# Can Symptom Search Trends Forecast Respiratory Virus Patient Counts? A Rhinovirus Case Study in Korean Syndromic Surveillance

**Anonymous Authors**[1]

## Abstract

Official respiratory-virus counts by governments are reliable but often delayed, while symptom-related search activity can appear before updated surveillance reports. We ask whether respiratory-virus patient counts can be forecast from online Relative Search Volume (RSV), a platform-normalized search-interest index, without using past patient counts. Using 270 weeks of Korea Disease Control and Prevention Agency (KDCA) surveillance data, we construct a structured dataset aligning patient counts with Naver and Google RSV for symptom queries expanded through an LLM-assisted pipeline. In a Rhinovirus case study, existing time-series backbones trained with patient counts but restricted to search-history inputs at prediction time produce RSV-only forecasts that partially track patient trajectories and achieve performance comparable to patient-history baselines.

## 1. Introduction

Respiratory-virus surveillance relies on patient counts reported through public-health systems. These counts remain the clinical gold standard, but they become available only after care seeking, testing or coding, aggregation, and quality control. For operational planning, such delays narrow the window for staffing, bed allocation, diagnostic preparation, and infection-control response. The practical forecasting problem is therefore not only to extrapolate a clinical time series, but also to identify timely signals observed before updated patient counts enter the surveillance system.

Online search activity is one observable form of preclinical health-information seeking. In this paper, Relative Search Volume (RSV) denotes a platform-normalized search-interest index, not an absolute query count; we write respiratory syncytial virus explicitly when referring to the virus. When individuals experience respiratory symptoms, they may search for symptom-related information through online tools. Aggregated RSV for such queries can therefore serve as a timely signal related to community illness activity. A key practical challenge is that clinical symptom terminology and public search vocabulary are not identical. The same symptom can appear in colloquial Korean, spacing variants, abbreviations, or related expressions, making query expansion a necessary step in constructing structured search-based surveillance covariates. Search-based syndromic surveillance has been studied for influenza and other outbreaks (Polgreen et al., 2008; Ginsberg et al., 2009), but also requires careful validation because query behavior can drift (Lazer et al., 2014).

Most forecasting backbones, by contrast, are formulated as endogenous forecasters that extrapolate a target from its own history (Zeng et al., 2023; Liu et al., 2024). This leaves an operational gap when historical patient labels exist for model development, but recent patient observations are delayed at prediction time. We study whether a search-history signal alone can forecast future patient counts under this RSV-to-patient setting. To do so, we construct a structured healthcare forecasting dataset linking weekly KDCA respiratory-virus surveillance to Korean search-query RSV collected from Naver and Google, and analyze Rhinovirus as the main case study.

Our contributions are summarized as follows:

- **LLM-assisted symptom-query construction.** We expand clinically motivated symptom terms into Korean search queries and collect aligned Naver/Google RSV for KDCA respiratory surveillance.
- **RSV-to-patient forecast modeling.** We formulate $x_{1:T} \rightarrow y_{T+1:T+H}$ modeling and evaluate representative time-series backbones using RSV histories as prediction-time inputs.
- **Early-warning feasibility analysis.** We show that selected RSV-only inputs can match or improve patient-history baselines in retrospective Rhinovirus forecasts, supporting RSV-only monitoring as an early-warning feasibility direction.

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

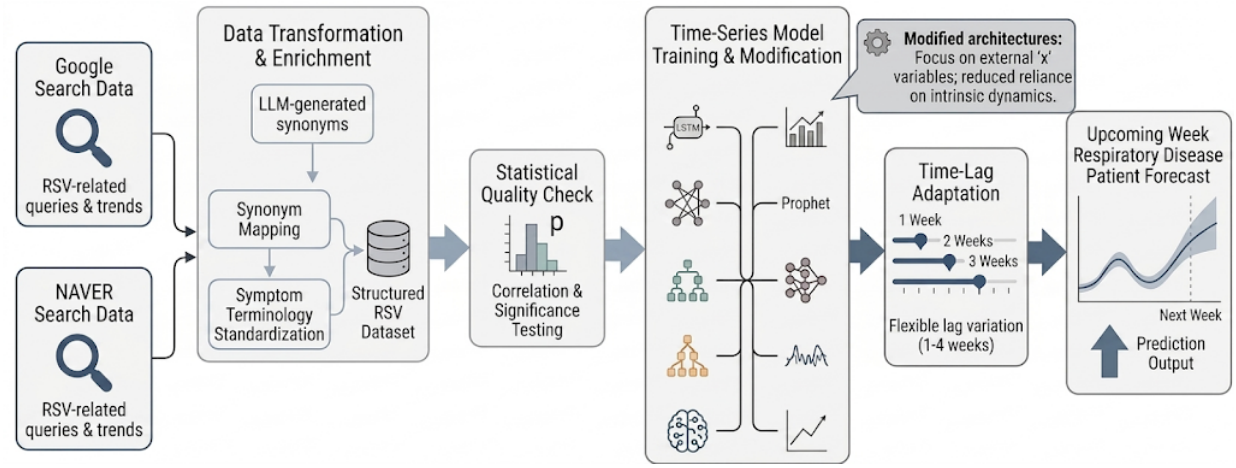



*Figure 1.* Overview of the study. Symptom keywords are expanded into Korean search queries, converted to Naver/Google RSV time series, filtered for sufficient nonzero observations, and used as RSV-only inputs to forecast KDCA respiratory-virus patient counts, with Rhinovirus analyzed as the main case study.

## 2. Data Collection and Preprocessing

### 2.1. KDCA Patient Counts

We use weekly acute respiratory infection surveillance counts publicly released by the Korea Disease Control and Prevention Agency (KDCA). The KDCA tables report virus-specific patient counts aggregated on the national surveillance calendar, with each reporting week ending on Saturday. After preprocessing, we use 270 weekly records from October 2019 to December 2024 and align all search-volume features to the same KDCA surveillance-week index. The main analysis focuses on Rhinovirus; Parainfluenza and respiratory syncytial virus are retained as descriptive context in this short paper. We use a chronological 70%/10%/20% train/validation/test split, with the final 54 weeks used for held-out testing.

### 2.2. Symptom Query Expansion

KDCA surveillance descriptions provide canonical symptom terms for acute respiratory infections, while public search behavior includes heterogeneous colloquial expressions, spelling variants, and spacing patterns. To capture this mismatch between clinical terminology and search behavior, we organize the initial Korean symptom keywords by virus and symptom group, then apply an LLM-assisted query expansion step to generate candidate derived expressions. In the implementation, GPT-4o generated Korean synonyms and spelling/spacing variants for each seed term; we then deduplicated the resulting query list before RSV collection. This LLM step was restricted to query construction; forecasting, evaluation, and result interpretation used the deterministic pipeline described below. The process produced 3,336 unique search terms across the three viruses; the Rhinovirus subset contains 1,207 terms per source, giving 2,414 possible keyword–source RSV series.

**Search RSV collection.** For each keyword, we collect source-specific Relative Search Volume from Naver DataLab and Google Trends. Each Korean query or synonym group is submitted over the same study period as the KDCA surveillance data. Both platforms report normalized search-interest indices rather than raw query counts, so RSV values are interpreted within a source, query, and time range. Naver provides daily relative ratios, which we aggregate to weekly RSV by averaging dates inside each KDCA surveillance week. Google Trends RSV is queried for the South Korea region with the Korean locale and returned as a weekly 0–100 index. Because RSV scales are platform-specific, each Naver or Google source–query pair is kept as a separate KDCA-aligned weekly feature column.

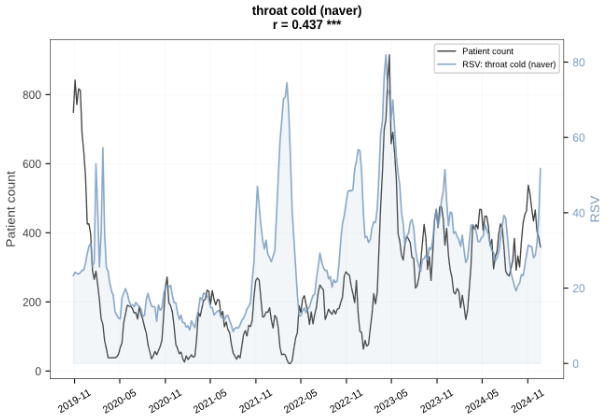
**Weekly RSV feature construction.** Let $\mathcal{V}$ denote the set of respiratory viruses, $\mathcal{M} = \{\text{Naver}, \text{Google}\}$ the search sources, and $\mathcal{Q}_v$ the deduplicated keyword set for virus $v \in \mathcal{V}$. For source $m$ and keyword $q$, we align the timestamped platform series $\tilde{x}_{m,q,\tau}$ to KDCA surveillance weeks $\{W_t\}_{t=1}^{T}$:

$$x_t^{(v,m,q)} = A_m\left(\{\tilde{x}_{m,q,\tau} : \tau \in W_t\}\right),$$

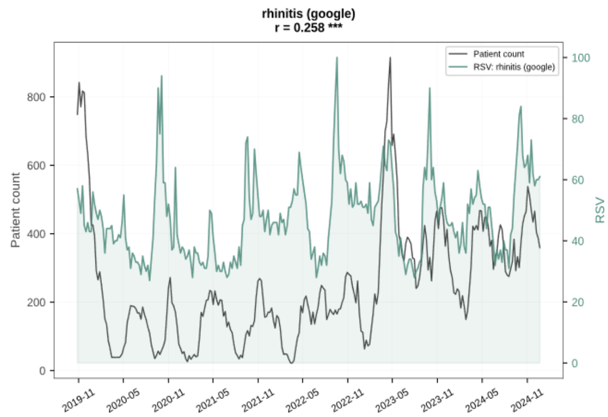
Here $A_m$ is a 7-day mean for daily Naver ratios and the platform-provided weekly value for Google Trends. The resulting long-format table contains $(v, t, m, q, g(q), x_t^{(v,m,q)}, y_t^{(v)})$ and is pivoted into a weekly matrix $\mathbf{X}^{(v)} \in \mathbb{R}^{T \times P_v}$ paired with patient counts $\mathbf{y}^{(v)} \in \mathbb{R}^{T}$. Implementation-level API fields and function calls are provided in Appendix A.

**Filtering and keyword selection.** A collected virus–source–query series was treated as usable only if

$$c_{v,m,q} = \sum_{t=1}^{T} \mathbb{I}\left[x_t^{(v,m,q)} > 0\right] \geq 5.$$




*(a)* Naver RSV




*(b)* Google RSV

*Figure 2.* Example Rhinovirus surveillance and search-trend time series. KDCA patient counts are shown with representative symptom-query RSV series from each online search platform.

For correlation-based feature analysis, we additionally removed all-NaN columns and selected columns whose Pearson correlation with the corresponding KDCA patient-count series exceeded $\rho \geq 0.3$. For Rhinovirus, this thresholded preprocessing retained 16 Google and 97 Naver single-keyword RSV columns. These retained columns define the candidate single-keyword RSV inputs used in the forecasting experiments.

## 3. RSV-to-Patient Forecast Modeling

### 3.1. Problem Formulation

For Rhinovirus, let $y_t$ denote the weekly KDCA patient count and let $p = (m, q)$ index one retained source–query RSV feature. We write its weekly value as $x_t^{(p)}$. Given a lookback length $L$ and prediction horizon $H$, define the target vector

$$\mathbf{y}_{t,H} = [y_{t+1}, \ldots, y_{t+H}].$$

The standard patient-history baseline forecasts this target from the recent endogenous history:

$$\hat{\mathbf{y}}_{t,H}^{\text{pat}} = f_\theta(y_{t-L+1:t})\,.$$

Our modeling setting keeps the same target but replaces the input with an RSV search-history window:

$$\hat{\mathbf{y}}_{t,H,p,s}^{\text{RSV}} = f_\theta\left(x_{t-L+1-s:t-s}^{(p)}\right),$$

where $s$ is an input lag shift. Thus, patient counts supervise training through $\mathbf{y}_{t,H}$, while prediction-time inputs are restricted to RSV histories. In all deep-learning experiments, we set $L = 48$ weeks, evaluate $H \in \{1, 2, 4, 8\}$, and grid-search $s \in \{0, 1, 2, 3\}$ for RSV-only runs. Reported retrospective results summarize the selected configurations from this keyword–lag–horizon grid.

**Keyword analysis.** For each retained source–query feature $p$, we compute Pearson correlation between RSV and patient counts across candidate temporal shifts:

$$r_p(\ell) = \text{corr}\left(x_{t-\ell}^{(p)}, y_t\right),$$

and use the maximum-correlation lag as a descriptive statistic; positive $\ell$ means the RSV series is shifted earlier relative to patient counts. This analysis is used to describe whether symptom-search behavior tends to move with or before patient counts, and to identify keywords worth visualizing. The main modeling results are based on held-out forecasting error rather than correlation alone.

**Models.** We evaluate 20 forecasting backbones and report five representative families in the main text: DLinear (Zeng et al., 2023), iTransformer (Liu et al., 2024), Mamba (Gu & Dao, 2024), LSTM (Hochreiter & Schmidhuber, 1997), and GRU (Cho et al., 2014). The full model list, citations, and hyperparameters are provided in Appendix B and Appendix C. Neural models are trained for up to 100 epochs with validation-loss early stopping using a patience of 30 epochs. Family-specific learning rates and schedulers are taken from the experimental configuration. Each model–keyword–lag–horizon configuration is repeated over 10 random seeds, and metrics are aggregated over these repeated runs.

**Metrics and ranking.** We compute RMSE, SMAPE, Pearson correlation, dynamic time warping distance (DTW), and mean directional accuracy (MDA). Keyword rankings are derived by averaging metric-wise ranks, using lower-is-better for RMSE/SMAPE/DTW and higher-is-better for Pearson/MDA. For the main text, RMSE is emphasized because it directly reflects patient-count scale; the full rank table uses all five metrics.

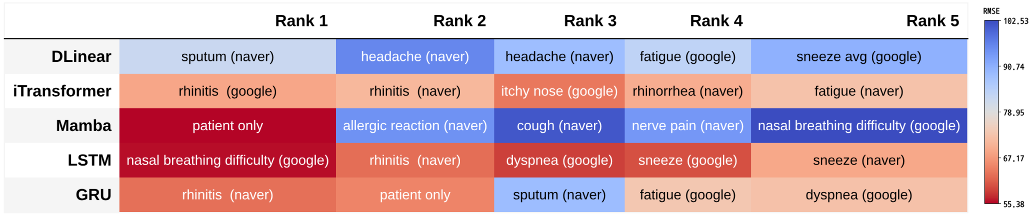



*Figure 3.* Top-5 keyword rankings per representative model, with patient-history baselines included where they rank among the top entries. Ranks are based on the five-metric average rank; cell color encodes RMSE on the held-out evaluation window, where lower is better.

*Table 1.* Main RMSE comparison for Rhinovirus between patient-history baselines and the best RSV-only keyword–source configuration after lag–horizon selection. Values are computed on the common overlapping evaluation window after aggregating predictions from 10 random seeds.

| Model | Patient | Best RSV | Keyword (source) |
|---|---|---|---|
| DLinear | 95.7 | 82.3 | sputum (naver) |
| iTransformer | 52.1 | 68.8 | rhinitis (google) |
| Mamba | 55.3 | 94.6 | allergic reaction (naver) |
| LSTM | 53.1 | 55.8 | nasal breathing difficulty (google) |
| GRU | 64.9 | 62.9 | rhinitis (naver) |

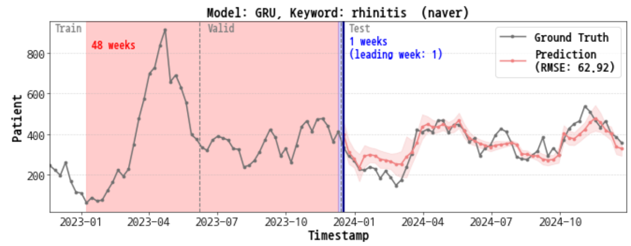



*Figure 4.* Representative held-out forecasts for Rhinovirus using the GRU model with the Naver rhinitis RSV input, emphasizing whether RSV-only forecasts track outbreak shape and turning points beyond aggregate error.

# 4. Experiments and Results

We analyze the retrospective Rhinovirus forecasts along three axes. First, we compare patient-history inputs and RSV-only inputs under the same representative forecasting backbones. Second, we inspect model-specific keyword rankings to identify which search terms drive RSV-only performance. Third, we compare forecast trajectories to determine whether selected search terms provide interpretable signals beyond aggregate error reductions.

## 4.1. Forecasting Performance

Table 1 is the primary quantitative result. It compares the patient-history baseline with the best RSV-only keyword–source configuration selected over the lag–horizon grid for each representative model. The comparison is interpreted within each model family, because the central question is not which backbone is globally strongest, but whether the same forecasting architecture can obtain a usable patient-count prediction when its input is restricted to RSV histories. When an RSV-only configuration approaches or improves the patient-history RMSE, it suggests that the corresponding search query contains temporal information about the upcoming Rhinovirus trajectory.

## 4.2. Keyword Analysis

The keyword analysis checks whether clinically plausible symptom terms co-vary with Rhinovirus counts and identifies which single RSV signals drive model performance. We report top-ranked keywords as a heatmap because useful queries are model-dependent: some backbones favor direct respiratory symptoms, while others select broader syndrome-related expressions. Consistent recurrence of a keyword family across models provides stronger evidence than a single best keyword from one architecture. Figure 3 ranks entries by the five-metric average rank, while using color only to show RMSE.

## 4.3. Qualitative Forecast Trajectories

Aggregate metrics summarize error but not outbreak timing. The trajectory analysis compares observed Rhinovirus counts with RSV-only predictions and patient-history predictions on the 2024 test interval. This analysis focuses on whether RSV-only models capture rises, peaks, and declines in the held-out outbreak trajectory. This qualitative view complements aggregate error: a model with similar RMSE can differ substantially in whether it anticipates the onset, misses the peak, or follows the decline after an outbreak.

# 5. Conclusion and Limitations

This retrospective Rhinovirus-focused study shows that selected online search trends contain predictive structure for patient-count trajectories in several model–keyword settings. The result motivates an early-warning workflow in which historical patient labels supervise model development, while prediction-time inputs are limited to search signals that may arrive closer to real time. Key limitations are platform-specific normalization, temporal drift in search behavior, retrospective keyword selection, and the current focus on single-keyword Rhinovirus forecasts in South Korea. Future work should validate the setting prospectively, extend it to multiple respiratory viruses and multi-keyword inputs, and test whether RSV-only forecasts support preparedness decisions under realistic reporting delays.

## References


Breiman, L. Random forests. *Machine Learning*, 45(1): 5–32, 2001.

Chen, S.-A., Li, C.-L., Arik, S. Ö., Yoder, N. C., and Pfister, T. TSMixer: An all-MLP architecture for time series forecasting. *Transactions on Machine Learning Research*, 2023.

Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pp. 1724–1734, 2014. doi: 10.3115/v1/D14-1179.

Das, A., Kong, W., Leach, A., Mathur, S. K., Sen, R., and Yu, R. Long-term forecasting with TiDE: Time-series dense encoder. *Transactions on Machine Learning Research*, 2023.

Ginsberg, J., Mohebbi, M. H., Patel, R. S., Brammer, L., Smolinski, M. S., and Brilliant, L. Detecting influenza epidemics using search engine query data. *Nature*, 457 (7232):1012–1014, 2009.

Gu, A. and Dao, T. Mamba: Linear-time sequence modeling with selective state spaces. In *First Conference on Language Modeling*, 2024.

Hochreiter, S. and Schmidhuber, J. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997. doi: 10.1162/neco.1997.9.8.1735.

Kitaev, N., Kaiser, L., and Levskaya, A. Reformer: The efficient transformer. In *International Conference on Learning Representations*, 2020.

Lazer, D., Kennedy, R., King, G., and Vespignani, A. The parable of Google Flu: traps in big data analysis. *Science*, 343(6176):1203–1205, 2014.

Lim, B., Arik, S. Ö., Loeff, N., and Pfister, T. Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, 37(4):1748–1764, 2021. doi: 10.1016/j.ijforecast.2021.03.012.

Liu, M., Zeng, A., Chen, M., Xu, Z., Lai, Q., Ma, L., and Xu, Q. SCINet: Time series modeling and forecasting with sample convolution and interaction. In *Advances in Neural Information Processing Systems*, volume 35, pp. 5816–5828, 2022.

Liu, Y., Li, C., Wang, J., and Long, M. Koopa: Learning non-stationary time series dynamics with Koopman predictors. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

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

Nie, Y., Nguyen, N. H., Sinthong, P., and Kalagnanam, J. A time series is worth 64 words: Long-term forecasting with transformers. In *International Conference on Learning Representations*, 2023.

Polgreen, P. M., Chen, Y., Pennock, D. M., and Nelson, F. D. Using Internet searches for influenza surveillance. *Clinical Infectious Diseases*, 47(11):1443–1448, 2008. doi: 10.1086/593098.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Woo, G., Liu, C., Sahoo, D., Kumar, A., and Hoi, S. C. H. ETSformer: Exponential smoothing transformers for time-series forecasting. *arXiv preprint arXiv:2202.01381*, 2022.

Zeng, A., Chen, M., Zhang, L., and Xu, Q. Are transformers effective for time series forecasting? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 11121–11128, 2023. doi: 10.1609/aaai.v37i9.26317.

Zhang, T., Zhang, Y., Cao, W., Bian, J., Yi, X., Zheng, S., and Li, J. Less is more: Fast multivariate time series forecasting with light sampling-oriented MLP structures. *arXiv preprint arXiv:2207.01186*, 2022.

Zhang, Y. and Yan, J. Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting. In *International Conference on Learning Representations*, 2023.

Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., and Zhang, W. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *AAAI Conference on Artificial Intelligence*, volume 35, pp. 11106–11115, 2021.

Zhou, T., Ma, Z., Wang, X., Wen, Q., Sun, L., Yao, T., Yin, W., and Jin, R. FiLM: Frequency improved legendre memory model for long-term time series forecasting. In *Advances in Neural Information Processing Systems*, volume 35, pp. 12677–12690, 2022a.

Zhou, T., Ma, Z., Wen, Q., Wang, X., Sun, L., and Jin, R. FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 27268–27286. PMLR, 2022b.

*Table 2.* Dataset statistics from the preprocessing artifacts. Candidate query rows are counted from the Korean-to-English mapping tables; retained columns are single-keyword RSV features after the $\rho \geq 0.3$ correlation threshold.

| Item | Value |
|---|---|
| Study period | 2019-10-27 to 2024-12-22 |
| Weekly KDCA patient records | 270 weeks |
| Patient series | Parainfluenza, Rhinovirus, respiratory syncytial virus |
| Median patient count (range) | Para: 42 (0–634); Rhino: 205.5 (22–915); respiratory syncytial virus: 57.5 (0–1078) |
| Unique LLM-expanded Korean query strings | 3,336 across three virus categories |
| Candidate query rows by virus | Para: 1,186; Rhino: 1,207; respiratory syncytial virus: 1,889 |
| Retained single-keyword RSV columns ($\rho \geq 0.3$) | Para: 89 Naver / 7 Google; Rhino: 97 Naver / 16 Google; respiratory syncytial virus: 83 Naver / 16 Google |

## A. Data Collection and Preprocessing Details

**KDCA patient-count table.** The patient-count table contains weekly counts for Parainfluenza, Rhinovirus, and respiratory syncytial virus from the KDCA acute respiratory infection surveillance system. KDCA reports these counts as surveillance-week totals ending on Saturday; after preprocessing, the merged table stores 270 aligned weekly rows spanning 2019-10-27 to 2024-12-22. These weekly patient counts define both the supervised target $y_t$ and the calendar bins used to aggregate or align Naver and Google RSV.

**Keyword table construction.** The keyword database starts from a structured table whose rows are KDCA-relevant symptom terms associated with a respiratory-virus category $v$ and a clinical symptom group $g$. For each seed symptom term, the same LLM expansion step is used to generate Korean synonyms, spelling variants, and spacing variants, which are then deduplicated. The resulting query table has schema

$$(v,\ g,\ q_{\text{seed}},\ q,\ \text{language},\ \mathcal{M}_q),$$

where $q$ is the final Korean query string and $\mathcal{M}_q$ is the set of search sources for which it is submitted. Across the three virus categories, this process yields 3,336 unique query strings; the Rhinovirus subset contains 1,207 candidate query strings per search source.

**Naver DataLab collection.** Naver RSV is collected with the Naver DataLab Search API endpoint `https://openapi.naver.com/v1/datalab/search`. For each keyword or keyword group, the request specifies `startDate`, `endDate`, `timeUnit="date"`, and a `keywordGroups` entry containing the Korean query terms. Authentication is provided through the Naver client identifier and secret headers. The API returns a date-indexed relative ratio series rather than raw query counts. Let $\tilde{x}^{\text{Naver}}_{q,\tau}$ be the daily ratio for query $q$ on calendar date $\tau$. For the KDCA surveillance week $W_t$, we define weekly Naver RSV as

$$x_t^{\text{Naver},q} = \frac{1}{|W_t|} \sum_{\tau \in W_t} \tilde{x}^{\text{Naver}}_{q,\tau}.$$

The resulting weekly series is reindexed to the 270 KDCA surveillance weeks before merging with patient counts.

**Google Trends collection.** Google RSV is collected from Google Trends using the Korean locale and South Korea as the geographic region. In the implementation, we use `pytrends` with `TrendReq(hl="ko-KR", tz=540)` and request each query through `build_payload` with `geo="KR"`, `cat=0`, and `gprop=""` for web search. Google Trends returns a normalized 0–100 interest series for each query over the requested window. The returned table is obtained with `interest_over_time()`. For the 270-week study period, Google returns weekly values directly; we remove the `isPartial` indicator, reindex the series to the KDCA week calendar, and store one RSV column per query. When spacing variants can be represented within the query-length limit, variants are grouped at collection time; otherwise, they are collected as separate query strings and handled during post-processing.

**Weekly alignment and database merge.** Both sources are converted to the same weekly index as the KDCA patient-count table. For each virus $v$, source $m$, and query $q$, the final aligned row for week $t$ contains

$$(v, t, m, q, g(q), x_t^{(v,m,q)}, y_t^{(v)}).$$

This long-format table is the canonical structured dataset. For forecasting, it is pivoted into the wide matrix $\mathbf{X}^{(v)} \in \mathbb{R}^{T \times P_v}$ and paired with $\mathbf{y}^{(v)} \in \mathbb{R}^T$. Naver and Google are not numerically pooled because their RSV scales are source-specific; instead, each source–query pair is treated as a separate feature column.

**Filtering and keyword selection.** A query is first retained only if it has nonzero RSV in at least five weeks:

$$\sum_{t=1}^{T} \mathbb{I}[x_t^{(v,m,q)} > 0] \geq 5.$$

Columns that are entirely missing or have no temporal variation are removed before correlation analysis and model fitting. For descriptive correlation screening, we compute Pearson correlation between each retained RSV column and the matched KDCA patient series and retain columns with $\rho \geq 0.3$ for visualization and keyword analysis. For Rhinovirus, this produces 16 Google and 97 Naver single-keyword columns at the $\rho \geq 0.3$ threshold. All forecasting windows are then constructed from the aligned weekly matrices as described in Section 3.1.

## B. Evaluated Forecasting Backbones

The retrospective benchmark evaluates 20 forecasting backbones covering four broad families. Transformer-style models include Transformer, Reformer, Informer, PatchTST, iTransformer, Crossformer, Temporal Fusion Transformer, FEDformer, and ETSformer (Vaswani et al., 2017; Kitaev et al., 2020; Zhou et al., 2021; Nie et al., 2023; Liu et al., 2024; Zhang & Yan, 2023; Lim et al., 2021; Zhou et al., 2022b; Woo et al., 2022). Linear, dense, and frequency-oriented models include DLinear, TiDE, LightTS, TSMixer, and FiLM (Zeng et al., 2023; Das et al., 2023; Zhang et al., 2022; Chen et al., 2023; Zhou et al., 2022a). State-space, Koopman, convolutional, and recurrent models include Mamba, Koopa, SCINet, LSTM, and GRU (Gu & Dao, 2024; Liu et al., 2023; 2022; Hochreiter & Schmidhuber, 1997; Cho et al., 2014). We also include Random Forest as a non-neural baseline (Breiman, 2001). The main text focuses on DLinear, iTransformer, Mamba, LSTM, and GRU because they span linear, Transformer, state-space, and recurrent forecasting families while keeping the short-paper analysis compact.

## C. Training Hyperparameters

**Common forecasting configuration.** All neural forecasting experiments use a lookback length of 48 weeks, label length of 48 weeks, patch length 1, time-feature embedding, one encoder layer, one decoder layer when applicable, and 8 attention heads when applicable. We evaluate prediction horizons $H \in \{1, 2, 4, 8\}$ and RSV lag shifts $s \in \{0, 1, 2, 3\}$. Each configuration is repeated over 10 random seeds. The maximum training epoch is 100 for applicable neural models, with validation-loss early stopping using a patience of 30 epochs. For the main Time-Series-Library configuration, the target column is the KDCA patient-count series; in RSV-only mode, `--use_feature_as_input` restricts the input to selected RSV feature columns while preserving the patient-count target.

**Evaluation range and ranking.** Metrics are computed on the held-out 2024 evaluation window used in the analysis figures. Because lag shifts and prediction horizons yield slightly different valid target timestamps, all reported metrics are computed only on the common overlapping target weeks, fixed to 2024-02-04 through 2024-11-03. For each model, keyword, lag, and horizon configuration, predictions from 10 random seeds are aggregated before computing RMSE, SMAPE, Pearson correlation, DTW, and MDA. Keyword rankings are computed from the average rank across these metrics, with RMSE/SMAPE/DTW minimized and Pearson/MDA maximized.

**Evaluation protocol and scope.** Patient-history baselines use the same chronological split, 48-week lookback, prediction horizons, random seeds, and evaluation window as the RSV-only runs; the lag grid applies only to RSV inputs. The correlation threshold is used as a retrospective feature screen in this feasibility study, not as a prospective deployment procedure. A prospective system should fix the query set and selection thresholds using only pre-evaluation data. The study uses aggregated weekly KDCA surveillance counts and platform-normalized aggregate RSV; no individual-level patient records or raw query logs are used.

*Table 3.* Model-specific optimizer and hidden-size settings used in the retrospective evaluation. N/A indicates non-neural baselines to which learning-rate schedules and hidden dimensions do not apply.

| Model family | Models | Learning rate | LR schedule | $(d_{\mathrm{model}}, d_{\mathrm{ff}})$ |
|---|---|---|---|---|
| Transformer | PatchTST, iTransformer, Crossformer, TemporalFusionTransformer | $1 \times 10^{-4}$ | type2 | $(512, 2048)$ |
| Linear/dense | FiLM, DLinear, TiDE, FEDformer, LightTS | $1 \times 10^{-3}$ | const | $(512, 2048)$ |
| RNN | GRU | $5 \times 10^{-3}$ | cosine | $(512, 2048)$ |
| RNN/conv. | LSTM, SCINet | $5 \times 10^{-4}$ | cosine | $(512, 2048)$ |
| SSM | Mamba | $5 \times 10^{-5}$ | cosine | $(64, 256)$ |
| Other neural | Koopa, Transformer, Reformer, ETSformer, TSMixer | $5 \times 10^{-5}$ | cosine | $(512, 2048)$ |
| Informer | Informer | $5 \times 10^{-6}$ | cosine | $(512, 2048)$ |
| Tree baseline | Random Forest | N/A | N/A | N/A |

## D. Rhinovirus Symptom-Query Lexicon

For the main Rhinovirus target, the table below lists the threshold-retained query expansion used to construct candidate RSV features. Each row gives the Korean root symptom term, its English symptom concept, and the derived Korean queries retained after the $\rho \geq 0.3$ correlation screen for Naver and Google separately. Derived Korean queries inherit the English concept of their root term rather than being assigned separate English translations.

**Rhinovirus symptom-query lexicon.** Listed derived queries are retained after the $\rho \geq 0.3$ screen. Derived Korean queries inherit the English concept of their root symptom term.

| Root Korean | English concept | Naver retained queries | Google retained queries |
|---|---|---|---|
| 발열 | fever | 열꽃 | – |
| 기침 | cough | 가래기침, 목감기, 기침발작, 심한기침, 콜록, 기침원인, 기침소리, 기침병, 지속적인기침, 알레르기성기침, 목기침, 기침멈추는법, 기침가래, 감기기침, 기침치료, 기침약, 기침심함, 기침멈추는방법, 기침감기, 기관지경련, 해소 | 가래기침, 목감기, 기침멈추는법, 기침가래, 감기기침, 기침감기, 감기 |
| 콧물 | rhinorrhea | 콧물염증, 콧물멈추는법, 콧물감기, 코염증, 감기콧물, 콧물약, 코질환, 코감기, 콧물계속, 콧물흐름, 물콧물, 맑은콧물, 감기초기증상 | 콧물감기, 감기콧물, 코감기, 콧물 |
| 가래 | sputum | 가래기침, 기침소리, 기침가래, 노란가래, 후비루, 기침할때가래, 가래섞인기침, 가래색깔, 가래감기, 진한가래, 가래배출, 가래덩어리, 가래끓는소리 | 가래기침, 기침가래, 후비루, 가래감기 |
| 인후통 | sore throat | 목감기, 목아픈이유, 목칼칼함, 후두염증, 목구멍, 후두염 | 목감기, 후두염 |
| 천명 | wheezing | 기침소리, 천식증상, 천식초기증상 | – |
| 쌕쌕거림 | wheezing | 천식증상, 기관지염증, 기관지소리, 천식소리, 폐질환증상 | – |
| 근육통 | myalgia | 근육경직, 근육통원인, 근육통증, 피로, 감기몸살 | 감기몸살 |
| 구토 | vomiting | 토함, 속토, 속이불편 | – |
| 컹컹 짖는 듯한 기침 | barking cough | 기침발작, 심한기침, 기침소리, 목기침, 감기기침, 숨쉴때기침, 쉿소리기침, 후두염증상, 깊은기침, 급성기침, 기침컹컹, 기침천식, 발작적기침, 후두염기침, 발작기침, 어린이기침 | 감기기침 |
| 설사 | diarrhea | 설사원인, 복부불편감, 설사멈추는법, 배아픔설사 | – |
| 크룹 | croup | 기침발작, 기침소리, 목기침, 후두염, 기침컹컹, 어린이기침, 크룹, 기도폐색, 컹컹거리는기침, 소아감기, 소아천식증상, 상기도감염, 아이기침, 소아후두염, 후두기관염, 크룹병, 기관지염, 밤에심한기침, 크룹증상, 급성후두염, 크루프, 어린이감기, 쉰목소리기침, 기관지부음 | 후두염, 기관지염 |