STM JOURNALS

Research & Reviews:
Discrete Mathematical Structures

ISSN: 2394-1979
Volume 13, Issue 1, 2026
January–April
DOI Journal: 10.37591/RRDMS

https://journals.stmjournals.com/rrdms

Review RRDMS

A SEIR-Informed Stacked Fusion of Prophet, XGBoost, and LSTM for Ward-Level Epidemic Forecasting in Amravati Municipal Corporation

Narendra J. Padole[1,*], Nitin S. Shrirao[2], Manish L. Jivtode[3]

Abstract

Municipal epidemic preparedness depends on accurate short-horizon forecasts at fine spatial granularity. Ward-level incidence series are typically nonstationary due to changing contact patterns, interventions, reporting delays, and heterogeneous demographic and environmental factors. This paper presents a mathematically formulated hybrid forecasting architecture designed for Amravati Municipal Corporation (AMC). The method decomposes observed incidence into (i) a mechanistic SEIR baseline that enforces epidemiological structure and (ii) a data-driven residual learned using Prophet (trend-seasonality decomposition), XGBoost (nonlinear covariate interactions), and LSTM (temporal memory). A constrained fusion estimator combines residual predictors on the probability simplex, improving stability and interpretability. We provide explicit model equations, parameter meanings, and optimization objectives, and we include experimental metrics extracted from the uploaded thesis evaluation for mathematics and discrete structures.

Keywords: Epidemic forecasting; SEIR; Hybrid modeling; Prophet; XGBoost; LSTM; Stacked fusion; Ward-level surveillance; Amravati.

INTRODUCTION

Urban local bodies require actionable predictions that support resource planning (testing, staffing, bed allocation, sanitation) and early-warning escalation. Compartmental models provide interpretable transmission dynamics but may not capture rapid behavioural changes, reporting artefacts, and nonlinear effects of covariates. Purely data-driven models can fit complex patterns but may lose epidemiological plausibility under distribution shift. A hybrid approach treats mechanistic modeling as a structured prior and uses machine learning to explain the remaining residual signal [1-11].

Contributions

- A baseline-plus-residual formulation linking SEIR dynamics to supervised residual learning.
- Explicit objectives for SEIR fitting, residual model learning, and constrained fusion weight estimation.
- Detailed interpretation of each equation and parameter, suitable for mathematical journal review.
- A deployment-aware pipeline for ward-wise inference compatible with cloud scaling.

PROBLEM FORMULATION AND NOTATION

Let $t = 1,2,\ldots,T$ index daily time steps and $w =$

***Author for Correspondence**
Narendra J. Padole
E-mail: njpadole@gmail.com

[1]Assistant Professor, Department of Computer Science and Technology, HVPM/DCPE, Amravati, Maharashtra, India
[2]Director, Siddhant Institute of Computer Application, Sudumbare, Pune, Maharashtra, India
[3]Professor, Janta mahavidyalaya, chandrapur, Maharashtra, India

Received Date: February 07, 2026
Accepted Date: February 26, 2026
Published Date: March 10, 2026

Citation: Narendra J. Padole, Nitin S. Shrirao, Manish L. Jivtode. A SEIR-Informed Stacked Fusion of Prophet, XGBoost, and LSTM for Ward-Level Epidemic Forecasting in Amravati Municipal Corporation. Research & Reviews: Discrete Mathematical Structures. 2026; 13(1): 17–23p.

$1,2,\ldots,W$ denote AMC wards. Let $y_{t,w} \geq 0$ be observed daily new cases (or incidence proxy) for ward w on day t, and let $x_{t,w} \in \mathrm{R}^d$ denote a covariate vector (meteorology, calendar effects, mobility proxies, demographic indicators, and lagged epidemiological features). For a forecast horizon $h \geq 1$, the aim is to produce $\hat{y}_{t+h,w}$ minimizing an empirical risk on a time-respecting evaluation window. [8], [15]

$$L(\hat{y}) = \frac{1}{|T_{\text{eval}}|\cdot W} \sum_{(t,w)\in T_{\text{eval}}\times\{1..W\}} \ell(y_{t,w}, \hat{y}_{t,w}) \tag{1}$$

Equation (1) is the ward-averaged loss over the evaluation period. The factor $\frac{1}{|T_{\text{eval}}|\cdot W}$ normalizes the total error so that results are comparable across different ward counts or test-window lengths. The function $\ell(\cdot,\cdot)$ can be chosen as squared error (for RMSE), absolute error (for MAE robustness), or a classification loss when predicting outbreak categories.

where $y_{t,w}$ denotes observed incidence at time t in ward w; $\hat{y}_{t,w}$ denotes predicted incidence; T_{eval} denotes evaluation time index set; $\ell(\cdot,\cdot)$ denotes loss function matching the prediction task.

MECHANISTIC BASELINE: WARD-WISE SEIR DYNAMICS

For each ward w, we use an SEIR compartmental model with state variables $S_w(t)$, $E_w(t)$, $I_w(t)$, $R_w(t)$ representing susceptible, exposed (infected but not yet infectious), infectious, and removed populations. Let $N_w = S_w + E_w + I_w + R_w$ denote the ward population (assumed constant over the forecast window). [1], [2], [6]

$$\frac{dS_w}{dt} = -\beta_w(t)\cdot\frac{S_w I_w}{N_w} \tag{2}$$

In Equation (2), the negative sign indicates that the susceptible compartment decreases as new infections occur. The product $S_w I_w$ approximates the number of potential encounters between susceptible and infectious individuals under homogeneous mixing; dividing by N_w normalizes by population size. The transmission rate $\beta_w(t)$ is allowed to vary over time to reflect interventions, seasonality, and behavioural change.

where $\beta_w(t)$ denotes time-varying transmission rate (effective contacts per unit time).

$$\frac{dE_w}{dt} = \beta_w(t)\cdot\frac{S_w I_w}{N_w} - \sigma_w E_w \tag{3}$$

Equation (3) increases E_w by the same infection flow that decreases S_w, and decreases E_w at rate σ_w as exposed individuals become infectious. Thus, $1/\sigma_w$ is the mean incubation period. The term $\sigma_w E_w$ is the flow from exposed to infectious.

where σ_w denotes progression rate from exposed to infectious (incubation inverse).

$$\frac{dI_w}{dt} = \sigma_w E_w - \gamma_w I_w \tag{4}$$

Equation (4) states that infectious individuals are gained from E_w at rate $\sigma_w E_w$ and removed from I_w at rate $\gamma_w I_w$. The removal term includes recovery, isolation, or any process that stops onward transmission. Hence $1/\gamma_w$ approximates the mean infectious period.

where γ_w denotes removal/recovery rate (infectious period inverse).

$$\frac{dR_w}{dt} = \gamma_w I_w \tag{5}$$

Equation (5) accumulates removed individuals. Since R_w does not feed back into new infections, it mainly serves to conserve population and to track cumulative outcomes under the model assumptions.

A standard interpretive quantity is the effective reproduction number:

$$R_{\mathrm{eff},w}(t) = \frac{\beta_w(t)}{\gamma_w} \cdot \frac{S_w(t)}{N_w} \tag{6}$$

Equation (6) expresses expected secondary infections per infectious individual at time t in ward w. When the susceptible fraction S_w/N_w declines, transmission potential reduces even if $\beta_w(t)$ is unchanged. When $R_{\mathrm{eff},w}(t) > 1$, infections tend to grow; when it is below 1, infections tend to shrink. [7]

Observation (Incidence) Mapping

Reported daily cases are linked to the latent SEIR trajectory using an observation equation. A common incidence proxy uses the flow from E to I and scales it by an ascertainment parameter ρ_w accounting for under-reporting and testing intensity.

$$\hat{y}_{t,w}^{SEIR} = \rho_w \cdot \sigma_w \cdot E_w(t) \tag{7}$$

In Equation (7), $\sigma_w E_w(t)$ is the model-implied number of individuals becoming infectious per unit time, and ρ_w maps this latent count to reported cases. When data are daily, $\hat{y}_{t,w}^{SEIR}$ may be interpreted as the expected number of new reported cases on day t. [15]

where ρ_w denotes ward-specific ascertainment/reporting rate.

HYBRID DECOMPOSITION: BASELINE + RESIDUAL

The mechanistic baseline captures structured transmission dynamics but may miss short-term deviations caused by interventions, mobility changes, weather shocks, or reporting artefacts. The hybrid strategy learns these deviations explicitly as residuals.

$$r_{t,w} = y_{t,w} - \hat{y}_{t,w}^{SEIR} \tag{8}$$

Equation (8) defines the residual series. If SEIR explains the coarse epidemic shape, $\boldsymbol{r_{t,w}}$ tends to have lower amplitude and more stationary behavior, which is beneficial for machine learning models.

$$\hat{y}_{t,w} = \hat{y}_{t,w}^{SEIR} + \hat{r}_{t,w} \tag{9}$$

Equation (9) reconstructs the final forecast by adding the predicted residual $\boldsymbol{\hat{r}_{t,w}}$ to the baseline. This preserves epidemiological plausibility while allowing flexible corrections. If needed, $\boldsymbol{\hat{y}_{t,w}}$ is truncated at zero to enforce nonnegativity.

RESIDUAL FORECASTING MODELS

Prophet (Trend + Seasonality)

Prophet models a time series as a sum of interpretable components. Applied to residuals, it primarily explains slow drifts and calendar-driven periodicity (e.g., weekly reporting cycles). [5], [8]

$$r_{t,w} = g_w(t) + s_w(t) + \varepsilon_{t,w} \tag{10}$$

In Equation (10), $g_w(t)$ is the trend component (often piecewise linear with change-points) and $s_w(t)$ captures seasonality. The noise $\varepsilon_{t,w}$ contains unexplained fluctuations.

$$s_w(t) = \sum_{k=1}^{K} \left[a_{w,k} \cos\left(\frac{2\pi kt}{P}\right) + b_{w,k} \sin\left(\frac{2\pi kt}{P}\right)\right] \tag{11}$$

Equation (11) uses a Fourier basis to represent seasonality with period P (e.g., $P = 7$ for weekly effects). Increasing K adds higher-frequency harmonics, allowing more complex seasonal shapes but increasing model variance.

where P denotes seasonal period (days); K denotes Fourier order controlling seasonality flexibility

XGBoost Residual Regression

XGBoost is a gradient-boosted tree ensemble suitable for tabular covariates. It learns nonlinear interactions (e.g., climate × density) and handles missing values via tree splits. [4], [9]

$$\hat{r}_{t,w}^{XGB} = \sum_{m=1}^{M} f_m\,(x_{t,w}), \text{ where } f_m \text{ are regression trees} \tag{12}$$

Equation (12) expresses the residual predictor as an additive model of M trees. Each tree f_m maps $x_{t,w}$ to a leaf score; the sum gives the final prediction.

$$\min_{f_1 \cdots f_M} \sum_{t,w} \ell\left(r_{t,w}, \sum_{m=1}^{M} f_m\,(x_{t,w})\right) + \sum_{m=1}^{M} \Omega(f_m) \tag{13}$$

Equation (13) is the regularized objective: the data-fit term encourages accurate residual prediction, while $\Omega(f_m)$ penalizes complexity to reduce overfitting.

$$\Omega(f) = \alpha \cdot (\#leaves) + \frac{\lambda}{2} \cdot \| w \|_2^2 \tag{14}$$

Equation (14) penalizes the number of leaves and the squared magnitude of leaf weights w. Larger α produces shallower trees; larger λ shrinks predictions, improving generalization.

where $x_{t,w}$ denotes covariate vector including lags and exogenous features; M denotes number of boosting trees.

LSTM Residual Sequence Model

LSTM networks model sequential dependence by maintaining a gated memory cell. Let z_t denote the input vector at time t (constructed from τ days of lags of residuals/covariates). The LSTM update equations are: [3]

$$i_t = \sigma(W_i z_t + U_i h_{t-1} + b_i), f_t = \sigma(W_f z_t + U_f h_{t-1} + b_f) \tag{15}$$

Equation (15) defines the input gate i_t (how much new information is written) and forget gate f_t (how much old memory is retained). The logistic function σ ensures gate values lie in $(0, 1)$.

$$o_t = \sigma(W_o z_t + U_o h_{t-1} + b_o), \hat{c}_t = \tanh\,(W_c z_t + U_c h_{t-1} + b_c) \tag{16}$$

Equation (16) defines the output gate o_t and candidate memory $\hat{c}_t$. The tanh nonlinearity bounds candidate values and stabilizes training.

$$c_t = f_t \odot c_{t-1} + i_t \odot \hat{c}_t, h_t = o_t \odot \tanh\,(c_t) \tag{17}$$

Equation (17) updates the cell state c_t using gated retention $(f_t \odot c_{t-1})$ and gated addition $(i_t \odot \hat{c}_t)$. The hidden state h_t is the exposed representation used for prediction. The operator $\odot$ is element-wise multiplication.

$$\hat{r}_{t,w}^{LSTM} = V h_t + b \tag{18}$$

Equation (18) maps the hidden state to a residual forecast through a linear layer. The parameters $\{W_*, U_*, b_*\}$ and $\{V, b\}$ are learned by minimizing a training loss over the residual series.

where $\sigma(\cdot)$ denotes logistic function; $\odot$ denotes Hadamard (element-wise) product; h_t denotes hidden state; c_t denotes cell state

CONSTRAINED FUSION OF RESIDUAL PREDICTORS

Each residual model specializes in a different structure. We combine them through a convex fusion so that the final residual remains stable and interpretable. [10], [11], [12]

$$\hat{r}_{t,w} = \alpha \cdot \hat{r}_{t,w}^{P} + \beta \cdot \hat{r}_{t,w}^{XGB} + \gamma \cdot \hat{r}_{t,w}^{LSTM} \tag{19}$$

Equation (19) is a weighted sum of component residual predictions. If α, β, γ are nonnegative and sum to one, the fused predictor cannot exceed the range implied by extreme component predictions (convex hull property).

$$\alpha \geq 0, \beta \geq 0, \gamma \geq 0, \text{and } \alpha + \beta + \gamma = 1 \tag{20}$$

Equation (20) is the simplex constraint. It also makes α, β, γ directly interpretable as mixture weights.

$$(\alpha^*, \beta^*, \gamma^*) = \arg \min_{(\alpha,\beta,\gamma)\in\Delta^2} \parallel R - \alpha R^P - \beta R^{XGB} - \gamma R^{LSTM} \parallel_2^2 \tag{21}$$

Equation (21) estimates fusion weights on a validation set by minimizing squared error between observed residuals $R$and the fused residual predictions. This is a small constrained quadratic program and is efficiently solvable.

where Rdenotes stacked vector of observed validation residuals; Δ^2denotes 2-simplex of nonnegative weights summing to one.

SEIR FITTING AND OVERALL TRAINING PIPELINE

Training is performed in three sequential steps:

(i) Estimate SEIR parameters,

(ii) Compute residuals and train residual models,

(iii) Fit fusion weights on a validation window.

This ordering reduces leakage and keeps the mechanistic part interpretable.

$$\min_{\theta_w} \sum_{t=1}^{T_{train}} \left(y_{t,w} - \rho_w \sigma_w E_w(t;\theta_w)\right)^2 + \lambda_\theta \Psi(\theta_w) \tag{22}$$

Equation (22) fits SEIR parameters θ_w(including initial states and $\beta_w(t)$parameterization) by minimizing squared deviation from observed incidence. The regularizer $\Psi(\theta_w)$constrains parameters to epidemiologically plausible ranges and prevents overfitting to noisy municipal reporting.

$$\beta_w(t) = \sum_{j=1}^{J} b_{w,j} \cdot 1_{t\in I_j}, \text{with } b_{w,j} \in [\beta_{min}, \beta_{max}] \tag{23}$$

Equation (23) is a practical parameterization for time-varying $\beta_w(t)$using Jintervals I_j(change points). It captures intervention phases while keeping estimation tractable

EVALUATION METRICS (REGRESSION AND EARLY-WARNING CLASSIFICATION)

The uploaded thesis reports MAE, RMSE, and classification metrics. For completeness, we state explicit formulas and interpret each metric operationally. [8], [13]

$$\text{MAE} = \frac{1}{(\mid T \mid \cdot W)} \sum_{t,w} \mid y_{t,w} - \hat{y}_{t,w} \mid \tag{24}$$

MAE in Equation (24) measures average absolute deviation. It is less sensitive to outliers than RMSE and is useful when occasional spikes occur.

$$\text{RMSE} = \sqrt{\frac{1}{(\mid T \mid \cdot W)} \sum_{t,w} \left(y_{t,w} - \hat{y}_{t,w}\right)^2} \tag{25}$$

RMSE in Equation (25) penalizes large errors more heavily, which is important when large forecasting misses could lead to insufficient preparedness.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \tag{26}$$

Accuracy in Equation (26) measures correct classification rate. Under class imbalance, precision and recall provide better operational insight.

$$\text{Precision} = \frac{TP}{TP+FP}, \text{Recall} = \frac{TP}{TP+FN} \tag{27}$$

Precision controls false alarms; recall controls missed outbreaks. AMC may prefer higher recall for early warning, subject to acceptable false-alarm costs.

$$F1 = 2 \cdot \frac{\text{Precision}\cdot\text{Recall}}{\text{Precision}+\text{Recall}} \tag{28}$$

F1 balances precision and recall as a harmonic mean

Table 1. Comparative performance metrics for evaluated models (from Thesis Table 6.2).

Model	MAE	RMSE	Accuracy
SEIR (Simplified)	0.0979	0.1345	0.5385
Prophet Model	0.0578	0.0652	1.0000
XGBoost	0.1220	0.2565	0.8750
LSTM	0.4734	0.4864	0.6184
Hybrid (XGBoost + LSTM)	0.2614	0.2980	0.8805
Hybrid (XGBoost + LSTM + Logistic Meta-Learner)	0.2602	0.2965	0.8821
Hybrid (Prophet + XGBoost + LSTM) (Stacked Fusion)	0.1262	0.2542	0.8803
Random Forest	0.1195	0.2500	0.8794

EXPERIMENTAL RESULTS (FILLED FROM UPLOADED THESIS TABLE 6.2)

Table 1 lists key models and their MAE, RMSE, and Accuracy exactly as reported in the uploaded thesis evaluation. The stacked fusion model (Prophet + XGBoost + LSTM) demonstrates competitive error and strong classification behavior compared to individual models.

Interpretation: Prophet shows extremely low RMSE in the thesis table, reflecting strong fit to trend/seasonality in the evaluated setting; however, its R^2 can be negative in nonstationary series, indicating that absolute fit does not always imply variance-explained improvement. The stacked fusion hybrid maintains strong accuracy and balanced error, benefiting from complementary components and constrained fusion.

From a municipal deployment perspective, the hybrid architecture is advantageous because it is modular: SEIR parameters can be updated as epidemiological assumptions change, while residual learners can be retrained more frequently as new covariates arrive or reporting patterns shift.

CLOUD DEPLOYMENT CONSIDERATIONS

For AMC-scale inference across wards, parallelization is essential. If $\tau_{SEIR}, \tau_P, \tau_{XGB}, \tau_{LSTM}$ are per-ward inference times and B is the number of parallel workers, then the end-to-end latency per forecast request can be approximated by: [14]

$$\text{Latency} \approx \left(\frac{W}{B}\right) \cdot (\tau_{SEIR} + \tau_P + \tau_{XGB} + \tau_{LSTM}) + \tau_{IO} \tag{29}$$

Equation (29) implies near-linear speedup with B until I/O dominates. Practical deployment should therefore use ward-level parallel jobs and caching of recent feature windows.

CONCLUSION

We presented a mathematical hybrid forecasting framework for ward-level epidemic prediction for Amravati Municipal Corporation. The approach uses an SEIR baseline to enforce epidemiological structure, learns residual dynamics using Prophet, XGBoost, and LSTM, and combines residual predictions using constrained simplex fusion. The paper provides explicit equations and detailed interpretation of each term to support mathematical journal review and reproducibility. Future work includes inter-ward coupling, uncertainty quantification, and multi-disease extensions.

REFERENCES

1. W.O. Kermack and A. G. McKendrick, “A contribution to the mathematical theory of epidemics,” Proc. Royal Soc. A, 1927.
2. H. W. Hethcote, “The mathematics of infectious diseases,” SIAM Review, 2000.
3. S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural Computation, 1997.
4. T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” Proc. KDD, 2016.
5. S. J. Taylor and B. Letham, “Forecasting at scale,” The American Statistician, 2018.
6. R. M. Anderson and R. M. May, Infectious Diseases of Humans: Dynamics and Control. Oxford,

U.K.: Oxford Univ. Press, 1991.
7. O. Diekmann, J. A. P. Heesterbeek, and M. G. Roberts, “The construction of next-generation matrices for compartmental epidemic models,” J. R. Soc. Interface, 2010.
8. R. J. Hyndman and G. Athanasopoulos, Forecasting: Principles and Practice, 2nd ed. Melbourne, Australia: OTexts, 2018.
9. J. H. Friedman, “Greedy function approximation: A gradient boosting machine,” Ann. Stat., 2001.
10. D. H. Wolpert, “Stacked generalization,” Neural Networks, 1992.
11. L. Breiman, “Stacked regressions,” Machine Learning, 1996.
12. S. Boyd and L. Vandenberghe, Convex Optimization. Cambridge, U.K.: Cambridge Univ. Press, 2004.
13. T. Fawcett, “An introduction to ROC analysis,” Pattern Recognition Letters, 2006.
14. J. Dean and S. Ghemawat, “MapReduce: Simplified data processing on large clusters,” Commun. ACM, 2008.
15. A. Höhle, “surveillance: An R package for the monitoring of infectious diseases,” Comput. Stat., 2007.

