CRIMEGPT: NUMERICALLY FAITHFUL CRIME FORECASTING WITH SITUATED LARGE LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Crime prediction is essential for proactive community safety and efficient police resource allocation. Beyond strong spatio-temporal periodicity, criminal activity is closely intertwined with regional functional characteristics and neighborhood environments, contextual factors that conventional statistical models largely overlook. Large language models (LLMs) offer a natural pathway to jointly reason over heterogeneous urban descriptors and historical signals within a unified semantic framework. However, their discrete, token-based generation paradigm renders them inherently insensitive to numerical values, a limitation further amplified in crime forecasting where competing multi-source urban signals prevent the model from forming confident numerical outputs. To this end, we propose **CrimeGPT**, which rethinks numerically faithful LLM-based crime prediction along two complementary axes: situating LLMs in urban dynamics and ensuring numerical faithfulness in their outputs. CrimeGPT introduces a Crime Dynamics Encoder that situates the LLM in heterogeneous urban signals. Bidirectional ordinal supervision then organizes numerical-token representations, which are mapped to continuous crime-count estimates by a distribution-guided regression head. Extensive experiments on real-world datasets from multiple cities show that CrimeGPT achieves superior performance over state-of-the-art methods. Our code and data will be made publicly available.

1 INTRODUCTION

Accurately predicting crime is a longstanding challenge at the intersection of urban computing and public safety, with direct implications for resource allocation, early warning systems, and community-level intervention strategies (Chen et al., 2004; Finkelhor, 2008; Huang et al., 2018).

Existing studies have leveraged traditional statistical models and graph neural networks to achieve promising results in tasks such as traffic flow forecasting and crime occurrence prediction (Huang et al., 2018; Rotaru et al., 2022; Liu et al., 2024). However, criminal activity is influenced not only by temporal dynamics, but also by environmental contexts, venue characteristics, and urban functional zoning, as suggested by classical criminology theories such as Routine Activity Theory (Cohen & Felson, 1979) and Broken Window Theory (Wilson & Kelling, 1982). Relying solely on statistical pattern recognition over historical data tends to overlook the latent influence of regional characteristics and neighborhood environments on crime, thereby limiting predictive accuracy.

The rise of large language models (LLMs) has injected new momentum into natural language processing and spurred their adoption across diverse domains. Recent studies have explored applying LLMs to urban spatio-temporal prediction tasks (Li et al., 2024; Jiang et al., 2024), achieving encouraging results. Beyond statistical pattern capture, LLMs offer the additional capacity to learn the causal relationships between auxiliary contextual information and the target prediction task at the semantic level. This naturally motivates the design of situated LLMs that are explicitly anchored in urban context. Nevertheless, recent studies have shown that LLMs remain inherently ill-suited for numerical prediction tasks (Tan et al., 2024; Merrill et al., 2024; Lukasik et al., 2025). This limitation primarily stems from their discrete token-based generation paradigm, which lacks an explicit numerical semantic space for modeling numerical magnitude and ordinal structure. This

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

shortcoming is further compounded in crime count forecasting. In this scenario, heterogeneous multi-source urban signals exert competing influences on the prediction, confounding the model and preventing it from forming confident, well-calibrated numerical outputs. Together, these challenges motivate two coupled requirements for LLM-based crime forecasting: spatio-temporal situatedness and numerical faithfulness.

To address these challenges, we propose CrimeGPT, an end-to-end framework that situates LLMs in urban dynamics while improving numerical faithfulness. CrimeGPT uses a Crime Dynamics Encoder (CDE) to extract heterogeneous urban features and project them into the LLM semantic space, providing spatio-temporal context beyond isolated crime-count sequences. Building on a standardized numerical vocabulary, we identify predictive diffidence, an empirical pattern in which fine-tuned LLMs produce diffuse, high-entropy numerical distributions in complex urban forecasting. We then propose Bidirectional CORN (BiCORN), which transfers conditional ordinal regression to LLM fine-tuning through bidirectional constraints over adjacent numerical-token logits. The resulting ordered representations are mapped to continuous crime-count estimates by a Distribution-Guided Regression Head (DGRH). Extensive experiments across multiple real-world cities show that CrimeGPT consistently outperforms state-of-the-art baselines and validate the complementary roles of urban dynamics encoding, BiCORN supervision, and distribution-guided regression.

2 Related Work

Spatio-temporal Prediction. Early methods relied on statistical models such as ARIMA (Box et al., 2015). Deep learning approaches, like RNN-based architectures (Yu et al., 2017b; Liu et al., 2016), substantially improved temporal dependency modeling. Graph neural networks further advanced the field by capturing irregular spatial topologies, with representative works such as DCRNN (Li et al., 2017), GMAN (Zheng et al., 2020) and MoSSL (Deng et al., 2024) achieving strong performance across traffic forecasting and other urban sensing tasks.

Crime Prediction. As a canonical urban computing task, crime prediction aims to forecast future crime occurrence or count based on historical urban data. A substantial body of research has employed attention-based and graph neural network models to capture temporal crime patterns and inter-region interaction dynamics (Zhao & Tang, 2017; Wu et al., 2020; Rotaru et al., 2022; Xia et al., 2022; Li et al., 2022; Wu et al., 2024), with subsequent works further incorporating auxiliary urban signals such as POI distributions, 311 service requests, and traffic flow to enrich the predictive context (Wang et al., 2016; Huang et al., 2018; Zhao et al., 2022; Du et al., 2026). However, according to classical criminological theories (Cohen & Felson, 1979; Brantingham & Brantingham, 1984; Wilson & Kelling, 1982), crime risk emerges from the semantic configuration of urban environments, encompassing regional functional characteristics, activity node arrangements, and socio-environmental dynamics that are mostly encoded in natural language descriptions of urban regions but remain invisible to models operating exclusively on numerical sequences. This motivates situating large language models in urban context, leveraging their ability to jointly process heterogeneous urban descriptors and historical signals within a unified semantic framework.

LLMs for Numerical Prediction. Large language models, endowed with extensive world knowledge and strong semantic reasoning capabilities acquired during pre-training, have shown great promise for knowledge-intensive prediction tasks (Li et al., 2024; Jiang et al., 2024; Zhao et al., 2025; Chen et al., 2025; Bumb et al., 2025). Nevertheless, their discrete, token-based output mechanism renders them inherently insensitive to precise numerical values, fundamentally limiting their effectiveness on quantitative prediction tasks (Zausinger et al., 2024; Ansari et al., 2024; Tan et al., 2024; Merrill et al., 2024; Park et al., 2025). While several efforts have partially alleviated this limitation through discretizing the prediction space with probability-weighted decoding (Ansari et al., 2024; Pei et al., 2025) or introducing auxiliary Wasserstein (Zausinger et al., 2024) or MSE objectives (Lukasik et al., 2025), they do not explicitly endow LLMs with ordinal relational structure among numerical outputs. xVal replaces discrete numbers with continuous embeddings (Golkar et al., 2023), while Moirai and S^2IP-LLM target generic time-series forecasting (Woo et al., 2024; Pan et al., 2024); in contrast, CrimeGPT preserves autoregressive numerical-token prediction and jointly models heterogeneous urban dynamics and textual semantics. Moreover, these approaches are predominantly designed and evaluated on low-ambiguity tasks such as mathematical reasoning or coarse-grained rating prediction, where unambiguous target signals enable the model to form confident predictions. For

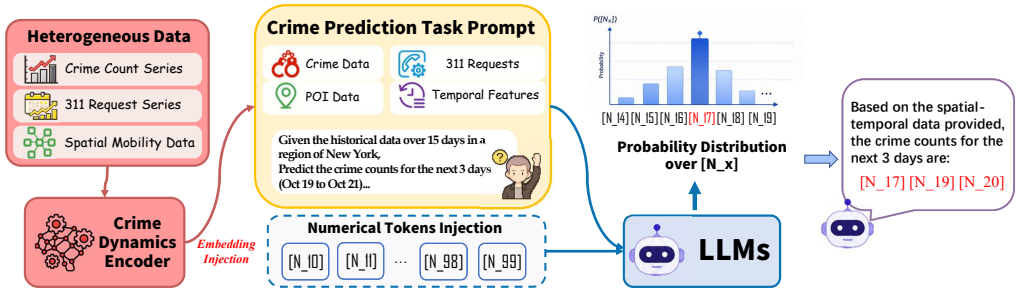


Figure 1: Applying LLMs to crime prediction, we align multi-source data with the LLM semantic space via a Crime Dynamics Encoder and inject numerical tokens into the vocabulary to analyze output distributions.

spatio-temporal tasks such as crime count forecasting, however, the heterogeneous and competing influences of multi-source urban signals induce output uncertainty that existing methods have yet to systematically recognize or address, leaving such tasks structurally underserved.

3 Preliminaries

3.1 Problem Formulation

Let $V = \{v_1, v_2, \ldots, v_N\}$ denote a set of N disjoint geographical regions in a city. For a given time step t (in days), we extract the historical observations over the past H days, including the crime records $\mathbf{X}_c \in \mathbb{R}^{N \times H}$ and the 311 service requests $\mathbf{X}_{311} \in \mathbb{R}^{N \times H}$. Additionally, we define an auxiliary urban context $\mathcal{A}$ to capture complex regional characteristics, which optionally incorporates spatial relationships (e.g., distance or adjacency), traffic flow, time features, and Points of Interest (POIs).

Problem Statement. Given the historical observations $\mathbf{X}_c$, $\mathbf{X}_{311}$, and the auxiliary context $\mathcal{A}$, our objective is to learn a mapping function $\mathcal{F}_\Theta$ with parameters Θ to forecast the crime occurrences for the next P days:

$$\mathbf{Y} = \mathcal{F}_\Theta(\mathbf{X}_c, \mathbf{X}_{311}, \mathcal{A}) \tag{1}$$

where $\mathbf{Y} \in \mathbb{R}^{N \times P}$ denotes the predicted crime counts across all N regions from day $t+1$ to $t+P$.

3.2 Datasets

We collected and curated real-world open-source urban datasets from four major cities: New York City (NYC), Chicago (CHI), Philadelphia (PHI), and Shanghai (SHA). The datasets for the three U.S. cities span a two-year period from 2022 to 2023, while the Shanghai dataset covers the year 2014 due to data availability constraints. Specifically, we gathered historical crime records, 311 service requests, Point of Interest (POI) information, and taxi flow data from the corresponding open data portals of these cities. The raw data was carefully cleaned, categorized, and aggregated according to the city's administrative divisions. Detailed information regarding the datasets and preprocessing procedures can be found in Appendix A.

4 CrimeGPT

4.1 Crime Dynamics Encoder

Crime dynamics reflect both temporal evolution within each region and spatial interactions across regions. Historical crime and 311 requests also follow different scales, while adjacency, geographic proximity, and mobility describe complementary spatial dependencies. The Crime Dynamics Encoder (CDE) aligns these heterogeneous signals before they enter the LLM.

CDE first applies feature-wise Z-score normalization to obtain $\tilde{\mathbf{X}} \in \mathbb{R}^{N \times H \times 2}$. It then stacks three spatio-temporal layers. Each layer uses gated temporal convolutions (Li et al., 2024) to capture

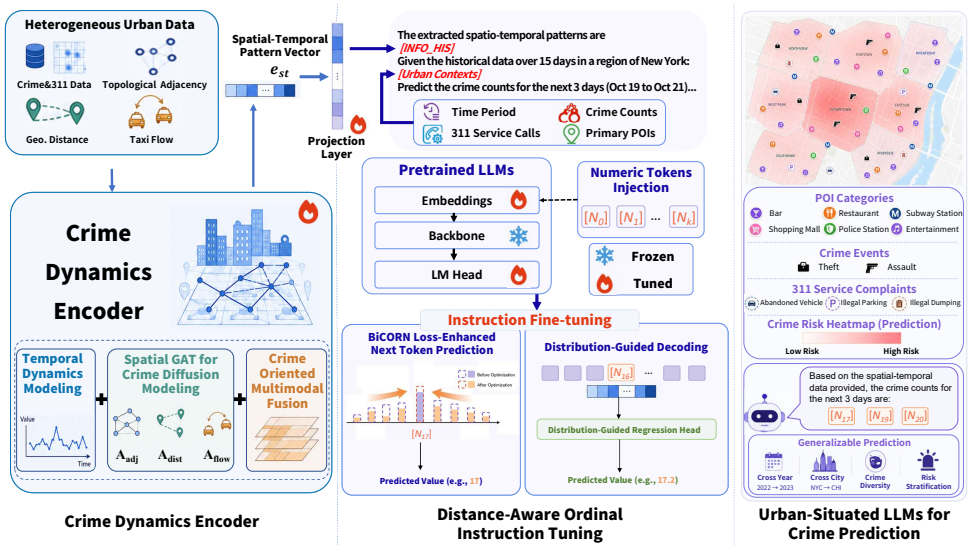


Figure 2: We propose **CrimeGPT**, which situates an LLM within urban dynamics via a Crime Dynamics Encoder that aligns heterogeneous multi-source signals with the LLM's semantic space. Numerical token injection and distance-aware ordinal supervision enforce numerical faithfulness, with final predictions obtained via token-level decoding or the Distribution-Guided Regression Head.

multi-scale temporal patterns, followed by a multi-view Graph Attention Network that aggregates topological adjacency ($\mathbf{A}_{adj}$), geographical distance ($\mathbf{A}_{dist}$), and taxi-flow transitions ($\mathbf{A}_{flow}$). Residual and skip connections preserve information across layers. The complete encoding process is summarized as

$$\mathbf{E}_{st} = \text{CDE}\left(\tilde{\mathbf{X}}, \mathbf{A}_{adj}, \mathbf{A}_{dist}, \mathbf{A}_{flow}\right) \tag{2}$$

where $\mathbf{E}_{st} \in \mathbb{R}^{N \times d_{st}}$ denotes the final spatio-temporal embedding aggregated across all layers. Detailed mathematical formulations of the CDE's internal architecture are provided in Appendix B.

The resulting embedding $\mathbf{E}_{st}$ is projected into the LLM semantic space and inserted into the [INFO_HIS] placeholders, grounding the language model in regional crime dynamics.

4.2 Task-Induced Predictive Diffidence

Standard LLM tokenizers decompose multi-digit counts into separate digits, obscuring count-level magnitude and preventing a direct distribution over candidate values. Dedicated numerical tokens instead provide a unified output space for probabilistic forecasting (Ansari et al., 2024). We therefore add 100 tokens [N_0], . . . , [N_99], each corresponding to an integer count. Each embedding is initialized as a normalized exponentially decaying weighted sum of its constituent digit embeddings:

$$\mathbf{e}_{[\text{N_x}]} = \sum_{j=1}^{d} \frac{2^{-j}}{\sum_{k=1}^{d} 2^{-k}} \cdot \mathbf{e}_{\text{digit}(x,j)} \tag{3}$$

Here, digit(x, j) denotes the pretrained embedding of the j-th digit of x. Higher-order digits receive greater weight, allowing the new tokens to inherit coarse magnitude semantics rather than starting from random embeddings. The resulting count-token space exposes the full probability distribution over candidate counts and makes model confidence directly measurable.

Using this distributional view, we compare confidence across three categories considered in LLM reasoning and regression studies (Lukasik et al., 2025; Zausinger et al., 2024; Li et al., 2024): **logical reasoning** (ARC (Clark et al., 2018), GSM8K (Cobbe et al., 2021)), **simple regression** (Amazon Reviews (Hou et al., 2024)), and **spatio-temporal prediction** (NYC Bike, Taxi, and Crime). These tasks range from deterministic single-answer problems to forecasting from heterogeneous signals.

As shown in Table 1, increasingly complex and heterogeneous forecasting settings produce broader, less confident output distributions. Logical reasoning tasks yield near-deterministic distributions

(top-1 prob > 0.89), whereas NYC Crime has a mean top-1 probability of 0.23 and entropy of 2.57. The pronounced uncertainty in crime forecasting reflects the challenge of reconciling heterogeneous urban signals within a unified numerical prediction. We use **predictive diffidence** to denote this diffuse-distribution pattern, for which standard token decoding is poorly suited. A more detailed analysis is provided in Appendix D.2.

Predictive diffidence exposes two limitations of the standard combination of CE training and argmax decoding. During training, CE treats count tokens as independent categories and explicitly supervises neither the ordinal relations among neighboring counts nor the numerical distance between a prediction and its target. During inference, argmax further collapses the diffuse distribution to a single token and discards the remaining probability mass. These limitations motivate BiCORN to impose ordinal structure, and a distribution-guided regression head to exploit the latent numerical information while providing explicit distance supervision.

4.3 Ordinally Structured Numerical Regression

Crime counts are ordered outcomes rather than unrelated categories. Criminological theories further suggest that historical incidents, human activity, and regional function combine to produce graded levels of risk (Cohen & Felson, 1979; Brantingham & Brantingham, 1984). These signals may not determine an exact count, but they can still support stable relations between neighboring risk levels. Numerical-token fine-tuning should therefore preserve ordinal structure that standard CE does not explicitly supervise.

We introduce this structure by adapting CORN (Shi et al., 2023) to LLM fine-tuning. Conventional CORN reformulates a K-class ordinal problem as $K-1$ conditional binary tasks, each estimating whether the target exceeds a successive boundary. It therefore requires $K-1$ conditional outputs, whereas an LLM provides K numerical-token logits without a separate ordinal head. Given logits $s_0, \ldots, s_{K-1}$, we construct each conditional boundary from adjacent logit odds:

$$q_k^+ = \sigma(s_{k+1} - s_k) = \frac{e^{s_{k+1}}}{e^{s_k} + e^{s_{k+1}}}, \qquad q_k^- = \sigma(s_k - s_{k+1}). \tag{4}$$

The forward objective supervises q_k^+ below the target. To constrain the otherwise weakly supervised upper side, we apply a reverse objective to q_k^- at and above the target:

$$\mathcal{L}_{\mathrm{BiCORN}} = \sum_{k<y^*} \ell_{\mathrm{BCE}}\big(q_k^+, \mathbf{1}[y^* > k]\big) + \sum_{k \geq y^*} \ell_{\mathrm{BCE}}\big(q_k^-, \mathbf{1}[y^* \leq k]\big). \tag{5}$$

After normalization over active boundaries, minimizing the two terms encourages $s_{k+1} > s_k$ below y^* and $s_{k+1} < s_k$ at and above y^*. The adjacent logit differences thus induce $s_0 \lesssim \cdots \lesssim s_{y^*} \gtrsim \cdots \gtrsim s_{K-1}$ without replacing the pretrained language-model head. We call this bidirectional objective **BiCORN**. Appendix E.2 provides the extended derivation.

Although BiCORN organizes probability mass into a clearer ordinal profile, it does not explicitly penalize the magnitude of count error. The resulting distribution can therefore remain broad, as shown in Figure 3. This motivates an explicit distance objective that also avoids reducing the diffuse distribution to one argmax token.

We therefore introduce the **Distribution-Guided Regression Head (DGRH)**, a two-layer MLP that maps the final-layer hidden state $\boldsymbol{\gamma}_t \in \mathbb{R}^{d_{\mathrm{LLM}}}$ at each numerical prediction position to a continuous count. The complete objective is

$$\hat{y}_t = g_\phi(\boldsymbol{\gamma}_t), \qquad \mathcal{L}_{\mathrm{DGRH}} = \frac{1}{T}\sum_{t=1}^{T} |\hat{y}_t - y_t|, \tag{6}$$

$$\mathcal{L} = \mathcal{L}_{\mathrm{CE}} + \lambda \mathcal{L}_{\mathrm{BiCORN}} + \alpha \mathcal{L}_{\mathrm{DGRH}}.$$

The L1 term supplies explicit distance supervision, while continuous decoding uses the latent numerical representation without the lossy argmax bottleneck. BiCORN and DGRH therefore provide complementary ordinal and metric constraints.

Overall, CrimeGPT combines numerical-token injection, BiCORN-enhanced model fine-tuning, and DGRH-based continuous decoding. Together, these components form an ordinally supervised regression model that preserves latent numerical evidence and produces crime-count predictions under explicit distance supervision.

Table 1: Output-distribution confidence across tasks. Normalized entropy is $H_{\text{model}}/H_{\text{label}}$, with H_{label} computed from test labels.

Dataset	Entropy	Norm. Ent.	Peak Prob.
Logical Reasoning			
ARC	0.0237	0.0171	0.9904
GSM8K	0.2576	0.0634	0.8989
Simple Regression			
Amazon Reviews (Health)	0.5745	0.3570	0.7622
Amazon Reviews (Games)	0.5839	0.3628	0.7599
Spatiotemporal Prediction			
NYC Bike	1.1042	0.5292	0.6510
NYC Taxi	1.6593	0.6021	0.4539
NYC Crime	2.5679	0.7429	0.2256

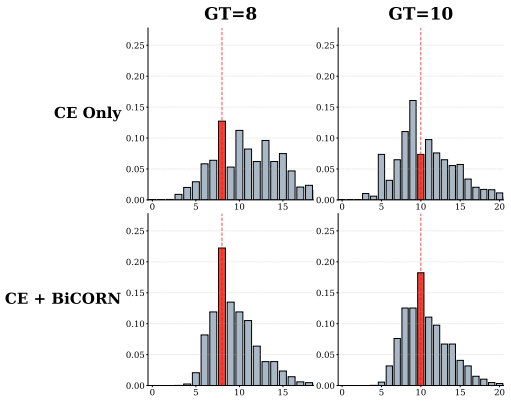


Figure 3: Comparison of probability distributions.

5 Experiments

5.1 Datasets and Baselines

Datasets. We conduct extensive experiments on crime datasets from four cities: New York City (NYC), Chicago (CHI), Philadelphia (PHI), and Shanghai (SHA). For NYC, CHI, and PHI, we evaluate *Theft*, *Assault*, and *All*, where *All* aggregates major crime types. SHA is evaluated on *All* only. Full dataset statistics are provided in Appendix A.

Baselines. We compare CrimeGPT against three categories of baselines, including general spatio-temporal prediction models, crime-specific forecasting methods, and recent LLM-based prediction frameworks. Detailed descriptions of all baselines are provided in Appendix G.3.

5.2 Tasks and Evaluation Metrics

Prediction Task. We set the historical observation window to $H = 15$ days and the prediction horizon to $P = 3$ days. A multi-step horizon is adopted to move beyond single-point estimation, enabling proactive rather than purely reactive crime prevention. We conduct two types of experiments: *cross-time evaluation*, where models are tested on unseen temporal intervals (including both in-year held-out splits and cross-year transfer from 2022 to 2023), and *zero-shot cross-city transfer* (trained on NYC, tested on CHI and PHI), assessing temporal robustness and spatial generalization respectively.

Evaluation Metrics. We adopt Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) as evaluation metrics, with all results averaged over all prediction horizons on the test set.

5.3 Implementation Details

We adopt Qwen2.5-7B (Qwen Team, 2024) as the backbone LLM, with 100 numerical tokens [N_0]–[N_99] injected into the vocabulary. The LLM backbone is fully frozen during fine-tuning; CDE, the projector, numerical embeddings, `lm_head`, and DGRH are jointly updated. Models are trained for 3 epochs on 4 NVIDIA RTX 5880 Ada Generation GPUs with batch size 4. Full implementation details are provided in Appendix G.4. All training-based results are averaged over three independent runs. The default numerical vocabulary $[0, 99]$ covers all observed counts; Appendix C evaluates a deliberately truncated $[0, 50]$ vocabulary on out-of-bound targets.

5.4 Main Results

Table 2 reports same-year cross-time results across NYC, CHI, PHI, and SHA. Cross-year transfer results are provided in Table 9 in the appendix.

LLM-Based vs. Traditional Models. Across most datasets and crime types, LLM-based methods consistently outperform or remain competitive with purely statistical and graph-based approaches, particularly on NYC where urban contextual information is richest. This suggests that semantic

Table 2: Same-year results (MAE/RMSE). Best and second-best results are bold and underlined.

Method	NYC			CHI			PHI			SHA
	All	Theft	Assault	All	Theft	Assault	All	Theft	Assault	All
Spatio-Temporal Forecasting Models										
ARIMA (Box et al., 2015)	3.924/5.460	2.591/3.685	2.231/3.121	2.746/3.854	1.897/2.690	1.332/1.861	3.466/4.790	2.299/3.164	1.670/2.271	0.993/1.744
DCRNN (Li et al., 2017)	4.447/5.777	2.845/3.802	2.387/3.066	2.666/3.744	1.822/2.610	1.298/1.840	3.141/4.276	2.027/2.767	1.718/2.264	1.059/1.426
GMAN (Zheng et al., 2020)	3.873/5.199	2.577/3.531	2.457/3.184	2.801/3.850	1.796/2.528	1.532/2.149	3.205/4.317	1.979/2.771	1.661/2.171	0.809/1.400
AGCRN (Bai et al., 2020)	3.808/5.214	2.497/3.487	2.160/2.917	2.609/3.619	1.762/2.515	1.320/1.857	3.113/4.190	1.980/2.723	1.665/2.181	0.762/1.402
xVal (Golkar et al., 2023)	4.223/5.691	3.367/4.229	2.613/3.830	2.929/4.143	1.982/2.798	1.623/2.456	3.676/5.222	2.094/2.921	2.115/3.191	0.935/1.390
MoSSL (Deng et al., 2024)	5.337/7.565	3.273/5.039	2.978/3.762	3.999/5.617	2.451/3.296	1.602/2.337	3.719/5.222	2.878/3.873	2.334/3.019	0.856/1.355
STBP (Liu & Zhang, 2026)	5.190/6.707	3.139/4.056	2.853/3.514	2.575/**3.595**	1.786/2.590	1.223/1.783	3.279/4.406	2.006/2.783	1.650/2.154	0.822/1.598
Crime Prediction Models										
DeepCrime (Huang et al., 2018)	4.189/5.698	2.607/3.631	2.135/2.974	2.729/3.802	1.939/2.896	1.285/1.815	3.180/4.300	2.018/2.714	1.643/2.255	1.275/2.065
ST-SHN (Xia et al., 2022)	4.405/5.685	2.922/3.840	2.449/3.108	2.633/3.680	1.788/2.540	1.264/1.763	3.165/4.301	2.030/2.780	1.595/2.158	0.953/1.494
STHSL (Li et al., 2022)	4.185/5.389	2.943/3.736	2.412/3.023	2.758/3.637	1.807/2.480	1.420/1.795	3.131/4.202	2.022/2.781	1.898/2.437	1.295/1.583
ST-HHOL (Du et al., 2026)	4.438/5.799	2.713/3.618	2.593/3.360	2.640/3.716	**1.704/2.448**	1.248/1.790	3.095/4.275	1.927/2.671	1.647/2.217	**0.674**/1.526
LLM-Based Prediction Models										
GPT4TS (Zhou et al., 2023)	3.794/5.236	2.496/3.527	2.164/3.002	2.661/3.720	1.831/2.570	1.286/1.793	3.228/4.368	2.060/2.790	1.591/2.134	0.874/1.420
Time-LLM (Jin et al., 2024)	3.782/5.212	2.504/3.518	2.270/3.139	2.587/3.629	1.798/2.524	1.295/1.825	3.458/4.681	2.203/3.010	1.607/2.159	0.887/1.458
UrbanGPT (Li et al., 2024)	3.833/5.396	2.891/4.475	2.875/3.848	2.738/3.869	1.996/2.797	1.324/1.884	3.893/5.538	2.674/3.913	1.996/2.844	0.837/1.626
CrimeGPT	**3.537/4.921** ±0.011/±0.007	**2.286/3.285** ±0.010/±0.020	**2.036/2.880** ±0.006/±0.006	**2.530**/3.598 ±0.014/±0.019	1.716/2.466 ±0.003/±0.001	**1.193/1.742** ±0.007/±0.007	**3.017/4.139** ±0.038/±0.048	**1.908/2.666** ±0.015/±0.012	**1.511/2.119** ±0.005/±0.009	0.735/**1.345** ±0.005/±0.015

reasoning over regional characteristics provides a meaningful complementary signal to numerical spatio-temporal crime patterns.

Advantages of CrimeGPT.

CrimeGPT achieves consistently strong performance across cities and crime types, with particularly clear advantages on NYC and PHI. We attribute this to three complementary mechanisms: the CDE grounds predictions in spatio-temporal crime dynamics, BiCORN-based instruction tuning organizes numerical-token representations through ordinal supervision, and DGRH adds explicit count-distance supervision while decoding them into continuous estimates. The advantage is most pronounced in high-magnitude settings (e.g., NYC *All*), where richer multi-source context and a wider prediction range amplify the benefits of improved numerical awareness and stronger alignment with crime dynamics. Performance is less pronounced on SHA, whose smaller numerical scale and sparser urban signals offer less scope for semantic grounding and ordinal constraints.

Zero-Shot Cross-City Transfer. Table 3a reports zero-shot cross-city results, where all models are trained exclusively on NYC and evaluated zero-shot on CHI and PHI, without target-city fine-tuning or parameter updates. Many baselines depend on region-specific graph structures and therefore do not transfer directly without architectural or parameter adaptation. STBP is designed for transfer, yet remains less effective under zero-shot evaluation. CrimeGPT achieves the best performance across all tasks and both cities by a substantial margin. The results indicate that CDE-grounded representations, BiCORN ordinal supervision, and DGRH-based explicit distance regression transfer more effectively to the two unseen cities than the evaluated baselines.

Zero-shot LLMs. Table 3b shows that off-the-shelf LLMs perform similarly despite substantial differences in scale and general capability. Generic next-token priors do not directly capture ordered count relationships or heterogeneous urban dynamics, so increasing model capacity alone provides little benefit for this task.

We further provide additional experiments under the commonly adopted crime prediction setting with $H = 10$ days and prediction horizon $P = 1$ day (Huang et al., 2018), together with a cross-year evaluation from 2022 to 2023. As reported in Appendix G.1, CrimeGPT achieves the best MAE and RMSE across all three cities and crime types, showing strong cross-year generalization.

5.5 Ablation Study

Loss functions ablation. Table 3c compares alternative objectives without and with DGRH. Without DGRH, BiCORN improves over CE only but remains weaker than MSE and W_1, because ordinal boundary supervision organizes adjacent logits without directly penalizing the magnitude of count errors. Once DGRH supplies explicit L1 distance supervision and continuous decoding, BiCORN achieves the best MAE and RMSE, surpassing both distance-aware alternatives. This reversal supports the complementary roles of BiCORN for ordinal structure and DGRH for explicit distance constraints.

Table 3: Zero-shot transfer and ablation analyses. Ablations are conducted on NYC All Crime 2022. MAE and RMSE are lower-is-better.

(a) Zero-shot cross-city transfer.

Method	CHI			PHI		
	All	Theft	Assault	All	Theft	Assault
UrbanGPT	3.528/4.893	2.525/3.413	2.111/3.756	3.771/4.932	2.569/3.444	1.985/2.608
Time-LLM	2.981/4.139	2.168/3.003	1.553/2.081	3.437/4.590	2.483/3.268	1.970/2.557
STBP	5.677/8.063	3.764/5.733	2.577/3.472	9.426/11.171	3.485/4.702	4.020/4.768
CrimeGPT	**2.675/3.714**	**1.785/2.553**	**1.269/1.789**	**3.113/4.275**	**1.961/2.749**	**1.545/2.150**

(b) Zero-shot LLMs and CrimeGPT.

LLM	MAE	RMSE
Qwen-2.5-7B (Qwen Team, 2024)	3.919	5.477
Qwen3.6-Plus (Qwen Team, 2026)	3.953	5.468
GPT-5.1 (OpenAI, 2025)	3.900	5.389
Gemini-3-Flash (Anil et al., 2023)	3.873	5.403
CrimeGPT	**3.537**	**4.921**

(c) Loss-function ablation.

Loss Configuration	MAE	RMSE
w/o DGRH		
CE only	3.815	5.328
CE + BiCORN	3.753	5.272
CE + MSE	3.688	5.135
CE + W_1	**3.614**	**5.025**
w/ DGRH		
CE only	3.571	4.963
CE + MSE	3.733	5.581
CE + W_1	3.569	4.971
CE + BiCORN	**3.537**	**4.921**

(d) Paradigm/component ablation.

Variant	MAE	RMSE
Paradigm Ablation		
Zero-shot LLM	3.919	5.477
SFT	3.864	5.407
SFT + RL	3.762	5.214
Component Ablation		
+ LoRA	3.649	5.041
w/o CDE & DGRH	3.766	5.288
w/o DGRH	3.767	5.289
w/o CDE	3.628	5.036
CrimeGPT	**3.537**	**4.921**

(e) Modality ablation.

Variant	MAE	RMSE
Encoding-Signal Ablation		
w/o Crime History	4.696	6.713
w/o 311 Requests	3.632	5.066
Auxiliary-Information Ablation		
w/o Geo. Distance	3.569	4.979
w/o Taxi Flow	3.553	4.966
w/o Spatial Modeling	3.587	4.985
w/o Context Prompts	3.582	4.989
w/o Spatial & Context	3.593	5.010
CrimeGPT	**3.537**	**4.921**

Note. In panel (d), SFT and SFT + RL are applied directly to the backbone LLM without numerical token injection, CDE, or DGRH.

Paradigm/component ablation. Table 3d presents a paradigm/component ablation study on NYC Crime. For paradigm comparison, *Zero-shot LLM* performs greedy decoding using the default Qwen2.5-7B configuration. *SFT* denotes supervised fine-tuning using LoRA (Hu et al., 2022), and *RL* denotes subsequent reinforcement learning using GRPO (Shao et al., 2024). Detailed configurations for SFT and RL are provided in Appendix G.5.

The results show that conventional instruction tuning offers only limited gains for crime prediction. Vanilla SFT slightly improves over zero-shot inference, and subsequent GRPO provides a further gain, but both remain behind our full framework. Numerical-token injection with ordinal-aware tuning alone (w/o CDE & DGRH) performs comparably to SFT + RL. This suggests that generic language adaptation and reinforcement learning do not fully capture the urban grounding and continuous count structure jointly modeled by CDE, BiCORN, and DGRH.

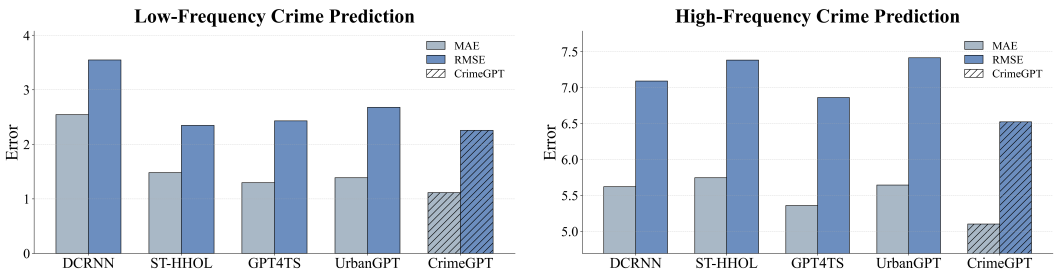
For component ablation, removing either the CDE or DGRH consistently degrades performance, while removing both causes the largest deterioration. This validates the complementary roles of urban spatio-temporal grounding and DGRH-based explicit distance regression within the BiCORN-supervised framework. Applying LoRA to the LLM backbone also degrades MAE/RMSE from 3.537/4.921 to 3.649/5.041, suggesting that broader low-rank adaptation introduces unnecessary degrees of freedom and can disturb useful pre-trained representations under the task-specific data.

Urban information ablation. Table 3e shows that crime history provides the primary forecasting signal: removing it causes a substantial performance collapse. Among the auxiliary sources, 311 requests contribute the largest individual improvement, while spatial relations and contextual prompts yield smaller but consistent gains. Removing both spatial and contextual information further degrades performance, indicating that these signals provide complementary refinements to the crime dynamics.

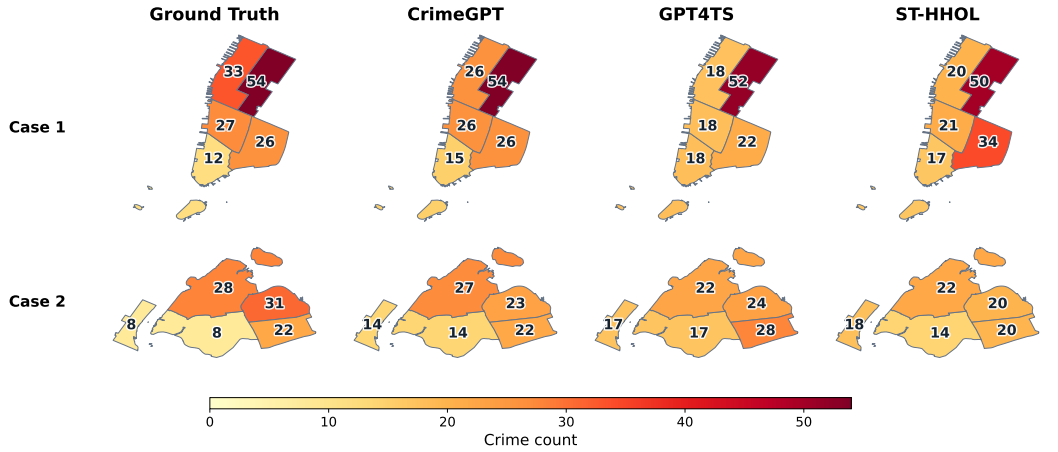
Appendix G.2 compares alternative spatial graph constructions; all improve over removing spatial modeling, while our multi-view design performs best.

5.6 Further Analysis

Figure 4(a) reports performance on the top and bottom 25% of NYC regions by crime frequency. CrimeGPT performs strongly in both groups, with clearer gains in high-frequency regions where larger counts and more complex dynamics make numerical ordering and count differences more



(a) Frequency-stratified errors.



(b) Local crime-count maps for two aligned cases.

Figure 4: Further analysis of CrimeGPT. Panel (a) compares forecasting errors in low- and high-frequency NYC regions. Panel (b) shows two local NYC cases. Numbers denote crime counts.

consequential. Figure 4(b) shows two local cases in which CrimeGPT follows the ground-truth hotspot pattern and count magnitudes more closely than GPT4TS and ST-HHOL. Table 11 further reports count errors on zero and non-zero targets and crime-occurrence detection results.

6 Conclusion and Limitations

In this paper, we present **CrimeGPT**, an end-to-end framework for numerically faithful urban crime forecasting. CrimeGPT situates a frozen LLM within urban crime dynamics through a Crime Dynamics Encoder that aligns heterogeneous multi-source spatio-temporal signals with the LLM's semantic space, enabling context-aware reasoning beyond purely statistical pattern mining. To address predictive diffidence, a phenomenon of diffuse output distributions induced by the heterogeneous nature of crime dynamics, CrimeGPT introduces numerical token injection, BiCORN-based ordinal supervision, and a Distribution-Guided Regression Head that adds explicit count-distance supervision and maps ordinally informed hidden representations directly to continuous count estimates, bypassing the information loss of token-level decoding. Extensive experiments across four cities demonstrate consistent improvements over statistical, spatio-temporal, crime-specific, and LLM-based baselines, together with strong generalization under fixed cross-year and zero-shot cross-city protocols. We hope this work contributes to more accurate crime forecasting and ultimately supports effective public safety management and urban resource allocation.

Despite these encouraging results, several limitations remain. CrimeGPT partly depends on auxiliary urban data whose availability, quality, and semantics vary across cities. Although the LLM backbone is frozen, a 7B model remains more computationally expensive than conventional forecasting architectures. Future work should investigate missing-modality adaptation and lighter backbones to improve transferability and deployment efficiency.

ETHICS STATEMENT

The authors have read and adhere to the ICLR Code of Ethics. This study uses only publicly available, aggregate or event-level urban records under their original licenses and terms; it recruits no human participants and uses no personally identifiable information.

Crime forecasts can nevertheless reproduce inequalities in reporting, service access, and historical policing, and deployment may create feedback loops that concentrate attention on already over-observed communities. CrimeGPT is intended for research and aggregate decision support, not individual risk assessment or automated enforcement. Any operational use would require local validation, uncertainty communication, human oversight, community and domain-expert consultation, and regular auditing for geographic disparities. Predictions should never be the sole basis for enforcement or resource-allocation decisions. We report the model's data, transfer, numerical-range, and computational limitations to support transparent and responsible assessment.

AI USE STATEMENT

In this work, we used generative AI tools to assist with dataset analysis and cleaning, the development of theoretical models, the formulation and writing of proof steps, and language editing for clarity and readability. All AI-assisted outputs were reviewed and verified by the authors, who take full responsibility for the final content, claims, and artifacts presented in this work.

REPRODUCIBILITY STATEMENT

Data sources, crime-category definitions, regional aggregation, and auxiliary urban signals are described in Section 5.1 and Appendix A. The CrimeGPT architecture, numerical-token construction, and training objectives are specified in Section 4.1–4.3, with further formulations in Appendices B and E. Training hyperparameters, baseline configurations, and evaluation protocols are documented in Section 5.3 and Appendix G.

REFERENCES

Rohan Anil et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

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. *arXiv preprint arXiv:2403.07815*, 2024.

Lei Bai, Lina Yao, Can Li, Xianzhi Wang, and Can Wang. Adaptive graph convolutional recurrent network for traffic forecasting. *Advances in neural information processing systems*, 33:17804–17815, 2020.

George EP Box, Gwilym M Jenkins, Gregory C Reinsel, and Greta M Ljung. *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

Paul J Brantingham and Patricia L Brantingham. *Patterns in crime*, volume 113. Macmillan New York, 1984.

Mayank Bumb, Anshul Vemulapalli, Sri Harsha Vardhan Prasad Jella, Anish Gupta, An La, Ryan Rossi, Hongjie Chen, Franck Dernoncourt, Nesreen K Ahmed, and Yu Wang. Forecasting time series with llms via patch-based prompting and decomposition. In *Proceedings of the 39th Pacific Asia Conference on Language, Information and Computation*, pp. 467–479, 2025.

Hsinchun Chen, Wingyan Chung, Jennifer Jie Xu, Gang Wang, Yi Qin, and Michael Chau. Crime data mining: a general framework and some examples. *computer*, 37(4):50–56, 2004.

Runfei Chen, Shuyang Jiang, and Wei Huang. Semob: Semantic synthesis for dynamic urban mobility prediction. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 15346–15366, 2025.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv:1803.05457v1*, 2018.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Lawrence E Cohen and Marcus Felson. Social change and crime rate trends: A routine activity approach. *American sociological review*, pp. 588–608, 1979.

Jiewen Deng, Renhe Jiang, Jiaqi Zhang, and Xuan Song. Multi-modality spatio-temporal forecasting via self-supervised learning. *arXiv preprint arXiv:2405.03255*, 2024.

Keqing Du, Yufan Kang, Xinyu Yang, and Wei Shao. ST-HHOL: Spatio-temporal hierarchical hypergraph online learning for crime prediction. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=Nc3dl43s5Z.

David Finkelhor. *Childhood victimization: Violence, crime, and abuse in the lives of young people*. oxford university Press, 2008.

Siavash Golkar, Mariel Pettee, Michael Eickenberg, Alberto Bietti, Miles Cranmer, Geraud Krawezik, Francois Lanusse, Michael McCabe, Ruben Ohana, Liam Parker, et al. xval: a continuous numerical tokenization for scientific language models. *arXiv preprint arXiv:2310.02989*, 2023.

Yupeng Hou, Jiacheng Li, Zhankui He, An Yan, Xiusi Chen, and Julian McAuley. Bridging language and items for retrieval and recommendation. *arXiv preprint arXiv:2403.03952*, 2024.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Chao Huang, Junbo Zhang, Yu Zheng, and Nitesh V Chawla. Deepcrime: Attentive hierarchical recurrent networks for crime prediction. In *Proceedings of the 27th ACM international conference on information and knowledge management*, pp. 1423–1432, 2018.

Yue Jiang, Qin Chao, Yile Chen, Xiucheng Li, Shuai Liu, and Gao Cong. Urbanllm: Autonomous urban activity planning and management with large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 1810–1825, 2024.

Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y. Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, and Qingsong Wen. Time-llm: Time series forecasting by reprogramming large language models. In *International conference on learning representations*, volume 2024, pp. 23857–23880, 2024.

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

Zhonghang Li, Chao Huang, Lianghao Xia, Yong Xu, and Jian Pei. Spatial-temporal hypergraph self-supervised learning for crime prediction. In *2022 IEEE 38th international conference on data engineering (ICDE)*, pp. 2984–2996. IEEE, 2022.

Zhonghang Li, Lianghao Xia, Jiabin Tang, Yong Xu, Lei Shi, Long Xia, Dawei Yin, and Chao Huang. Urbangpt: Spatio-temporal large language models. In *Proceedings of the 30th ACM SIGKDD conference on knowledge discovery and data mining*, pp. 5351–5362, 2024.

Aoyu Liu and Yaying Zhang. A general spatio-temporal backbone with scalable contextual pattern bank for urban continual forecasting. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=LHSea6DI8U.

Chenxi Liu, Sun Yang, Qianxiong Xu, Zhishuai Li, Cheng Long, Ziyue Li, and Rui Zhao. Spatial-temporal large language model for traffic prediction. In *2024 25th IEEE international conference on mobile data management (MDM)*, pp. 31–40. IEEE, 2024.

Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. Predicting the next location: A recurrent model with spatial and temporal contexts. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30, 2016.

Michal Lukasik, Zhao Meng, Harikrishna Narasimhan, Yin-Wen Chang, Aditya Krishna Menon, Felix Yu, and Sanjiv Kumar. Better autoregressive regression with llms via regression-aware fine-tuning. In *The Thirteenth International Conference on Learning Representations*, 2025.

Mike A Merrill, Mingtian Tan, Vinayak Gupta, Thomas Hartvigsen, and Tim Althoff. Language models still struggle to zero-shot reason about time series. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 3512–3533, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.201. URL https://aclanthology.org/2024.findings-emnlp.201/.

OpenAI. Gpt-5.1. https://platform.openai.com/docs/models/gpt-5.1, 2025. Accessed: 2026-05-25.

OpenStreetMap contributors. Planet dump retrieved from https://planet.osm.org. https://www.openstreetmap.org, 2022. Accessed: 2026-05-02.

Zijie Pan, Yushan Jiang, Sahil Garg, Anderson Schneider, Yuriy Nevmyvaka, and Dongjin Song. s^2IP-LLM: Semantic space informed prompt learning with LLM for time series forecasting. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 39135–39153, 2024.

Junwoo Park, Hyuck Lee, Dohyun Lee, Daehoon Gwak, and Jaegul Choo. Revisiting llms as zero-shot time series forecasters: Small noise can break large models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 906–922, 2025.

Yuhang Pei, Tao Ren, Yifan Wang, Zhipeng Sun, Wei Ju, Chong Chen, Xian-Sheng Hua, and Xiao Luo. Leaf: Large language diffusion model for time series forecasting. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 1076–1091, 2025.

Qwen Team. Qwen2.5: A party of foundation models, September 2024. URL https://qwenlm.github.io/blog/qwen2.5/.

Qwen Team. Qwen3.6-Plus: Towards real world agents, April 2026. URL https://qwen.ai/blog?id=qwen3.6.

Victor Rotaru, Yi Huang, Timmy Li, James Evans, and Ishanu Chattopadhyay. Event-level prediction of urban crime reveals a signature of enforcement bias in us cities. *Nature human behaviour*, 6(8): 1056–1068, 2022.

Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523, 1988.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y.K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Xintong Shi, Wenzhi Cao, and Sebastian Raschka. Deep neural networks for rank-consistent ordinal regression based on conditional probabilities. *Pattern Analysis and Applications*, 26(3):941–955, 2023.

Mingtian Tan, Mike A Merrill, Vinayak Gupta, Tim Althoff, and Thomas Hartvigsen. Are language models actually useful for time series forecasting? *Advances in Neural Information Processing Systems*, 37:60162–60191, 2024.

Hongjian Wang, Daniel Kifer, Corina Graif, and Zhenhui Li. Crime rate inference with big data. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 635–644, 2016.

James Q Wilson and George L Kelling. Broken windows. 1982.

Gerald Woo, Chenghao Liu, Akshat Kumar, Caiming Xiong, Silvio Savarese, and Doyen Sahoo. Unified training of universal time series forecasting transformers. *arXiv preprint arXiv:2402.02592*, 2024.

Xian Wu, Chao Huang, Chuxu Zhang, and Nitesh V Chawla. Hierarchically structured transformer networks for fine-grained spatial event forecasting. In *Proceedings of the web conference 2020*, pp. 2320–2330, 2020.

Ziyang Wu, Fan Liu, Jindong Han, Yuxuan Liang, and Hao Liu. Spatial-temporal mixture-of-graph-experts for multi-type crime prediction. *arXiv preprint arXiv:2409.15764*, 2024.

Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, and Cheng Zhang. Graph WaveNet for deep spatial-temporal graph modeling. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pp. 1907–1913, 2019.

Lianghao Xia, Chao Huang, Yong Xu, Peng Dai, Liefeng Bo, Xiyue Zhang, and Tianyi Chen. Spatial-temporal sequential hypergraph network for crime prediction with dynamic multiplex relation learning. *arXiv preprint arXiv:2201.02435*, 2022.

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

Rose Yu, Yaguang Li, Cyrus Shahabi, Ugur Demiryurek, and Yan Liu. Deep learning: A generic approach for extreme condition traffic forecasting. In *Proceedings of the 2017 SIAM international Conference on Data Mining*, pp. 777–785. SIAM, 2017b.

Jonas Zausinger, Lars Pennig, Anamarija Kozina, Sean Sdahl, Julian Sikora, Adrian Dendorfer, Timofey Kuznetsov, Mohamad Hagog, Nina Wiedemann, Kacper Chlodny, Vincent Limbach, Anna Ketteler, Thorben Prein, Vishwa Mohan Singh, Michael Morris Danziger, and Jannis Born. Regress, don't guess–a regression-like loss on number tokens for language models. *arXiv preprint arXiv:2411.02083*, 2024.

Yan Zhang, Mei-Po Kwan, and Libo Fang. An llm driven dataset on the spatiotemporal distributions of street and neighborhood crime in china. *Scientific Data*, 12(1):467, 2025.

Shuai Zhao, Ruiqiang Liu, Bo Cheng, and Daxing Zhao. Classification-labeled continuousization and multi-domain spatio-temporal fusion for fine-grained urban crime prediction. *IEEE Transactions on Knowledge and Data Engineering*, 35(7):6725–6738, 2022.

Xiangyu Zhao and Jiliang Tang. Modeling temporal-spatial correlations for crime prediction. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pp. 497–506, 2017.

Yusheng Zhao, Xiao Luo, Haomin Wen, Zhiping Xiao, Wei Ju, and Ming Zhang. Embracing large language models in traffic flow forecasting. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 8108–8123, 2025.

Chuanpan Zheng, Xiaoliang Fan, Cheng Wang, and Jianzhong Qi. Gman: A graph multi-attention network for traffic prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 1234–1241, 2020.

Tian Zhou, Peisong Niu, Xue Wang, Liang Sun, and Rong Jin. One fits all: Power general time series analysis by pretrained lm. *Advances in neural information processing systems*, 36:43322–43355, 2023.

APPENDIX OVERVIEW

This supplementary material provides extended technical details, implementation specifics, and additional experimental analyses to support the main manuscript. The contents are organized as follows:

- **Datasets (§A):** Dataset construction details, including data sources, crime category selection, statistical summaries of crime and auxiliary urban signals, and the administrative division structures of the four cities.
- **Crime Dynamics Encoder (§B):** Detailed formulation of the proposed Crime Dynamics Encoder.
- **Numerical Vocabulary Boundary Analysis (§C):** Stress tests of the bounded numerical vocabulary on in-bound and out-of-bound crime counts.
- **Predictive Diffidence Analysis (§D):** Analysis of limitations in token-level numerical modeling, comparisons of different loss functions, and visualization-based evaluation of predictive distribution behavior.
- **Auxiliary Losses and BiCORN Analysis (§E):** Additional formulations of the MSE and Wasserstein-1 alternatives, together with an extended analysis of the BiCORN objective introduced in Section 4.3.
- **Prompt Design (§F):** Prompt templates for crime prediction tasks and a TF-IDF-based strategy for selecting representative POI categories to encode urban functional context.
- **Experimental Setup (§G):** Detailed descriptions of baseline models, implementation configurations, training strategies (SFT and RL), and evaluation protocols.
- **Additional Results (§G.1):** Cross-year evaluation, different prediction horizon settings, occurrence detection, and spatial modeling analyses.

A DATASETS

A.1 DATA STATISTICS

Data for NYC[1], CHI[2], and PHI[3] are collected from official open-data portals of the corresponding cities. The SHA dataset is derived from the ChinaCrime dataset (Zhang et al., 2025).

For NYC, CHI, and PHI, we select *Theft* and *Assault* because they are prevalent and consistently defined across municipal recording systems. We additionally construct *All* by aggregating major crime types. For SHA, we report *All* only.

Table 4: Crime record statistics for NYC, CHI, PHI, and SHA.

City	Year	All	Theft	Assault
NYC	2022	449,130	200,040	181,642
	2023	452,074	193,296	186,894
CHI	2022	216,541	101,039	71,475
	2023	238,052	111,763	78,571
PHI	2022	76,128	33,116	25,360
	2023	81,340	37,054	25,986
SHA	2014	5,285	4,143	1,058

Table 5: Statistics of 311 requests, POI, and taxi flow across the four cities.

City	Year	311 Requests	POI	Taxi Flow
NYC	2022	3,169,960	34,891	39,602,680
	2023	3,224,721	42,114	38,336,504
CHI	2022	1,883,858	22,067	5,471,228
	2023	1,781,009	23,830	5,786,019
PHI	2022	560,647	9,262	–
	2023	528,596	9,509	–
SHA	2014	–	1,961	–

Table 4 summarizes the crime records by year, including *All*, *Theft*, and *Assault* for the three U.S. cities and *All* for SHA. Table 5 summarizes the 311 requests, POI and taxi flow statistics of our datasets. Missing values are due to the unavailability of corresponding data sources.

[1] https://opendata.cityofnewyork.us/

[2] https://data.cityofchicago.org/

[3] https://opendataphilly.org/



Figure 5: Administrative division structures of NYC, CHI, PHI, and SHA used for crime prediction task.

A.2 City Division Structure

To ensure spatial consistency across datasets, we aggregate all records according to the official administrative division systems of each city. As shown in Figure 5, NYC, CHI, PHI, and SHA are partitioned based on their respective district-level administrative boundaries, with 71, 77, 18, and 16 regions respectively.

A.3 Data Usage and Compliance

All datasets used in this work are collected from publicly available official open-data portals and are used strictly for academic research purposes. NYC, CHI, and PHI data are obtained from their respective municipal open-data platforms, while SHA is derived from the publicly released ChinaCrime dataset (Zhang et al., 2025). We adhere to the terms of use and licensing agreements provided by each data source, and do not redistribute any raw proprietary data in this paper.

B Detailed Formulation of the Crime Dynamics Encoder

This appendix provides the detailed mathematical formulations for the internal architecture of the Crime Dynamics Encoder (CDE) introduced in Section 4.1.

Heterogeneous Signal Normalization. Prior to deep representation learning, the raw heterogeneous urban signals (crime records and 311 requests) are standardized via feature-wise Z-score normalization:

$$\tilde{\mathbf{X}}^f = \frac{\mathbf{X}^f - \mu^f}{\sigma^f}, \quad \forall f \in \{c, 311\} \tag{7}$$

where μ^f and σ^f denote the empirical mean and standard deviation of feature f computed from the training set. The normalized features are stacked along the feature dimension to form the input tensor $\tilde{\mathbf{X}} = [\tilde{\mathbf{X}}^c \parallel \tilde{\mathbf{X}}^{311}] \in \mathbb{R}^{N\times H\times 2}$.

Layer-wise Computations. The CDE consists of $L = 3$ stacked Spatio-Temporal Layers with hidden dimension $d_h = 32$ and skip dimension $d_s = 64$. The initial hidden state $\mathbf{H}^{(0)}$ is obtained by projecting $\tilde{\mathbf{X}}$ into d_h via a 1×1 convolution.

Within each layer l, the Gate-Conv employs four parallel dilated 1D convolutions with kernel widths $\{2, 3, 6, 7\}$ and dilation factors increasing as $\{1, 2, 4\}$ across layers to capture multi-scale temporal dependencies:

$$\begin{aligned}\mathbf{T}^{(l)} = \tanh\Big(\Theta_1 *_{\text{dil}(l)} \mathbf{H}^{(l-1)}\Big) \\ \odot \sigma\Big(\Theta_2 *_{\text{dil}(l)} \mathbf{H}^{(l-1)}\Big)\end{aligned} \tag{8}$$

The SpatialGAT then applies multi-head graph attention ($h = 4$ heads, each of dimension $d_h/h = 8$) over the multi-view urban graphs, operating on the channel dimension to remain independent of node count N:

$$\tilde{\mathbf{H}}^{(l-1)} = \text{SpatialGAT}(\mathbf{T}^{(l)}, \mathbf{A}_{adj}, \mathbf{A}_{dist}, \mathbf{A}_{flow}) \tag{9}$$

Residual, Skip Connections, and Aggregation. A residual connection with a 1×1 convolution wraps the temporal-spatial transformation, while a separate projection with kernel size $1 \times T_l$ (where T_l depends on the layer's receptive field) maps the hidden states to the skip dimension $d_s = 64$:

$$\begin{aligned}\mathbf{H}^{(l)} &= \text{ResConv}^{1\times 1}(\tilde{\mathbf{H}}^{(l-1)}) + \mathbf{H}^{(l-1)}, \\ \mathbf{Z}^{(l)} &= \text{SkipConv}^{1\times T_l}(\tilde{\mathbf{H}}^{(l-1)}),\end{aligned} \tag{10}$$

An additional skip pathway $\mathbf{Z}^{(0)}$ is extracted directly from the normalized input via a 1×1 convolution. All skip representations are aggregated through summation and fed into a two-layer convolutional MLP with intermediate dimension $d_e = 128$ to produce the final spatio-temporal embedding $\mathbf{E}_{st} \in \mathbb{R}^{N\times d_s}$:

$$\mathbf{E}_{st} = \text{ReLU}(\mathbf{Z}_E^{(L)} + \sum_{l=0}^{L} \mathbf{Z}^{(l)}) \tag{11}$$

where $\mathbf{Z}_E^{(L)}$ is a final skip projection from the last layer's hidden state.

C Robustness to Numerical Vocabulary Boundaries

The injected vocabulary determines which integer values can be generated directly by the LLM, but it does not impose a hard bound on the continuous output of DGRH. The default range $[0, 99]$ is selected from the dataset statistics and covers every training, validation, and test sample. To test sensitivity to this choice, we deliberately truncate the vocabulary to $[0, 50]$ while keeping the architecture, objectives, and optimization settings unchanged. As shown in Table 6, 675 of 74,124 NYC samples (0.91%) then fall outside the token vocabulary.

Table 7 further shows that out-of-bound samples assign more probability to the boundary token and its immediate neighborhood than samples whose target is exactly 50. This concentration provides a latent indication that the target exceeds the representable token range. Because DGRH is jointly optimized with the LLM and is not clipped to the token vocabulary, it can exploit such boundary-exceeding information in the hidden state to regress values above 50. This experiment does not establish robustness to arbitrarily large values; hierarchical interval prediction or continuous residual regression remains an important direction for truly unbounded targets.

Table 6: Sample statistics with the truncated $[0, 50]$ vocabulary.

Split	Samples	Max.	> 50
Train	59,214	70	521
Validation	7,455	68	91
Test	7,455	75	63
Total	74,124	75	675 (0.91%)

Table 7: Boundary-token probabilities under the $[0, 50]$ vocabulary.

Target / Statistic	Probability
GT $= 50$: P([N_50])	0.4221
GT > 50: P([N_50])	**0.5718**
GT $= 50$: P([N_45]:[N_50])	0.6518
GT > 50: P([N_45]:[N_50])	**0.8879**

Table 8: MAE/RMSE on in-bound and out-of-bound NYC samples.

Method	**GT** ≤ 50	**GT** > 50
DCRNN	4.388 / 5.676	10.447 / 12.276
GMAN	3.832 / 5.117	8.737 / 11.257
DeepCrime	4.149 / 5.634	8.895 / 10.870
STHSL	4.114 / 5.255	12.288 / 13.915
Time-LLM	3.742 / 5.140	8.311 / 10.559
UrbanGPT	3.762 / 5.245	12.157 / 14.758
Ours (0–99)	**3.484 / 4.834**	**7.984 / 10.614**
Ours (0–50)	**3.539 / 4.901**	**8.079 / 10.548**

Table 8 compares performance on in-bound and out-of-bound test samples. All methods incur larger absolute errors for targets above 50. Nevertheless, truncating our vocabulary from $[0, 99]$ to $[0, 50]$ changes MAE on out-of-bound samples from 7.984 to 8.079, a relative degradation of approximately 1.2%. Thus, although the smaller vocabulary prevents direct generation of tokens above [N_50], it does not impose an equivalent upper bound on the final regression output.

D Further Analysis of Predictive Diffidence

D.1 Limitations of Token-level Losses in Numerical Prediction

Figure 6 illustrates the limitations of discrete digit-level tokenization and standard cross-entropy (CE) loss for numerical prediction tasks. Digit-level tokenization treats different digit positions independently, causing predictions with different numerical deviations from the ground truth to yield identical losses. For a target count of 55, predictions 47 and 36 have the same summed per-digit deviation of 3, although their count errors are 8 and 19. Similarly, CE loss only considers the probability assigned to the target token and ignores the actual numerical distance between predictions and targets. In Figure 6(b), both distributions assign probability 0.2 to the target and thus receive CE loss $-\log 0.2 = 1.61$, while their modes are one and three units away. As a result, numerically distant predictions may receive the same optimization signal.

In contrast, after introducing dedicated numerical tokens to avoid the limitations of digit-level tokenization, the distance-aware W_1 loss further preserves the ordinal relationships between numerical values, producing more faithful supervision for structured numerical prediction. It assigns losses of 0.80 and 2.20 to the nearby and distant distributions, respectively.

D.2 Probability Distribution Comparison

Figure 7 further compares the output probability distributions under different loss configurations. We observe that CE-only training exhibits severe predictive diffidence, where probability mass remains broadly dispersed across candidate values and meaningful ordinal structure is largely absent. In this setting, the final prediction heavily relies on an argmax operation, discarding substantial uncertainty and distributional information.

Adding MSE regularization significantly sharpens the probability distribution by imposing penalties proportional to squared numerical distance. However, this often leads to "confidently wrong" predictions, where the model assigns excessively concentrated probability mass to incorrect values far from the ground truth. Even when expectation-based decoding is adopted, noticeable prediction shifts still remain. In contrast, W1 loss achieves a better trade-off by compressing the distribution through distance-proportional penalties, producing more confident yet smoother predictions.

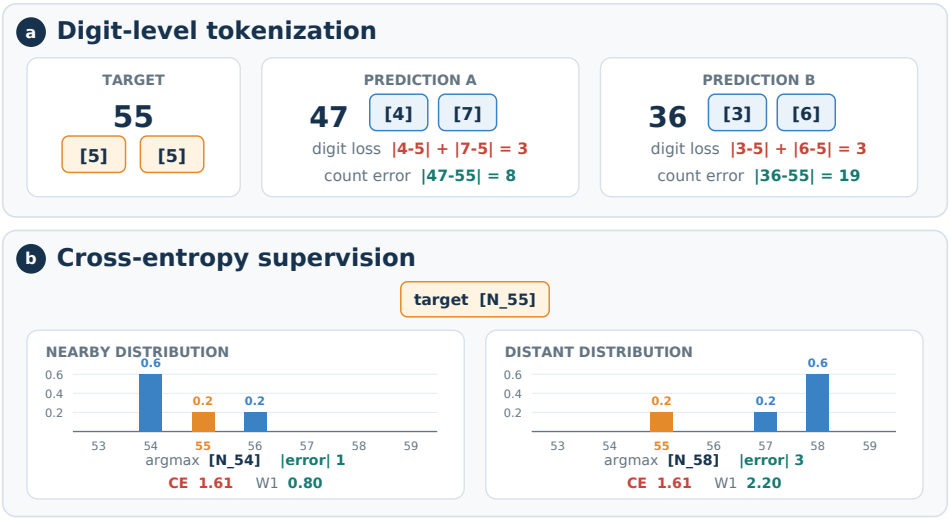


Figure 6: Failure modes of token-level numerical modeling: (a) digit-level tokenization and (b) cross-entropy supervision.

BiCORN produces the clearest ordinal structure but only weakly sharpens the token distribution. As discussed in Section 4.3, DGRH complements this ordering with explicit L1 distance supervision and continuous decoding.

The loss-function comparison is reported in Table 3c. BiCORN improves token-level prediction over CE only, while DGRH complements its ordinal structure with explicit L1 distance supervision and continuous decoding.

E Auxiliary Losses

E.1 Distance-Aware Comparison Objectives

Section 4.3 presents BiCORN as our ordinal objective. Here we detail the MSE and W_1 alternatives used to distinguish implicit ordinal structure from explicit distance supervision, followed by an extended analysis of BiCORN.

MSE Loss. The MSE loss penalizes the squared deviation of the predicted expected value from the true count:

$$\mathcal{L}_{\text{MSE}} = \left(\sum_{k=0}^{K-1} k \cdot p(k) - y^* \right)^2 \tag{12}$$

Since the optimization target is the expected value of the distribution, the corresponding decoding strategy applies expected-value decoding $\hat{y} = \sum_k k \cdot p(k)$.

Wasserstein-1 Loss. The W_1 loss minimizes the Earth Mover's Distance between the predicted cumulative distribution $F(k) = \sum_{j \le k} p(j)$ and the empirical CDF of the true label:

$$\mathcal{L}_{W_1} = \sum_{k=0}^{K-1} |F(k) - \mathbf{1}[y^* \le k]| \tag{13}$$

Unlike MSE, W_1 directly penalizes probability mass proportionally to its ordinal distance from y^*, providing explicit distance-aware supervision.

Bidirectional CORN (BiCORN) Loss. The original CORN framework (Shi et al., 2023) decomposes ordinal prediction into $K-1$ independent binary classifiers, each estimating the conditional exceedance probability $P(Y > k \mid Y \ge k)$ over the conditional training subset $\mathcal{S}_k = \{i : y_i \ge k\}$. We adapt this to the LLM setting by parameterizing each conditional boundary score via adjacent

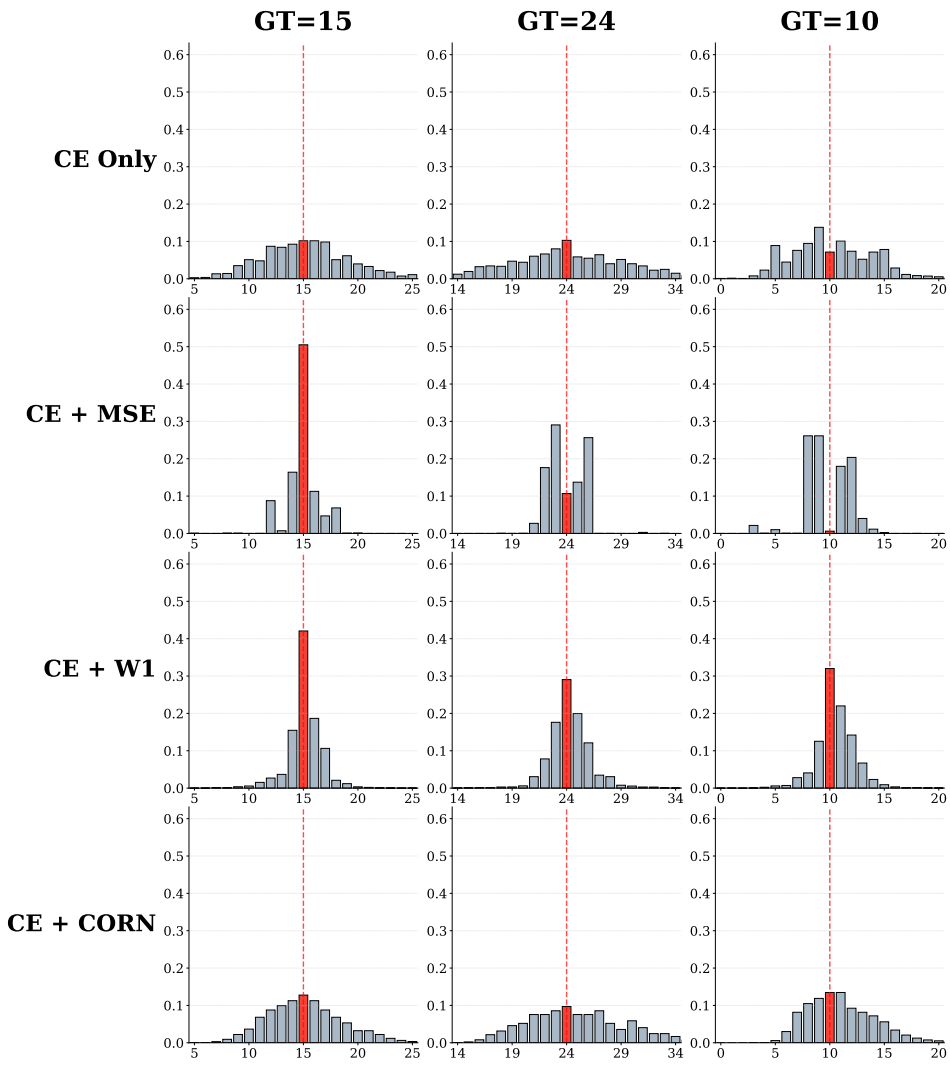


Figure 7: Comparison of probability distributions under different loss configurations. The ordinal configuration uses BiCORN.

logit differences:

$$P(Y > k \mid Y \geq k) \approx \sigma(r_k), \quad r_k = s_{k+1} - s_k \tag{14}$$

where s_k is the logit of token [N_k]. Restricting supervision to $\mathcal{S}_k$ via a binary mask yields the forward loss, using the positive-BCE convention $\ell_{\text{BCE}}(p, y) = -[y \log p + (1 - y) \log(1 - p)]$ defined above:

$$\mathcal{L}_{\text{fwd}} = \frac{1}{\sum_j |\mathcal{S}_j|} \sum_k \sum_{i \in \mathcal{S}_k} \ell_{\text{BCE}}(\sigma(r_k),\, \mathbf{1}[y_i > k]) \tag{15}$$

However, the original CORN mask leaves boundaries $k > y^*$ entirely unsupervised: overestimation by any degree is corrected only by the single term at $k = y^*$, while underestimation by a margin d is corrected by d active terms, informally introducing a systematic bias toward overestimation.

To restore symmetry, we introduce a backward pass that supervises exactly the boundaries left uncovered by $\mathcal{L}_{\text{fwd}}$, using the *same* adjacent-pair index as the forward term:

$$\sigma(-r_k) = \sigma(s_k - s_{k+1}) = 1 - \sigma(r_k), \tag{16}$$

with a strict conditional mask $\mathcal{S}_k^- = \{i : y_i < k\}$ that is disjoint from $\mathcal{S}_k$, and backward loss

$$\mathcal{L}_{\text{bwd}} = \frac{1}{\sum_j |\mathcal{S}_j^-|} \sum_k \sum_{i \in \mathcal{S}_k^-} \ell_{\text{BCE}}(\sigma(-r_k),\, \mathbf{1}[y_i \leq k])\,. \tag{17}$$

Because $\mathcal{S}_k \cup \mathcal{S}_k^-$ covers every boundary exactly once for each sample ($|\mathcal{S}_k| + |\mathcal{S}_k^-| = K-1$ per sample), $\mathcal{L}_{\text{fwd}}$ and $\mathcal{L}_{\text{bwd}}$ supervise disjoint sets of boundaries with no redundant double-counting. The full BiCORN loss is:

$$\mathcal{L}_{\text{BiCORN}} = \mathcal{L}_{\text{fwd}} + \mathcal{L}_{\text{bwd}} \tag{18}$$

E.2 Convergence Analysis of BiCORN

In this section, we provide an intuitive explanation of how BiCORN encourages an ordered logit structure. We emphasize that, due to finite model capacity and non-convex optimization, neither CE nor CORN-type objectives can be guaranteed to fully converge to an ideal discrete optimum in practice. Instead, we focus on the inductive bias induced by different loss designs.

Limitations of CE. Although cross-entropy loss theoretically optimizes a one-hot distribution, in practice LLMs trained with CE often produce *diffuse* and *unordered* logits under predictive uncertainty. This is because CE only supervises the correct class and does not explicitly impose any ordinal structure among neighboring logits.

Forward CORN behavior. CORN-style forward losses model cumulative probabilities via

$$\begin{aligned} P(Y > k \mid Y \geq k) &\approx \sigma(r_k), \\ r_k &= s_{k+1} - s_k. \end{aligned} \tag{19}$$

Minimizing BCE encourages $\sigma(r_k)$ to match the binary target $\mathbf{1}[y^* > k]$ for every $k \leq y^*$, i.e.,

$$\begin{aligned} \sigma(r_k) &\to 1 \quad \text{for } k < y^*, \\ \sigma(r_k) &\to 0 \quad \text{for } k = y^*. \end{aligned} \tag{20}$$

Intuitively, this pushes the logit sequence to satisfy a *partial ordering structure* up to one boundary past the target:

$$s_0 \lesssim s_1 \lesssim \cdots \lesssim s_{y^*} \gtrsim s_{y^*+1}. \tag{21}$$

However, the forward objective leaves every boundary from $k = y^* + 1$ onward entirely unsupervised, which may still lead to asymmetric supervision and overestimation bias.

Backward correction in BiCORN. To address this asymmetry, BiCORN introduces a backward objective using the *same* adjacent-pair index as the forward term:

$$\begin{aligned} P(Y \leq k \mid Y > k-1) &\approx \sigma(r_k^-), \\ r_k^- &= s_k - s_{k+1} = -r_k. \end{aligned} \tag{22}$$

This loss supplies exactly the supervision the forward objective leaves out, covering each remaining boundary once by encouraging:

$$\sigma(r_k^-) \to 1, \quad \text{for } k > y^*, \tag{23}$$

which induces a reverse ordering constraint starting one boundary past the target:

$$s_{y^*+1} \gtrsim s_{y^*+2} \gtrsim \cdots \gtrsim s_{K-1}. \tag{24}$$

Combined effect. By combining forward and backward objectives, BiCORN imposes a bidirectional ordering constraint on logits, with each boundary supervised by exactly one of the two objectives (forward covers $k = 0, \ldots, y^*$; backward covers $k = y^* + 1, \ldots, K-2$):

$$s_0 \lesssim \cdots \lesssim s_{y^*} \gtrsim \cdots \gtrsim s_{K-1}. \tag{25}$$

This does not guarantee strict convergence or perfect unimodality, but it *encourages a unimodal-like structure* centered at y^*.

Discussion. Compared to CE, which tends to produce diffuse and locally unstructured logits, CORN introduces partial ordinal consistency in one direction. BiCORN further improves this by enforcing symmetry, reducing the systematic bias toward overestimation and promoting more stable ordinal structure in practice, even under non-convergent training dynamics.

Beyond output logits, we further observe that BiCORN implicitly induces richer ordinal information in the hidden states of LLMs when generating numerical tokens. Although standard token-level decoding

struggles to fully exploit such structured representations, our proposed decoding strategy, *DGRH*, is specifically designed to leverage this hidden ordinality, leading to state-of-the-art performance.

Finally, the strong empirical results of our model, *CrimeGPT*, across multiple datasets further validate the effectiveness of the proposed BiCORN method and its associated decoding mechanism.

F Prompts

F.1 Prompt Template

Crime Prediction Prompt Example

Given the historical data over 15 days in a specific region of New York, the recorded number of all crimes is [27, 13, 13, 12, 15, 23, 11, 22, 19, 18, 16, 11, 13, 18, 19], and the recorded number of 311 service calls is [66, 80, 54, 42, 75, 72, 67, 75, 87, 65, 60, 91, 83, 77, 95].
The recording period is 'October 6, 2022, Saturday' to 'October 20, 2022, Saturday'. This region encompasses: Residential, Restaurant & Café, School & University.
Predict the values for the next 3 days during 'October 21, 2022, Sunday' to 'October 23, 2022, Tuesday'.
The spatio-temporal encoder uses [INFO_HIS] for these patterns.
Output Format Example: [N_1] [N_2] [N_3]

F.2 Representative POI Selection

To provide compact yet informative urban context in the prompts, we construct representative POI labels for each spatial region using a TF-IDF-based functional characterization strategy. Specifically, we first collect OpenStreetMap (OpenStreetMap contributors, 2022) (OSM) POIs from yearly city snapshots via the ohsome API and group them into nine semantic categories, including *Restaurant & Café*, *School & University*, *Park & Playground*, *Hospital*, *Railway Station*, *Shopping*, *Police*, *Entertainment*, and *Residential*. After spatially joining POIs to regions, we obtain a region-category count matrix.

For each region, we compute TF-IDF (Salton & Buckley, 1988) scores over POI categories, where the term frequency (TF) measures the relative proportion of a POI category within the region, and the inverse document frequency (IDF) measures the rarity of that category across the entire city. As a result, categories that are both locally dominant and globally distinctive receive higher TF-IDF scores, while ubiquitous categories are down-weighted.

Finally, for each region, we select the Top-3 POI categories with the highest TF-IDF values as representative functional labels and incorporate them into the prompts. This strategy enables the prompts to capture region-specific urban semantics while avoiding redundancy from overly common POI categories.

G Experiment Details

G.1 Additional Results

Cross-Year Evaluation.

We train on 2022 and evaluate the resulting fixed models directly on 2023, without updating their parameters on the target year. CrimeGPT maintains strong performance across all reported metrics under this temporal shift.

Additional Evaluation under $H = 10$, $P = 1$. Table 10 reports additional experiments under the commonly adopted crime prediction setting with a historical window size of 10 and prediction horizon of 1. CrimeGPT achieves the best MAE and RMSE across all three crime categories, consistently outperforming spatio-temporal forecasting models, specialized crime prediction methods, and recent LLM-based baselines.

Compared with the default $H = 15$, $P = 3$ setting, the performance gains become relatively smaller under the shorter prediction horizon. This is consistent with our earlier observation that combining

Table 9: Cross-year results (MAE/RMSE) on NYC, CHI, and PHI. Best and second-best results are bold and underlined.

Method	NYC			CHI			PHI		
	All	Theft	Assault	All	Theft	Assault	All	Theft	Assault
Spatio-Temporal Forecasting Models									
ARIMA (Box et al., 2015)	3.832/5.279	2.447/3.542	2.393/3.324	2.640/3.709	1.801/2.524	1.396/1.963	3.669/4.869	2.286/2.990	1.845/2.480
DCRNN (Li et al., 2017)	5.945/8.639	3.252/5.138	2.852/3.866	2.615/3.744	1.820/2.481	1.354/1.876	3.509/4.669	2.159/2.796	1.723/2.294
GMAN (Zheng et al., 2020)	3.846/5.142	2.654/3.655	2.552/3.318	2.774/3.908	1.794/2.625	1.651/2.301	3.337/4.461	2.150/2.830	1.826/2.383
AGCRN (Bai et al., 2020)	3.660/5.012	2.506/3.580	2.268/3.064	2.567/3.665	1.714/2.476	1.379/1.935	3.328/4.501	2.173/2.934	1.912/2.551
xVal (Golkar et al., 2023)	4.254/5.704	3.504/4.334	2.826/4.089	2.945/4.206	1.923/2.736	1.798/2.711	4.017/5.331	2.232/3.037	1.984/2.788
MoSSL (Deng et al., 2024)	9.319/11.245	8.004/8.940	7.686/8.489	4.757/5.949	3.637/4.313	2.471/2.890	4.802/6.202	5.833/6.630	4.842/5.638
STBP (Liu & Zhang, 2026)	4.635/6.006	3.254/4.278	2.687/3.416	2.707/3.931	1.734/2.569	1.420/2.101	4.588/5.857	2.303/2.992	1.956/2.501
Crime Prediction Models									
DeepCrime (Huang et al., 2018)	3.881/5.259	2.639/3.677	2.388/3.401	2.615/3.723	2.147/3.170	2.369/3.021	3.404/4.634	2.160/2.870	1.760/2.379
ST-SHN (Xia et al., 2022)	4.229/5.494	3.818/4.863	2.875/3.633	2.542/3.659	1.765/2.516	1.352/1.879	3.786/5.041	2.188/2.865	1.749/2.326
STHSL (Li et al., 2022)	4.024/5.175	2.867/3.675	2.482/3.147	2.675/3.676	1.878/2.472	1.536/1.934	3.360/4.352	2.159/2.764	1.955/2.463
ST-HHOL (Du et al., 2026)	4.284/5.564	2.716/3.639	2.604/3.356	2.539/3.660	2.841/4.584	1.355/1.908	3.495/4.743	2.316/3.070	1.754/2.331
LLM-Based Prediction Models									
GPT4TS (Zhou et al., 2023)	3.635/4.984	2.347/3.338	2.322/3.225	2.606/3.724	1.741/2.471	1.367/1.908	3.391/4.461	2.120/2.780	1.758/2.291
Time-LLM (Jin et al., 2024)	3.743/5.138	2.478/3.497	2.425/3.325	2.787/4.001	1.845/2.672	1.393/1.994	3.695/4.904	2.366/3.179	1.772/2.341
UrbanGPT (Li et al., 2024)	3.821/5.313	2.850/4.569	3.045/4.105	2.707/3.842	2.077/2.876	1.420/2.032	4.321/5.861	3.010/4.306	2.106/2.974
CrimeGPT	**3.483/4.824**	**2.232/3.237**	**2.183/3.061**	**2.408/3.371**	**1.691/2.454**	**1.257/1.829**	**3.300/4.268**	**1.995/2.670**	**1.700/2.272**

Table 10: NYC results for $H = 10$, $P = 1$ (MAE/RMSE). Best and second-best results are bold and underlined.

Method	All	Theft	Assault
Spatio-Temporal Forecasting Models			
ARIMA (Box et al., 2015)	4.129/5.803	2.747/3.955	2.506/3.541
DCRNN (Li et al., 2017)	4.592/5.988	2.930/3.889	2.603/3.337
GMAN (Zheng et al., 2020)	3.948/5.308	2.748/3.684	2.484/3.196
AGCRN (Bai et al., 2020)	3.943/5.358	2.503/3.487	2.314/3.110
xVal (Golkar et al., 2023)	4.199/5.834	2.586/3.805	2.614/3.539
MoSSL (Deng et al., 2024)	5.587/7.783	3.018/4.533	2.895/3.766
STBP (Liu & Zhang, 2026)	4.935/6.251	2.865/3.780	2.827/3.515
Crime Prediction Models			
DeepCrime (Huang et al., 2018)	3.922/5.400	2.912/4.421	2.327/3.349
ST-SHN (Xia et al., 2022)	4.646/5.857	2.674/3.955	2.490/3.034
STHSL (Li et al., 2022)	4.659/5.760	3.192/3.992	3.185/3.769
ST-HHOL (Du et al., 2026)	4.250/5.702	2.703/3.563	2.593/3.466
LLM-Based Prediction Models			
GPT4TS (Zhou et al., 2023)	3.768/5.220	2.470/3.487	2.177/3.028
Time-LLM (Jin et al., 2024)	3.671/5.057	2.405/3.437	2.151/3.017
UrbanGPT (Li et al., 2024)	3.961/5.671	2.757/4.217	2.445/3.355
CrimeGPT	**3.572/4.993**	**2.264/3.249**	**2.074/2.928**

BiCORN ordinal supervision with DGRH-based explicit distance regression provides larger benefits in more challenging long-range and higher-uncertainty forecasting scenarios. Nevertheless, the consistent improvements across both configurations demonstrate that CrimeGPT generalizes robustly across different historical observation windows and forecasting horizons, indicating strong adaptability to diverse crime prediction settings commonly adopted in prior studies.

Table 11: Zero/non-zero count errors and occurrence detection of CrimeGPT on NYC. Occurrence is predicted when the non-negative rounded count is greater than zero.

Task	Zero MAE ↓	Non-zero MAE ↓	Macro F1 ↑	True-zero Recall ↑	Occurrence Recall ↑
All	0.042	4.112	0.905	0.958	0.954
Theft	0.159	2.641	0.947	0.967	0.972
Assault	0.198	2.544	0.946	0.953	0.974

Zero/non-zero targets and occurrence detection. Table 11 complements the region-level frequency analysis with a target-level breakdown. Zero targets constitute 12.3–15.4% of the evaluated NYC samples. Without a separate occurrence classifier, CrimeGPT achieves macro-F1 of 0.905–0.947, true-zero recall of 0.953–0.967, and occurrence recall of 0.954–0.974. Its low errors on both zero and non-zero targets together with high occurrence recall show that the regression gains do not arise from collapsing predictions toward either class.

G.2 Comparison of Spatial Relationship Modeling Strategies

Table 12: Comparison of spatial relationship modeling strategies on NYC All Crime 2022.

Spatial Modeling Strategy	**MAE ↓**	**RMSE ↓**
w/o Geographical Distance	3.569	4.979
w/o Taxi Flow	3.553	4.966
w/o Spatial Modeling	3.587	4.985
Top-K Graph (Yu et al., 2017a; Li et al., 2017)	3.579	4.976
Distance-weighted Graph (Yu et al., 2017a; Li et al., 2017)	3.577	4.984
Adaptive Graph Learning (Wu et al., 2019)	3.560	4.960
Multi-view Spatial Modeling (Ours)	**3.537**	**4.921**

Table 12 isolates the effect of spatial relation construction on NYC All Crime 2022. Removing spatial modeling increases MAE from 3.537 to 3.587, confirming that inter-region dependencies provide useful context beyond temporal histories. Removing either geographical distance or taxi-flow connectivity also degrades performance, indicating that physical proximity and human mobility encode complementary relationships. Top-K, distance-weighted, and adaptively learned graphs all outperform the non-spatial variant, while their narrow performance range suggests that explicitly exposing inter-region information is more important than any single graph construction. Nevertheless, jointly modeling adjacency, geographical proximity, and mobility achieves the best MAE and RMSE.

G.3 Baseline Descriptions

General Spatio-Temporal Prediction Models.

ARIMA (Box et al., 2015) is a classical autoregressive integrated moving average model that captures linear temporal dependencies in univariate time series. Since ARIMA is not inherently designed for multimodal inputs, it is trained and evaluated using only crime data in our experiments.

DCRNN (Li et al., 2017) integrates diffusion convolution with recurrent sequence modeling for spatio-temporal forecasting.

GMAN (Zheng et al., 2020) employs a graph multi-attention network with spatial and temporal attention mechanisms to model long-range spatio-temporal correlations.

AGCRN (Bai et al., 2020) introduces adaptive graph convolutional recurrent networks that automatically infer node-specific graph structures without manual definition.

MoSSL (Deng et al., 2024) is a multi-modality spatio-temporal learning framework via self-supervised learning that captures urban dynamics across different temporal granularities.

STBP (Liu & Zhang, 2026) proposes a novel framework that integrates a general spatio-temporal backbone with a scalable contextual pattern bank for improved spatio-temporal prediction.

Crime-Specific Prediction Models.

DeepCrime (Huang et al., 2018) leverages deep learning with auxiliary urban features including POI distributions to augment crime prediction.

ST-SHN (Xia et al., 2022) constructs spatial-temporal hypergraphs to capture high-order inter-region interactions for crime forecasting.

STHSL (Li et al., 2022) proposes a spatial-temporal self-supervised hypergraph learning framework that adaptively infers crime-relevant relational structures.

ST-HHOL (Du et al., 2026) proposes a spatio-temporal hierarchical hypergraph online learning framework to capture multi-scale spatial dependencies in crime dynamics.

LLM-Based Prediction Models.

GPT4TS (Zhou et al., 2023) powers time series forecasting by fine-tuning only the LayerNorm and positional embeddings of a frozen pretrained GPT-2.

UrbanGPT (Li et al., 2024) is an LLM-based urban spatio-temporal prediction framework that aligns spatio-temporal encodings with the semantic space of a large language model via instruction tuning.

Time-LLM (Jin et al., 2024) is a reprogramming-based framework that repurposes frozen large language models for general time series forecasting by aligning time series inputs with natural language representations through prompt-guided transformations.

G.4 Implementation Details

CrimeGPT. We adopt Qwen2.5-7B (Qwen Team, 2024) as the backbone LLM. A total of 100 numerical tokens [N_0], [N_1], . . . , [N_99] are injected into the vocabulary. Token embeddings are initialized using the digit-weighted strategy described in Section 4.2. During instruction tuning, all LLM backbone parameters are frozen;

the CDE, ST projector, numerical token embeddings, language modeling head (`lm_head`), and DGRH are jointly optimized end-to-end, with gradient masking used to restrict updates within the pretrained LLM. No separate pre-training stage is used for the CDE.

The Crime Dynamics Encoder (CDE) stacks $L = 3$ spatio-temporal layers with a gated temporal convolution dilation of 1 and a hidden dimension of 64. The ST projector is a single linear layer mapping from CDE output dimension to the LLM hidden dimension (3584 for Qwen2.5-7B (Qwen Team, 2024)). The Distribution-Guided Regression Head (DGRH) is a two-layer MLP with hidden dimension 512 and ReLU activation, with shared weights applied across all $P = 3$ prediction steps.

The combined training objective is $\mathcal{L} = \mathcal{L}_{\text{CE}} + \lambda\mathcal{L}_{\text{BiCORN}} + \alpha\mathcal{L}_{\text{DGRH}}$, where $\lambda = 1.0$ and $\alpha = 1.0$. The historical window is $H = 15$ and prediction horizon $P = 3$. Z-score normalization is applied to CDE inputs using statistics computed from the training set only to prevent data leakage. Training is conducted for 3 epochs on 4 NVIDIA RTX 5880 GPUs using AdamW optimizer with learning rate 2×10^{-3} and batch size 4, requiring approximately 1.5 hours per run. All reported results for training-based methods are averaged over three independent runs.

LLM-based Baselines. UrbanGPT and other LLM-based baselines are trained using their official open-source implementations on NVIDIA RTX 5880 GPUs, following the hyperparameter configurations reported in their respective papers.

Other Baselines. All spatio-temporal and crime-specific prediction baselines are trained on a single NVIDIA RTX 4090 GPU using their official implementations, with hyperparameters tuned on the validation set following original paper settings.

G.5 Training Details for SFT and RL

Supervised Fine-Tuning (SFT). We apply Low-Rank Adaptation (Hu et al., 2022) to Qwen2.5-7B-Instruct with rank $r = 16$, scaling factor $\alpha = 32$, and dropout 0.05. LoRA modules are inserted into all linear projections of the multi-head attention layers (query, key, value, and output) and feed-forward network layers (gate, up, and down projections).

The model is trained for 3 epochs using the AdamW optimizer with a cosine learning rate scheduler and warmup ratio 0.03. The peak learning rate is set to 2×10^{-4} with batch size 4.

Reinforcement Learning via GRPO. Starting from the SFT checkpoint, we further apply Group Relative Policy Optimization (Shao et al., 2024). The reward is defined as the negative Mean Absolute Error between generated numerical predictions and ground-truth crime counts. Each prompt generates `num_generations` $= 4$ completions for intra-group reward comparison and policy optimization. The RL stage is trained for 1 epoch with learning rate 1×10^{-5} and batch size 32, while reusing the same LoRA configuration as in the SFT stage.