ExoC2T: An exogenous-driven spatio-temporal learning framework for cross-city transfer

Hailong Yu[1] **Zhengyang Zhou**[1,2*] **Liwen Zhang**[1] **Qihe Huang**[1]
Kuo Yang[1] **Yudong Zhang**[1,2] **Yang Wang**[1,2,*]
[1]University of Science and Technology of China (USTC), Hefei, Anhui, China
[2]Suzhou Institute for Advanced Research, USTC, Suzhou, Jiangsu, China
{yuhlong,zhangliwen,hqh,yangkuo}@mail.ustc.edu.cn
{zzy0929,yudong.zhang,angyan}@ustc.edu.cn

Abstract

Urban traffic forecasting supports infrastructure planning and resource allocation, yet dense traffic sensing remains costly for many cities. Existing cross-city methods and spatio-temporal foundation models typically require recent target-city traffic observations. We study exogenous-driven cross-city transfer, predicting traffic from static and dynamic urban context with limited offline target labels and no historical flow input at query time. The challenge is to infer predictive traffic-state information from external evidence while separating transferable responses from city-specific variation. We propose **ExoC2T**, a plasticity-inspired spatio-temporal learning framework with two coupled components. The Exogenous-Infused Spatio-Temporal Network (**EXIST**) combines semantic spatial context, temporal regimes, and weather-conditioned expert responses in shared prototype coordinates. Stability-Plasticity Adaptation (**SPAda**) preserves a stable exogenous-to-flow core and regulates adaptive responses through a support-conditioned latent posterior, invariant discrepancy regularization, and posterior correction. Experiments on a nine-city benchmark show consistent improvements over strong forecasting and transfer baselines, supporting exogenous-driven prediction for cities with limited traffic sensing.

1 Introduction

Urban traffic forecasting is fundamental to infrastructure optimization, anomaly detection, and resource allocation [19, 58, 53, 3]. Yet the deployment and long-term maintenance of dense sensing systems remain prohibitively expensive for many cities, making large-scale traffic observation an unrealistic prerequisite for practical urban prediction. For example, deploying only 100 traffic cameras or loop detectors can cost roughly $500K and $300K, respectively [6]. This challenge is especially acute in under-instrumented target cities, newly monitored regions, and other settings where reliable historical traffic records are scarce and explicit spatial graphs may be unavailable. A practical cross-city predictor must therefore operate even when the target environment cannot satisfy the sensing assumptions built into conventional flow-driven models.

Most existing forecasting, cross-city transfer, and recent spatio-temporal foundation-model approaches are formulated around historical traffic observations [19, 58, 53, 15, 40, 55, 27, 3, 52, 12, 24, 20]. These models usually rely on recent target-side traffic values as direct observations of the current endogenous traffic state and extrapolate future flow from this observed state. Cross-city transfer is then achieved by adapting source-trained flow dynamics, region correspondences, graph structures, or meta-learned initialization to the target city, while static urban context and external

*Corresponding authors: Zhengyang Zhou (zzy0929@ustc.edu.cn) and Yang Wang (angyan@ustc.edu.cn).

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

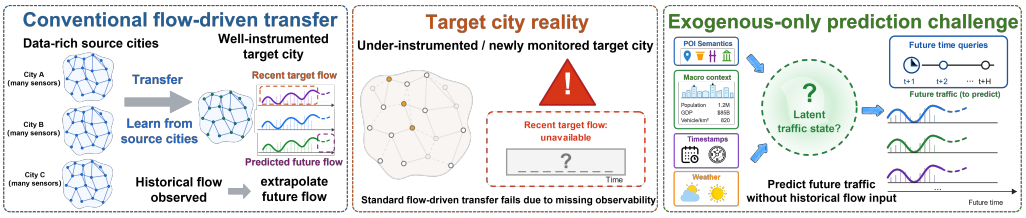


Figure 1: Exogenous-only cross-city transfer challenge. Conventional cross-city transfer assumes recent target-side traffic observations as prediction context, whereas data-scarce target cities may lack such query-time flow evidence. ExoC2T addresses the resulting setting by recovering latent traffic-state information from static and dynamic exogenous urban context and predicting future traffic without recent target flow input.

signals mainly play auxiliary roles [15, 40, 8, 13]. This formulation is effective when target-side traffic histories are available, but it is poorly matched to cities where traffic flow cannot serve as the prediction context.

We study exogenous-driven cross-city transfer, where traffic is predicted from static and dynamic exogenous urban context without historical flow input. Under exogenous-only observability, recent flow no longer provides the observed state for extrapolation, so the model must infer traffic-state information from partial external evidence and transfer the exogenous-to-flow response across cities. This becomes difficult because similar urban functions, temporal regimes, and environmental conditions can induce different traffic responses in different cities, while the target adaptation set is too small to support unrestricted model updates. The central problem is therefore to form a shared mechanism space that makes indirect evidence predictive, separates transferable response structure from local shift, and permits target adaptation without destabilizing the learned structure. Figure 1 illustrates the resulting gap between conventional flow-driven transfer and the exogenous-only prediction challenge addressed in this paper.

We propose **ExoC2T**, which couples representation learning with target adaptation. The Exogenous-Infused Spatio-Temporal Network (**EXIST**) extracts semantic spatial context, temporal regimes, and environment-conditioned responses, then organizes them through prototype-routed expert memory. Stability-Plasticity Adaptation (**SPAda**) combines a stable exogenous-to-flow core with adaptive responses regulated by a support-conditioned posterior and invariant discrepancy regularization. Homeostatic plasticity and metaplasticity motivate this separation between preserving shared responses and regulating target-specific change [41, 2, 1].

Our contributions are summarized as follows. **First**, we formalize exogenous-driven cross-city transfer as a dynamic node-level forecasting setting for data-scarce target cities, where prediction is performed without recent traffic-state observations at query time. This setting targets traffic forecasting in cities with limited sensing infrastructure by reducing dependence on dense traffic sensors. **Second**, we introduce EXIST, a shared exogenous mechanism representation that organizes semantic relations, temporal regimes, and environment-conditioned responses in shared prototype coordinates for stable-adaptive forecasting. **Third**, we develop SPAda, which combines stable-adaptive mechanism separation, support-conditioned latent posterior inference, and discrepancy-regularized target-risk minimization within a two-stage learning pipeline. Extensive experiments on a multi-city benchmark show that, in data-scarce target cities, ExoC2T consistently outperforms strong flow-input forecasting and transfer baselines while using only exogenous inputs at prediction time.

2 Preliminary

This section formalizes the exogenous-driven cross-city transfer setting addressed by ExoC2T. We adopt a graph-theoretic view of urban systems. The forward model receives static and dynamic exogenous urban context, and historical traffic flow is excluded from the query input.

Definition 1 (Urban Context Graph). We represent an urban system as a graph $\mathcal{G} = (\mathcal{V}, \mathbf{A})$, where $\mathcal{V} = \{v_1, \ldots, v_N\}$ denotes a set of N traffic regions or sensors, and $\mathbf{A} \in \mathbb{R}^{N \times N}$ encodes spatial or

semantic relations between nodes. When an explicit adjacency matrix is unavailable, $\mathbf{A}$ is replaced by a semantic prior graph $\mathbf{A}^{sem}$ constructed from urban semantics, enabling graph-missing cities to be handled within the same framework.

Definition 2 (Static and Dynamic Urban Signals). Let $\mathcal{Y} \in \mathbb{R}^{H\times N}$ denote the target traffic sequence over a forecast horizon of length H and N traffic nodes. For city c, the static input consists of a node-level semantic matrix $\mathbf{P} = [\mathbf{p}_1, \ldots, \mathbf{p}_N]^\top \in \mathbb{R}^{N\times d_p}$ together with a macro context vector $\mathbf{m}_c \in \mathbb{R}^{d_m}$. The dynamic input consists of timestamp features $\mathbf{T}_{1:S} \in \mathbb{R}^{S\times d_\tau}$ and weather features $\mathbf{W}_{1:S} \in \mathbb{R}^{S\times d_w}$ over the observation horizon of length S, together with future timestamp queries $\mathbf{T}^{+}_{1:H} \in \mathbb{R}^{H\times d_\tau}$ over the forecast horizon. These variables provide the transferable context used to infer endogenous traffic dynamics at prediction time.

Problem 1 (Exogenous-driven Cross-city Transfer). Given a set of data-rich source cities $\mathcal{C}_S$ and a data-scarce target city d_t, let $\mathcal{D}_S = \{D_d\}_{d\in\mathcal{C}_S}$ denote the source-city data, where each D_d contains labeled exogenous-context/traffic-target pairs from city d. The target city provides a limited offline adaptation set $\mathcal{S}_t = \{(x_j, y_j)\}_{j=1}^{K_t}$ and a query set $\mathcal{Q}_t = \{(x_q, y_q)\}$ used for evaluation. Here $K_t = |\mathcal{S}_t|$ counts labeled support examples, independently of the prototype count K below. The query input contains only exogenous urban context and forecast-horizon timestamp queries, and excludes recent historical traffic flow. The objective is to learn a transferable forecasting model

$$f_t(\cdot) = f(\cdot; \mathcal{D}_S, \mathcal{S}_t, d_t), \tag{1}$$

where the target-city input is

$$x_t = \left(\mathbf{P}_t, \mathbf{m}_t, \mathbf{T}_{1:S}, \mathbf{W}_{1:S}, \mathbf{T}^{+}_{1:H}, \mathbf{A}_t\right), \tag{2}$$

and the learning objective is

$$\min_f \ \mathbb{E}_{(d_t,\mathcal{S}_t,\mathcal{Q}_t)}\left[R_{\mathcal{Q}_t}(f_t)\right]. \tag{3}$$

Here $\mathbf{P}_t$, $\mathbf{m}_t$, and $\mathbf{A}_t$ denote the semantic matrix, macro context, and adjacency matrix of target city d_t, respectively. The quantity $R_{\mathcal{Q}_t}(f_t)$ denotes prediction risk on the target-city query set, and f_t is the predictor instantiated for target city d_t after adaptation. The goal is to recover transferable exogenous-to-flow mechanisms that support target-city prediction without recent traffic observations as query input.

A standard state-space view of traffic prediction posits an unobserved traffic-generating state that governs the evolution of future flow [17, 44, 37]. Following this view, we let u_c denote the latent traffic-generating state of city c and model the city-level data generation process as

$$x_c \sim p(x \mid c), \qquad u_c \sim p(u \mid x_c, c), \qquad y_c \sim p(y \mid u_c, x_c, c), \tag{4}$$

where x_c denotes the observable exogenous context and y_c denotes the target traffic sequence. Under this formulation, exogenous urban signals condition the latent state that drives traffic generation. The statistical challenge is therefore to recover traffic-state information from partial exogenous evidence under city shift. This motivates a mechanism-centered formulation in which external context provides an observable proxy for the latent traffic state, while the target adaptation set identifies city-specific modulation around a transferable exogenous-to-flow core.

3 Methodology

3.1 Framework overview

ExoC2T predicts target-city traffic from exogenous urban context when recent traffic flow is unavailable at query time. Given query context x_t, source-city supervision $\mathcal{D}_S$, and a limited target adaptation set $\mathcal{S}_t$, the framework first uses EXIST to organize external evidence into exogenous mechanism states and then applies SPAda to preserve a stable exogenous-to-flow core while modulating target-conditioned deviation. The final prediction is decomposed into stable and adaptive components,

$$\mathbf{H}_{1:S} = \mathrm{EXIST}(x_t), \qquad \hat{\mathcal{Y}}_t = f_\Theta(x_t; \mathcal{D}_S, \mathcal{S}_t, d_t) = \hat{\mathcal{Y}}_t^{\mathrm{inv}} + \hat{\mathcal{Y}}_t^{\mathrm{var}}. \tag{5}$$

In Eq. (5), f_Θ produces $f_t^{\mathrm{stat}}(x_t)$; the notation $\mathrm{EXIST}(x_t)$ suppresses the fixed target-support conditioning. Here Θ denotes the full framework parameter collection; θ below denotes the stable/adaptive parameter blocks used in the adaptation analysis. Each $\mathbf{H}_{i,s}^{(t)} \in \mathbb{R}^{d_h}$ is a node-time feature column; $\mathbf{H}_{1:S}$ collects these features over S steps and N nodes (time, node, feature axes). Section 3.2

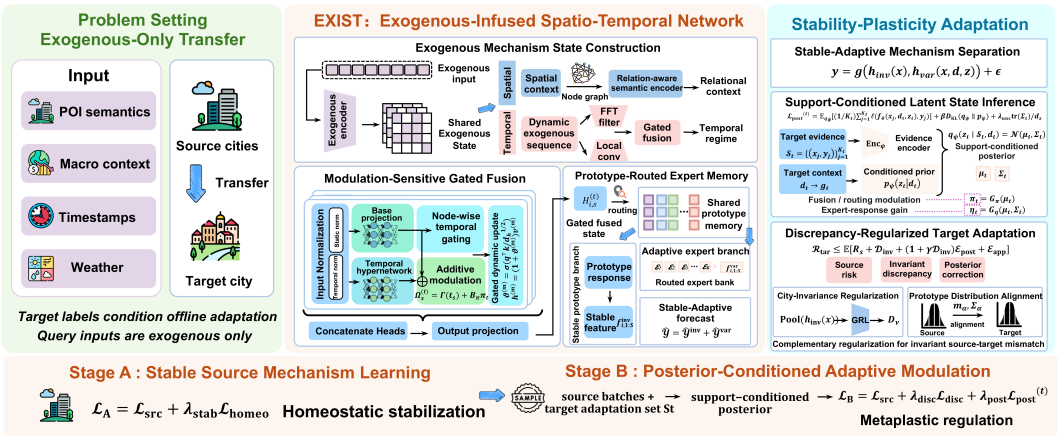


Figure 2: Overall framework of ExoC2T. The model predicts target-city traffic from exogenous urban context without recent flow input. EXIST builds a shared exogenous mechanism representation through three coupled steps: exogenous mechanism state construction, modulation-sensitive gated fusion, and prototype-routed expert memory. SPAda then performs Stage A: Stable Source Mechanism Learning and Stage B: Posterior-Conditioned Adaptive Modulation.

details how EXIST constructs the shared exogenous mechanism representation, and Section 3.3 describes how the target adaptation set regulates stable-adaptive transfer. Figure 2 summarizes the architecture and two-stage learning pipeline.

3.2 Exogenous-Infused Spatio-Temporal Network (EXIST)

Under exogenous-only observability, recent traffic flow is unavailable as direct query-time evidence of the traffic state. EXIST converts heterogeneous external urban context into a shared mechanism space for recovering traffic-state information from indirect external evidence. It first constructs exogenous mechanism states by combining observable external evidence with relational and temporal contexts, then transforms these states through a modulation-sensitive gated fusion rule, and finally organizes the resulting fused states with a prototype-routed expert memory.

Exogenous mechanism state construction. EXIST first builds an exogenous mechanism state as an observable proxy for the latent traffic state. We denote semantic, node, city, timestamp, weather, and forecast-query context by $\mathbf{P}$, $\boldsymbol{\rho}$, $\mathbf{m}_c$, $\mathbf{T}_{1:S}$, $\mathbf{W}_{1:S}$, and $\mathbf{T}^{+}_{1:H}$, respectively. Each $\boldsymbol{\rho}_i \in \mathbb{R}^{d_\rho}$ is a deterministic, parameter-free sinusoidal encoding of node i in the city's fixed node ordering. Heterogeneous external variables are projected into static, dynamic, and city-level exogenous states,

$$\mathbf{E}^s, \mathbf{E}^d_{1:S}, \mathbf{g}_c = \mathcal{P}_{\text{exo}}(\mathbf{P}, \boldsymbol{\rho}, \mathbf{m}_c, \mathbf{T}_{1:S}, \mathbf{W}_{1:S}; c). \tag{6}$$

Here $\mathbf{E}^s$ stores node-level semantic evidence, $\mathbf{E}^d_{1:S}$ stores timestamp-weather evidence over the observation window, and $\mathbf{g}_c$ encodes city-level macro context. We use feature columns and row-stacked sequences: $\mathbf{E}^s \in \mathbb{R}^{N\times d_s}$ and $\mathbf{E}^d_{1:S} \in \mathbb{R}^{S\times d_d}$, with feature widths d_s, d_d; $\mathbf{g}_c \in \mathbb{R}^{d_g}$ has macro-context width d_g. The projection uses feature-conditioned node-basis retrieval and city modulation, which make the state construction independent of learned city-specific node-ID tables and compatible with different target graph sizes; the expanded tensor form is given in Appendix C.1.

The projected state still mixes factors with different relevance to traffic generation. EXIST therefore extracts a semantic graph-conditioned spatial context and a spectral-local temporal context. Since traffic-flow correlation is unavailable as an input signal, the spatial context is derived from urban semantics, node-state similarity, and available graph support,

$$\mathbf{A}^\star = \text{Graph}_{\text{sem}}(\mathbf{E}^s, \mathbf{P}, \mathbf{g}_c, \mathbf{A}), \qquad \mathbf{s}_i = \text{Attn}_{\mathbf{A}^\star}\big(\mathbf{e}^s_i, \{\mathbf{e}^s_j : j \in \mathcal{N}_i(\mathbf{A}^\star)\}, \mathbf{g}_c\big). \tag{7}$$

The operator $\text{Graph}_{\text{sem}}$ returns the observed graph when available and otherwise constructs a semantic prior graph; $\text{Attn}_{\mathbf{A}^\star}$ produces the spatial mechanism context $\mathbf{s}_i$. Here $(\mathbf{e}^s_i)^\top$ is row i of

$\mathbf{E}^s$, $\mathcal{N}_i(\mathbf{A}^\star)$ is its graph-supported neighbor set, and $\mathbf{s}_i \in \mathbb{R}^{d_h}$. Detailed graph construction and relation-aware attention are provided in Appendix C.2.

On the temporal side, exogenous dynamics contain recurrent regimes and local disturbances. EXIST decomposes the dynamic exogenous state into spectral and local components,

$$\mathbf{t}_{1:S} = \omega\, \mathcal{F}^{-1}\big(\mathcal{F}(\mathbf{E}^d_{1:S}) \odot \mathcal{K}\big) + (1-\omega)\mathrm{Local}(\mathbf{E}^d_{1:S}). \tag{8}$$

Here $\mathcal{F}$ and $\mathcal{F}^{-1}$ act along the S-step temporal axis, $\odot$ is elementwise multiplication, and $\mathcal{K}$ is a shape-compatible spectral filter. Both branches return $S \times d_d$ states, with $\mathbf{t}_s \in \mathbb{R}^{d_d}$. The local temporal operator preserves this shape, and $\omega \in (0,1)$ is an input-dependent scalar gate defined in Appendix C.2. The resulting pair $(\mathbf{s}_i, \mathbf{t}_{1:S})$ forms the exogenous mechanism state used by modulation-sensitive gated fusion and prototype-routed memory. Appendix C.2 provides the expanded operator-level form of the context extraction stage.

Modulation-sensitive gated fusion. Without recent flow trends, the current operating state must be inferred from temporal and environmental context. EXIST therefore places modulation inside a gated fusion rule, allowing temporal regimes and the support-conditioned latent state to control how spatial semantic states are activated into predictive fused states,

$$\mathbf{H}^{(t)}_{i,s} = \mathcal{G}_{\mathrm{mod}}\left(\mathbf{s}_i, \mathbf{t}_s; \Gamma(\mathbf{t}_s) + \mathbf{B}_\pi \boldsymbol{\pi}_t\right). \tag{9}$$

Here $\mathcal{G}_{\mathrm{mod}}$ denotes the modulation-sensitive gated fusion operator, $\Gamma(\mathbf{t}_s)$ generates temporal gating perturbations, and $\boldsymbol{\pi}_t$ is induced by the support-conditioned latent state inferred from the target adaptation set in Section 3.3.2. Writing d_π for the width of $\boldsymbol{\pi}_t$ and d_Ω for the perturbation width, $\Gamma : \mathbb{R}^{d_d} \to \mathbb{R}^{d_\Omega}$ and $\mathbf{B}_\pi \in \mathbb{R}^{d_\Omega \times d_\pi}$ map into the same gating space. The superscript (t) identifies the target task, while s indexes time. The target adaptation set modulates the shared fused states by changing how exogenous context is activated in this layer. Appendix C.3 provides the expanded operator form of this layer.

Prototype-routed expert memory. The modulation-sensitive fused states must support both cross-city comparison and city-conditioned response variation. EXIST organizes these states with a shared prototype-routed expert memory, where prototype anchors define the shared coordinate system and routed experts parameterize localized deviations around it,

$$\mathcal{M} = \{(\mathbf{c}_k, \mathcal{E}_k)\}_{k=1}^{K}, \qquad \mathbf{C} = [\mathbf{c}_1, \ldots, \mathbf{c}_K]^\top. \tag{10}$$

Here K is the number of prototype–expert pairs, $\mathbf{c}_k \in \mathbb{R}^{d_c}$ is a prototype anchor, and $\mathcal{E}_k : \mathbb{R}^{d_h} \to \mathbb{R}^{d_h}$ is its learnable low-rank nonlinear expert; $\mathbf{C} \in \mathbb{R}^{K \times d_c}$ stacks the anchors. The stable component reads prototype coordinates, while the adaptive component combines prototype routing with weather- and modulation-conditioned expert responses. We summarize the component-specific memory readout as

$$\mathbf{b}^{\mathrm{inv}}_{i,s}, \mathbf{b}^{\mathrm{var}}_{i,s} = \mathcal{R}_{\mathcal{M}}\left(\mathbf{H}^{(t)}_{i,s}, \mathbf{C}, \boldsymbol{\pi}_t, \boldsymbol{\eta}_t, \mathbf{u}_s\right), \qquad \mathbf{u}_s = \Psi_w(\mathbf{W}_s), \tag{11}$$

where $\mathcal{R}_{\mathcal{M}}$ denotes prototype routing and expert response generation, and $\boldsymbol{\eta}_t \in \mathbb{R}^K$ is the posterior-generated expert-response gain. This vector is distinct from the parameter-update matrix $\mathbf{M}_t$ in Section 3.3.1. The responses have widths d_c and d_h, respectively; Ψ_w maps weather features to $\mathbf{u}_s \in \mathbb{R}^{d_u}$, shared across nodes at step s. Both branches receive the shared modulated states. The stable memory response remains anchored to prototype coordinates, while the adaptive memory response additionally uses weather-conditioned routing and expert responses together with posterior routing shifts and expert gains. Detailed Entmax routing, weather-controlled expert responses, and component injection are provided in Appendix C.4.

After memory responses are injected, forecasting keeps the node dimension throughout the readout. Let $\mathbf{f}^{\mathrm{inv}}_{i,1:S}$ and $\mathbf{f}^{\mathrm{var}}_{i,1:S}$ denote the stable and adaptive node-wise feature sequences. EXIST produces stable and adaptive forecasts with node-shared readouts,

$$\hat{\mathbf{y}}_{i,1:H} = \hat{\mathbf{y}}^{\mathrm{inv}}_{i,1:H} + \hat{\mathbf{y}}^{\mathrm{var}}_{i,1:H}, \qquad \hat{\mathbf{y}}^{\mathrm{inv}}_{i,1:H} = \mathcal{F}_{\mathrm{sta}}(\mathbf{f}^{\mathrm{inv}}_{i,1:S}, \mathbf{T}^{+}_{1:H}), \quad \hat{\mathbf{y}}^{\mathrm{var}}_{i,1:H} = \mathcal{F}_{\mathrm{ada}}(\mathbf{f}^{\mathrm{var}}_{i,1:S}, \mathbf{u}_{1:S}). \tag{12}$$

The weather-controller sequence $\mathbf{u}_{1:S}$ is an explicit input of the adaptive readout, as expanded in Appendix C.4. Because the same readout functions are applied to every node, the model produces $\hat{\mathcal{Y}} \in \mathbb{R}^{H \times N}$ while remaining valid for cities with different graph sizes. The final forecast combines a stable component anchored by prototype coordinates with an adaptive component produced by routed experts. Appendix C.4 gives the operator-level formulation of the memory responses and the stable-adaptive forecasting interface.

3.3 Stability-Plasticity Adaptation (SPAda)

Once EXIST has mapped external urban context into a shared mechanism space, target adaptation uses $\mathcal{S}_t$ to specialize the adaptive component while stable exogenous-to-flow structure is retained. SPAda assigns these roles to two coupled mechanisms. Homeostatic stabilization constrains the stable component, and metaplastic regulation controls posterior-dependent adaptive gain. Appendix D gives the local perturbation analysis for stable-adaptive mechanism separation and the plasticity-separation bound, Appendix E details support-conditioned latent state inference and adaptive gain, and Appendix F formalizes the target-risk decomposition with discrepancy regularization.

3.3.1 Stable-adaptive mechanism separation

Without historical traffic flow as query-time evidence, exogenous-driven transfer relies on stable exogenous-to-flow regularities that remain valid beneath urban variation. A limited target adaptation set makes unconstrained adaptation unreliable because target gradients can entangle transferable response structure with local shift. ExoC2T therefore separates the adaptation pathways within the shared mechanism space formed by EXIST. The stable component represents the exogenous-to-flow core, while the adaptive component carries city-conditioned deviation around that core. Within the prototype-routed memory, the stable component is parameterized by prototype coordinates, while the adaptive component is driven by routed expert responses.

This stable-adaptive split admits a plasticity interpretation. Homeostatic plasticity stabilizes functional responses, whereas metaplasticity regulates how adaptation is expressed [41, 49, 56, 2, 1]. In our setting, homeostatic stabilization is assigned to the stable component and metaplastic regulation is assigned to the adaptive component. We therefore use the following local stable-adaptive approximation of traffic generation:

$$y = g\big(h_{\mathrm{inv}}(x),\, h_{\mathrm{var}}(x, d, z)\big) + \epsilon, \tag{13}$$

Here x, y denote an exogenous input and its traffic target, d is the city, g is the response map, and ϵ is additive observation noise in the target space. The map h_{inv} denotes the stable predictive core and h_{var} denotes city- and state-conditioned modulation. The latent state z is later instantiated by the support-conditioned posterior in Section 3.3.2. The corresponding local mechanism has the first-order form

$$y^\star \approx y^{\mathrm{inv}}_{\mathrm{loc}} + y^{\mathrm{var}}_{\mathrm{loc}}, \qquad y^{\mathrm{inv}}_{\mathrm{loc}} = g\big(h_{\mathrm{inv}}(x), 0\big), \tag{14}$$

where $y^{\mathrm{inv}}_{\mathrm{loc}}$ is the zeroth-order mechanism term and $y^{\mathrm{var}}_{\mathrm{loc}}$ is its first-order adaptive term. Appendix D shows that this additive form corresponds to a first-order expansion around the stable component, with a residual that is second order in the adaptive component. Here $y^\star = g(h_{\mathrm{inv}}(x), h_{\mathrm{var}}(x, d, z))$ is the noise-free local mechanism. The learned heads in Section 3.2 implement the additive structure motivated by this local expansion.

This structural split induces a homeostatic regularization principle on the stable component. Let $\phi(x) = h_{\mathrm{inv}}(x)$ and let w denote the predictor acting on $\phi(x)$. We define

$$\mathcal{L}_{\mathrm{homeo}} = \sum_{c \in \mathcal{C}_S} \big\|\nabla_w R^{\mathrm{inv}}_c(w, \phi)\big\|^2, \qquad R^{\mathrm{inv}}_c(w, \phi) = \mathbb{E}_{(x,y)\sim P_c}\big[\ell\big(w(\phi(x)), y\big)\big], \tag{15}$$

Here P_c is the joint input–target distribution in source city c, ℓ is the prediction loss, and the gradient uses the Euclidean norm of the vectorized predictor parameters. This regularizer encourages the stable component to support a shared predictor across source cities, reducing source-city-specific response patterns before target adaptation.

Once the stable component has been estimated, target adaptation acts primarily through the adaptive component. Let $\theta = \{\theta_{\mathrm{inv}}, \theta_{\mathrm{var}}\}$ denote the parameters associated with the two components. Let $p_{\mathrm{inv}} = \dim \theta_{\mathrm{inv}}$ and $p_{\mathrm{var}} = \dim \theta_{\mathrm{var}}$ denote the vectorized parameter-block sizes; neither is identified with the latent width d_z or expert count K. To analyze relative plasticity at the parameter level, we consider the following local block-gradient update model:

$$\theta^{(a+1)} = \theta^{(a)} - \eta\, \mathbf{\Pi}_t\, \nabla_\theta \mathcal{L}_t, \qquad \mathbf{\Pi}_t = \begin{bmatrix} \rho_{\mathrm{inv}}\mathbf{I} & 0 \\ 0 & \mathbf{M}_t \end{bmatrix}, \tag{16}$$

Here a indexes optimization steps, $\eta > 0$ is the scalar learning rate, and $\mathcal{L}_t$ denotes the target adaptation objective in this generic block update. The scalar η is distinct from the expert-gain vector

$\boldsymbol{\eta}_t$. Here $\mathbf{I}$ is $p_{\text{inv}} \times p_{\text{inv}}$, $\mathbf{M}_t$ is $p_{\text{var}} \times p_{\text{var}}$, and $\mathbf{\Pi}_t$ acts on their concatenated parameter vector. In this analytical update model, $\rho_{\text{inv}} \ll 1$ scales stable-component gradients and $\mathbf{M}_t$ supplies posterior-dependent adaptive gain. Appendix D bounds the resulting relative update magnitudes under its stated conditions.

3.3.2 Support-conditioned latent state inference

The target adaptation set provides evidence about how the stable exogenous-to-flow core should be modulated in the target city. ExoC2T treats this target-conditioned deviation as a latent state z_t and infers a support-conditioned posterior from adaptation evidence and city context,

$$q_\psi(z_t \mid \mathcal{S}_t, d_t) = \mathcal{N}(\mu_t, \Sigma_t), \qquad (\mu_t, \Sigma_t) = \text{Enc}_\psi\left(\xi(\mathcal{S}_t), \mathbf{g}_t\right). \tag{17}$$

Here $\xi(\mathcal{S}_t)$ is a permutation-invariant support summary, and Enc_ψ includes covariance decoding. The latent state and mean lie in $\mathbb{R}^{d_z}$, $\Sigma_t \in \mathbb{R}^{d_z \times d_z}$ is positive definite, ψ denotes inference parameters, and $\mathbf{g}_t = \mathbf{g}_{d_t}$ is the encoded target-city context. The mean shifts adaptive routing and expert preference; the marginal variances $\text{diag}(\Sigma_t)$ regulate adaptive gain. For fixed adapted parameters and covariance, inference uses $f_t^{\text{stat}}(x) = f_\theta(x, d_t, \mu_t; \Sigma_t)$, while $\bar{f}_t(x) = \mathbb{E}_{z\sim q_\psi}[f_\theta(x, d_t, z; \Sigma_t)]$ denotes the analytical posterior average. Appendix E defines this forward family and its variance-aware gain maps. The task covariance remains fixed when its argument is suppressed below. The posterior objective combines support fitting, a city-conditioned prior, and mean-variance regularization,

$$\begin{aligned}\mathcal{L}_{\text{post}}^{(t)} = \mathbb{E}_{z_t\sim q_\psi}\left[\frac{1}{K_t}\sum_{j=1}^{K_t}\ell(f_\theta(x_j, d_t, z_t), y_j)\right] \\ + \beta D_{\text{KL}}(q_\psi(z_t \mid \mathcal{S}_t, d_t)\|p_\psi(z_t \mid d_t)) + \lambda_{\text{unc}}\frac{\text{tr}(\Sigma_t)}{d_z}.\end{aligned} \tag{18}$$

Here $K_t = |\mathcal{S}_t|$, j indexes support examples, and the training loss $\ell(\hat{y}, y) = \|\hat{y} - y\|_F^2/(HN)$ averages squared errors over forecast steps and nodes. The prior is $p_\psi(z_t \mid d_t)$; $\beta \geq 0$ weights the KL divergence, and $\lambda_{\text{unc}} \geq 0$ weights the mean latent variance $\text{tr}(\Sigma_t)/d_z$. All three terms are scaled together by λ_{post} in Eq. (21). This objective estimates a support-conditioned latent state anchored to source-learned cross-city structure. Appendix E further shows that the posterior-induced adaptation gap is controlled by the posterior mean error and covariance trace, which links evidence fitting and KL regularization to target adaptation reliability. It also records how the latent state controls prediction, routing, and variance-aware adaptive gain, together with the PAC-Bayes-style justification.

3.3.3 Target-risk decomposition with discrepancy regularization

Comparable stable representations across source and target cities support reliable posterior correction. For the analytical posterior-averaged predictor $\bar{f}_t$, let $\mathcal{R}_{\text{tar}} = \mathbb{E}[R_{\mathcal{Q}_t}(\bar{f}_t)]$. Under the cross-predictor transfer and coupling conditions stated in Appendix F, the discrepancy-based analysis [4] gives the following target-risk control relation:

$$\mathcal{R}_{\text{tar}} \leq \mathbb{E}_{(d_t,\mathcal{S}_t)}\left[R_s(\bar{f}_s) + \mathcal{D}_{\text{inv}}(P_s^\phi, P_t^\phi) + \left(1 + \gamma\,\mathcal{D}_{\text{inv}}(P_s^\phi, P_t^\phi)\right)\mathcal{E}_{\text{post}}(\mathcal{S}_t, d_t) + \mathcal{E}_{\text{app}}\right], \tag{19}$$

where P_s^ϕ and P_t^ϕ are the source and target feature distributions induced by $\phi = h_{\text{inv}}$, $R_s(\bar{f}_s)$ is the source risk of the corresponding source posterior-averaged predictor $\bar{f}_s$, $\mathcal{D}_{\text{inv}}$ measures invariant source-target mismatch, $\mathcal{E}_{\text{post}}$ denotes posterior estimation error, and $\mathcal{E}_{\text{app}}$ collects approximation, ideal joint-error, and source–oracle predictor-mismatch residuals. The query-set expectation is absorbed into the risk terms. Appendix F relates this averaged-predictor risk to the risk of f_t^{stat} using the posterior-statistics bridge. Under the coupling condition in Appendix F, the posterior-gap envelope scales linearly with invariant source-target mismatch.

In practice, we control invariant mismatch through two complementary regularizers, City-Invariance Regularization and Prototype Distribution Alignment. City-Invariance Regularization uses a gradient-reversal domain discriminator on pooled invariant features to suppress city-identifying directions. Prototype Distribution Alignment matches prototype-occupancy moments in the shared prototype coordinates. The adversarial regularizer is optimized with gradient reversal, so the discriminator minimizes city-classification loss while the invariant encoder receives the reversed gradient.

We group the encoder-side discrepancy signals into a single penalty,

$$\mathcal{L}_{\mathrm{disc}} = \mathcal{L}_{\mathrm{adv}}^{\mathrm{inv}} + \mathcal{L}_{\mathrm{align}}^{\mathrm{inv}}, \tag{20}$$

with Appendix F giving the expected-risk definition and the concrete forms of both regularizers. The two terms act at different levels. City-Invariance Regularization targets domain-identifying feature directions, and Prototype Distribution Alignment keeps samples organized consistently in the shared prototype coordinates.

The resulting two-stage objectives use one coefficient for each modeling role,

$$\begin{aligned} \mathcal{L}_{\mathrm{A}} &= \mathcal{L}_{\mathrm{src}} + \lambda_{\mathrm{stab}}\, \mathcal{L}_{\mathrm{homeo}}, \\ \mathcal{L}_{\mathrm{B}} &= \mathcal{L}_{\mathrm{src}} + \lambda_{\mathrm{disc}}\, \mathcal{L}_{\mathrm{disc}} + \lambda_{\mathrm{post}}\, \mathcal{L}_{\mathrm{post}}^{(t)}. \end{aligned} \tag{21}$$

Here $\mathcal{L}_{\mathrm{src}}$ is the supervised source prediction loss on $\mathcal{D}_S$. The coefficients λ_{stab}, λ_{disc}, and λ_{post} respectively control stable-component stabilization, invariant discrepancy regularization, and posterior-conditioned adaptation. These objectives correspond to Stage A: Stable Source Mechanism Learning and Stage B: Posterior-Conditioned Adaptive Modulation. Query-time prediction still uses exogenous context and excludes historical target flow.

4 Experiment

In this section, we evaluate ExoC2T on exogenous-driven cross-city transfer, with the primary emphasis on target adaptation ability under heterogeneous city splits and graph-missing conditions.

4.1 Experimental settings

We evaluate ExoC2T on 9 heterogeneous urban environments, including eight full-year regions and one shorter-span region, with hourly traffic flow as the prediction target and only exogenous urban context as model input. We use leave-one-city-out evaluation, where each city is held out as the target and all remaining cities serve as sources. All models use the same 12-step context length and 12-step forecasting horizon, while input modalities follow each model category. We evaluate 24- and 72-shot target-evidence regimes and report MAE, RMSE, and SMAPE, with city-wise metrics macro-averaged over leave-one-city-out runs. Additional dataset information, implementation details, split definitions, and metric formulas are provided in Appendix B.

4.2 Results and analysis

Baselines. We compare ExoC2T with statistical methods, flow-input spatio-temporal predictors, exogenous-driven variants, cross-city transfer frameworks, and time-series or spatio-temporal foundation models. Full baseline descriptions and implementation details are provided in Appendix B.

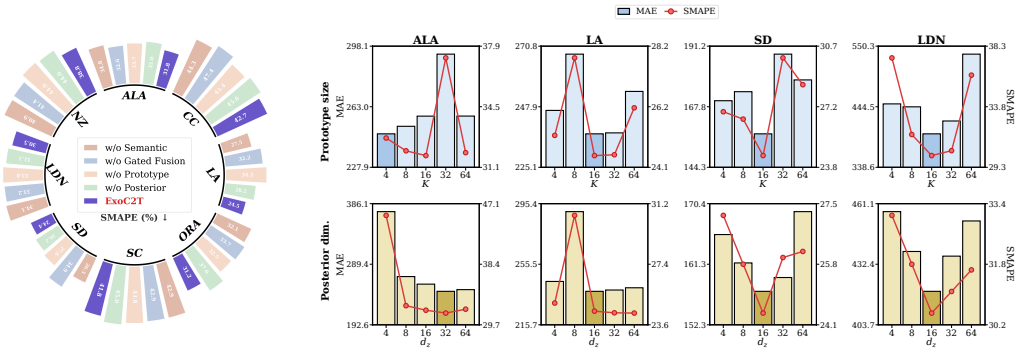
Performance comparison. Table 1 reports 24-shot performance on the eight full-year target regions; the shorter-span SIP target and 72-shot results appear in Appendix B.2. ExoC2T achieves the best MAE and SMAPE on all eight regions and the best RMSE on seven. Relative to the city-wise strongest non-ExoC2T results, its eight-city average MAE and SMAPE are lower by 16.8% and 17.9%, respectively. These results demonstrate the effectiveness of exogenous-only query inputs under scarce target supervision.

Ablation study and sensitivity analysis. Figure 3 summarizes the component-level ablation and hyperparameter sensitivity analysis. The ablation panel reports SMAPE for four variants that match the figure legend: w/o Semantic removes the semantic graph-conditioned spatial context, w/o Gated Fusion removes the modulation-sensitive gated fusion layer, w/o Prototype removes the prototype-routed expert memory, and w/o Posterior removes posterior-conditioned adaptive modulation. The w/o Gated Fusion variant causes the largest average degradation, followed by w/o Posterior and w/o Prototype, indicating that ExoC2T relies on both context-conditioned gated fusion and stable-adaptive memory organization. The sensitivity panel varies the number of prototype anchors K and the posterior latent dimension d_z. Moderate values give the strongest results across representative cities, while overly small or large settings tend to reduce accuracy, suggesting that the shared

Table 1: City-wise performance comparison under the 24-shot target-evidence regime. **Best** results are bolded and second-best results are underlined.

Category	Method	Alameda			Contra Costa			Los Angeles			Orange		
		MAE↓	RMSE↓	SMAPE↓	MAE↓	RMSE↓	SMAPE↓	MAE↓	RMSE↓	SMAPE↓	MAE↓	RMSE↓	SMAPE↓
Stat.	HA	700.38	844.64	71.59	589.73	727.55	79.45	686.38	823.28	59.62	674.44	842.56	78.37
	ARIMA	687.48	896.23	64.07	572.47	774.37	68.94	678.80	880.58	52.96	688.89	973.01	67.96
Flow ST	STGCN	308.77	454.31	37.81	225.13	359.48	47.82	519.17	655.88	49.06	249.72	359.21	44.91
	DCRNN	367.48	486.23	44.16	260.44	358.78	54.53	333.32	435.09	35.08	329.12	460.47	51.70
	Graph WaveNet	345.24	472.97	42.07	278.20	387.79	54.98	291.01	392.90	30.23	293.45	419.05	45.78
Exo.	STID-E	313.33	432.82	37.58	258.33	382.22	50.02	368.04	490.33	35.31	393.10	550.20	56.72
Transfer	DASTNet	269.46	369.44	35.43	213.70	299.44	46.11	248.69	387.38	25.67	294.52	410.84	53.04
	ST-GFSL	441.94	611.76	48.77	360.46	505.35	63.60	323.89	436.66	34.90	307.07	425.06	47.61
	STGP	282.47	388.25	36.29	244.29	339.58	50.51	274.10	**368.42**	28.98	272.48	375.83	46.63
FM	UniTS	330.82	477.05	46.51	241.29	373.91	55.97	289.46	415.66	32.48	275.49	408.74	44.70
	UniTime	362.16	537.04	46.37	278.73	434.50	59.09	347.70	499.07	36.26	361.61	527.52	53.22
	OpenCity	371.73	548.36	51.06	287.48	470.88	62.11	321.95	479.36	37.61	324.79	502.37	53.07
Ours	**ExoC2T**	**257.56**	**367.45**	**31.77**	**186.43**	**277.72**	**42.68**	**237.70**	373.94	**24.48**	**171.52**	**261.11**	**31.22**

Category	Method	Santa Clara			San Diego			London			New Zealand		
		MAE↓	RMSE↓	SMAPE↓	MAE↓	RMSE↓	SMAPE↓	MAE↓	RMSE↓	SMAPE↓	MAE↓	RMSE↓	SMAPE↓
Stat.	HA	557.11	675.13	79.98	675.41	820.88	75.07	1541.52	1974.11	80.66	618.59	926.54	97.46
	ARIMA	594.18	779.11	70.67	670.18	888.14	63.44	1620.20	2241.85	71.25	659.18	1085.94	80.69
Flow ST	STGCN	426.74	541.76	64.73	218.61	317.58	34.27	534.13	856.67	38.95	200.15	380.32	54.51
	DCRNN	284.50	389.59	51.19	295.63	412.17	43.16	687.11	1010.60	46.97	333.20	586.22	72.04
	Graph WaveNet	274.44	387.36	46.28	285.95	398.31	43.17	763.33	1120.47	48.16	314.30	589.80	57.65
Exo.	STID-E	301.56	412.96	50.04	302.19	426.88	42.34	982.33	1344.37	59.92	462.28	793.09	80.46
Transfer	DASTNet	247.56	340.48	48.41	229.89	315.09	40.59	773.13	1091.98	61.03	269.65	446.53	68.59
	ST-GFSL	369.49	519.93	60.16	334.66	451.73	52.22	842.39	1252.57	61.47	307.62	613.21	56.74
	STGP	285.30	389.52	47.71	263.62	368.54	38.03	713.08	977.36	50.29	347.61	662.46	71.77
FM	UniTS	290.02	426.90	54.90	252.05	368.88	41.61	715.87	1087.74	49.29	215.06	385.57	52.26
	UniTime	302.27	447.06	57.23	297.85	429.06	47.21	766.00	1159.97	48.39	273.35	524.91	56.18
	OpenCity	315.81	481.62	60.89	287.54	424.00	48.86	728.18	1082.91	52.43	249.77	465.08	55.03
Ours	**ExoC2T**	**224.74**	**328.05**	**41.76**	**157.28**	**221.52**	**24.44**	**419.57**	**647.09**	**30.51**	**160.60**	**285.66**	**38.76**



(a) Ablation study.

(b) Sensitivity analysis.

Figure 3: Ablation and sensitivity analysis under the 24-shot target-evidence setting. Left: component removal study for the main modules of ExoC2T. Right: city-wise sensitivity to prototype memory size K and posterior latent dimension d_z, with bars showing MAE on the left axis and red curves showing SMAPE on the right axis.

mechanism space needs sufficient capacity without excessive fragmentation. Additional 72-shot ablations and sensitivity studies for expert rank, semantic graph Top-k, and discrepancy-regularization strength are provided in Appendix B.2.

4.3 Case study

Figure 4 examines three aspects of target-city prediction. Panel (a) shows that ExoC2T remains unchanged when recent-flow inputs used by flow-dependent baselines are corrupted. Panel (b)

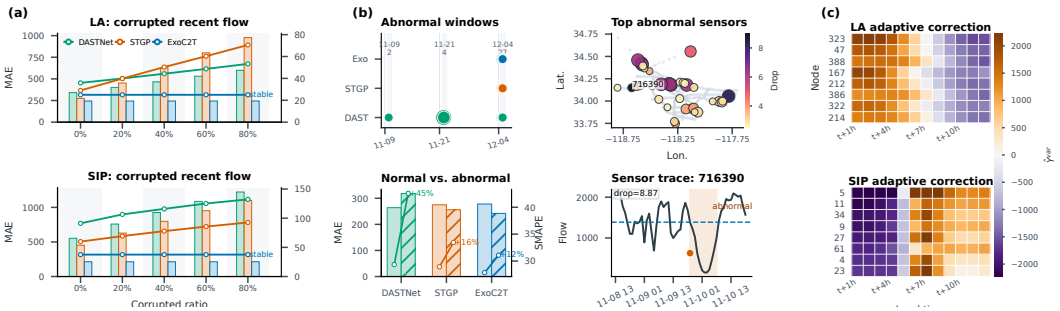


Figure 4: Case studies for exogenous-driven cross-city transfer. (a) ExoC2T remains stable when recent-flow inputs used by flow-conditioned baselines are corrupted. (b) Los Angeles abnormal-period evidence, including abnormal windows, normal-versus-abnormal performance, high-impact sensors, and a representative sensor trace. (c) Node-wise adaptive component patterns in Los Angeles and SIP.

evaluates automatically detected abnormal traffic windows: all methods become less accurate, but ExoC2T has a smaller degradation on LA than the compared flow-input transfer baselines. Panel (c) shows structured node- and horizon-specific adaptive corrections in LA and SIP. Procedures and further analysis are provided in Appendix B.3.

5 Conclusion

This paper introduced ExoC2T for exogenous-driven cross-city transfer, where target-city traffic is predicted from external urban context and no historical flow is used as query-time input. ExoC2T combines EXIST, which organizes heterogeneous external context into a shared exogenous mechanism representation, with SPAda, which separates a stable exogenous-to-flow core from adaptive target-conditioned variation. With only external context and limited target-city labels, ExoC2T consistently outperforms strong flow-input forecasting, transfer, and foundation-model baselines. More broadly, these results suggest a shift from sensor-dependent traffic forecasting toward context-grounded urban intelligence. Reducing reliance on dense real-time traffic instrumentation can make predictive mobility services more accessible to under-instrumented cities and support a more scalable deployment model for urban AI.

Limitations. ExoC2T's response to sudden disruptions depends on whether they are reflected in the available exogenous signals. Prediction quality also depends on POI coverage, weather alignment, and macro-context resolution. Target adaptation requires a small labeled offline support set. Appendix B.4 provides further discussion.

Acknowledgments and Disclosure of Funding

This paper is partially supported by the National Natural Science Foundation of China (No.62502488, No.12227901), Natural Science Foundation of Jiangsu Province (BK20240460), the grant from State Key Laboratory of Resources and Environmental Information System. The AI-driven experiments, simulations and model training were performed on the robotic AI-Scientist platform of Chinese Academy of Sciences. The authors declare no competing interests.

References

[1] Wickliffe C Abraham. Metaplasticity: tuning synapses and networks for plasticity. *Nature Reviews Neuroscience*, 9(5):387–387, 2008.

[2] Wickliffe C Abraham and Mark F Bear. Metaplasticity: the plasticity of synaptic plasticity. *Trends in neurosciences*, 19(4):126–130, 1996.

[3] Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, Jasper Zschiegner, Danielle C. Maddix, Hao Wang, Michael W. Mahoney, Kari Torkkola, Andrew Gordon Wilson, Michael Bohlke-Schneider, and Yuyang Wang. Chronos: Learning the language of time series, 2024. URL https://arxiv.org/abs/2403.07815.

[4] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79(1):151–175, 2010.

[5] Xiaojun Bi, Wudi Li, Yiwen Sun, Siyuan Qi, and Zheng Wang. Leveraging multiple source cities in selective transfer learning for traffic prediction with limited data. *IEEE Transactions on Intelligent Transportation Systems*, 26(10):15646–15659, October 2025. ISSN 1558-0016. doi: 10.1109/tits.2025.3596309. URL http://dx.doi.org/10.1109/TITS.2025.3596309.

[6] G. A. Bodvarsson and S. T. Muench. Effects of loop detector installation on the portland cement concrete pavement lifespan: Case study on i-5. https://digitalarchives.wa.gov, 2024. Accessed 2026-05-07.

[7] Akhilan Boopathy, Zhiyan Yi, Gil Wolff, and Amber Roy Chowdhury. Fast few-shot graph flow prediction, 2025. URL https://openreview.net/forum?id=rrn6XXB3u0.

[8] Kehua Chen, Yuxuan Liang, Jindong Han, Siyuan Feng, Meixin Zhu, and Hai Yang. Semantic-fused multi-granularity cross-city traffic prediction. *Transportation Research Part C: Emerging Technologies*, 162:104604, 2024.

[9] Hakan Erdol, Robert Klar, Vangelis Angelakis, James Pope, Robert Piechocki, Theo Tryfonas, and George Oikonomou. City-agnostic demand prediction: A graph attention approach for urban transfer learning. In *2025 IEEE International Smart Cities Conference (ISC2)*, pages 1–6. IEEE, 2025.

[10] Yuchen Fang, Hao Miao, Yuxuan Liang, Liwei Deng, Yue Cui, Ximu Zeng, Yuyang Xia, Yan Zhao, Torben Bach Pedersen, Christian S Jensen, et al. Unraveling spatio-temporal foundation models via the pipeline lens: A comprehensive review. *IEEE Transactions on Knowledge and Data Engineering*, 2026.

[11] Jie Feng, Jun Zhang, Tianhui Liu, Xin Zhang, Tianjian Ouyang, Junbo Yan, Yuwei Du, Siqi Guo, and Yong Li. Citybench: Evaluating the capabilities of large language models for urban tasks. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pages 5413–5424, 2025.

[12] Shanghua Gao, Teddy Koker, Owen Queen, Thomas Hartvigsen, Theodoros Tsiligkaridis, and Marinka Zitnik. Units: A unified multi-task time series model. *Advances in Neural Information Processing Systems*, 37:140589–140631, 2024.

[13] Junfeng Hu, Xu Liu, Zhencheng Fan, Yifang Yin, Shili Xiang, Savitha Ramasamy, and Roger Zimmermann. Prompt-based spatio-temporal graph transfer learning. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 890–899, 2024.

[14] Guangyin Jin, Yuxuan Liang, Yuchen Fang, Zezhi Shao, Jincai Huang, Junbo Zhang, and Yu Zheng. Spatio-temporal graph neural networks for predictive learning in urban computing: A survey. *IEEE transactions on knowledge and data engineering*, 36(10):5388–5408, 2023.

[15] Yilun Jin, Kai Chen, and Qiang Yang. Selective cross-city transfer learning for traffic prediction via source city region re-weighting. In *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*, pages 731–741, 2022.

[16] Yilun Jin, Kai Chen, and Qiang Yang. Transferable graph structure learning for graph-based traffic forecasting across cities. In *Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining*, pages 1032–1043, 2023.

[17] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45, 1960. doi: 10.1115/1.3662552.

[18] Bo Li, Ruotao Yu, Zijun Chen, Yingzhe Ding, Mingxia Yang, Jinghua Li, Jianxiao Wang, and Haiwang Zhong. High-resolution multi-source traffic data in new zealand. *Scientific Data*, 11(1):1216, 2024. doi: 10.1038/s41597-024-04067-5. URL https://doi.org/10.1038/s41597-024-04067-5.

[19] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926*, 2017.

[20] Zhonghang Li, Long Xia, Lei Shi, Yong Xu, Dawei Yin, and Chao Huang. Open spatio-temporal foundation models for traffic prediction. *ACM Transactions on Intelligent Systems and Technology*, October 2025. ISSN 2157-6912. doi: 10.1145/3773912. URL http://dx.doi.org/10.1145/3773912.

[21] Ziqian Lin, Jie Feng, Ziyang Lu, Yong Li, and Depeng Jin. Deepstn+: Context-aware spatial-temporal neural network for crowd flow prediction in metropolis. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):1020–1027, 2019.

[22] Jiayue Liu, Zhongchao Yi, Zhengyang Zhou, Qihe Huang, Kuo Yang, Xu Wang, and Yang Wang. Synevo: A neuro-inspired spatiotemporal evolutional framework for cross-domain adaptation. *arXiv preprint arXiv:2505.16080*, 2025.

[23] Xu Liu, Yutong Xia, Yuxuan Liang, Junfeng Hu, Yiwei Wang, Lei Bai, Chao Huang, Zhenguang Liu, Bryan Hooi, and Roger Zimmermann. Largest: A benchmark dataset for large-scale traffic forecasting. *Advances in Neural Information Processing Systems*, 36:75354–75371, 2023.

[24] Xu Liu, Junfeng Hu, Yuan Li, Shizhe Diao, Yuxuan Liang, Bryan Hooi, and Roger Zimmermann. Unitime: A language-empowered unified model for cross-domain time series forecasting. In *Proceedings of the ACM Web Conference 2024*, pages 4095–4106, 2024.

[25] Zhanyu Liu, Guanjie Zheng, and Yanwei Yu. Cross-city few-shot traffic forecasting via traffic pattern bank. In *Proceedings of the 32nd ACM international conference on information and knowledge management*, pages 1451–1460, 2023.

[26] Zhidan Liu, Zhengze Sun, Junru Zhang, Bolin Zhang, and Panrong Tong. Regional knowledge transfer for urban traffic flow prediction via satellite imagery assisted contrastive domain adaptation. *IEEE Transactions on Intelligent Transportation Systems*, 26(11):19120–19134, November 2025. ISSN 1558-0016. doi: 10.1109/tits.2025.3589218. URL http://dx.doi.org/10.1109/TITS.2025.3589218.

[27] Bin Lu, Xiaoying Gan, Weinan Zhang, Huaxiu Yao, Luoyi Fu, and Xinbing Wang. Spatio-temporal graph few-shot learning with cross-city knowledge transfer. In *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*, pages 1162–1172, 2022.

[28] Vishal Mahajan, Guido Cantelmo, Raoul Rothfeld, and Constantinos Antoniou. Predicting network flows from speeds using open data and transfer learning. *IET Intelligent Transport Systems*, 17(4):804–824, 2023.

[29] National Highways. WebTRIS: Traffic flow data system. https://webtris.highwaysengland.co.uk/, 2026. Accessed 2026-05-04.

[30] New Zealand Ministry of Transport. Vehicle fleet statistics. https://www.transport.govt.nz/statistics-and-insights/fleet-statistics/, 2026. Accessed 2026-05-04.

[31] NOAA National Centers for Environmental Information. Integrated surface database. https://www.ncei.noaa.gov/products/land-based-station/integrated-surface-database, 2026. Accessed 2026-05-04.

[32] OpenStreetMap contributors. Openstreetmap. https://www.openstreetmap.org/, 2026. Accessed 2026-05-04.

[33] Xiaocao Ouyang, Yan Yang, Yiling Zhang, Wei Zhou, Jihong Wan, and Shengdong Du. Domain adversarial graph neural network with cross-city graph structure learning for traffic prediction. *Knowledge-Based Systems*, 278:110885, 2023.

[34] Zheyi Pan, Yuxuan Liang, Weifeng Wang, Yong Yu, Yu Zheng, and Junbo Zhang. Urban traffic prediction from spatio-temporal data using deep meta learning. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1720–1730, 2019.

[35] Junqi Shao, Shen Li, Ke Zhang, Anyou Wang, and Meng Li. Cross-city traffic prediction based on deep domain adaptive transfer learning. *Transportation Research Part C: Emerging Technologies*, 176:105152, 2025.

[36] Zezhi Shao, Zhao Zhang, Fei Wang, Wei Wei, and Yongjun Xu. Spatial-temporal identity: A simple yet effective baseline for multivariate time series forecasting. In *Proceedings of the 31st ACM international conference on information & knowledge management*, pages 4454–4458, 2022.

[37] Anthony Stathopoulos and Matthew G Karlaftis. A multivariate state space approach for urban traffic flow modeling and prediction. *Transportation Research Part C: Emerging Technologies*, 11(2):121–135, 2003.

[38] Stats NZ Tatauranga Aotearoa. Official statistics and data for new zealand. `https://www.stats.govt.nz/`, 2026. Accessed 2026-05-04.

[39] Suzhou Municipal Bureau of Statistics. Suzhou statistical yearbook and statistical data. `https://tjj.suzhou.gov.cn/`, 2026. Accessed 2026-05-04.

[40] Yihong Tang, Ao Qu, Andy HF Chow, William HK Lam, Sze Chun Wong, and Wei Ma. Domain adversarial spatial-temporal network: A transferable framework for short-term traffic forecasting across cities. In *Proceedings of the 31st ACM international conference on information & knowledge management*, pages 1905–1915, 2022.

[41] Gina G Turrigiano. The dialectic of hebb and homeostasis. *Philosophical transactions of the royal society B: biological sciences*, 372(1715), 2017.

[42] UK Government Digital Service. data.gov.uk. `https://www.data.gov.uk/`, 2026. Accessed 2026-05-04.

[43] United States Census Bureau. American community survey data. `https://www.census.gov/programs-surveys/acs/data.html`, 2026. Accessed 2026-05-04.

[44] JWC Van Lint, Serge P Hoogendoorn, and Henk J van Zuylen. Freeway travel time prediction with state-space neural networks: Modeling state-space dynamics with recurrent neural networks. *Transportation Research Record*, 1811(1):30–39, 2002.

[45] Beibei Wang, Youfang Lin, Shengnan Guo, and Huaiyu Wan. Gsnet: Learning spatial-temporal correlations from geographical and semantic aspects for traffic accident risk forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(5):4402–4409, 2021.

[46] Binwu Wang, Jiaming Ma, Pengkun Wang, Xu Wang, Yudong Zhang, Zhengyang Zhou, and Yang Wang. Stone: A spatio-temporal ood learning framework kills both spatial and temporal shifts. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2948–2959, 2024.

[47] Leye Wang, Xu Geng, Xiaojuan Ma, Feng Liu, and Qiang Yang. Cross-city transfer learning for deep spatio-temporal prediction. *arXiv preprint arXiv:1802.00386*, 2018.

[48] Yuxuan Wang, Haixu Wu, Jiaxiang Dong, Guo Qin, Haoran Zhang, Yong Liu, Yunzhong Qiu, Jianmin Wang, and Mingsheng Long. Timexer: Empowering transformers for time series forecasting with exogenous variables. *Advances in Neural Information Processing Systems*, 37:469–498, 2024.

[49] Alanna J Watt and Niraj S Desai. Homeostatic plasticity and stdp: keeping a neuron's cool in a fluctuating world. *Frontiers in synaptic neuroscience*, 2:1486, 2010.

[50] Ying Wei, Yu Zheng, and Qiang Yang. Transfer knowledge between cities. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1905–1914, 2016.

[51] Billy M Williams and Lester A Hoel. Modeling and forecasting vehicular traffic flow as a seasonal arima process: Theoretical basis and empirical results. *Journal of transportation engineering*, 129(6):664–672, 2003.

[52] Gerald Woo, Chenghao Liu, Akshat Kumar, Caiming Xiong, Silvio Savarese, and Doyen Sahoo. Unified training of universal time series forecasting transformers. In *Forty-first International Conference on Machine Learning*, 2024.

[53] Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, and Chengqi Zhang. Graph wavenet for deep spatial-temporal graph modeling. *arXiv preprint arXiv:1906.00121*, 2019.

[54] Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, Xiaojun Chang, and Chengqi Zhang. Connecting the dots: Multivariate time series forecasting with graph neural networks. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 753–763, 2020.

[55] Huaxiu Yao, Yiding Liu, Ying Wei, Xianfeng Tang, and Zhenhui Li. Learning from multiple cities: A meta-learning approach for spatial-temporal prediction. In *The world wide web conference*, pages 2181–2191, 2019.

[56] Ada X Yee, Yu-Tien Hsu, and Lu Chen. A metaplasticity view of the interaction between homeostatic and hebbian plasticity. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 372(1715), 2017.

[57] Zhongchao Yi, Zhengyang Zhou, Qihe Huang, Yanjiang Chen, Liheng Yu, Xu Wang, and Yang Wang. Get rid of isolation: A continuous multi-task spatio-temporal learning framework. *Advances in Neural Information Processing Systems*, 37:136701–136726, 2024.

[58] Bing Yu, Haoteng Yin, and Zhanxing Zhu. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv preprint arXiv:1709.04875*, 2017.

[59] Yuan Yuan, Chenyang Shao, Jingtao Ding, Depeng Jin, and Yong Li. Spatio-temporal few-shot learning via diffusive neural network generation. *arXiv preprint arXiv:2402.11922*, 2024.

[60] Yudong Zhang, Xu Wang, Xuan Yu, Zhaoyang Sun, Kai Wang, and Yang Wang. Drawing informative gradients from sources: A one-stage transfer learning framework for cross-city spatiotemporal forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39 (1):1147–1155, April 2025. ISSN 2159-5399. doi: 10.1609/aaai.v39i1.32102. URL `http://dx.doi.org/10.1609/aaai.v39i1.32102`.

[61] Yuhao Zheng, Jinyang Wu, Zihao Cai, Senzhang Wang, and Jianxin Wang. Adatm: fine-grained urban flow inference with adaptive knowledge transfer across multiple cities. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 3424–3432, 2024.

[62] Zhengyang Zhou, Yang Wang, Xike Xie, Lianliang Chen, and Hengchang Liu. Riskoracle: A minute-level citywide traffic accident forecasting framework. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(01):1258–1265, 2020.

[63] Xingchen Zou, Yibo Yan, Xixuan Hao, Yuehong Hu, Haomin Wen, Erdong Liu, Junbo Zhang, Yong Li, Tianrui Li, Yu Zheng, et al. Deep learning for cross-domain data fusion in urban computing: Taxonomy, advances, and outlook. *Information Fusion*, 113:102606, 2025.

Contents

1 Introduction **1**

2 Preliminary **2**

3 Methodology **3**

3.1 Framework overview 3

3.2 Exogenous-Infused Spatio-Temporal Network (EXIST) 4

3.3 Stability-Plasticity Adaptation (SPAda) 6

3.3.1 Stable-adaptive mechanism separation 6

3.3.2 Support-conditioned latent state inference 7

3.3.3 Target-risk decomposition with discrepancy regularization 7

4 Experiment **8**

4.1 Experimental settings 8

4.2 Results and analysis 8

4.3 Case study 9

5 Conclusion **10**

A Related work **16**

B Experimental details **17**

B.1 Experimental setup 17

B.2 Additional experimental results 20

B.2.1 Recent transfer baselines under the 24-shot protocol 20

B.2.2 Matched exogenous-input baseline comparison 21

B.2.3 Additional native-input results and sensitivity analysis 21

B.3 Case study details 24

B.4 Limitations 26

C Expanded formulation of EXIST **26**

C.1 Expanded shared exogenous state space 26

C.2 Expanded mechanism-relevant context extraction 27

C.3 Expanded modulation-sensitive gated fusion 28

C.4 Expanded prototype-routed expert memory for stable-adaptive forecasting 29

D Stable-adaptive mechanism separation: local perturbation analysis **31**

E Support-conditioned latent state inference and adaptive gain **32**

F Target-risk decomposition with discrepancy regularization **34**

A Related work

Spatio-temporal forecasting and foundation models

Spatio-temporal forecasting has progressed from statistical and state-space traffic models such as ARIMA [51], Kalman filtering [17], and state-space neural traffic predictors [44, 37] to deep architectures that capture temporal dynamics and spatial dependencies. Representative Spatio-Temporal Graph Neural Networks (STGNNs), including DCRNN [19], STGCN [58], Graph WaveNet [53], MTGNN [54], and STID [36], predict future traffic by conditioning on observed flow sequences and explicit or learned graph structure. Related urban forecasting studies also incorporate geographical, semantic, and risk-oriented contexts for tasks such as crowd flow and accident risk prediction [21, 45, 62], while large-scale benchmarks and surveys summarize the expanding traffic forecasting landscape [23, 14, 10]. Recent time-series and spatio-temporal foundation models, including Chronos [3], Moirai [52], UniTS [12], UniTime [24], OpenCity [20], and exogenous-aware sequence models such as TimeXer [48], further demonstrate the value of large-scale pretraining across heterogeneous temporal domains. Broader urban benchmark efforts also study how general-purpose models behave on urban tasks [11]. These models provide powerful sequence priors, but their forecasting formulation still relies on informative target-side histories. Recent traffic observations serve as direct evidence of the current endogenous traffic state, and forecasting is performed by extrapolating from this observed state. ExoC2T studies a different forecasting setting in which historical traffic flow is unavailable as model input and prediction must be mediated by static and dynamic exogenous urban context.

Cross-city transfer and few-shot adaptation

Cross-city transfer learning uses data-rich source cities to improve prediction in data-scarce target cities. Early urban transfer studies show that heterogeneous city data can support knowledge sharing across urban tasks [50, 47]. Representative traffic transfer methods, including DASTNet [40], CrossTReS [15], TransGTR [16], semantic-fused transfer [8], ADATM [61], STONE [46], domain-adversarial graph transfer [33], deep domain-adaptive transfer [35], TSJT [60], SynEVO [22], and broader cross-domain urban data-fusion studies [63], reduce inter-city discrepancy through invariant representation learning, region re-weighting, transferable graph learning, adaptive knowledge transfer, or source-target alignment. Meta-learning and few-shot adaptation methods, including MetaST [55], urban deep meta learning [34], ST-GFSL [27], Traffic Pattern Bank [25], STGP [13], diffusive neural generation [59], and continuous multi-task spatio-temporal learning [57], improve rapid specialization by learning transferable initializations, traffic pattern memories, prompts, or generative adaptation priors. The dominant transfer object in these methods remains the representation of observed traffic dynamics. Target-city flow histories are often used as prediction context, adaptation signals, graph-estimation cues, or pattern-matching evidence. ExoC2T shifts the transfer object to exogenous-to-flow mechanisms and uses the target adaptation set to infer a support-conditioned latent state while keeping query-time prediction conditioned on exogenous context.

Open-data and exogenous mobility prediction

External urban variables such as POI semantics, weather, temporal indicators, and road-network descriptors have long been used to enhance traffic prediction [21]. These signals are especially important when dense traffic instrumentation is costly to deploy or maintain [6]. In cross-city scenarios, semantic-fused transfer frameworks exploit functional similarity and environmental context to support adaptation [8], while recent sequence models such as TimeXer [48] provide stronger mechanisms for incorporating exogenous covariates. Open-data mobility prediction further shows that publicly available or weakly sensed urban signals can support traffic and demand estimation. Mahajan et al. [28] estimate link-level traffic volume from floating-car speed observations, road attributes, and temporal features, transferring from Paris to Madrid under limited target flow labels. Erdol et al. [9] use OpenStreetMap POI distributions and graph attention for city-agnostic bike-sharing OD demand level classification. Fast few-shot graph flow prediction [7] estimates average edge-level flow or AADT proxies in unseen road networks from node and edge attributes. These studies demonstrate the value of open urban evidence across different observation and prediction interfaces. Speed remains a direct traffic-state measurement, OD demand levels define categorical tasks, and AADT-style graph flow is closer to static average-flow estimation than horizon-based forecasting.

ExoC2T differs by targeting dynamic node-level traffic forecasting from exogenous context at query time, where recent traffic-state measurements such as flow, speed, and occupancy are unavailable and limited target labels are used only to infer adaptive exogenous-to-flow modulation. Under this exogenous-only observability, static and dynamic exogenous context becomes the primary evidence for recovering traffic-state information.

Invariant transfer with plasticity control

Invariant representation learning and discrepancy-based adaptation provide a common foundation for transfer under domain shift [4]. In cross-city traffic prediction, such principles are often instantiated through domain-invariant features, graph transfer, or source-target alignment. ExoC2T uses invariance in a mechanism-level form. The stable component represents the exogenous-to-flow core, while the adaptive component carries city-conditioned deviation around that core. This design is also motivated by neural plasticity. Homeostatic plasticity stabilizes functional responses [41, 49], and metaplasticity regulates the future sensitivity of adaptation [2, 56, 1]. In ExoC2T, homeostatic stabilization constrains drift in the stable component, whereas metaplastic regulation maps the support-conditioned posterior to adaptive modulation and uncertainty-aware gain. This perspective links stable representation learning with posterior-dependent adaptation for exogenous-only transfer under city shift.

B Experimental details

B.1 Experimental setup

Dataset. The benchmark contains six California counties in North America, namely Alameda County, Contra Costa County, Los Angeles County, Orange County, Santa Clara County, and the County of San Diego, together with London in Europe, New Zealand in Oceania, and Suzhou Industrial Park in Asia. The California traffic series are derived from LargeST [23]; the New Zealand traffic series are derived from the Scientific Data dataset of Li et al. [18]; the SIP traffic series follow the RiskOracle data source [62]; and the London traffic series are collected from WebTRIS [29]. Weather variables are obtained from the NOAA Integrated Surface Database [31], including temperature, dew point, relative humidity, and wind speed. POI semantics are extracted from OpenStreetMap [32]. Macro indicators are collected from data.gov.uk for London [42], the American Community Survey for U.S. counties [43], Stats NZ and the New Zealand Ministry of Transport for New Zealand [38, 30], and the Suzhou Municipal Bureau of Statistics for SIP [39]. Unless otherwise stated, each city is chronologically partitioned into training, validation, and testing segments with a 80%/10%/10% ratio.

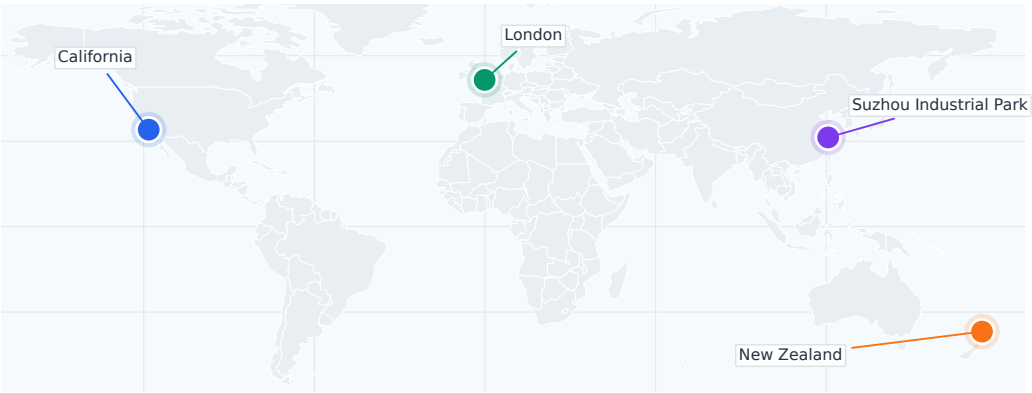


Figure 5: Geographic distribution of the benchmark target regions.

Leave-one-city-out protocol. Let $\mathcal{C}$ denote the full city set. In each run, one city $d_t \in \mathcal{C}$ is held out as the target city, and all remaining cities are used as source cities,

$$\mathcal{C}_S^{(t)} = \mathcal{C} \setminus \{d_t\}. \tag{22}$$

Table 2: Dataset statistics for the 9-city exogenous-driven cross-city benchmark. Explicit graph support indicates whether an observed adjacency matrix is available; otherwise the semantic prior graph is used.

Region	Traffic source	Nodes	Hourly records	Time span	Temporal coverage	Explicit graph
Alameda	LargeST	103	8,760	2018-01-01–2018-12-31	Full year	Yes
Contra Costa	LargeST	105	8,760	2018-01-01–2018-12-31	Full year	Yes
Los Angeles	LargeST	437	8,760	2018-01-01–2018-12-31	Full year	Yes
Orange	LargeST	205	8,760	2018-01-01–2018-12-31	Full year	Yes
Santa Clara	LargeST	128	8,760	2018-01-01–2018-12-31	Full year	Yes
San Diego	LargeST	215	8,760	2018-01-01–2018-12-31	Full year	Yes
London	WebTRIS	145	8,760	2018-01-01–2018-12-31	Full year	No
New Zealand	NZ multi-source traffic	224	8,760	2018-01-01–2018-12-31	Full year	Yes
Suzhou Industrial Park	RiskOracle	108	2,160	2017-01-01–2017-03-31	Short span	No

Table 3: External variables used by ExoC2T.

Variable group	Variables
Weather	Temperature, dew point, relative humidity, wind speed.
Temporal covariates	Hour-of-day, day-of-week, holiday or calendar indicators when available, and forecast-horizon timestamp queries.
Macro context	Year-end population, gross domestic product in USD, resident population per-capita GDP in USD, population density, and total motor vehicles.
POI semantics	Six OpenStreetMap-derived functional categories counted within a 5 km radius around each node.

Table 4: POI categories and OpenStreetMap sub-categories used to construct node-level urban semantic context.

Category	Sub-category	Category	Sub-category
Sustenance	bar, biergarten, cafe, fast food, food court, ice cream, pub, restaurant	Education	college, dancing school, driving school, first aid school, kindergarten, language school, library, research institute, training, music school, traffic park, university
Healthcare	baby hatch, clinic, dentist, doctors, hospital, nursing home, pharmacy, social facility, veterinary	Entertainment	arts centre, casino, cinema, community centre, conference centre, events venue, gambling, music venue, nightclub, planetarium
Service	courthouse, fire station, police, post box, post depot, post office, prison, ranger station, townhall	Place	city, borough, suburb, quarter, neighbourhood, city block, plot, town, village, hamlet, isolated dwelling, farm, allotments

In Stage A: Stable Source Mechanism Learning, the model is trained on the union of source-city training data from $\mathcal{C}_S^{(t)}$. In Stage B: Posterior-Conditioned Adaptive Modulation, the model uses d_t under the specified target-evidence budget. Repeating this procedure over all cities yields one evaluation run per target city.

Implementation. ExoC2T is implemented in PyTorch and trained on an NVIDIA V100 GPU with 16GB memory. Each run uses a 12-step context window and predicts the next 12 traffic steps. In Stage A: Stable Source Mechanism Learning, AdamW optimizes source prediction together with homeostatic stable-component stabilization. In Stage B: Posterior-Conditioned Adaptive Modulation, training jointly samples source batches and target adaptation batches, optimizing source prediction, posterior correction, and invariant discrepancy regularization. Source and target prediction losses use mean squared error averaged over batch samples, forecast steps, and nodes. The posterior objective combines target prediction, KL regularization, and mean posterior variance as in Eq. (18). The KL and variance coefficients are internal to this objective; its overall weight is λ_{post}. During Stage B, the overall discrepancy and posterior weights are linearly warmed up to their full values

over the first 100 steps, scaling all posterior terms together. We evaluate 24- and 72-shot target-evidence regimes. For graph-missing cities, the semantic prior graph selects $k_{\mathrm{graph}} = 15$ neighbors using the mixed POI and node-state similarity score in Eq. (29).

Table 5 summarizes the main ExoC2T hyperparameters used in the reported runs. Algorithm 1 summarizes the training and prediction procedure. The capacity-related parameters are further examined in the sensitivity analysis in Appendix B.2.

Table 5: Key hyperparameter settings of ExoC2T.

Group	Hyperparameter	Value
Forecasting protocol	Context length S / prediction horizon H	12 / 12
Forecasting protocol	Target-evidence budget	24-shot and 72-shot
Forecasting protocol	Target adaptation batch / evaluation batch	24 / 24
Training schedule	Stage A / Stage B epochs	30 / 30
Training schedule	Source batch size	168 in reported runs; 192 for the Stage-B source stream
Training schedule	Warm-up steps for auxiliary terms	100
Training schedule	Early stopping patience / minimum delta	8 / 10^{-4}
Optimization	Optimizer	AdamW
Optimization	Backbone / middle / head learning rates	5×10^{-4} / 10^{-3} / 2×10^{-3}
Optimization	Weight decay / gradient clipping	10^{-5} / 1.0
Representation dimensions	Node / POI / macro / time / weather embeddings	32 / 32 / 16 / 16 / 8
Representation dimensions	Shared hidden and spatial context dimensions	64 / 64
Representation dimensions	Stable / adaptive component dimensions	32 / 16
Adaptive memory and posterior	Prototype anchors and experts K	16
Adaptive memory and posterior	Expert rank r	8
Adaptive memory and posterior	Posterior latent dimension d_z	16
Adaptive memory and posterior	Prototype temperature τ_p / dropout	1.0 / 0.1
Adaptive memory and posterior	Semantic prior graph Top-k	15
Regularization weights	Stable-component regularizer λ_{stab}	0.01
Regularization weights	City-Invariance Regularization / Prototype Distribution Alignment	0.1 / 0.1
Regularization weights	Overall posterior weight λ_{post}	0.1
Regularization weights	Internal posterior KL / mean-variance weights β / λ_{unc}	10^{-3} / 10^{-3}

Baselines. We group baselines into five categories. Statistical methods include HA (Historical Average), which estimates traffic from repeated historical patterns, and ARIMA [51], which models linear temporal autocorrelation. Flow-input spatio-temporal predictors include STGCN [58], which combines graph convolution with temporal convolution, DCRNN [19], which models directed traffic diffusion with recurrent forecasting, and Graph WaveNet [53], which learns adaptive graph dependencies with dilated temporal convolutions. Exogenous-driven variants include STID-E, adapted from STID [36] by replacing historical traffic input with static and dynamic exogenous context. Cross-city transfer frameworks include DASTNet [40], which performs domain-adaptive spatio-temporal transfer, ST-GFSL [27], which targets few-shot graph flow prediction, and STGP [13], which adapts spatio-temporal predictors through transferable prompts. Foundation-model baselines include UniTS [12], a unified multi-task time-series model, UniTime [24], a language-empowered cross-domain time-series forecaster, and OpenCity [20], an open spatio-temporal foundation model for traffic prediction. All baselines are evaluated under the same target-city split and target-evidence protocol as ExoC2T.

Table 6: Model size comparison. Parameter counts are reported in thousands (K). Cached embeddings are counted separately when applicable.

Model	Parameters (K)	With cached embeddings (K)
DASTNet	117.4	127.6
ExoC2T	209.9	–
ST-GFSL (GWN)	877.6	–
STGP	1243.9	–
UniTS	9491.2	–
OpenCity	26026.0	–
UniTime	96120.6	–

Table 6 reports the model-size comparison. ExoC2T uses 209.9K parameters and remains lightweight among transfer and foundation-model baselines. Although it is slightly larger than DASTNet, DASTNet additionally relies on cached embeddings; ExoC2T uses about 24% of the parameters of ST-GFSL, 17% of STGP, and only a small fraction of UniTS, OpenCity, or UniTime.

The same cross-city model is used in both learning stages. Stage A learns shared encoders, prototypes, experts, and readouts from source supervision with stable-component regularization. Stage

B uses source batches and labeled target-support batches with Eq. (21); the support posterior conditions fusion, routing, and adaptive gain. Equation (16) is a local analytical model of stable/adaptive parameter-block updates. Experimental training uses AdamW with the parameter-group learning rates, weight decay, and gradient clipping listed in Table 5. The inferred posterior summary is reused at query time without query labels. The mean-variance term in Eq. (18) is an explicit training regularizer; its coefficient λ_{unc} is distinct from the analytical coefficient λ_{Σ} in Eq. (77).

Metrics. We report Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and Symmetric Mean Absolute Percentage Error (SMAPE), defined as

$$\begin{aligned} \mathrm{MAE} &= \frac{1}{n_{\mathrm{eval}}} \sum_{v=1}^{n_{\mathrm{eval}}} \left| \hat{y}_v - y_v \right|, \\ \mathrm{RMSE} &= \sqrt{\frac{1}{n_{\mathrm{eval}}} \sum_{v=1}^{n_{\mathrm{eval}}} \left(\hat{y}_v - y_v \right)^2}, \\ \mathrm{SMAPE} &= \frac{100\%}{n_{\mathrm{eval}}} \sum_{v=1}^{n_{\mathrm{eval}}} \frac{2 \left| \hat{y}_v - y_v \right|}{|\hat{y}_v| + |y_v| + \epsilon_0}, \end{aligned} \tag{23}$$

where n_{eval} is the number of evaluated target entries and ϵ_0 is a small constant for numerical stability. Metrics are first computed for each target city and then macro-averaged over leave-one-city-out runs.

B.2 Additional experimental results

This appendix extends the evaluation with recent transfer baselines and matched exogenous-input comparisons under the 24-shot protocol, followed by the additional native-input results and sensitivity analyses. The main text focuses on the 24-shot regime because it is the most data-scarce setting; the 72-shot results below evaluate whether the same ranking is preserved with a larger target adaptation set. Since SIP has a shorter temporal span, its 24- and 72-shot results are reported separately in Table 9.

B.2.1 Recent transfer baselines under the 24-shot protocol

We additionally compare the recent transfer methods SERT [26] and CMSVC [5] under the same 24-shot protocol. Table 7 reports all nine target cities, including SIP, with MAE, RMSE, and SMAPE. ExoC2T obtains the lowest SMAPE on every target city. The MAE and RMSE rankings are not uniform: SERT has lower MAE on Santa Clara and lower RMSE on Los Angeles. The nine-city averages summarize each metric with equal weight per target city.

Table 7: Comparison with SERT and CMSVC under the same 24-shot protocol. SMAPE is reported in percent. Best results within this comparison are bolded and second-best results are underlined.

City	SERT			CMSVC			ExoC2T		
	MAE	RMSE	SMAPE	MAE	RMSE	SMAPE	MAE	RMSE	SMAPE
Alameda	<u>267.9</u>	<u>383.8</u>	<u>36.0</u>	289.6	414.2	37.7	**257.56**	**367.45**	**31.77**
Contra Costa	<u>203.0</u>	<u>316.2</u>	<u>46.9</u>	215.4	321.3	48.2	**186.43**	**277.72**	**42.68**
Los Angeles	<u>269.6</u>	**370.6**	<u>29.0</u>	273.5	374.2	29.3	**237.70**	<u>373.94</u>	**24.48**
Orange	<u>224.5</u>	<u>327.8</u>	<u>38.8</u>	241.7	348.6	41.6	**171.52**	**261.11**	**31.22**
Santa Clara	**220.3**	<u>337.1</u>	<u>43.8</u>	226.6	339.3	44.0	<u>224.74</u>	**328.05**	**41.76**
San Diego	<u>201.3</u>	<u>291.2</u>	<u>32.1</u>	219.3	310.1	35.2	**157.28**	**221.52**	**24.44**
London	<u>518.1</u>	826.3	<u>34.8</u>	537.1	<u>813.4</u>	37.8	**419.57**	**647.09**	**30.51**
New Zealand	<u>188.0</u>	<u>319.1</u>	<u>49.0</u>	206.4	350.1	52.6	**160.60**	**285.66**	**38.76**
SIP	<u>247.2</u>	<u>387.0</u>	<u>44.0</u>	298.1	446.9	54.7	**213.67**	**323.18**	**37.58**
Average	<u>260.0</u>	<u>395.5</u>	<u>39.4</u>	278.6	413.1	42.3	**225.45**	**342.86**	**33.69**

B.2.2 Matched exogenous-input baseline comparison

To isolate the effect of model design under the same query-time observability, we adapt representative spatio-temporal, transfer, and foundation-model baselines to use the same POI, macro, timestamp, and weather inputs as ExoC2T, replacing their recent-flow input. The suffix "-E" denotes these exogenous-input variants; SERT-E is adapted from SERT [26]. All methods use the same source-city split, 24-shot target support, query split, forecast horizon, and evaluation metrics. This matched-input comparison complements the flow-input comparison in Table 1.

Table 8 reports the city-wise results. ExoC2T achieves the lowest MAE, RMSE, and SMAPE for every target city among these matched-input baselines. Relative to the corresponding strongest baseline averages, its MAE is 29.85% lower than SERT-E (321.38), its RMSE is 27.37% lower than DASTNet-E (472.05), and its SMAPE is 22.71% lower than SERT-E (43.59%). These comparisons control the input modalities and provide evidence for the contribution of the shared and target-adaptive model design.

Table 8: City-wise comparison under matched exogenous inputs and the 24-shot protocol. The nine target cities are arranged in three blocks. SMAPE is reported in percent; best results are bolded and second-best results are underlined.

Method	Alameda			Contra Costa			Los Angeles		
	MAE	RMSE	SMAPE	MAE	RMSE	SMAPE	MAE	RMSE	SMAPE
STGCN-E	385.96	508.62	45.32	299.08	417.44	54.57	409.60	531.02	40.92
DCRNN-E	383.18	503.14	47.05	267.11	367.76	55.26	399.87	520.13	41.01
Graph WaveNet-E	373.85	496.05	43.60	312.26	427.00	55.67	413.21	533.10	40.09
DASTNet-E	273.41	<u>375.47</u>	36.91	207.87	<u>300.81</u>	<u>47.55</u>	273.76	<u>389.68</u>	29.13
SERT-E	<u>265.47</u>	402.16	<u>34.95</u>	<u>198.37</u>	324.67	48.37	<u>270.97</u>	402.87	<u>28.49</u>
ST-GFSL-E	480.69	616.47	56.12	994.72	1418.02	153.10	444.74	576.30	42.67
UniTS-E	350.51	479.88	42.80	277.79	384.03	55.48	350.19	463.94	37.18
OpenCity-E	933.80	1149.89	128.96	655.36	853.21	118.04	1085.76	1300.40	129.99
ExoC2T	**257.56**	**367.45**	**31.77**	**186.43**	**277.72**	**42.68**	**237.70**	**373.94**	**24.48**
Method	Orange			Santa Clara			San Diego		
	MAE	RMSE	SMAPE	MAE	RMSE	SMAPE	MAE	RMSE	SMAPE
STGCN-E	428.28	579.18	62.48	378.51	489.55	61.23	367.94	492.19	49.15
DCRNN-E	370.06	515.93	55.88	316.81	424.52	53.99	352.16	474.27	48.74
Graph WaveNet-E	429.27	579.33	59.05	338.42	452.83	54.44	386.74	515.24	50.78
DASTNet-E	264.09	361.11	50.63	249.09	<u>362.33</u>	<u>46.71</u>	233.28	323.73	42.99
SERT-E	<u>186.15</u>	<u>279.30</u>	<u>33.20</u>	<u>241.63</u>	382.37	47.39	<u>173.30</u>	<u>250.90</u>	<u>29.43</u>
ST-GFSL-E	968.17	1225.54	146.95	646.36	820.60	114.00	559.41	774.27	76.08
UniTS-E	334.50	468.86	51.52	375.60	470.58	60.40	308.36	406.87	46.81
OpenCity-E	902.76	1190.75	121.09	637.20	827.83	112.21	792.21	1032.69	117.14
ExoC2T	**171.52**	**261.11**	**31.22**	**224.74**	**328.05**	**41.76**	**157.28**	**221.52**	**24.44**
Method	London			New Zealand			SIP		
	MAE	RMSE	SMAPE	MAE	RMSE	SMAPE	MAE	RMSE	SMAPE
STGCN-E	982.70	1372.14	63.70	515.94	952.08	85.00	553.83	873.02	73.85
DCRNN-E	1046.60	1425.52	62.62	488.93	866.17	79.54	556.25	875.07	67.18
Graph WaveNet-E	1042.72	1432.56	62.87	508.59	907.97	82.25	551.22	882.03	66.95
DASTNet-E	<u>633.08</u>	<u>903.97</u>	56.25	396.03	551.78	96.95	436.44	679.55	69.95
SERT-E	900.99	1321.14	<u>55.00</u>	<u>321.96</u>	<u>509.13</u>	<u>67.98</u>	<u>333.61</u>	<u>564.70</u>	<u>47.53</u>
ST-GFSL-E	1049.47	1447.12	65.78	504.95	900.89	88.07	694.50	1031.80	88.50
UniTS-E	691.97	952.41	56.30	334.99	544.63	70.23	532.03	813.56	77.25
OpenCity-E	1938.65	2650.68	142.63	663.60	1165.28	120.45	879.53	1305.71	115.27
ExoC2T	**419.57**	**647.09**	**30.51**	**160.60**	**285.66**	**38.76**	**213.67**	**323.18**	**37.58**

B.2.3 Additional native-input results and sensitivity analysis

Additional performance comparison. Across the 24-shot full-year targets in Table 1, the strongest competing method varies by city and metric, including transfer baselines such as DASTNet

and STGP, flow-input models such as STGCN and Graph WaveNet, and foundation-model baselines such as UniTS. ExoC2T obtains the best MAE and SMAPE on every full-year target city. Its average MAE over the eight full-year targets is 226.93, compared with 272.75 for the city-wise second-best results, corresponding to a 16.8% relative reduction. Its average SMAPE is 33.20%, compared with 40.46% for the city-wise second-best results, corresponding to a 17.9% relative reduction. Increasing the target-evidence budget from 24 to 72 shots generally improves both transfer and flow-input baselines, especially on large California targets where limited target labels can calibrate local dynamics. ExoC2T also improves under the larger target-evidence budget and keeps the same ranking pattern, with the best MAE and SMAPE on all eight full-year targets and the best RMSE on seven of them in Table 10. This indicates that a larger target adaptation set improves adaptation, and the advantage of organizing external context into a shared mechanism space remains visible under the larger target-evidence budget.

Table 9: SIP target-city performance under 24- and 72-shot target-evidence regimes. **Best** results are bolded and second-best results are underlined.

Category	Method	24-shot			72-shot		
		MAE↓	RMSE↓	SMAPE↓	MAE↓	RMSE↓	SMAPE↓
Stat.	HA	840.26	1108.16	90.15	840.26	1108.16	90.15
	ARIMA	1023.10	1470.34	113.19	849.95	1167.83	93.73
Flow ST	STGCN	468.35	757.68	66.73	399.31	652.60	58.43
	DCRNN	391.33	659.81	55.72	354.34	603.42	53.51
	Graph WaveNet	477.54	795.66	60.94	511.17	842.22	63.99
Exo.	STID-E	530.41	862.11	64.55	527.17	874.74	61.77
Transfer	DASTNet	475.24	696.82	76.58	296.51	450.35	52.80
	ST-GFSL	426.01	799.89	51.16	387.62	695.36	46.07
	STGP	452.62	780.64	59.85	398.89	669.41	57.18
FM	UniTS	438.90	672.39	63.99	350.72	582.28	54.51
	UniTime	396.43	611.15	65.58	333.85	525.24	52.11
	OpenCity	445.63	752.92	63.60	303.62	516.25	48.05
Ours	**ExoC2T**	**213.67**	**323.18**	**37.58**	**172.31**	**278.85**	**29.16**

The non-California targets indicate that the gains are not limited to California-to-California transfer. In the 24-shot full-year table, ExoC2T reaches 419.57 MAE and 30.51% SMAPE on London, compared with the strongest non-ExoC2T results of 534.13 MAE and 38.95% SMAPE. On New Zealand, ExoC2T reaches 160.60 MAE and 38.76% SMAPE, while the strongest competing results are 200.15 MAE and 52.26% SMAPE. The same trend remains under 72 shots: ExoC2T obtains 451.39 MAE / 32.10% SMAPE on London and 150.22 MAE / 36.54% SMAPE on New Zealand, still ahead of the corresponding second-best results. SIP provides a stronger distribution shift because it is a shorter-span target from a different urban system. Under 24 shots, ExoC2T obtains 213.67 MAE and 37.58% SMAPE, whereas the best non-ExoC2T results are 391.33 MAE and 51.16% SMAPE. With 72 shots, ExoC2T further improves to 172.31 MAE and 29.16% SMAPE, while the strongest competing results remain 296.51 MAE and 46.07% SMAPE. These results suggest that the proposed adaptation remains effective when the target city differs substantially from the dominant source regions.

Additional ablation analysis. Figure 6 extends the main-text ablation analysis by reporting the core component-removal study under the 72-shot target-evidence regime. All values are reported with SMAPE so that the comparison is scale-normalized across cities. The main-text ablation removes four components. Removing semantic graph-conditioned spatial context disables the semantic graph evidence used to replace flow-derived spatial correlation. The gated-fusion ablation removes the context-conditioned gated fusion layer. The prototype-memory ablation removes the shared prototype-routed expert memory that organizes stable and adaptive responses. The posterior-modulation ablation removes posterior-conditioned adaptive modulation. In the 24-shot setting, removing the gated fusion layer gives the largest average SMAPE increase, while removing posterior modulation or prototype memory also causes clear degradation. Removing any component increases SMAPE across all eight target cities, while the component with the largest impact varies by city. A larger target adaptation set lowers the absolute error, while the model still loses accuracy when the gated fusion layer, prototype memory, or support-conditioned posterior is removed. These results indicate that extra target labels improve adaptation, while exogenous gated fusion and prototype-routed adaptive modulation remain necessary for accurate transfer.

Table 10: City-wise performance comparison under the 72-shot target-evidence regime. **Best** results are bolded and second-best results are underlined.

Category	Method	Alameda			Contra Costa			Los Angeles			Orange		
		MAE↓	RMSE↓	SMAPE↓	MAE↓	RMSE↓	SMAPE↓	MAE↓	RMSE↓	SMAPE↓	MAE↓	RMSE↓	SMAPE↓
Stat.	HA	700.38	844.64	71.59	589.73	727.55	79.45	686.38	823.28	59.62	674.44	842.56	78.37
	ARIMA	626.23	780.73	69.50	519.39	669.07	76.84	622.48	778.55	56.18	608.05	786.06	75.61
Flow ST	STGCN	302.22	447.34	37.41	216.53	344.80	47.75	499.01	637.96	47.47	236.58	343.81	42.81
	DCRNN	369.84	490.40	44.22	262.02	361.09	53.22	324.25	421.67	34.30	325.02	442.46	50.35
	Graph WaveNet	332.95	458.14	40.82	268.89	376.81	54.39	274.53	374.08	28.46	282.81	413.32	46.50
Exo.	STID-E	306.38	425.22	36.90	253.19	374.18	48.74	360.80	483.59	34.35	384.80	544.35	55.32
Transfer	DASTNet	241.66	347.12	33.16	177.85	265.46	43.47	229.18	366.48	24.24	220.42	311.99	42.55
	ST-GFSL	339.76	479.54	42.89	242.20	353.40	53.04	276.83	380.06	29.09	272.41	387.21	43.91
	STGP	260.65	366.58	33.94	228.88	324.51	48.94	242.29	**329.09**	26.06	243.97	350.14	43.34
FM	UniTS	285.45	436.22	40.00	215.02	353.56	51.19	247.99	363.82	28.39	223.04	346.05	38.17
	UniTime	347.27	521.81	44.39	262.54	410.76	56.56	313.80	460.40	33.67	310.71	469.26	46.33
	OpenCity	328.44	536.14	46.36	242.13	411.98	55.99	274.13	416.60	31.88	273.75	437.65	47.65
Ours	**ExoC2T**	**240.52**	**331.34**	**30.88**	**171.97**	**255.52**	**41.43**	**228.91**	365.96	**24.19**	**168.56**	**253.54**	**31.00**

Category	Method	Santa Clara			San Diego			London			New Zealand		
		MAE↓	RMSE↓	SMAPE↓	MAE↓	RMSE↓	SMAPE↓	MAE↓	RMSE↓	SMAPE↓	MAE↓	RMSE↓	SMAPE↓
Stat.	HA	557.11	675.13	79.98	675.41	820.88	75.07	1541.52	1974.11	80.66	618.59	926.54	97.46
	ARIMA	503.76	639.60	76.77	590.97	750.50	70.33	1405.36	1890.90	77.88	541.15	866.27	93.39
Flow ST	STGCN	429.44	546.09	64.75	212.41	307.58	35.54	503.35	813.38	36.86	189.30	365.33	51.32
	DCRNN	283.12	385.94	51.19	286.72	396.60	42.39	680.92	995.94	48.06	313.44	541.30	69.82
	Graph WaveNet	264.09	374.31	45.30	271.10	386.91	41.24	775.90	1181.64	48.41	320.32	604.72	67.83
Exo.	STID-E	283.76	394.05	48.83	297.57	423.93	41.09	969.24	1329.44	59.04	458.54	796.56	78.73
Transfer	DASTNet	199.37	295.94	39.81	183.42	265.78	31.13	513.07	787.01	40.72	195.43	334.25	61.66
	ST-GFSL	250.53	365.71	45.78	237.20	335.43	37.55	552.78	846.19	40.65	282.73	478.09	59.90
	STGP	243.19	341.45	43.18	226.58	332.49	32.66	562.14	814.87	44.83	287.39	518.32	67.68
FM	UniTS	225.54	357.90	46.72	210.16	311.38	37.12	592.29	896.95	41.83	183.45	338.15	43.52
	UniTime	268.54	408.13	54.08	256.81	381.02	40.42	679.90	1021.99	45.41	254.48	491.85	51.37
	OpenCity	258.73	421.87	52.16	232.32	364.21	40.60	580.60	901.36	44.22	206.47	398.12	52.66
Ours	**ExoC2T**	**196.49**	**290.44**	**38.03**	**158.16**	**219.25**	**25.18**	**451.39**	**670.39**	**32.10**	**150.22**	**266.71**	**36.54**

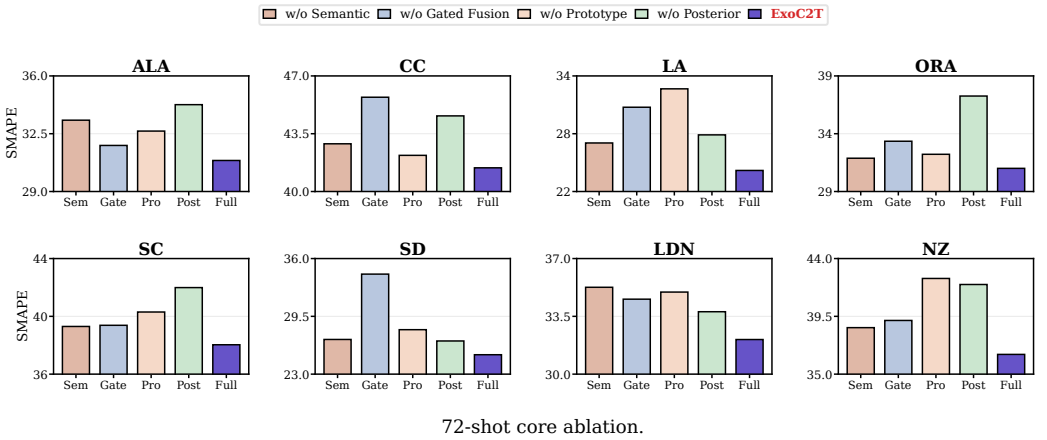


Figure 6: Additional SMAPE-based 72-shot core ablation over the main modules across eight full-year target cities.

Additional hyperparameter sensitivity. Figure 7 reports SMAPE sensitivity for three additional capacity and regularization choices: expert rank r, semantic prior graph Top-k, and invariant discrepancy-regularization strength λ_{disc}. The discrepancy coefficient λ_{disc} scales the grouped city-invariance and prototype-distribution alignment penalty in Eq. (20), thereby controlling the overall strength of invariant discrepancy regularization. Together with the main-text sensitivity to prototype memory size K and posterior latent dimension d_z, these experiments cover the main capacity and adaptation knobs of ExoC2T. For the prototype memory size in the main-text sensitivity figure, $K = 16$ is the best setting on LA, London, and San Diego, and it gives the best SMAPE on Alameda

although a smaller K slightly improves Alameda MAE. For the posterior latent dimension, $d_z = 16$ is best on London and San Diego, while LA has nearly identical results at 16 and 32 and Alameda slightly favors 32. We use $d_z = 16$ because it keeps the posterior compact and reduces adaptation cost with little loss in accuracy.

The appendix sensitivity study shows a similar cost–performance tradeoff for the remaining knobs. Expert rank improves clearly from $r = 1$ to $r = 8$, but larger ranks give only small or inconsistent gains; therefore the experiments use $r = 8$ to avoid unnecessary expert computation. The semantic prior graph is strongest at $k_{\text{graph}} = 15$ on all four representative cities in Figure 7, while smaller neighborhoods lose semantic coverage and $k_{\text{graph}} = 20$ begins to include weaker relations. For discrepancy regularization, $\lambda_{\text{disc}} = 0.1$ is best on Alameda, LA, and San Diego, and London is only slightly better at 0.3. We therefore keep 0.1 as the default because it improves transfer alignment without letting the discrepancy penalty dominate the forecasting objective.

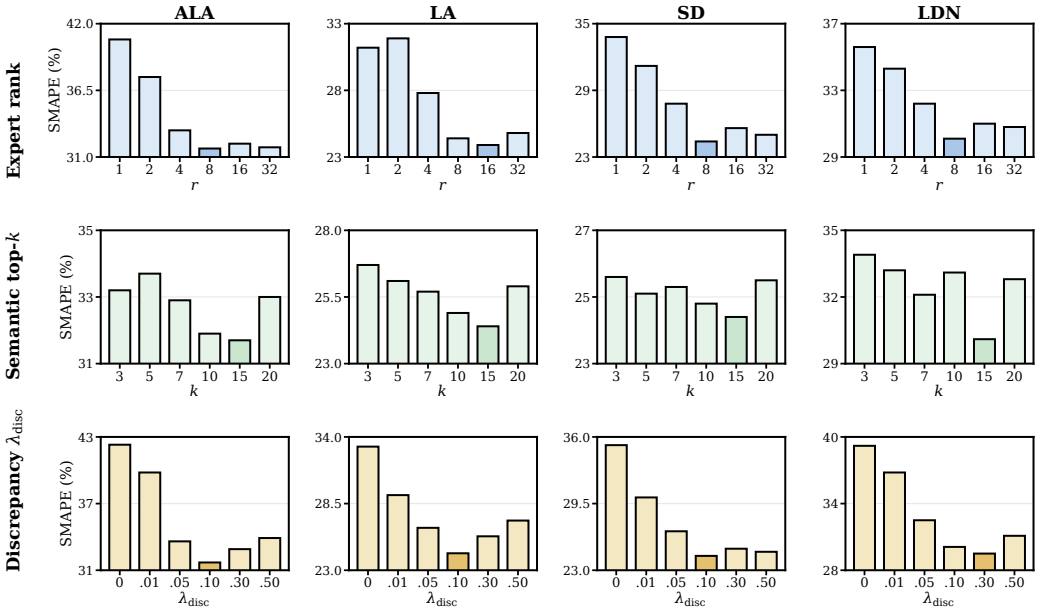


Figure 7: Additional SMAPE-based hyperparameter sensitivity under the 24-shot target-evidence regime. Rows vary expert rank r, semantic prior graph Top-k, and invariant discrepancy-regularization strength λ_{disc} across four representative target cities.

B.3 Case study details

Robustness to corrupted recent-flow inputs. Figure 4(a) evaluates a common sensing failure mode in which recent traffic measurements become unavailable, zero-filled, or contaminated by abnormal sensor readings. For flow-input baselines, we corrupt a fraction of the recent-flow context with outage-like zeros and noisy perturbations, increasing the corruption ratio from 0% to 80%. ExoC2T is not modified in this study because its query-time input consists only of exogenous context. The figure shows a clear separation between flow-dependent methods and ExoC2T. On LA, DASTNet degrades from 341.31 MAE / 35.90% SMAPE at 0% corruption to 599.26 MAE / 53.26% SMAPE at 80% corruption, and STGP increases from 274.10 MAE / 28.98% SMAPE to 976.79 MAE / 70.55% SMAPE. ExoC2T remains at 237.70 MAE / 24.48% SMAPE across all corruption levels. SIP shows the same qualitative pattern, with ExoC2T remaining unchanged while flow-input baselines degrade as the query context becomes less reliable. The result highlights a practical advantage of exogenous-only prediction in unreliable sensing environments, where forecast quality is insulated from corrupted recent-flow measurements.

Robustness to degraded exogenous inputs. The recent-flow corruption study above evaluates the benefit of avoiding unreliable flow measurements. We additionally test the external inputs that ExoC2T itself uses. Frozen 24-shot models are evaluated on Alameda, Los Angeles, San Diego, and

SIP after clean training and adaptation, without retraining under the perturbations. Macro tests apply a 20% population bias or mean-impute missing population/GDP. POI tests corrupt counts or mask 30% of the features at 20% of the nodes. Weather tests add a $0.2\sigma_w$ bias or noise, where σ_w denotes the corresponding feature standard deviation, or introduce 20% missingness. Calendar features are derived from timestamps.

For each checkpoint and metric Err, we compute the relative change against its paired clean result,

$$\Delta\mathrm{Err} = 100\,\frac{\mathrm{Err}_{\mathrm{corrupt}} - \mathrm{Err}_{\mathrm{clean}}}{\mathrm{Err}_{\mathrm{clean}}}. \tag{24}$$

Table 11 reports the four-city macro-average of these relative changes in percent. These are relative increases in the metrics, including SMAPE, rather than percentage-point changes. POI-count corruption produces the largest average degradation among the tested conditions: 3.91% in MAE, 3.24% in RMSE, and 2.92% in SMAPE. The remaining conditions produce smaller average changes. These results complement the recent-flow corruption study by showing robustness to the tested partial degradation of exogenous context.

Table 11: Four-city macro-average relative metric changes under degraded exogenous inputs. Each frozen 24-shot checkpoint is compared with its paired clean evaluation without retraining. Positive values indicate degradation.

Exogenous-input degradation	ΔMAE (%)	ΔRMSE (%)	ΔSMAPE (%)
Macro bias	+0.40	+0.22	+0.50
Macro missing	+1.76	+1.17	+1.19
POI error	+3.91	+3.24	+2.92
POI missing	+1.49	+1.10	+0.99
Weather bias	+1.23	+1.13	+1.17
Weather missing	+1.11	+1.24	+1.50
Weather noise	+1.01	+1.03	+1.07

Automatically detected abnormal traffic periods. Figure 4(b) examines naturally occurring abnormal traffic windows without using external incident annotations. We identify abnormal test windows from the observed traffic sequence using a robust down-spike criterion based on deviations from the previous 24-hour median; a window is marked abnormal when a sufficient fraction of sensors exhibits large robust deviation. The timeline reports the detected LA windows, including examples around November 9, November 21, and December 4, while the sensor map and trace make the evidence traceable to concrete sensor locations and time periods. The normal-versus-abnormal comparison shows that all methods degrade during abnormal windows, while ExoC2T has the smallest degradation on LA among the compared methods. DASTNet increases from 29.35% normal SMAPE to 42.58% abnormal SMAPE, and STGP increases from 28.93% to 33.44%. ExoC2T has the lowest abnormal-period SMAPE, increasing from 27.85% to 31.08%. This case shows that, during naturally occurring abnormal traffic periods, ExoC2T undergoes a smaller performance drop than baselines that take traffic flow as query-time input.

Adaptive contribution heatmaps. Figure 4(c) visualizes the adaptive component of ExoC2T by decomposing the prediction into a stable component and an adaptive component. The heatmaps show node-wise and horizon-wise adaptive contributions for LA and SIP, with structured corrections whose signs and magnitudes vary across sensors, forecast steps, and target cities. The visualization is consistent with the stable-adaptive design, where transferable exogenous-to-flow structure is retained in the stable component and target-city adjustment is carried by localized adaptive responses.

Support-conditioned posterior modulation. Figure 8 further analyzes whether the target adaptation set changes the adaptive component of ExoC2T. We compare SIP under four conditions, namely no posterior conditioning, 24-, 72-, and 168-shot target adaptation sets. Figures 8(a) and (b) show that posterior conditioning changes the routing distribution over adaptive prototypes and induces non-uniform expert amplification and suppression across target-evidence budgets. These observations indicate that the support-conditioned posterior modulates the adaptive component through prototype routing and expert gain.

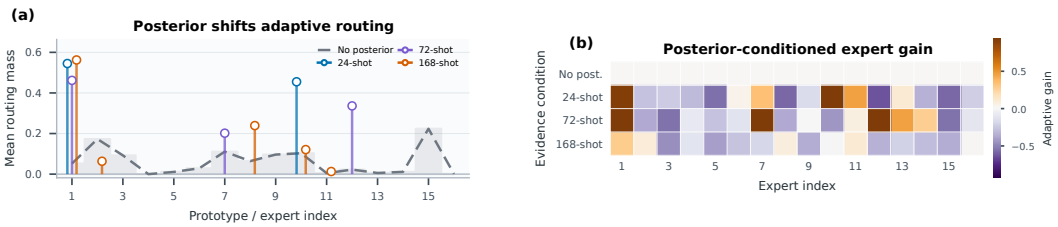


Figure 8: Support-conditioned posterior modulation on SIP. (a) Posterior-induced shifts in adaptive prototype routing. (b) Posterior-conditioned expert gain across evidence conditions, shown with equal-aspect heatmap cells. Colors indicate expert-gain increments relative to the unmodulated response, with zero denoting no modulation.

B.4 Limitations

ExoC2T focuses on exogenous-only transfer, so its query-time response is naturally limited for sudden accidents, temporary closures, special events, or control actions that are only visible through real-time traffic sensors. The case studies show a complementary strength: during abnormal traffic windows and corrupted recent-flow inputs, ExoC2T is more stable than flow-input baselines because it avoids unreliable query-time flow observations. This limitation is also tied to the intended setting of the paper, where under-instrumented target cities often lack the same real-time sensing stream that conventional transfer models require; in such settings, flow-input transfer models also have difficulty receiving and capturing sudden dynamic disruptions. The method can also be affected by sparse POI records, mismatched weather stations, or coarse macro indicators, although these are common constraints in low-cost urban deployment and richer event or road-operation feeds can be added as extra exogenous variables. Target-specific adaptation also requires labeled offline support windows to infer the posterior.

C Expanded formulation of EXIST

This appendix expands the Exogenous-Infused Spatio-Temporal Network formulation in Section 3.2 into the tensor-level operators used in implementation. It details exogenous mechanism state construction, modulation-sensitive gated fusion, and prototype-routed memory.

Notation summary. Table 12 summarizes the symbols used in the method and theoretical appendices. Bold uppercase letters denote matrices or tensors unless they are standard neural-network parameters; city, time, node, and prototype indices are respectively denoted by c, s, i, and k.

C.1 Expanded shared exogenous state space

Building on the exogenous mechanism state construction in Section 3.2, the shared exogenous state space is assembled in tensor form through feature-conditioned node-basis retrieval, city modulation, node-level semantic projection, and sequence-wise dynamic exogenous encoding. Here $\boldsymbol{\rho}_i$ denotes transferable node descriptors, $\mathbf{p}_i$ denotes the POI semantic vector of node i given by the i-th row of $\mathbf{P}$, and $\mathbf{a}_c = \Phi_a(\mathbf{g}_c)$ denotes a city-conditioned modulation vector generated from macro context. The node basis and city-specific modulation are

$$\mathbf{r}_i = \mathbf{B}_r^\top \operatorname{Softmax}(\chi_r([\boldsymbol{\rho}_i \oplus \mathbf{p}_i])), \qquad \Delta\mathbf{h}_{i,c} = \mathbf{U}_a(\mathbf{r}_i \odot \mathbf{a}_c), \qquad \mathbf{h}_i^{\text{node}} = \text{MLP}(\boldsymbol{\rho}_i) + \Delta\mathbf{h}_{i,c}. \tag{25}$$

Static and dynamic exogenous states are then stacked as

$$\mathbf{E}^s = [\mathbf{e}_1^s, \ldots, \mathbf{e}_N^s]^\top, \qquad \mathbf{e}_i^s = \mathbf{h}_i^{\text{node}} \oplus \Phi_p(\mathbf{p}_i), \tag{26}$$

$$\mathbf{E}_{1:S}^d = [\mathbf{e}_1^d, \ldots, \mathbf{e}_S^d]^\top, \qquad \mathbf{e}_s^d = \Psi_d(\mathbf{T}_s, \mathbf{W}_s), \qquad \mathbf{g}_c = \Phi_c(\mathbf{m}_c, c), \tag{27}$$

where $\mathbf{E}^s \in \mathbb{R}^{N \times d_s}$ collects node-level semantic states, $\mathbf{E}_{1:S}^d \in \mathbb{R}^{S \times d_d}$ collects dynamic exogenous states over the observation window, and $\mathbf{g}_c$ is the encoded macro context. For subsequent city-conditioned operations, the global context is broadcast across nodes as

$$\mathbf{G}_c = \mathbf{1}_N \mathbf{g}_c^\top, \tag{28}$$

Table 12: Summary of main notation used by ExoC2T.

Symbol	Meaning
$\mathcal{C}_S$, d_t	Source-city set and held-out target city.
$\mathcal{D}_S$, $\mathcal{S}_t$, $\mathcal{Q}_t$	Source data, limited target adaptation set, and target query set.
x_t, $\mathcal{Y}$, $\hat{\mathcal{Y}}$	Target-city exogenous query input, ground-truth traffic sequence, and predicted traffic sequence.
$\mathbf{P}$, $\mathbf{p}_i$, $\boldsymbol{\rho}_i$, $\mathbf{m}_c$	Node-level semantic matrix, POI semantic vector of node i, fixed-order sinusoidal descriptor of node i, and macro context vector of city c.
$\mathbf{T}_{1:S}$, $\mathbf{W}_{1:S}$, $\mathbf{T}^{+}_{1:H}$	Observation-window timestamp features, observation-window weather features, and forecast-horizon timestamp queries.
$\mathbf{A}$, $\mathbf{A}^{sem}$, $\mathbf{A}^{\star}$	Observed graph support, semantic prior graph, and the graph used by semantic aggregation.
$\mathbf{E}^{s}$, $\mathbf{E}^{d}_{1:S}$, $\mathbf{g}_c$	Row-stacked static ($N \times d_s$) and dynamic ($S \times d_d$) states, and city context (d_g).
$\mathbf{s}_i$, $\mathbf{t}_{1:S}$, $\mathbf{H}^{(t)}_{i,s}$	Spatial mechanism context, temporal regime descriptor, and fused node-time mechanism state.
$\mathbf{C}$, $\mathbf{c}_k$, $\mathcal{E}_k$	Prototype matrix, prototype anchor k, and expert operator attached to prototype k.
$\boldsymbol{\alpha}^{\mathrm{inv}}$, $\widetilde{\boldsymbol{\alpha}}^{\mathrm{var}}$	Stable and adaptive prototype occupancy vectors.
$\mathbf{b}^{\mathrm{inv}}_{i,s}$, $\mathbf{b}^{\mathrm{var}}_{i,s}$	Stable and adaptive memory responses for node i at step s.
z_t, μ_t, Σ_t	Support-conditioned latent state, posterior mean, and posterior covariance.
p_{inv}, p_{var}	Vectorized stable/adaptive parameter-block sizes, distinct from d_z and K.
f^{stat}_t, $\bar{f}_t$	Statistics-conditioned point predictor and analytical posterior prediction average.
$\mathbf{R}^{\mathrm{weather}}_{\mathcal{E}}$, $\widetilde{\mathbf{R}}^{(t)}_{\mathcal{E}}$	Weather-modulated intermediate and final posterior-scaled response stacks.
$\boldsymbol{\pi}_t$, $\mathbf{M}_t$, $\boldsymbol{\Pi}_t$	Latent-state-induced routing or modulation vector, adaptive gain matrix, and block plasticity operator.
$\boldsymbol{\delta}_t$, $\boldsymbol{\eta}_t$	Posterior-generated routing-shift and expert-gain vectors.
d_h	Hidden width of attention, fused states, and expert outputs; individual features are columns.
S, H, K_t, K	Context length, forecast length, support count, and prototype/expert count.
a, j, m	Optimization step in Eq. (16), support-example index, and attention-head index; n_{head} counts heads.
n_{eval}, k_{graph}	Evaluated scalar-entry count and semantic-graph neighbor count (Top-k in figures).
$\boldsymbol{\varpi}$, H, $\bar{\lambda}_{\mathrm{cov}}$	Domain-prior vector, its entropy, and fixed covariance-moment weight.
ρ_{inv}, θ_{inv}, θ_{var}	Stable-component plasticity scale, stable-component parameters, and adaptive-component parameters.
$\mathcal{L}_{\mathrm{homeo}}$, $\mathcal{L}_{\mathrm{post}}$, $\mathcal{L}_{\mathrm{disc}}$	Homeostatic stabilization, posterior correction, and invariant discrepancy losses.
P^{ϕ}_s, P^{ϕ}_t, $\mathcal{D}_{\mathrm{inv}}$	Source and target invariant feature distributions and their discrepancy. Source/target node–time occupancy laws are denoted P^{α}_s, P^{α}_t.
$\mathcal{E}_{\mathrm{post}}$, $\mathcal{E}_{\mathrm{app}}$	Loss-scale posterior-gap envelope and transfer residual covering approximation, ideal joint error, and source–oracle predictor mismatch.
$y^{\mathrm{inv/var}}_{\mathrm{loc}}$	Zeroth-/first-order local mechanism terms; distinct from learned forecasts $\hat{y}^{\mathrm{inv/var}}$.
$h_{\mathrm{inv/var}}$, $\mathrm{Head}_{\mathrm{inv/var}}$	Full branch representations and local feature transforms, respectively.

with $\mathbf{1}_N$ denoting the all-ones vector. These objects form the shared exogenous representation on which the later spatial extraction, gated fusion, and prototype routing are defined. Here $\mathbf{B}_r$ is the shared node basis, χ_r is the basis-retrieval network, Φ_p, Φ_c, and Φ_a are feature encoders, and $\mathbf{U}_a$ projects city-conditioned node modulation.

C.2 Expanded mechanism-relevant context extraction

The exogenous mechanism state construction in Section 3.2 summarizes semantic graph construction, relation-aware attention, and temporal-regime extraction with high-level operators. The expanded form specifies city-conditioned affine modulation, graph-supported attention, and spectral-local temporal decomposition used to build mechanism-relevant context.

The spatial branch first generates a semantic prior graph and city-conditioned node modulation,

$$\begin{aligned}
\tilde{A}^{sem}_{ij} &= \alpha_c \,\mathrm{sim}(\mathbf{p}_i, \mathbf{p}_j) + (1-\alpha_c)\,\mathrm{sim}(\mathbf{h}^{\mathrm{node}}_i, \mathbf{h}^{\mathrm{node}}_j),\\
\mathbf{A}^{sem} &= \mathrm{Norm}\big(\mathrm{TopK}_{k_{\mathrm{graph}}}(\tilde{\mathbf{A}}^{sem})\big), \qquad \mathbf{A}^{\star} = \mathbf{A} \text{ or } \mathbf{A}^{sem},\\
[\gamma_c, \beta_c] &= \Upsilon(\mathbf{g}_c), \qquad \tilde{\mathbf{e}}^s_i = \gamma_c \odot \mathbf{e}^s_i + \beta_c,
\end{aligned} \tag{29}$$

where $\alpha_c \in [0,1]$ is a city-conditioned semantic-structural mixing weight and $\Upsilon(\cdot)$ is the city-to-affine generator. The operator sim computes pairwise feature similarity, and Norm normalizes the retained edge weights. Here k_{graph} is the retained neighbor count (Top-k in the sensitivity figures), distinct from the expert index k. Relation-aware spatial aggregation is then expanded as

$$\begin{aligned}
\mathbf{q}_i &= \mathbf{W}_q \tilde{\mathbf{e}}^s_i,\\
\mathbf{k}_j &= \mathbf{W}_k \tilde{\mathbf{e}}^s_j,\\
\kappa_{ij} &= \mathcal{R}\big([\mathbf{p}_i, \mathbf{p}_j, \mathbf{h}^{\mathrm{node}}_i, \mathbf{h}^{\mathrm{node}}_j]\big),\\
\alpha_{ij} &= \mathrm{Softmax}_{j\in\mathcal{N}_i(\mathbf{A}^\star)}\left(\frac{\mathbf{q}_i^\top \mathbf{k}_j}{\sqrt{d_h}} + \kappa_{ij} + \lambda_A A^\star_{ij}\right),\\
\mathbf{s}_i &= \sum_{j\in\mathcal{N}_i(\mathbf{A}^\star)} \alpha_{ij} \mathbf{W}_v \tilde{\mathbf{e}}^s_j,
\end{aligned} \tag{30}$$

where $\mathcal{R}(\cdot)$ is the semantic relation generator and λ_A controls the structural graph bias.

The temporal branch decomposes dynamic exogenous context into spectral and local components,

$$\mathbf{z}^{\mathrm{spec}}_{1:S} = \mathcal{F}^{-1}\big(\mathcal{F}(\mathbf{E}^d_{1:S}) \odot \mathcal{K}\big), \qquad \mathbf{z}^{\mathrm{loc}}_{1:S} = \mathrm{Local}(\mathbf{E}^d_{1:S}), \tag{31}$$

followed by gated mixing

$$\omega = \sigma(\mathbf{w}_\omega^\top \bar{\mathbf{e}}^d + b_\omega), \qquad \mathbf{t}_{1:S} = \omega\, \mathbf{z}^{\mathrm{spec}}_{1:S} + (1-\omega)\, \mathbf{z}^{\mathrm{loc}}_{1:S}, \tag{32}$$

where $\bar{\mathbf{e}}^d = S^{-1}\sum_{s=1}^{S} \mathbf{e}^d_s$ is the observation-window mean exogenous state. In the gating and gain maps, σ denotes the elementwise logistic sigmoid, whereas $\sigma_{\min}$ in the perturbation analysis denotes a minimum singular value. The resulting pair $(\mathbf{s}_i, \mathbf{t}_{1:S})$ forms the mechanism-relevant context passed to the gated fusion layer.

C.3 Expanded modulation-sensitive gated fusion

The modulation-sensitive gated fusion paragraph in Section 3.2 introduces the perturbation state $\mathbf{\Omega}^{(t)}_s = \Gamma(\mathbf{t}_s) + \mathbf{B}_\pi \boldsymbol{\pi}_t$, where $\boldsymbol{\pi}_t$ is induced by the support-conditioned latent state in Section 3.3.2. The superscript (t) indexes the target task rather than a time step. A compatible multi-head instantiation expands this layer into head-wise query, key, and value modulation. Each head has query, key, and value width d_h. For head $m \in \{1, \dots, n_{\mathrm{head}}\}$,

$$\begin{aligned}
\mathbf{q}^{(m)}_{s,t} &= \mathbf{W}^{(m)}_q \mathbf{t}_s + \mathbf{U}^{(m)}_q \mathbf{\Omega}^{(t)}_s,\\
\mathbf{k}^{(m)}_{i,s,t} &= \mathbf{W}^{(m)}_k \mathbf{s}_i + \mathbf{U}^{(m)}_k \mathbf{\Omega}^{(t)}_s,\\
\mathbf{v}^{(m)}_{i,s,t} &= \mathbf{W}^{(m)}_v \mathbf{s}_i + \mathbf{U}^{(m)}_v \mathbf{\Omega}^{(t)}_s,
\end{aligned} \tag{33}$$

where $\mathbf{W}^{(m)}_q$, $\mathbf{W}^{(m)}_k$, and $\mathbf{W}^{(m)}_v$ are base projections and $\mathbf{U}^{(m)}_q$, $\mathbf{U}^{(m)}_k$, and $\mathbf{U}^{(m)}_v$ inject the support-conditioned perturbation into the gated fusion operator. Consistent with the implementation, the node response uses independent sigmoid gates in place of a softmax competition across nodes:

$$\vartheta^{(m)}_{i,s,t} = \sigma\left(\frac{(\mathbf{q}^{(m)}_{s,t})^\top \mathbf{k}^{(m)}_{i,s,t}}{\sqrt{d_h}}\right), \qquad \mathbf{h}^{(m)}_{i,s,t} = \mathbf{v}^{(m)}_{i,s,t} + \vartheta^{(m)}_{i,s,t} \odot \mathbf{v}^{(m)}_{i,s,t}, \tag{34}$$

and the final modulation-sensitive fused state is obtained by output projection over all heads,

$$\mathbf{H}^{(t)}_{i,s} = \mathbf{W}_o\big[\mathbf{h}^{(1)}_{i,s,t} \oplus \cdots \oplus \mathbf{h}^{(n_{\mathrm{head}})}_{i,s,t}\big]. \tag{35}$$

Here $\oplus$ concatenates features, and $\mathbf{W}_o \in \mathbb{R}^{d_h \times (n_{\mathrm{head}} d_h)}$ projects the concatenated heads to the fused-state width. These equations show that posterior-conditioned state information and temporal exogenous context enter the same gated fusion interface before the stable-adaptive split.

C.4 Expanded prototype-routed expert memory for stable-adaptive forecasting

The prototype-routed expert memory paragraph in Section 3.2 summarizes memory definition, component-specific memory readout, and stable-adaptive forecasting with high-level operators. The expanded formulation specifies adaptive routing logits, weather-controlled expert gain, component injection, node-wise temporal readout, and the final stable-adaptive forecasting interface.

The adaptive component receives city-level weather information through a controller state that is shared across nodes at the same time step,

$$\mathbf{u}_s = \Psi_w(\mathbf{W}_s) \in \mathbb{R}^{d_u}, \tag{36}$$

and component-specific occupancies are obtained from prototype routing logits,

$$\begin{aligned}
\boldsymbol{\ell}_{i,s}^{\mathrm{inv}} &= \frac{\mathbf{C}\phi_{\mathrm{inv}}(\mathbf{H}_{i,s}^{(t)})}{\tau_p\sqrt{d_c}}, \\
\boldsymbol{\ell}_{i,s}^{\mathrm{var}} &= \frac{\mathbf{C}\phi_{\mathrm{var}}(\mathbf{H}_{i,s}^{(t)})}{\tau_p\sqrt{d_c}}, \\
\boldsymbol{\ell}_{i,s}^{\mathrm{var,weather}} &= \boldsymbol{\ell}_{i,s}^{\mathrm{var}} + \mathbf{A}_u\mathbf{u}_s, \qquad \boldsymbol{\delta}_t = \mathbf{A}_\pi\boldsymbol{\pi}_t, \\
\widetilde{\boldsymbol{\ell}}_{i,s}^{\mathrm{var}} &= \boldsymbol{\ell}_{i,s}^{\mathrm{var,weather}} + \boldsymbol{\delta}_t, \\
\boldsymbol{\alpha}_{i,s}^{\mathrm{inv}} &= \mathrm{Entmax}(\boldsymbol{\ell}_{i,s}^{\mathrm{inv}}), \qquad \widetilde{\boldsymbol{\alpha}}_{i,s}^{\mathrm{var}} = \mathrm{Entmax}(\widetilde{\boldsymbol{\ell}}_{i,s}^{\mathrm{var}}).
\end{aligned} \tag{37}$$

Here $\tau_p > 0$ is the prototype-routing temperature, $\mathbf{C} \in \mathbb{R}^{K\times d_c}$ has prototype rows, each scoring projection returns a d_c-column, and logits and occupancies are K-columns. Thus $\mathbf{A}_u \in \mathbb{R}^{K\times d_u}$ and $\mathbf{A}_\pi \in \mathbb{R}^{K\times d_\pi}$. The weather-conditioned logits contain $\mathbf{A}_u\mathbf{u}_s$ once; the task offset $\boldsymbol{\delta}_t$ contains only the posterior routing contribution. Each expert is parameterized as a low-rank operator,

$$\mathcal{E}_k(\mathbf{H}) = \mathbf{U}_k\,\sigma_{\mathcal{E}}\big(\mathbf{V}_k^\top\mathbf{H} + \mathbf{b}_k\big), \qquad \mathbf{U}_k \in \mathbb{R}^{d_h\times r},\ \mathbf{V}_k \in \mathbb{R}^{d_h\times r},\ r \ll d_h, \tag{38}$$

where $\sigma_{\mathcal{E}}$ denotes the elementwise expert activation (distinguished from the sigmoid gates), r is the expert rank and $\mathbf{b}_k \in \mathbb{R}^r$. The controller-dependent gain and shift are d_h-vectors, with $\mathbf{W}_k^\gamma, \mathbf{W}_k^\beta \in \mathbb{R}^{d_h\times d_u}$:

$$\boldsymbol{\gamma}_k(\mathbf{u}_s) = \sigma(\mathbf{W}_k^\gamma\mathbf{u}_s + \mathbf{b}_k^\gamma), \qquad \boldsymbol{\beta}_k(\mathbf{u}_s) = \mathbf{W}_k^\beta\mathbf{u}_s + \mathbf{b}_k^\beta, \tag{39}$$

so that the weather-modulated expert response becomes

$$\widetilde{\mathcal{E}}_k(\mathbf{H}_{i,s}^{(t)}, \mathbf{u}_s) = \boldsymbol{\gamma}_k(\mathbf{u}_s) \odot \mathcal{E}_k(\mathbf{H}_{i,s}^{(t)}) + \boldsymbol{\beta}_k(\mathbf{u}_s), \tag{40}$$

The corresponding expert-response stack and component memory responses are

$$\begin{aligned}
\mathbf{R}_{\mathcal{E}}^{\mathrm{weather}}(\mathbf{H}_{i,s}^{(t)}, \mathbf{u}_s) &= \begin{bmatrix} \widetilde{\mathcal{E}}_1(\mathbf{H}_{i,s}^{(t)}, \mathbf{u}_s)^\top \\ \vdots \\ \widetilde{\mathcal{E}}_K(\mathbf{H}_{i,s}^{(t)}, \mathbf{u}_s)^\top \end{bmatrix} \in \mathbb{R}^{K\times d_h}, \\
\widetilde{\mathbf{R}}_{\mathcal{E}}^{(t)}(\mathbf{H}_{i,s}^{(t)}, \mathbf{u}_s) &= \mathrm{Diag}(\boldsymbol{\eta}_t)\,\mathbf{R}_{\mathcal{E}}^{\mathrm{weather}}(\mathbf{H}_{i,s}^{(t)}, \mathbf{u}_s), \\
\mathbf{b}_{i,s}^{\mathrm{inv}} &= \mathbf{C}^\top\boldsymbol{\alpha}_{i,s}^{\mathrm{inv}}, \\
\mathbf{b}_{i,s}^{\mathrm{var}} &= \widetilde{\mathbf{R}}_{\mathcal{E}}^{(t)}(\mathbf{H}_{i,s}^{(t)}, \mathbf{u}_s)^\top\widetilde{\boldsymbol{\alpha}}_{i,s}^{\mathrm{var}}.
\end{aligned} \tag{41}$$

The superscript $\mathrm{weather}$ denotes the intermediate stack, and the tilde with task superscript (t) denotes its final posterior-scaled form. The gain scales the entire weather-modulated response, including its shift term. Posterior scaling occurs once, before routing-weighted aggregation. The injection matrices satisfy $\mathbf{W}_{\mathrm{sta}} \in \mathbb{R}^{d_h\times d_c}$ and $\mathbf{W}_a \in \mathbb{R}^{d_h\times d_h}$, so both residual sums have width d_h. These memory responses are injected into component-specific states through

$$\begin{aligned}
\widetilde{\mathbf{H}}_{i,s}^{\mathrm{inv}} &= \mathrm{LN}\Big(\mathbf{H}_{i,s}^{(t)} + \mathbf{W}_{\mathrm{sta}}\mathbf{b}_{i,s}^{\mathrm{inv}}\Big), \\
\widetilde{\mathbf{H}}_{i,s}^{\mathrm{var}} &= \mathrm{LN}\Big(\mathbf{H}_{i,s}^{(t)} + \mathbf{W}_a\mathbf{b}_{i,s}^{\mathrm{var}}\Big), \\
\mathbf{f}_{i,s}^{\mathrm{inv}} &= \mathrm{Head}_{\mathrm{inv}}\Big(\widetilde{\mathbf{H}}_{i,s}^{\mathrm{inv}}\Big), \qquad \mathbf{f}_{i,s}^{\mathrm{var}} = \mathrm{Head}_{\mathrm{var}}\Big(\widetilde{\mathbf{H}}_{i,s}^{\mathrm{var}}\Big).
\end{aligned} \tag{42}$$

Here LN denotes layer normalization, and $\mathrm{Head}_{\mathrm{inv}}$ and $\mathrm{Head}_{\mathrm{var}}$ are the local feature transforms. Their composition with the upstream state construction gives the branch representations denoted by h_{inv} and h_{var} in Section 3.3. Both branch representations inherit the shared modulation, with task conditioning suppressed in the invariant notation. After node-wise mechanism features are obtained, the forecasting interface preserves spatial resolution by applying shared temporal readouts independently at each node. For node i and forecast step q, the stable component uses the future timestamp query to attend over the stable history,

$$\begin{aligned} \mathbf{Q}_q &= \mathbf{W}_Q \psi_T(\mathbf{T}_q^+), \qquad \mathbf{K}_{i,s} = \mathbf{W}_K \mathbf{f}_{i,s}^{\mathrm{inv}}, \qquad \mathbf{V}_{i,s} = \mathbf{W}_V \mathbf{f}_{i,s}^{\mathrm{inv}}, \\ a_{i,q,s}^{\mathrm{sta}} &= \mathrm{Softmax}_{s=1:S}\left(\frac{\mathbf{Q}_q^\top \mathbf{K}_{i,s}}{\sqrt{d_h}}\right), \qquad \hat{y}_{i,q}^{\mathrm{inv}} = \mathrm{Out}_{\mathrm{sta}}\left(\sum_{s=1}^{S} a_{i,q,s}^{\mathrm{sta}} \mathbf{V}_{i,s}\right). \end{aligned} \tag{43}$$

Here ψ_T encodes future timestamp queries, $\mathbf{W}_Q$, $\mathbf{W}_K$, and $\mathbf{W}_V$ are shared projections, and $\mathrm{Out}_{\mathrm{sta}}$ is the stable readout head. All three projected features are d_h-columns; the projection matrices have d_h rows and as many columns as their respective input widths. The adaptive map is implemented as a shared lightweight temporal-convolutional readout over the adaptive history and the broadcast weather controller states,

$$\begin{aligned} \mathbf{z}_i^{\mathrm{var}} &= \mathrm{TempConv}_{\mathrm{var}}\big([\mathbf{f}_{i,1}^{\mathrm{var}} \oplus \mathbf{u}_1, \ldots, \mathbf{f}_{i,S}^{\mathrm{var}} \oplus \mathbf{u}_S]\big)_S, \\ \hat{\mathbf{y}}_{i,1:H}^{\mathrm{var}} &= \mathbf{W}_o^{\mathrm{var}} \mathbf{z}_i^{\mathrm{var}} + \mathbf{b}_o^{\mathrm{var}}, \qquad \hat{\mathbf{y}}_{i,1:H} = \hat{\mathbf{y}}_{i,1:H}^{\mathrm{inv}} + \hat{\mathbf{y}}_{i,1:H}^{\mathrm{var}}. \end{aligned} \tag{44}$$

The input to $\mathrm{TempConv}_{\mathrm{var}}$ is arranged as feature channels by time, with one concatenated feature column per time step; the subscript S selects the output at the last context position. The readout parameters are shared across nodes, which keeps the forecasting interface valid for cities with different graph sizes. Short-timescale environmental adjustment enters through weather-conditioned routing and expert gain, while the stable component remains anchored to prototype coordinates.

Two-stage learning procedure. Algorithm 1 summarizes the source-learning and target-adaptation stages using the objectives defined in Section 3.3.

Algorithm 1 ExoC2T training and target-city prediction

Require: Source data $\mathcal{D}_S$, labeled target support $\mathcal{S}_t$, target city d_t, and exogenous query inputs.
Ensure: Target-city forecasts $\hat{\mathcal{Y}}_t$.

Initialize the shared predictor, posterior encoder, and city discriminator.
Stage A: Stable Source Mechanism Learning
for each source-training step **do**
  Sample a source batch from $\mathcal{D}_S$.
  Compute source prediction loss $\mathcal{L}_{\mathrm{src}}$ and city-wise stabilization $\mathcal{L}_{\mathrm{homeo}}$ in Eq. (15).
  Form $\mathcal{L}_{\mathrm{A}}$ using Eq. (21).
  Update the shared encoders, prototypes, experts, and readouts with AdamW.
end for
Stage B: Posterior-Conditioned Adaptive Modulation
for each target-adaptation step **do**
  Sample source and labeled target-support batches.
  Infer the support-conditioned posterior and its statistics using Eq. (17).
  Compute source predictions; compute target predictions with posterior-conditioned fusion, adaptive routing, and expert responses.
  Compute $\mathcal{L}_{\mathrm{post}}^{(t)}$ in Eq. (18) and $\mathcal{L}_{\mathrm{disc}}$ in Eq. (20).
  Form $\mathcal{L}_{\mathrm{B}}$ with the scheduled discrepancy and posterior weights described in Appendix B.1.
  Reverse the city-classification gradient at the pooled invariant features; retain its minimizing direction for the discriminator.
  Update the shared predictor, posterior encoder, and city discriminator with AdamW.
end for
Query prediction
Infer (μ_t, Σ_t) from $\mathcal{S}_t$ and reuse these statistics for the target queries.
Return $\hat{\mathcal{Y}}_t = f_t^{\mathrm{stat}}(x_t)$ for each exogenous query input x_t.

D Stable-adaptive mechanism separation: local perturbation analysis

This section derives the local perturbation form used to justify the stable-adaptive mechanism separation in Section 3.3.1.

Proposition 1 (Local stable-adaptive perturbation analysis). Let

$$a = h_{\text{inv}}(x), \qquad b = h_{\text{var}}(x, d, z), \tag{45}$$

so that the prediction rule takes the form $y = g(a, b) + \epsilon$. Assume that for each fixed a, the mapping $b \mapsto g(a, b)$ is twice continuously differentiable in a neighborhood of $b = \mathbf{0}$ and that its second derivative with respect to that argument, viewed as a bilinear map, is uniformly bounded by L_g in operator norm. Take b such that the segment $\{\tau b : 0 \le \tau \le 1\}$ lies in this neighborhood. Then the local mechanism output can be expanded into a stable term, a first-order adaptive term, and a quadratic residual, as shown in Eq. (49) and Eq. (51).

Proof of Proposition: A second-order Taylor expansion around $b = \mathbf{0}$ gives

$$g(a, b) = g(a, \mathbf{0}) + J_2 g(a, \mathbf{0})\, b + r(a, b), \tag{46}$$

where $J_2 g(a, \mathbf{0})$ is the Jacobian with respect to the second argument and the remainder satisfies

$$\|r(a, b)\|_2 \le \frac{L_g}{2} \|b\|_2^2. \tag{47}$$

Let

$$y^\star(x, d, z) = g\big(h_{\text{inv}}(x), h_{\text{var}}(x, d, z)\big) \tag{48}$$

denote the local mechanism output used in this perturbation analysis. Substituting back yields the local stable-adaptive separation

$$y^\star(x, d, z) = y_{\text{loc}}^{\text{inv}} + y_{\text{loc}}^{\text{var}} + r_t, \tag{49}$$

with

$$y_{\text{loc}}^{\text{inv}} = g\big(h_{\text{inv}}(x), \mathbf{0}\big), \qquad y_{\text{loc}}^{\text{var}} = J_2 g\big(h_{\text{inv}}(x), \mathbf{0}\big)\, h_{\text{var}}(x, d, z), \tag{50}$$

and residual $r_t = r(h_{\text{inv}}(x), h_{\text{var}}(x, d, z))$. Therefore,

$$\big\| y^\star(x, d, z) - y_{\text{loc}}^{\text{inv}} - y_{\text{loc}}^{\text{var}} \big\|_2 \le \frac{L_g}{2} \, \|h_{\text{var}}(x, d, z)\|_2^2 \,. \tag{51}$$

Corollary: Stable core as a local approximation

Eq. (51) formalizes the adaptive component as a structured perturbation around a stable predictive core:

$$\big\| y^\star(x, d, z) - y_{\text{loc}}^{\text{inv}} - y_{\text{loc}}^{\text{var}} \big\|_2 \le \frac{L_g}{2} \, \|h_{\text{var}}(x, d, z)\|_2^2 \,. \tag{52}$$

Here $y^\star(x, d, z)$ denotes the local mechanism used in the Taylor expansion. The implemented network outputs $\hat{y} = \hat{y}^{\text{inv}} + \hat{y}^{\text{var}}$ using its learned readouts. The remainder bounds the error of the local mechanism expansion. The residual is second order in the adaptive component, so small target-specific modulation leads to a small local approximation error.

Proposition 2 (Plasticity-separation bound). Consider the local block-gradient update in Eq. (16). If $\mathbf{M}_t$ is full rank on the adaptive subspace, the ratio between stable and adaptive update magnitudes is controlled by ρ_{inv} and the minimum singular value of $\mathbf{M}_t$, as in Eq. (56). Under the additional gradient separation in Eq. (57), the two-sided ratio in Eq. (59) follows.

Proof of Proposition: Writing the analytical update of Eq. (16) in block form,

$$\Delta\theta_{\text{inv}} = -\eta \rho_{\text{inv}} \nabla_{\theta_{\text{inv}}} \mathcal{L}_t, \qquad \Delta\theta_{\text{var}} = -\eta \mathbf{M}_t \nabla_{\theta_{\text{var}}} \mathcal{L}_t, \tag{53}$$

which gives the one-sided update controls

$$\|\Delta\theta_{\text{inv}}\|_2 \le \eta \rho_{\text{inv}} \, \|\nabla_{\theta_{\text{inv}}} \mathcal{L}_t\|_2 \,, \qquad \|\Delta\theta_{\text{var}}\|_2 \le \eta \|\mathbf{M}_t\|_2 \, \|\nabla_{\theta_{\text{var}}} \mathcal{L}_t\|_2 \,. \tag{54}$$

If $\mathbf{M}_t$ is full rank on the adaptive subspace with $\sigma_{\min}(\mathbf{M}_t) > 0$, the adaptive update also satisfies

$$\|\Delta\theta_{\text{var}}\|_2 \geq \eta\sigma_{\min}(\mathbf{M}_t)\,\|\nabla_{\theta_{\text{var}}}\mathcal{L}_t\|_2\,. \tag{55}$$

Consequently, whenever $\nabla_{\theta_{\text{var}}}\mathcal{L}_t \neq \mathbf{0}$,

$$\frac{\|\Delta\theta_{\text{inv}}\|_2}{\|\Delta\theta_{\text{var}}\|_2} \leq \frac{\rho_{\text{inv}}}{\sigma_{\min}(\mathbf{M}_t)} \cdot \frac{\|\nabla_{\theta_{\text{inv}}}\mathcal{L}_t\|_2}{\|\nabla_{\theta_{\text{var}}}\mathcal{L}_t\|_2}. \tag{56}$$

If there exist constants $G_{\text{inv}}, G_{\text{var}} > 0$ such that

$$\|\nabla_{\theta_{\text{inv}}}\mathcal{L}_t\|_2 \leq G_{\text{inv}}, \qquad \|\nabla_{\theta_{\text{var}}}\mathcal{L}_t\|_2 \geq G_{\text{var}}, \tag{57}$$

then

$$\|\Delta\theta_{\text{inv}}\|_2 \leq \eta\rho_{\text{inv}}G_{\text{inv}}, \qquad \|\Delta\theta_{\text{var}}\|_2 \geq \eta\sigma_{\min}(\mathbf{M}_t)G_{\text{var}}, \tag{58}$$

and therefore

$$\frac{\|\Delta\theta_{\text{inv}}\|_2}{\|\Delta\theta_{\text{var}}\|_2} \leq \frac{\rho_{\text{inv}}G_{\text{inv}}}{\sigma_{\min}(\mathbf{M}_t)G_{\text{var}}}. \tag{59}$$

Corollary: Homeostatic stabilization localizes target updates

For this block-gradient model, under the stated gradient bounds, the stable component evolves on a much slower scale when the full ratio bound $\rho_{\text{inv}}G_{\text{inv}}/[\sigma_{\min}(\mathbf{M}_t)G_{\text{var}}]$ is much smaller than one:

$$\frac{\|\Delta\theta_{\text{inv}}\|_2}{\|\Delta\theta_{\text{var}}\|_2} \leq \frac{\rho_{\text{inv}}G_{\text{inv}}}{\sigma_{\min}(\mathbf{M}_t)G_{\text{var}}}. \tag{60}$$

Under this quantitative condition, stable-block increments are small relative to adaptive-block increments in Eq. (16).

E Support-conditioned latent state inference and adaptive gain

This section records the posterior formulation behind support-conditioned latent state inference in Section 3.3.2. The posterior defines the following target-conditioned predictive distribution,

$$p_\theta(y \mid x, \mathcal{S}_t, d_t) = \int p_\theta(y \mid x, d_t, z_t)q_\psi(z_t \mid \mathcal{S}_t, d_t)\,dz_t. \tag{61}$$

Equation (61) describes the analytical posterior predictive distribution. Point prediction uses the posterior mean and covariance directly. For a fixed adapted parameter state θ and covariance Σ_t, define $f_{t,z}(x) := f_\theta(x, d_t, z; \Sigma_t)$ and $f_t^{\text{stat}}(x) := f_{t,\mu_t}(x)$. For $f_{t,z}$, evaluate the forward modulation maps as $\boldsymbol{\pi}(z) = G_\pi(z)$ and $\boldsymbol{\eta}(z) = G_\eta(z, \Sigma_t)$, and use $\boldsymbol{\delta}(z) = \mathbf{A}_\pi\boldsymbol{\pi}(z)$ in routing. These values enter the existing fusion and expert-response operators. The comparison holds θ, Σ_t, and the exogenous input fixed; $\mathbf{M}_t$ acts in the separate analytical parameter-update rule. Substituting $z = \mu_t$ recovers the statistics-conditioned forward pass. We suppress the covariance argument in expressions such as $f_\theta(x, d_t, z)$ below. The posterior-statistics maps are

$$\boldsymbol{\pi}_t = G_\pi(\mu_t), \qquad \mathbf{M}_t = \mathrm{Diag}(\sigma(G_\mu(\mu_t) - G_\Sigma(\mathrm{diag}(\Sigma_t))))\,. \tag{62}$$

Here diag extracts a matrix diagonal as a vector and Diag constructs a diagonal matrix from a vector. For the parameter-update matrix in the analytical model of Eq. (16), G_μ and G_Σ have p_{var} outputs. When each output of G_Σ is coordinatewise nondecreasing, increasing any posterior variance coordinate makes each diagonal gain non-increasing, with μ_t and the gain-map parameters held fixed. The nonnegative diagonal weights in Eq. (71) satisfy this condition. The expert-response gain $\boldsymbol{\eta}_t \in \mathbb{R}^K$ is separately named below; no equality between $\mathbf{M}_t$ and $\mathrm{Diag}(\boldsymbol{\eta}_t)$ is assumed.

Proposition 3 (Posterior error bound). Let

$$\bar{f}_t(x) = \mathbb{E}_{q_\psi(z_t|\mathcal{S}_t,d_t)}\left[f_\theta(x, d_t, z_t)\right] \tag{63}$$

be the analytical posterior-averaged predictor, distinct from f_t^{stat}, and let $z_t^\star$ denote the latent state that would specialize the target predictor under complete target-state information. Assume that for each target city d_t, the mapping $z \mapsto f_\theta(x, d_t, z)$ is L_z-Lipschitz uniformly in x, so that

$$\|f_\theta(x, d_t, z) - f_\theta(x, d_t, z')\|_2 \leq L_z\|z - z'\|_2. \tag{64}$$

Under the Gaussian posterior in Eq. (17), the posterior-conditioned predictor satisfies the error bound in Eq. (68).

Proof of Proposition: Jensen's inequality and Eq. (64) give

$$\begin{aligned}\left\|\bar{f}_t(x) - f_\theta(x, d_t, z_t^\star)\right\|_2 &= \left\|\mathbb{E}_{q_\psi}\left[f_\theta(x, d_t, z) - f_\theta(x, d_t, z_t^\star)\right]\right\|_2 \\ &\le \mathbb{E}_{q_\psi}\left[\|f_\theta(x, d_t, z) - f_\theta(x, d_t, z_t^\star)\|_2\right] \\ &\le L_z\, \mathbb{E}_{q_\psi}\left[\|z - z_t^\star\|_2\right].\end{aligned} \tag{65}$$

For the Gaussian posterior in Eq. (17),

$$q_\psi(z_t \mid \mathcal{S}_t, d_t) = \mathcal{N}(\mu_t, \Sigma_t), \tag{66}$$

and therefore

$$\mathbb{E}_{q_\psi}\left[\|z - z_t^\star\|_2\right] \le \|\mu_t - z_t^\star\|_2 + \sqrt{\mathrm{tr}(\Sigma_t)}. \tag{67}$$

Combining Eq. (65) and Eq. (67) yields

$$\left\|\bar{f}_t(x) - f_\theta(x, d_t, z_t^\star)\right\|_2 \le L_z\left(\|\mu_t - z_t^\star\|_2 + \sqrt{\mathrm{tr}(\Sigma_t)}\right). \tag{68}$$

Corollary: Posterior mean and covariance control adaptation error

Eq. (68) separates target posterior error into a bias term governed by the posterior mean and an uncertainty term governed by posterior covariance:

$$\left\|\bar{f}_t(x) - f_\theta(x, d_t, z_t^\star)\right\|_2 \le L_z\left(\|\mu_t - z_t^\star\|_2 + \sqrt{\mathrm{tr}(\Sigma_t)}\right). \tag{69}$$

The adaptation gap of $\bar{f}_t$ is controlled by the posterior mean error and covariance trace.

For the statistics-conditioned point predictor, the same L_z-Lipschitz assumption in Eq. (64) gives the following connection, conditional on the same adapted θ and Σ_t:

$$\begin{aligned}\|f_t^{\mathrm{stat}}(x) - \bar{f}_t(x)\|_2 &= \left\|\mathbb{E}_{z\sim q_\psi}[f_{t,\mu_t}(x) - f_{t,z}(x)]\right\|_2 \\ &\le \mathbb{E}_{z\sim q_\psi}\|f_{t,\mu_t}(x) - f_{t,z}(x)\|_2 \\ &\le L_z\, \mathbb{E}_{z\sim q_\psi}\|\mu_t - z\|_2 \le L_z\sqrt{\mathrm{tr}(\Sigma_t)}.\end{aligned}$$

The first inequality is the norm triangle inequality under expectation; the last follows from Cauchy–Schwarz and $\mathbb{E}\|z - \mu_t\|_2^2 = \mathrm{tr}(\Sigma_t)$. The Euclidean norm applies to vectorized forecasts when the output is a tensor. This bounds the difference between the two predictors. The Lipschitz condition must cover the latent values in the Gaussian expectation, with task covariance held fixed.

The routed memory uses these posterior statistics through routing shifts and expert gains. A concrete modulation rule consistent with Eq. (62) is

$$\begin{aligned}\widetilde{\boldsymbol{\alpha}}_{i,s}^{\mathrm{var}} &= \mathrm{Entmax}(\boldsymbol{\ell}_{i,s}^{\mathrm{var,weather}} + \boldsymbol{\delta}_t), \\ \widetilde{\mathbf{R}}_{\mathcal{E}}^{(t)}(\mathbf{H}_{i,s}^{(t)}, \mathbf{u}_s) &= \mathrm{Diag}(\boldsymbol{\eta}_t)\, \mathbf{R}_{\mathcal{E}}^{\mathrm{weather}}(\mathbf{H}_{i,s}^{(t)}, \mathbf{u}_s).\end{aligned} \tag{70}$$

Here $\boldsymbol{\delta}_t = \mathbf{A}_\pi \boldsymbol{\pi}_t$ and $\boldsymbol{\eta}_t = G_\eta(\mu_t, \Sigma_t) \in \mathbb{R}^K$, where G_η is the posterior-to-expert-gain map. This equation restates the single gain application in Eqs. (37) and (41). Weather remains explicit in the intermediate logits and response stack. The parameter-update matrix $\mathbf{M}_t$ retains its separate role in the analytical update model of Eq. (16).

Proposition 4 (Uncertainty-controlled metaplasticity). Under the variance-aware gain parameterization in Eq. (71), with nonnegative variance weights, each diagonal adaptive gain is monotone non-increasing with respect to the corresponding posterior variance.

Proof of Proposition: A concrete uncertainty-aware metaplasticity operator consistent with Eq. (62) is

$$\mathbf{M}_t = \mathrm{Diag}(\sigma(W_\mu \mu_t - W_\Sigma\, \mathrm{diag}(\Sigma_t) + \mathbf{b}_m)), \tag{71}$$

Here the diagonal W_μ, W_Σ form is the coordinate-aligned case $p_{\text{var}} = d_z$; both matrices have non-negative entries and $\mathbf{b}_m$ has length p_{var}. The general generators in Eq. (62) map their d_z-dimensional inputs to p_{var} update coordinates. The expert count K specifies the separate response-gain space. For the j-th diagonal entry,

$$\frac{\partial [\mathbf{M}_t]_{jj}}{\partial [\Sigma_t]_{jj}} = -\sigma'(\cdot)\,[W_\Sigma]_{jj} \le 0. \tag{72}$$

Corollary: Posterior uncertainty yields conservative plasticity

Eq. (72) implies that larger posterior uncertainty produces a smaller adaptive gain:

$$\frac{\partial [\mathbf{M}_t]_{jj}}{\partial [\Sigma_t]_{jj}} \le 0. \tag{73}$$

With the posterior mean, gain-map parameters, and current gradient held fixed, this monotonicity reduces the allowed adaptive gain and the corresponding update-norm upper bound in the block-gradient model of Eq. (54).

A conditional PAC-Bayes analysis relates posterior fitting to generalization within a fixed predictive family. For this calculation, fix θ, Σ_t, and the city-conditioned prior using information independent of the support sample, and let $\mathcal{S}_t$ contain $n = K_t \ge 2$ independent draws from the target distribution P_t. Infer $q_\psi = \mathcal{N}(\mu_t(\mathcal{S}_t), \Sigma_t)$ within this fixed-covariance family. All risks in this paragraph use a bounded analysis loss $\ell_{\text{PB}} \in [0,1]$; R_t denotes its population risk and $\hat{R}_{\mathcal{S}_t}$ its support-sample average. Let

$$f_{t,z}(x) = f_\theta(x, d_t, z), \qquad z \sim q_\psi(z_t \mid \mathcal{S}_t, d_t), \tag{74}$$

and define the Gibbs risk and empirical Gibbs risk as

$$R_G(q_\psi) = \mathbb{E}_{z\sim q_\psi}\left[R_t(f_{t,z})\right], \qquad \hat{R}_G(q_\psi) = \mathbb{E}_{z\sim q_\psi}\left[\hat{R}_{\mathcal{S}_t}(f_{t,z})\right]. \tag{75}$$

For $0 < \delta < 1$, the PAC-Bayes binary-kl bound and Pinsker's inequality give, with probability at least $1-\delta$ over $\mathcal{S}_t$, simultaneously for posteriors with finite KL divergence,

$$R_G(q_\psi) \le \hat{R}_G(q_\psi) + \sqrt{\frac{D_{\text{KL}}(q_\psi(z_t \mid \mathcal{S}_t, d_t) \,\|\, p_\psi(z_t \mid d_t)) + \log\frac{2\sqrt{n}}{\delta}}{2n}}. \tag{76}$$

The binary-kl bound follows from the exponential moment bound $2\sqrt{n}$ for independent bounded observations, change of measure, and Markov's inequality; Pinsker gives the displayed square-root form. For $\bar{f}_t(x) = \mathbb{E}_{z\sim q_\psi}[f_{t,z}(x)]$, convexity of ℓ_{PB} in its prediction argument yields $R_t(\bar{f}_t) \le R_G(q_\psi)$. More generally, if ℓ_{PB} has an L_{sm}-Lipschitz gradient throughout the convex hull of the predictions and $f_{t,z}$ is L_z-Lipschitz in z, then

$$R_t(\bar{f}_t) \le R_G(q_\psi) + \lambda_\Sigma \operatorname{tr}(\Sigma_t), \tag{77}$$

where $\lambda_\Sigma = L_{\text{sm}} L_z^2/2$: the smoothness remainder is bounded by $L_{\text{sm}}\mathbb{E}\|f_{t,z} - \bar{f}_t\|_2^2/2$, and this predictive variance is at most $L_z^2 \operatorname{tr}(\Sigma_t)$. Combining Eq. (76) and Eq. (77) yields the PAC-Bayes-style interpretation

$$R_t(\bar{f}_t) \le \hat{R}_G(q_\psi) + \sqrt{\frac{D_{\text{KL}}(q_\psi(z_t \mid \mathcal{S}_t, d_t) \,\|\, p_\psi(z_t \mid d_t)) + \log\frac{2\sqrt{n}}{\delta}}{2n}} + \lambda_\Sigma \operatorname{tr}(\Sigma_t). \tag{78}$$

Under the fixed-family and sampling conditions above, Eq. (78) controls population prediction risk through empirical fitting, posterior complexity, and predictive uncertainty. These terms motivate the fitting and regularization roles in the training objective.

F Target-risk decomposition with discrepancy regularization

This section gives the target-risk decomposition with discrepancy regularization used in Eq. (19). The target risk $\mathcal{R}_{\text{tar}}$ refers to the analytical posterior-averaged predictor $\bar{f}_t$ for target problem $(d_t, \mathcal{S}_t, \mathcal{Q}_t)$:

$$\mathcal{R}_{\text{tar}} := \mathbb{E}_{(d_t,\mathcal{S}_t,\mathcal{Q}_t)}\left[R_{\mathcal{Q}_t}(\bar{f}_t)\right]. \tag{79}$$

Proposition 5 (Discrepancy-regularized target-risk decomposition). Assume the support-conditioned posterior gap in Eq. (68), the cross-predictor transfer condition in Eq. (82), and the coupling condition in Eq. (83). Then the expected target risk is controlled by source risk, invariant discrepancy, posterior error, their product term, and a transfer residual, matching Eq. (19).

Proof of Proposition: Let

$$f_t^\star(x) = f_\theta(x, d_t, z_t^\star) \tag{80}$$

denote the oracle target predictor specialized by the latent state $z_t^\star$. For the MSE analysis, assume that the residuals of the compared predictions are uniformly bounded in Frobenius norm by B_{res}. Then $\ell(a, y) = \|a - y\|_F^2/(HN)$ is Lipschitz on this prediction domain with $L_\ell = 2B_{\mathrm{res}}/(HN)$. For any loss ℓ that is L_ℓ-Lipschitz in its prediction argument,

$$R_{\mathcal{Q}_t}(\bar{f}_t) \le R_{\mathcal{Q}_t}(f_t^\star) + L_\ell \mathbb{E}_{x\sim\mathcal{Q}_t}\left[\left\|\bar{f}_t(x) - f_t^\star(x)\right\|_2\right]. \tag{81}$$

The second term is the posterior-induced adaptation gap for the sampled target problem. Eq. (68) controls it through posterior mean error and posterior uncertainty. The quantity $\mathcal{E}_{\mathrm{post}}(\mathcal{S}_t, d_t)$ in the main text is a nonnegative loss-scale envelope for this contribution under the coupling condition below. For example, the uniform Lipschitz bound supplies the envelope $L_\ell L_z(\|\mu_t - z_t^\star\|_2 + \sqrt{\mathrm{tr}\,\Sigma_t})$. This analytical envelope measures deviation from the oracle predictor.

The first term in Eq. (81) is the oracle target risk for the same realization of $(d_t, \mathcal{S}_t, \mathcal{Q}_t)$. Let $\phi = h_{\mathrm{inv}}$ be the invariant representation and suppose that the source predictor $\bar{f}_s$ and the oracle target predictor $f_t^\star$ belong to a common hypothesis family over the invariant space. We assume the following cross-predictor transfer condition for this source–oracle pair:

$$R_{\mathcal{Q}_t}(f_t^\star) \le R_s(\bar{f}_s) + \mathcal{D}_{\mathrm{inv}}(P_s^\phi, P_t^\phi) + \mathcal{E}_{\mathrm{app}}, \tag{82}$$

Here $\mathcal{D}_{\mathrm{inv}}$ is a nonnegative discrepancy on the common invariant space, and $\mathcal{E}_{\mathrm{app}}$ is a nonnegative transfer residual accounting for approximation, joint error, and source–oracle predictor mismatch. This condition supplies the transfer step in the risk decomposition.

Posterior correction is applied after invariant representation learning, so source-target mismatch in ϕ can also affect the posterior correction term. We model this dependence with the following coupling condition:

$$L_\ell \mathbb{E}_{x\sim\mathcal{Q}_t}\left[\left\|\bar{f}_t(x) - f_t^\star(x)\right\|_2\right] \le \left(1 + \gamma \mathcal{D}_{\mathrm{inv}}(P_s^\phi, P_t^\phi)\right) \mathcal{E}_{\mathrm{post}}(\mathcal{S}_t, d_t), \tag{83}$$

for some coupling constant $\gamma > 0$. This coupling assumption allows the posterior-gap envelope to scale linearly with invariant mismatch. The uniform envelope above also satisfies this condition for nonnegative discrepancy. Substituting Eq. (82) and Eq. (83) into Eq. (81) gives a conditional risk-control relation for the sampled target problem. Taking the same marginal expectation over $(d_t, \mathcal{S}_t, \mathcal{Q}_t)$ as in Eq. (79) yields Eq. (19); the explicit expectation in the main text is over $(d_t, \mathcal{S}_t)$ because the query-set expectation is already absorbed into $R_{\mathcal{Q}_t}(\cdot)$.

Corollary: Stage B: Posterior-Conditioned Adaptive Modulation

Eq. (19) implies that target adaptation should jointly reduce invariant mismatch and posterior error:

$$\mathcal{R}_{\mathrm{tar}} \le \mathbb{E}_{(d_t,\mathcal{S}_t)}\left[R_s + \mathcal{D}_{\mathrm{inv}} + \mathcal{E}_{\mathrm{post}} + \gamma \mathcal{D}_{\mathrm{inv}} \mathcal{E}_{\mathrm{post}} + \mathcal{E}_{\mathrm{app}}\right]. \tag{84}$$

The Stage B objective follows these roles: City-Invariance Regularization and Prototype Distribution Alignment provide empirical signals for invariant mismatch, while posterior correction fits support evidence and regularizes uncertainty. These terms form a training surrogate for the roles identified in the risk bound.

For the actual statistics-conditioned output, define $\mathcal{R}_{\mathrm{tar}}^{\mathrm{stat}} := \mathbb{E}_{(d_t,\mathcal{S}_t,\mathcal{Q}_t)}[R_{\mathcal{Q}_t}(f_t^{\mathrm{stat}})]$. Under the same prediction-argument L_ℓ-Lipschitz loss condition used in Eq. (81), the connection in Appendix E yields

$$\mathcal{R}_{\mathrm{tar}}^{\mathrm{stat}} \le \mathcal{R}_{\mathrm{tar}} + \mathbb{E}_{(d_t,\mathcal{S}_t)}\left[L_\ell L_z \sqrt{\mathrm{tr}(\Sigma_t)}\right].$$

The query expectation is inside each risk. Under the stated Lipschitz conditions, the posterior-averaged risk bound connects to the statistics-conditioned predictor through this discrepancy term. The risk expression in Figure 2 abbreviates the $\mathcal{R}_{\mathrm{tar}}$ bound.

Section 3.3.3 implements invariant discrepancy regularization through City-Invariance Regularization and Prototype Distribution Alignment. City-Invariance Regularization uses a gradient-reversal city-classification objective. Its empirical adversarial loss is

$$\mathcal{L}_{\mathrm{adv}}^{\mathrm{inv}} := \mathbb{E}_{(x,d)}\left[\ell_{\mathrm{ce}}\big(D_\nu(\bar{\phi}(x)), d\big)\right]. \tag{85}$$

Let $\bar{\phi}(x) = \mathrm{Pool}(h_{\mathrm{inv}}(x))$ be the pooled invariant feature and let D_ν be a gradient-reversal domain discriminator over the observed cities in the current run, namely the source-city set together with the target city. Let $\boldsymbol{\varpi} = (\varpi_d)_d$ be the empirical domain-prior vector in the adversarial batches, with ϖ_d its city-d probability, and $\mathsf{H}(\boldsymbol{\varpi}) = -\sum_d \varpi_d \log \varpi_d$ denote its entropy. For each city $d \in \mathcal{C} = \mathcal{C}_S^{(t)} \cup \{d_t\}$, let $P_d^{\bar{\phi}}$ denote its pooled-feature distribution. The city-invariance discrepancy depends on these city-specific laws and their prior:

$$\begin{aligned}\mathcal{D}_{\mathrm{cls}}\Big(\{P_d^{\bar{\phi}}\}_{d\in\mathcal{C}}, \boldsymbol{\varpi}\Big) = \mathsf{H}(\boldsymbol{\varpi}) \\ - \inf_{\nu} \mathbb{E}_{(x,d)}\left[\ell_{\mathrm{ce}}\big(D_\nu(\bar{\phi}(x)), d\big)\right].\end{aligned} \tag{86}$$

Here d is drawn with probability ϖ_d, and conditional on d, the pooled feature $\bar{\phi}(x)$ follows $P_d^{\bar{\phi}}$. The discriminator family includes the constant predictor $\boldsymbol{\varpi}$. When invariant features contain no domain information, the optimal discriminator predicts the domain prior and the cross-entropy reaches $\mathsf{H}(\boldsymbol{\varpi})$, so the discrepancy collapses toward zero. Under balanced domain mini-batches, $\mathsf{H}(\boldsymbol{\varpi}) = \log(|\mathcal{C}_S^{(t)}|+1)$. The adversarial term in Eq. (85) estimates the classification risk in Eq. (86), whose discrepancy is defined using the optimized discriminator and prior entropy. With gradient reversal, the discriminator still minimizes classification loss, while the invariant encoder receives the reversed gradient and removes domain information from $\bar{\phi}(x)$.

Prototype Distribution Alignment treats each node–time occupancy $\boldsymbol{\alpha}_{i,s}^{\mathrm{inv}} \in \Delta^{K-1}$ as one observation, where Δ^{K-1} is the probability simplex. For a batch of B windows with N nodes and S time steps, the routing tensor is flattened to $BNS \times K$ without within-window averaging. Source and target moments are computed separately, giving each node–time observation equal weight; empirical covariance uses the sample-count-minus-one denominator (clamped to one for a singleton). Let P_s^α and P_t^α be the resulting source and target occupancy laws, distinct from the feature distributions P_s^ϕ, P_t^ϕ in the risk bound. For an occupancy law P, define

$$\mathbf{m}_\alpha(P) = \mathbb{E}_{\boldsymbol{a}\sim P}\left[\boldsymbol{a}\right]. \tag{87}$$

The first-order prototype distribution discrepancy is

$$\mathcal{D}_{\mathrm{geo}}^{(1)}(P_s^\alpha, P_t^\alpha) = \|\mathbf{m}_\alpha(P_s^\alpha) - \mathbf{m}_\alpha(P_t^\alpha)\|_2^2. \tag{88}$$

The covariance term records the spread of prototype occupancy:

$$\boldsymbol{\Sigma}_\alpha(P) = \mathrm{Cov}_{\boldsymbol{a}\sim P}\left[\boldsymbol{a}\right], \tag{89}$$

and the second-order prototype distribution discrepancy

$$\mathcal{D}_{\mathrm{geo}}^{(2)}(P_s^\alpha, P_t^\alpha) = \|\boldsymbol{\Sigma}_\alpha(P_s^\alpha) - \boldsymbol{\Sigma}_\alpha(P_t^\alpha)\|_F^2. \tag{90}$$

The composite prototype distribution discrepancy is

$$\mathcal{D}_{\mathrm{geo}}(P_s^\alpha, P_t^\alpha) = \mathcal{D}_{\mathrm{geo}}^{(1)}(P_s^\alpha, P_t^\alpha) + \bar{\lambda}_{\mathrm{cov}}\mathcal{D}_{\mathrm{geo}}^{(2)}(P_s^\alpha, P_t^\alpha), \tag{91}$$

where $\bar{\lambda}_{\mathrm{cov}} \geq 0$ is a fixed internal moment-balance constant absorbed into $\mathcal{L}_{\mathrm{align}}^{\mathrm{inv}}$ and kept fixed in all experiments. Eq. (91) measures mismatch in the shared prototype coordinates induced by the shared prototype anchors of the routed memory. It complements $\mathcal{D}_{\mathrm{cls}}$ by preserving prototype distribution structure alongside city-invariance regularization. City-Invariance Regularization targets domain-identifying directions in pooled invariant features, while Prototype Distribution Alignment constrains how samples occupy the shared prototype coordinates. The moment penalty $\mathcal{L}_{\mathrm{align}}^{\mathrm{inv}}$ is the empirical counterpart of these occupancy-moment differences, with the same choice of occupancy sampling and aggregation. The quantities $\mathcal{D}_{\mathrm{cls}}$ and $\mathcal{D}_{\mathrm{geo}}$ provide empirical surrogates for invariant mismatch; their calibration to $\mathcal{D}_{\mathrm{inv}}$ is an additional theoretical condition.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the exogenous-driven cross-city transfer setting, the lack of historical-flow query input, and the main methodological contributions. The empirical claims are supported by the main tables, ablations, case studies, and the limitations in the appendix.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper includes a concise Limitations paragraph in the Conclusion and a detailed discussion in Appendix B.4.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: Appendices D–F state the assumptions, derivations, corollaries, and intended interpretation for stable-adaptive mechanism separation, support-conditioned latent state inference with adaptive gain, and target-risk decomposition with discrepancy regularization.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper specifies the leave-one-city-out protocol, data sources, target-evidence regimes, metrics, baselines, implementation details, model size, and key hyperparameters in Appendix B. These details provide the information needed to reproduce the main experimental setting.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The supplementary material will provide an anonymized code package and partial preprocessed data.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The experimental setup is described in Section 4 and Appendix B, including data splits, leave-one-city-out evaluation, target-evidence budgets, baselines, metrics, optimizer settings, training schedule, and key hyperparameters.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: The reported results are macro-averaged over target-city runs and include extensive city-level comparisons, ablations, sensitivity analysis, and case studies, but the paper does not report multi-seed error bars or formal statistical significance tests.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No]

   Justification: Appendix B reports the implementation platform, including the GPU type and memory, and also reports model sizes. The current version does not provide per-experiment wall-clock runtime or total project compute.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The study uses public or previously released traffic, weather, POI, and macro-level urban data sources and reports aggregate forecasting results. No personally identifiable information or human-subject intervention is involved.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The introduction and conclusion discuss the positive impact of reducing reliance on dense sensing infrastructure, including more accessible and equitable smart-mobility forecasting for under-instrumented cities. The limitations discuss potential reliability risks when abnormal events or sudden disruptions are not reflected in available external variables.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The work does not release a high-risk generative model, scraped media dataset, or dual-use system requiring special controlled-release safeguards.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The paper credits and cites the original data sources and baselines, including LargeST, WebTRIS, NOAA ISD, OpenStreetMap, RiskOracle, and macro-data providers. The supplementary material includes an asset manifest documenting licenses or terms of use for reused data and code assets.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: The supplementary material will include an anonymized code package and partial preprocessed data. The released files will be documented in the supplementary asset manifest.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The work does not involve crowdsourcing experiments or research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The work does not involve crowdsourcing experiments or research with human subjects, so IRB approval is not applicable.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method, experiments, and analysis do not use LLMs as an original or non-standard methodological component. Any routine writing or formatting assistance, if used, does not affect the core methodology or scientific claims.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.