted one), and their harmonic mean, Switch F1. 'Timing quality' is reported as AbsDelay, the mean absolute timing error (hours) of matched transitions. 'Frequency quality' is reported as False Alarms (unmatched predicted transitions per 24-hour forecast) and Count MAE ($|\#\text{predicted} - \#\text{true}|$ per forecast).

6.3 Direction Prediction Results

As a baseline, we consider a persistence model ('NoPred') that assumes the current operating direction remains unchanged throughout the next 24 hours, i.e., $y_{t+h} = y_t$ for $h = 1, \ldots, H$. This baseline is nontrivial: with only 0.85 direction changes per day, simply repeating the current direction already achieves 76.34% accuracy averaged over the 24-hour horizon.

The proposed model improves over NoPred, reaching 88.34% accuracy and 0.85 Macro-F1 (Table 2). The improvement is larger on the southbound direction than on the northbound one (F1 +0.21 vs. +0.08), and this gain is precision-driven: southbound precision rises substantially more than recall (+0.27 vs. +0.16). This pattern suggests that the model tends to predict southbound only when the evidence is strong, rather than switching aggressively under uncertainty, since the latter would typically increase recall at the cost of precision. The precision-favored profile suits operational use: false southbound predictions imply spurious transitions from the dominant northbound direction, triggering unnecessary planning adjustments, whereas missed periods are recovered by the next hourly forecast update.

Table 2 – Direction prediction performance. Bold font denotes the best result for each metric.

Model	Accuracy	Macro-F1	Northbound			Southbound		
			Precision	Recall	F1	Precision	Recall	F1
NoPred	76.34%	0.70	0.84	0.84	0.84	0.56	0.56	0.56
Proposed	**88.34%**	**0.85**	**0.90**	**0.95**	**0.92**	**0.83**	**0.72**	**0.77**

Figure 4 shows accuracy as a function of forecast lead time (1–24 hours ahead). The persistence baseline degrades steadily as lead time increases, because the likelihood of a direction change increases over longer horizons. In contrast, the proposed model maintains relatively stable accuracy across the 24-hour horizon. This behavior is important for strategic planning, where forecasts several hours ahead are more useful than immediate predictions alone.

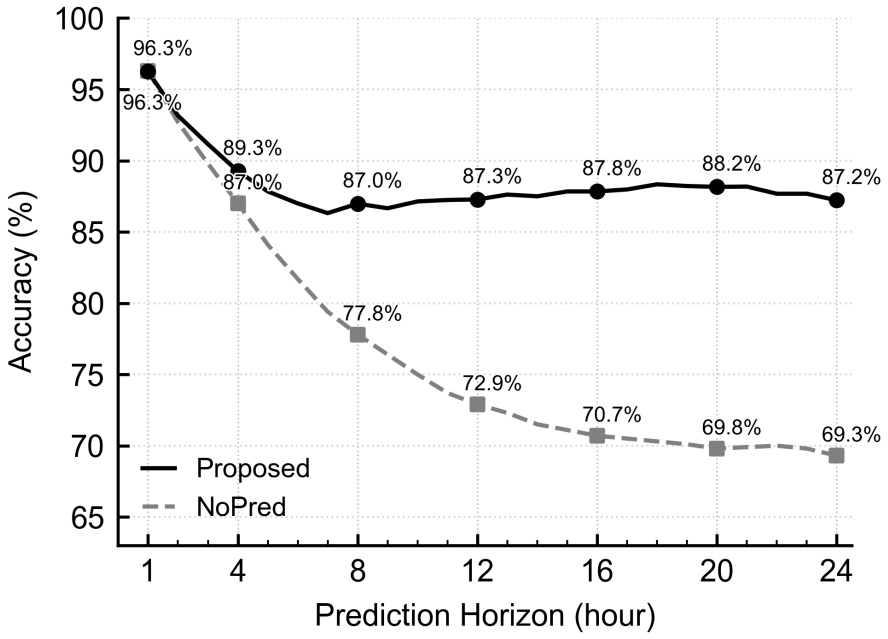


Figure 4 – Prediction accuracy as a function of forecast lead time (1 to 24 hours ahead).

6.4 Transition Prediction Results

The proposed objective L (‘*Composite*’) is compared against three reduced variants that isolate the contribution of each loss term: *State-Only* uses L_{state} only; *State+Switch* adds L_{sw}; *State+Rate* adds L_{rate}. All four variants share the loss coefficients and hyperparameters specified in Section 6.1, so that performance differences reflect only the presence or absence of L_{sw} or L_{rate}.

Tables 3 and 4 report per-hour direction prediction and transition-event prediction performance, respectively; the latter is evaluated under the tight (± 1 h) and mid (± 2 h) matching criteria. In Table 4, Switch F1 summarizes transition detection, AbsDelay measures timing localization among matched transitions, and False Alarms and Count MAE measure whether the model predicts too many or too few transitions.

At the direction level (Table 3), *Composite* achieves the best mean on every metric, but the absolute improvements over *State-Only* are modest—roughly half a percentage point in Accuracy. This is expected: direction transitions are rare events, so direction-level metrics that average across all hours are necessarily dominated by the predominant non-transition hours. The transition-specific metrics in Table 4, however, evaluated only over the sparse transition events themselves, reveal noteworthy improvements consistent with the design intent of each loss term, which we examine row by row below.

State-Only’s Switch F1 is already respectable, indicating that the per-hour loss alone partially recovers transitions as a byproduct of getting individual hours right. But on most transition metrics it is the weakest of the four variants, which is consistent with the motivation that per-hour training alone does not prioritize transition events. *State+Switch* improves where L_{sw} is designed to act: Switch Recall rises and AbsDelay shrinks, as the focal weighting concentrates pressure on the true-transition hours and aligns the predicted transition timing with the true timing. False Alarms and Count MAE, however, are essentially unchanged relative to *State-Only*, because L_{sw} acts on individual hour-pairs but not on the total number of transitions. *State+Rate* shows the complementary pattern: substantial reductions in False Alarms and Count MAE with little change in AbsDelay, since L_{rate} constrains the total transition count without controlling when transitions should occur.

Composite achieves the best mean on Switch F1, Switch Precision, Switch Recall, AbsDelay, and Count MAE under both tight and mid criteria, and remains competitive on False Alarms. Its AbsDelay matches *State+Switch* and its Count MAE matches *State+Rate*, indicating that the two auxiliary terms address genuinely complementary failure modes—L_{sw} controls transition timing, L_{rate} controls transition count—and combining them captures both gains without one term degrading the other. *Composite* therefore provides a balanced operational profile that improves both timing and frequency

together. Crucially, the improvement persists under the tight ± 1 h criterion as well as the mid ± 2 h criterion. A model that placed transitions loosely—hitting the right hour within ± 2 hours but not within ± 1 hour—would gain only under the mid window; *Composite*'s gain under both criteria is consistent with predicted transitions clustering tightly around the true transition hour rather than being scattered nearby.

Table 3 – Direction prediction performance across loss-term variants. Bold denotes the best result per metric, underline the second best; all metrics are higher-is-better.

Variant	Accuracy	Macro-F1	Northbound F1	Southbound F1
State-Only	$87.78 \pm 0.68\%$	$0.833 \pm .015$	$0.92 \pm .003$	$0.75 \pm .026$
State+Switch	$\underline{88.12 \pm 0.25}\%$	$\underline{0.840 \pm .006}$	$\underline{0.92 \pm .001}$	$\underline{0.76 \pm .011}$
State+Rate	$88.02 \pm 0.46\%$	$0.837 \pm .009$	$0.92 \pm .003$	$0.75 \pm .016$
Composite	$\mathbf{88.34 \pm 0.13}\%$	$\mathbf{0.85 \pm .002}$	$\mathbf{0.92 \pm .001}$	$\mathbf{0.77 \pm .004}$

Table 4 – Transition-event prediction across loss-term variants. Bold denotes the best result per metric, underline the second best; AbsDelay, FalseAlarms, and Count MAE are lower-is-better, all other metrics higher-is-better.

Variant	Criteria	Detection Quality			Timing	Frequency Quality	
		Sw. F1	Sw. Precision	Sw. Recall	AbsDelay	FalseAlarms	Count MAE
State-Only	Tight	$0.257 \pm .020$	$0.296 \pm .011$	$0.227 \pm .025$	$0.672 \pm .008$	$0.476 \pm .040$	$0.564 \pm .019$
	Mid	$0.377 \pm .032$	$0.435 \pm .019$	$0.334 \pm .039$	$1.139 \pm .020$	$0.381 \pm .029$	
State+Switch	Tight	$\underline{0.263 \pm .015}$	$0.302 \pm .010$	$\underline{0.233 \pm .018}$	$\underline{0.664 \pm .012}$	$0.475 \pm .025$	$0.550 \pm .016$
	Mid	$\underline{0.389 \pm .020}$	$0.447 \pm .014$	$\underline{0.345 \pm .025}$	$1.138 \pm .022$	$0.377 \pm .021$	
State+Rate	Tight	$0.260 \pm .022$	$\underline{0.308 \pm .018}$	$0.225 \pm .025$	$0.667 \pm .015$	$\mathbf{0.447 \pm .029}$	$\underline{0.539 \pm .026}$
	Mid	$0.385 \pm .031$	$\underline{0.456 \pm .026}$	$0.334 \pm .035$	$\underline{1.136 \pm .021}$	$\mathbf{0.351 \pm .025}$	
Composite	Tight	$\mathbf{0.266 \pm .014}$	$\mathbf{0.310 \pm .017}$	$\mathbf{0.233 \pm .013}$	$\mathbf{0.652 \pm .010}$	$\underline{0.459 \pm .024}$	$\mathbf{0.537 \pm .024}$
	Mid	$\mathbf{0.394 \pm .014}$	$\mathbf{0.459 \pm .020}$	$\mathbf{0.345 \pm .013}$	$\mathbf{1.128 \pm .024}$	$\underline{0.360 \pm .024}$	

7. Conclusion and Future Work

This study presented a weather-driven deep learning framework for hourly runway operating direction forecasting over a 24-hour horizon. The framework combines AMOS observations and LDAPS forecasts with an encoder-only inverted Transformer and a transition-sensitive training objective. The proposed objective supplements per-hour classification with auxiliary terms for transition timing and transition frequency.

At Incheon International Airport, the proposed model improves per-hour accuracy from 76.34% for the persistence baseline to 88.34%, and Macro-F1 from 0.70 to 0.85. The largest improvement occurs for the minority southbound direction. Ablation experiments show that the transition timing and frequency terms provide complementary benefits: the timing term improves localization of predicted transitions, while the frequency term improves the predicted transition count. Their combination produces the most balanced performance across transition detection, timing, and frequency metrics.

These results suggest that runway operating direction can be forecast from standardized weather observations and numerical weather prediction data while remaining sensitive to rare transition events. This sensitivity is important because such transitions, although infrequent, are operationally significant for airport planning and traffic flow management.

Future work will proceed in three directions. First, cross-airport validation will be conducted at airports with different runway layouts and direction distributions, such as Jeju International Airport, to evaluate model portability. Second, broader baseline comparisons, including gradient-boosted trees and conventional time-step Transformers, will clarify the relative contribution of the proposed objec-

tive and architecture. Third, interpretability analyses, including horizon-aware pooling-weight analysis and permutation-based feature importance, will be used to examine how the model changes its reliance on persistence and forecast weather features across lead times. Incorporating operational variables such as traffic demand and airport-specific tactical constraints is also a promising extension, particularly when wind conditions are close to the direction-change threshold.

8. Acknowledgments

This work is supported by the Korea Agency for Infrastructure Technology Advancement (KAIA) grant funded by the Ministry of Land, Infrastructure and Transport (Grant RS-2021-KA163373 and RS-2024-00398152).

9. Contact Author Email Address

Corresponding Author: Hyewook Kim (hyewookkim@kari.re.kr)

10. Copyright Statement

The authors confirm that they, and/or their company or organization, hold copyright on all of the original material included in this paper. The authors also confirm that they have obtained permission, from the copyright holder of any third party material included in this paper, to publish it as part of their paper. The authors confirm that they give permission, or have obtained permission from the copyright holder of this paper, for the publication and distribution of this paper as part of the ICAS proceedings or as individual off-prints from the proceedings.

References

[1] Bertsimas D, Frankovich M and Odoni A. Optimal selection of airport runway configurations. *Operations Research*, Vol. 59, No. 6, pp 1407-1419, 2011.

[2] Jacquillat A, Odoni A R and Webster M D. Dynamic control of runway configurations and of arrival and departure service rates at JFK airport under stochastic queue conditions. *Transportation Science*, Vol. 51, No. 1, pp 155-176, 2017.

[3] Avery J and Balakrishnan H. Predicting airport runway configuration: a discrete-choice modeling approach. *13th USA/Europe Air Traffic Management Research and Development Seminar (ATM2015)*, Lisbon, Portugal, pp 1-10, 2015.

[4] Avery J and Balakrishnan H. Data-driven modeling and prediction of the process for selecting runway configurations. *Transportation Research Record: Journal of the Transportation Research Board*, Vol. 2600, No. 1, pp 1-11, 2016.

[5] Puranik T G, Memarzadeh M and Kalyanam K M. Predicting airport runway configurations for decision-support using supervised learning. *2023 IEEE/AIAA 42nd Digital Avionics Systems Conference (DASC)*, Barcelona, Spain, pp 1-9, 2023.

[6] Dalmau R and Herrema F. Encoder-decoder approach to predict airport operational runway configuration. *9th SESAR Innovation Days (SID)*, Athens, Greece, pp 1-8, 2019.

[7] Wang Y and Zhang Y. Prediction of runway configurations and airport acceptance rates for multi-airport system using gridded weather forecast. *Transportation Research Part C: Emerging Technologies*, Vol. 125, pp 103049, 2021.

[8] Wang X, Liu L, Li S, Xu Y, Low K H and Wu W. Prediction of multiple runway configurations for efficient flight operations by hybrid LSTM and transformer model. *Scientific Reports*, Vol. 15, No. 1, pp 37764, 2025.

[9] Hesselink H and Nibourg J. Probabilistic 2-day forecast of runway use: efficient and safe runway allocation based on weather forecast. *9th USA/Europe Air Traffic Management Research and Development Seminar (ATM2011)*, Berlin, Germany, pp 1-9, 2011.

[10] Lau M E C, Lam A J G and Alam S. Predicting runway configuration transition timings using machine learning methods. *2021 Winter Simulation Conference (WSC)*, Phoenix, AZ, USA, pp 1-12, 2021.

[11] Republic of Korea. *RKSI AD 2 1.5 Preferential RWY System*. Aeronautical Information Publication, 2026.

[12] Republic of Korea. *RKSI AD 2.20 Local Aerodrome Regulations*. Aeronautical Information Publication, 2026.

[13] Republic of Korea. *RKSI AD CHART 2-1*. Aeronautical Information Publication, 2026.

[14] Liu Y, Hu T, Zhang H, Wu H, Wang S, Ma L and Long M. iTransformer: inverted transformers are effective for time series forecasting. *The Twelfth International Conference on Learning Representations*, 2024.

[15] Lin T Y, Goyal P, Girshick R B, He K and Dollár P. Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 42, pp 318-327, 2017.