From Marks to Narratives: Language-Augmented Spatio-Temporal Point Processes

Zheng Dong
zdong76@gatech.edu
Georgia Institute of Technology
Atlanta, Georgia, USA

Xiaoyue Liu
xliu800@gatech.edu
Georgia Institute of Technology
Atlanta, Georgia, USA

Abstract

Spatio-temporal point processes (STPPs) provide a principled framework for forecasting discrete events in continuous time and space, yet most existing models represent additional event information as fixed categorical marks. This abstraction is increasingly restrictive for modern event streams, where events are often accompanied by rich free-form textual descriptions and reducing them into categorical labels can lose valuable information about the underlying event dynamics. Large language models emerge as powerful tools for handling textual context. However, they still lack principled mechanisms for modeling complex spatio-temporal dynamics. We introduce language-augmented spatio-temporal point process (LA-STPP), a framework that leverages rich texts in the past events for future event prediction. More importantly, our framework enables free-form text generation as part of the event forecasting. LA-STPP couples an STPP-based forecaster with a fine-tuned language model via a shared history representation that encodes past event times, locations, and textual content. By conditioning language generation on spatio-temporal dynamics, LA-STPP predicts not only the timing and location of future events but also their semantic content. Experiments show that LA-STPP largely improves text prediction quality over text-only baselines while preserving superior spatio-temporal forecasting capability, suggesting a path toward language-capable world models that forecast when, where, and what will happen next.

Keywords

Spatio-temporal point processes, free-form text generation, semantic event forecasting

ACM Reference Format:
Zheng Dong and Xiaoyue Liu. 2026. From Marks to Narratives: Language-Augmented Spatio-Temporal Point Processes. In *The 12th Mining and Learning from Time Series, August 10, 2026, Jeju, Korea.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

Point processes are widely used to model asynchronous event data in real-world applications. These data typically consist of sequences of events, each recording when and where the event occurred, along with additional information that goes beyond the spatio-temporal coordinates, commonly referred to as *marks*. For example, these marks can represent the specific type of offense in crime forecasting [7], earthquake magnitudes in seismology [28], or patient vitals in medical research [17]. To effectively capture the complex dependencies inherent in these marked spatio-temporal sequences, recent deep learning-based point process models [2, 5, 27, 33, 35, 41, 43] have been developed. These neural frameworks have demonstrated considerable success in modeling intricate event dynamics and serve as principled tools for event forecasting.

Nonetheless, existing neural point process models face significant challenges when applied to modern applications. The rise of complex systems increasingly involve event marks that take the form of textual descriptions, encoding rich information about the underlying event dynamics. For instance, police reports details criminal activity [42] and selling product reviews describe user experiences. This unstructured textual data demands more expressive architectures for capturing the mechanisms that drive event occurrence. Existing neural point process models cannot accommodate such data because they handle only categorical or continuous scalar marks. Reducing rich textual content to categorical labels discards information that may be critical for accurate forecasting. A more fundamental limitation is that current neural point processes predict only event times, locations, or categorical marks and possess no capability to generate free-form text. This hinders their potential for wider applications in modern forecasting tasks, where text generation is indispensable for the development of language-based world models.

Large language models (LLMs) provide powerful text understanding and generation capabilities, yet they lack principled mechanisms for modeling complex spatio-temporal event dynamics. Recent efforts to bridge two paradigms [13, 26] improve event representations through language embeddings. However, they still rely on the LLM's innate capability to capture spatio-temporal structure and treat event marks as discrete types without handling rich event-level text.

We introduce LA-STPP, a language-augmented spatio-temporal point process to address this gap. LA-STPP enables the model to take full advantage of past textual information in event marks for spatio-temporal forecasting, while simultaneously generating free-form descriptions for future events. The framework achieves this by coupling principled STPP components with a decoder-only LLM through a shared causal history encoding. Our contributions are:

(1) We introduce the first spatio-temporal point process that integrates free-form event text for spatio-temporal event

KDD '26, Jeju, Korea.

ACM ISBN 979-8-4007-2258-5/26/08
https://doi.org/10.1145/nnnnnnn.nnnnnnn

forecasting, learning the underlying event dynamics implicitly captured by rich semantic information. This extends the mark space from categorical labels to unrestricted natural language.

(2) The framework generates time-aware natural language event descriptions that encode complex spatio-temporal dependencies, rather than producing text independently of the event dynamics.

(3) Experiments on spatio-temporal event datasets with textual marks demonstrate that our spatio-temporal-conditioned text prediction outperforms text-only LLMs, and that text-driven history encoding achieves superior forecasting performance compared to categorical-mark baselines.

2 Related Work

Neural spatio-temporal point processes. Early attempts [10, 27] in neural point processes [32] use RNNs for intensity modeling. Subsequent work adopted transformer architectures [35, 38, 43] to model the event intensity and capture long-range event dependencies, overcoming the limitations of RNN-based architectures. In the spatio-temporal setting, NSTPP [2] uses neural ODEs to model continuous-space dynamics, while DSTPP [33] employs diffusion-based density estimation. All these methods represent event marks as fixed categorical labels. Another line of research [3, 5, 8, 9, 40, 41] adopts kernel-based method in point processes for modeling the intricate event dependencies via a so-called influence kernel, leveraging the statistically principled self-exciting architecture in Hawkes processes [11]. Early attempt on building text-aware influence kernel [42] also demonstrates the benefits of incorporating text information for capturing the underlying event dynamics. However, the reliance on the influence kernel restricts the model capability to handle free-form text, with a more fundamental limitation in generating high-dimensional textual event marks. The application of generative models to point processes has received increasing attention. RNN [15, 31, 34] and diffusion models [6, 18] have been adopted for enhancing both the generation capability and scalability of point process models. However, while these approaches have demonstrated success in event sequence generation, none of them have accounted for the event-level narrative text generation in natural language.

Language models for forecasting. TPP-LLM [26] embeds event type descriptions via an LLM backbone but still classifies into K discrete categories. Language-TPP [13] encodes time intervals as byte tokens for seamless LLM integration, yet predicts categorical marks. LA-STPP uniquely occupies the intersection of principled spatio-temporal forecasting and natural language generation. It differs fundamentally from existing models in its capability to handle rich semantic knowledge encapsulated in the complex event data from modern applications. LA-STPP allows the principled modeling of event dynamics while generates unrestricted natural language descriptions conditioned on learned spatio-temporal dynamics, a key step towards the development of language-based world models.

Spatio-temporal event forecasting. Beyond point process modeling, spatio-temporal event forecasting is a broader research area encompassing graph neural network-based approaches for traffic and urban computing [16, 36], attention-based models for multi-variate time series [39], and physics-informed methods for environmental prediction [29]. It also serves as a critical input to operational planning under uncertainty, such as stochastic capacity planning [20–22], resource allocation [19, 23, 25], and network design [14, 24], where richer predictive signals over space and time can directly improve downstream decision quality. Our work contributes to this ecosystem by producing not only quantitative forecasts but also interpretable textual descriptions of predicted events.

3 Background

Spatio-temporal point processes (STPPs) [30] model a sequence of discrete events $\mathcal{H} = \{(t_i, s_i)\}_{i=1}^n$, where $t_i \in [0, T]$ is the time and $s_i \in \mathcal{S} \subset \mathbb{R}^{d_S}$ is the location of ith event. The event number n is also random. Given the history $\mathcal{H}_t = \{(t_i, s_i) \in \mathcal{H} | t_i < t\}$, an STPP is fully characterized by the conditional intensity function

$$\lambda\left(t, s \mid \mathcal{H}_t\right) = \lim_{\Delta t \downarrow 0, \Delta s \downarrow 0} \frac{\mathbb{E}\left[\mathbb{N}([t, t+\Delta t] \times B(s, \Delta s)) \mid \mathcal{H}_t\right]}{|B(s, \Delta s)| \Delta t},$$

where $B(s, \Delta s)$ is a ball centered at $s \in \mathbb{R}^{d_S}$ with radius Δs, and the counting measure $\mathbb{N}$ is defined as the number of events occurring in $[t, t+\Delta t] \times B(s, \Delta s) \subset \mathbb{R}^{d_S+1}$. Naturally $\lambda\left(t, s|\mathcal{H}_t\right) \geq 0$ for any arbitrary t and s. By conditioning on time aspect, we can re-write the intensity function as $\lambda(t, s|\mathcal{H}_t) = \lambda(t|\mathcal{H}_t) \cdot f(s|t, \mathcal{H}_t)$, where $\lambda(t|\mathcal{H}_t)$ is the ground intensity of the temporal process, and f is the conditional event spatial density. The log-likelihood of observing $\mathcal{H}$ on $[0, T] \times \mathcal{S}$ is given by [4]

$$\ell(\mathcal{H}) = \sum_{i=1}^{n} \log \lambda\left(t_i, s_i | \mathcal{H}_{t_i}\right) - \int_0^T \lambda(t|\mathcal{H}_t) dt \tag{1}$$

4 Method: language-augmented STPP

We consider event sequences with text information for each event, denoted as $\mathcal{H} = \{(t_i, s_i, \text{text}_i)\}_{i=1}^n$ where text_i is a variable-length free-form textual description from vocabulary $\mathcal{V}$. Our goal is two-fold: (1) model the event spatio-temporal dynamics via the intensity λ, and (2) predict the next event's time, location, and text given history.

Given history, we model the joint event distribution via decomposed marked point process intensity [30]:

$$\lambda(t, s, \text{text} \mid \mathcal{H}_t) = \lambda(t \mid \mathcal{H}_t) \cdot f(s \mid t, \mathcal{H}_t) \cdot p(\text{text} \mid t, s, \mathcal{H}_t).$$

where $\lambda(t \mid \mathcal{H}_t)$ and f are the conditional temporal intensity and spatial density, respectively, and $p(\text{text} \mid \cdot)$ is the conditional text likelihood. This factorization separates the modeling challenge into three components that operates on a shared history vector representation. Figure 1 shows an overview of the LA-STPP architecture.

4.1 History Vector

The history encoder jointly represent temporal dynamics, spatial patterns, and textual semantics from the event history. We adopt a decoder-only LLM and inject spatio-temporal event information directly at the embedding level. This preserves the continuous nature of times and locations while leveraging the LLM's capacity for sequential reasoning.

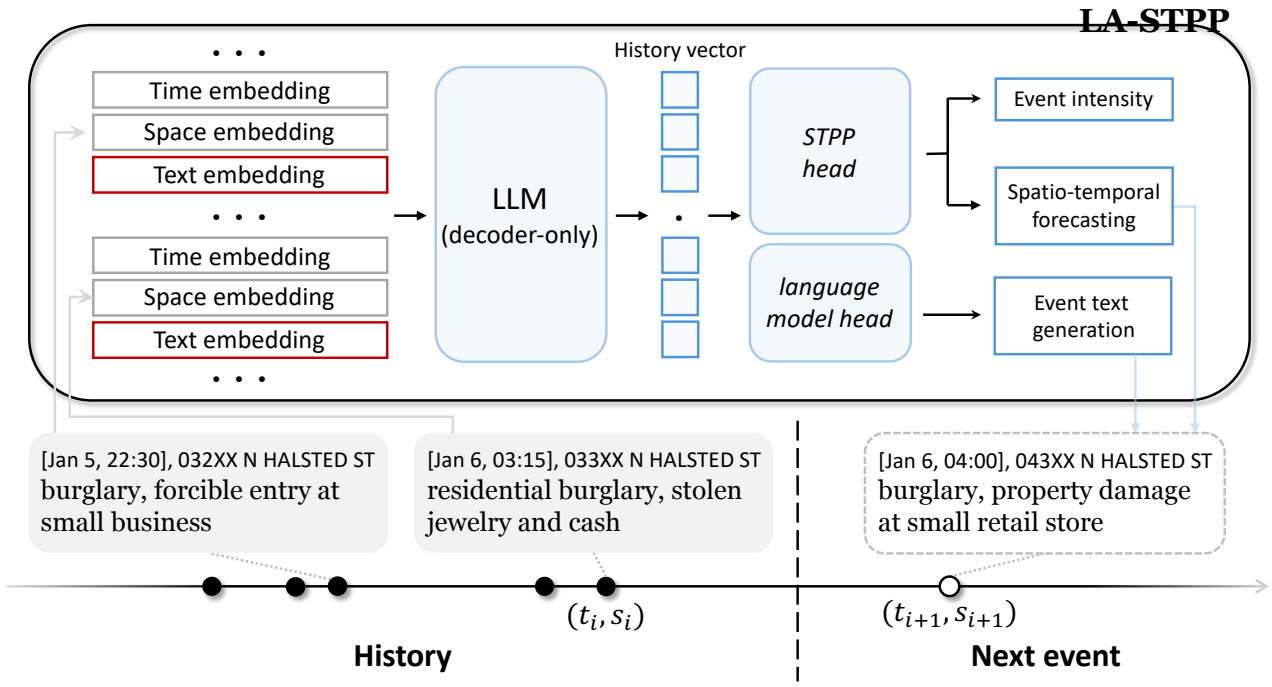

Figure 1: LA-STPP architecture.

Event embedding. Each event $(t_i, s_i, \text{text}_i)$ is mapped to: $\mathbf{E}_i = [\text{TE}(t_i),\ \text{SE}(\mathbf{s}_i),\ \text{LLM\_Emb}(\text{text}_i)]$. The temporal embedding $\text{TE}(t_i) \in \mathbb{R}^D$ encodes the continuous timestamp using sinusoidal positional encoding:

$$\text{TE}(t)_{2j} = \sin(t/10000^{2j/D}),$$

$$\text{TE}(t)_{2j+1} = \cos(t/10000^{2j/D}),$$

where $j = 0, 1, \ldots, D/2-1$. The spatial embedding $\text{SE}(s_i) = \mathbf{W}_s s_i + \mathbf{b}_s \in \mathbb{R}^D$ is a learned linear projection, where $\mathbf{W}_s \in \mathbb{R}^{D\times 2}$ and $\mathbf{b}_s \in \mathbb{R}^D$ are trainable parameters. The text embedding $\text{LLM\_Emb}(\text{text}_i) \in \mathbb{R}^{L_i \times D}$ is obtained by tokenizing text_i and passing the token IDs through the LLM's frozen embedding layer. Each event contributes L_i text tokens (truncated to a maximum of $L_{\max}$ tokens).

History vector extraction. We concatenate embeddings of all events in the sequence and get $\mathbf{X} = [\mathbf{E}_1, \mathbf{E}_2, \ldots, \mathbf{E}_N] \in \mathbb{R}^{(\sum_{i=1}^{n} L_i + 2n)\times D}$. We then input it to the LLM, producing hidden states at every position. The history vector $\mathbf{h}_i \in \mathbb{R}^H$ for observed history up to i-th event is defined as the hidden state at the last token position (the final text token) of event i. This vector encodes the previously observed history and serves as the shared representation for modeling event intensity and predicting next event i+1. Note that we apply causal attention masking to prevent future events from influencing previous events.

4.2 Spatio-Temporal Intensity

Given history vector $\mathbf{h}_i$, the conditional intensity for the next event at time $t > t_i$ is modeled as:

$$\lambda(t \mid \mathcal{H}_t) = \text{softplus}\big(\alpha \cdot (t - t_i) + \mathbf{w}^\top \mathbf{h}_i + b\big),$$

where $\alpha \in \mathbb{R}$, $\mathbf{w} \in \mathbb{R}^H$, and $b \in \mathbb{R}$ are learnable parameters. The softplus activation ensures the practicality of an non-negative intensity. To model conditional spatial event density, we model it as a K-component Gaussian mixture:

$$f(s \mid t, \mathcal{H}_t) = \sum_{k=1}^{K} \pi_k \cdot \mathcal{N}(s;\, \boldsymbol{\mu}_k,\, \Sigma_k).$$

To reflect the dependency on the history, all parameters in the mixture model are predicted based the history vector $\mathbf{h}_i$. For each component k, the weight π_k and the mean vector $\boldsymbol{\mu}_k = [\mu_k^{(1)}, \mu_k^{(2)}]^\top \in \mathbb{R}^2$ are directly learned. We model the covariance matrix Σ_k via Cholesky decomposition as

$$\mathbf{L}_k = \begin{bmatrix} e^{a_k} & 0 \\ b_k & e^{c_k} \end{bmatrix}, \quad \Sigma_k = \mathbf{L}_k \mathbf{L}_k^\top,$$

where (a_k, b_k, c_k) are learnable parameters. We use a single linear network $\mathbf{W}_{\text{gmm}} \in \mathbb{R}^{H\times 6K}$ that outputs $K \times 6$ values as all the mixture parameters.

4.3 Conditional Text Generation

The design of the shared representation in LA-STPP eliminates the need for a separate model or LLM forward pass for the text generation capability. During the forward pass of history vector computation, the LLM produces hidden states at every position in the input sequence, including all text token positions. Thus, we can apply the language model head to project the embeddings to vocabulary at all text token position, predicting the next text token

Table 1: Spatio-temporal event modeling and forecasting performance. Time MAE in hours. Location MAE in km. Best results in bold. The numbers in the parentheses are standard deviation from three independent runs.

Method	*Amazon reviews*		*GitHub events*		*Chicago Crime*		
	Testing ℓ (↑)	Time MAE (↓)	Testing ℓ (↑)	Time MAE (↓)	Testing ℓ (↑)	Time MAE (↓)	Loc. MAE (↓)
NHP	-4.597 (0.102)	9.48	0.373 (0.069)	1.41	-5.560 (0.027)	8.87	/
THP	-9.942 (0.084)	8.93	-0.469 (0.055)	1.24	-4.341 (0.032)	7.62	/
SAHP	-6.638 (0.061)	8.25	-0.024 (0.021)	0.83	-4.942 (0.018)	7.49	/
TPP-LLM	-4.127 (0.072)	7.10	0.185 (0.030)	1.17	-4.578 (0.012)	7.02	/
NSTPP	/	/	/	/	-3.050 (0.065)	7.54	0.89
DSTPP	/	/	/	/	**-2.824** (0.037)	7.35	0.73
LA-STPP	**-3.931** (0.039)	**6.42**	**0.535** (0.031)	**0.62**	-3.172 (0.010)	**6.81**	**0.68**

based on:

$$p(x_{i,j+1} \mid x_{i,\le j}, t_i, s_i, \mathcal{H}_{t_i}) = \text{softmax}(\mathbf{W}_{\text{lm}} \cdot \mathbf{h}_{i,j})$$

where $x_{i,j}$ is the j-th text token in i-th event, $\mathbf{h}_{i,j}$ is the hidden state at the corresponding token position, and $\mathbf{W}_{\text{lm}}$ is the language model head. Therefore, the text generation considers not only the text tokens but also the spatio-temporal knowledge of previous events. The temporal and spatial embeddings preceding each event's text tokens provide spatio-temporal conditioning to the text predictions at no additional computational cost.

4.4 Model Training

The STPP model is trained through Maximum likelihood estimation (MLE) [30]. The final log-likelihood of observing $\mathcal{H}$ can be expressed as

$$\ell(\mathcal{H}) = \sum_{i=1}^{n} \log \lambda\left(t_i|\mathcal{H}_{t_i}\right) + \sum_{i=1}^{n} f(s|t, \mathcal{H}_t) - \int_0^T \lambda(t|\mathcal{H}_t)dt + \sum_{(i,j)\in\mathcal{T}} \log p(x_{i,j+1} \mid x_{i,\le j}, t_i, s_i, \mathcal{H}_{t_i}),$$

where $\mathcal{T}$ is the set of all valid text prediction positions. A higher log-likelihood suggests a better model goodness-of-fit of the event dynamics. For the prediction purpose, we add auxiliary losses $\mathcal{L}_{\text{time}}$ and $\mathcal{L}_{\text{loc}}$ for time and location. The time prediction loss is defined as

$$\mathcal{L}_{\text{time}} = \frac{1}{n-1}\sum_{i=2}^{n}(\log \Delta t_i - \hat{y}_i)^2,$$

where $\Delta t_i = t_i - t_{i-1}$ and $\hat{y}_i = \mathbf{w}_{\text{time}}^\top \mathbf{h}_{i-1} + b_{\text{time}}$ is the next-event time interval prediction. The location prediction loss measures the error between the predicted and true spatial coordinates:

$$\mathcal{L}_{\text{loc}} = \frac{1}{n-1}\sum_{i=2}^{n}\|s_i - \hat{s}_i\|^2,$$

where $\hat{s}_i = \mathbf{W}_{\text{loc}}\mathbf{h}_{i-1} + \mathbf{b}_{\text{loc}} \in \mathbb{R}^2$. The final training objective combines both as:

$$\mathcal{L} = -\ell(\mathcal{H}) + \beta_{\text{time}}\mathcal{L}_{\text{time}} + \beta_{\text{loc}}\mathcal{L}_{\text{loc}}.$$

The weights β_{time} and β_{loc} balance the contribution of the two-fold goal during training.

Fine-tuning strategy. The LLM backbone is fine-tuned with LoRA [12], keeping the majority of parameters frozen while adapting the model to encode domain-specific event dynamics. Other parameters for embedding and intensity are all trained from scratch. The LM head is shared with the pretrained LLM output projection.

5 Experiments

Datasets. We evaluate on three event sequence datasets spanning different domains and modalities: (1) *Amazon Reviews*: user review sequences with timestamps and review text; (2) *GitHub Events*: repository activity streams with timestamps and event descriptions; (3) *Chicago Crime*: crime incident reports with timestamps, geographic coordinates, and textual descriptions.

Implementation. We use TinyLlama-1.1B [37] as the default backbone with LoRA (rank 16, α=16). Training uses Adam with learning rate 5×10^{-4}, batch size 4, gradient checkpointing, and 20 MC samples for the compensator. More details of the hyper-parameter selection and sensitivity analysis are provided in Appendix A and B.

Baselines. For event modeling and forecasting, we compare against state-of-the-art neural point process models, including NHP [27], THP [43], SAHP [38], NSTPP [2], DSTPP [33], and TPP-LLM [26]. We also compare against three text-only baselines on model text generation quality: Random, Claude Sonnet 4 [1], and LoRA Fine-tuned TinyLlama on event text without point process structure.

Metrics. We report log-likelihood on test set to reflect model goodness-of-fit of the data. We use time and location MAE to measure the model prediction power. For generated text quality assessment, we use a retrieval-based protocol since exact-match on free-form text is unreliable: given history and true timestamp/location, we rank the correct next event text against K=50 distractors sampled from the test split, reporting Mean Reciprocal Rank (MRR), Hits@1, and Hits@5. Higher values suggests closer text prediction to the ground truth. More metric details and evaluation protocol are provided in Appendix A.

5.1 Baseline Comparisons

Spatio-temporal forecasting. Table 1 presents the quantitative results on spatio-temporal modeling when handling discrete events

Table 2: Text generation performance. Best results in bold.

Method	MRR (↑)	Hits@1 (↑)	Hits@5 (↑)
Amazon Reviews			
Random	0.090	0.020	0.100
Claude Sonnet 4	0.278	0.172	0.336
TinyLlama-Fine-tuned	0.491	0.359	0.447
LA-STPP	**0.700**	**0.628**	**0.782**
GitHub Events			
Random	0.090	0.020	0.100
Claude Sonnet 4	**0.806**	**0.778**	0.838
TinyLlama-Fine-tuned	0.723	0.655	0.804
LA-STPP	0.770	0.700	**0.842**
Chicago Crime			
Random	0.090	0.020	0.100
Claude Sonnet 4	0.665	0.528	0.852
TinyLlama-Fine-tuned	0.479	0.312	0.583
LA-STPP	**0.683**	**0.536**	**0.896**

associated with free-form texts. On temporal-only textual dataset, LA-STPP achieves the best log-likelihood and time MAE among all methods, outperforming both classical neural point processes and TPP-LLM. This confirms that our text-aware history encoding actively leverages the rich semantic content in event texts, providing informative features for modeling temporal dynamics that categorical marks cannot capture. On the spatio-temporal Chicago Crime dataset, LA-STPP produces competitive goodness-of-fit to the data dynamics against dedicated neural STPP models (NSTPP, DSTPP). While they have no capability to digest textual information, the superior forecasting power of LA-STPP indicates the benefits of incorporating semantic knowledge into event forecasting.

Next-event text generation. Table 2 compares the text generation performance of different methods. As we can see, LA-STPP achieves the best overall performance and has superior performance on Amazon and Chicago Crime across all metrics. These results demonstrate the benefits of our STPP-principled LLM architecture against text-only approaches in providing more accurate text prediction for events that exhibit intricate spatio-temporal dynamics. The frozen pre-trained LLM (Claude Sonnet 4) performs well on GitHub. A possible reason is that repository-specific terminology provides strong lexical cues even without temporal conditioning. The comparison between the fine-tuned LLM and LA-STPP is also valuable, as it showcases that the lack of spatio-temporal knowledge of the event history can result in consistently lower text prediction accuracy.

5.2 Ablation Study

We further study two architectural choices in LA-STPP to showcase the benefits of a fine-tuned LLM adapted to event text and the performance robustness over the choice of LLM backbone. We first remove the LoRA LLM fine-tuning, making LLM backbone frozen as a fixed feature extractor, while the temporal and spatial embedding layers, the intensity head, the spatial GMM head, and the auxiliary prediction heads remains trainable. As shown in Table 3(a), the impact on text prediction is substantial with a significant drop in MRR. This confirms that domain-adapted representations are essential for the model to discriminate between temporally plausible event descriptions. The degradation in testing log-likelihood also suggests the benefits of fine-tuning on learning shared history vector to better capture the underlying event dynamics for intensity modeling. We then evaluate the effect of the pretrained backbone on downstream performance. We varies the LLM backbone while keeping other components identical. As shown in Table 3(b), the three backbones achieve comparable forecasting log-likelihood, suggesting that the STPP component provides a robust mechanism for modeling spatio-temporal dynamics. The comparable MRR scores, with Phi-3.5-mini achieving the best retrieval performance, indicates that stronger backbones can provide modest gains in semantic discrimination.

Table 3: Ablation study results on Chicago crime data.

Variant	Testing ℓ	MRR
(a) Text fine-tuning ablation		
LA-STPP	-3.172	0.683
LA-STPP (no text fine-tuning)	-3.455	0.309
(b) LLM backbone comparison		
TinyLlama-1.1B	-3.172	0.683
Qwen2.5-1.5B	-3.178	0.677
Phi-3.5-mini	-3.168	0.696

6 Conclusion

We presented LA-STPP, a language-augmented spatio-temporal point process framework that bridges principled event dynamics modeling with free-form text generation. LA-STPP leverages rich textual information in past events to improve spatio-temporal forecasting, while simultaneously generating semantically meaningful descriptions of future events conditioned on the learned dynamics. Experiments confirm the advantages of LA-STPP in both event forecasting and text generation in spatio-temporal context.

References

[1] Anthropic. 2024. The Claude Model Family. https://www.anthropic.com.

[2] Ricky T. Q. Chen, Brandon Amos, and Maximilian Nickel. 2021. Neural Spatio-Temporal Point Processes. In *Proceedings of the 9th International Conference on Learning Representations*.

[3] Xiuyuan Cheng, Zheng Dong, and Yao Xie. 2025. Deep Spatiotemporal Point Processes: Advances and New Directions. *Annual Review of Statistics and Its Application* 13 (2025).

[4] Daryl J Daley and David Vere-Jones. 2008. *An Introduction to the Theory of Point Processes. Volume II: General Theory and Structure.* Springer.

[5] Zheng Dong, Xiuyuan Cheng, and Yao Xie. 2023. Spatio-temporal point processes with deep non-stationary kernels. In *The Eleventh International Conference on Learning Representations*.

[6] Zheng Dong, Zekai Fan, and Shixiang Zhu. 2025. Conditional generative modeling for high-dimensional marked temporal point processes. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1*. 248–259.

[7] Zheng Dong, Jorge Mateu, and Yao Xie. 2024. Spatio-temporal-network point processes for modeling crime events with landmarks. *arXiv preprint arXiv:2409.10882* (2024).

[8] Zheng Dong, Matthew Repasky, Xiuyuan Cheng, and Yao Xie. 2026. Deep Graph Kernel Point Processes over Networks. *Journal of Computational and Graphical Statistics* (2026), 1–34.

[9] Zheng Dong, Shixiang Zhu, Yao Xie, Jorge Mateu, and Francisco J Rodríguez-Cortés. 2023. Non-stationary spatio-temporal point process modeling for high-resolution COVID-19 data. *Journal of the Royal Statistical Society Series C: Applied Statistics* 72, 2 (2023), 368–386.
[10] Nan Du, Hanjun Dai, Rakshit Trivedi, Utkarsh Upadhyay, Manuel Gomez-Rodriguez, and Le Song. 2016. Recurrent marked temporal point processes: Embedding event history to vector. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 1555–1564.
[11] Alan G. Hawkes. 1971. Spectra of some self-exciting and mutually exciting point processes. *Biometrika* 58, 1 (1971), 83–90.
[12] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *Proceedings of the 10th International Conference on Learning Representations*.
[13] Quyu Kong, Yixuan Zhang, Yang Liu, Panrong Tong, Enqi Liu, and Feng Zhou. 2025. Language-tpp: Integrating temporal point processes with language models for event analysis. *arXiv preprint arXiv:2502.07139* (2025).
[14] Jingze Li, Xiaoyue Liu, Mathieu Dahan, and Benoit Montreuil. 2024. Stochastic service network design with different operational patterns for hyperconnected relay transportation. *Proceedings of 9th International Physical Internet Conference (IPIC)* (2024).
[15] Shuang Li, Shuai Xiao, Shixiang Zhu, Nan Du, Yao Xie, and Le Song. 2018. Learning temporal point processes via reinforcement learning. *Advances in neural information processing systems* 31 (2018).
[16] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. 2018. Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting. In *Proceedings of the 6th International Conference on Learning Representations*.
[17] Che-Yi Liao, Zheng Dong, Gian-Gabriel P Garcia, Kamran Paynabar, Yao Xie, and Mohammad S Jalali. 2026. Tides Need STEMMED: A Locally Operating Spatiotemporal Mutually Exciting Point Process with Dynamic Network for Improving Opioid Overdose Death Prediction. *Manufacturing & Service Operations Management* 28, 2 (2026), 577–593.
[18] Haitao Lin, Lirong Wu, Guojiang Zhao, Pai Liu, and Stan Z Li. 2022. Exploring generative neural temporal point process. *arXiv preprint arXiv:2208.01874* (2022).
[19] Xiaoyue Liu, Walid Klibi, and Benoit Montreuil. 2026. Modular and Mobile Capacity Planning for Hyperconnected Supply Chain Networks. *arXiv preprint arXiv:2601.11107* (2026).
[20] Xiaoyue Liu, Jingze Li, Mathieu Dahan, and Benoit Montreuil. 2024. Multi-Period Stochastic Logistic Hub Capacity Planning for Relay Transportation. In *Institute of Industrial and Systems Engineers (IISE)*.
[21] Xiaoyue Liu, Jingze Li, Mathieu Dahan, and Benoit Montreuil. 2025. Dynamic hub capacity planning in hyperconnected relay transportation networks under uncertainty. *Transportation Research Part E: Logistics and Transportation Review* 194 (2025), 103940.
[22] Xiaoyue Liu, Jingze Li, and Benoit Montreuil. 2023. Logistics Hub Capacity Deployment in Hyperconnected Transportation Network Under Uncertainty. In *IISE Annual Conference Proceedings*. Institute of Industrial and Systems Engineers (IISE).
[23] Xiaoyue Liu and Benoit Montreuil. 2026. Two-Echelon Delivery Vehicle Sharing and Repositioning in Hyperconnected Urban Logistic Networks. *arXiv preprint arXiv:2606.17608* (2026).
[24] Xiaoyue Liu, Praveen Muthukrishnan, and Benoit Montreuil. 2025. Network design and capacity management in hyperconnected urban logistic networks. *Proceedings of 11th International Physical Internet Conference (IPIC)* (2025).
[25] Xiaoyue Liu, Yujia Xu, and Benoit Montreuil. 2024. Dynamic Containerized Modular Capacity Planning and Resource Allocation in Hyperconnected Supply Chain Ecosystems. In *Proceedings of 10th International Physical Internet Conference (IPIC)*.
[26] Zefang Liu and Yinzhu Quan. 2024. TPP-LLM: Modeling Temporal Point Processes by Efficiently Fine-Tuning Large Language Models. *arXiv preprint arXiv:2410.02062* (2024).
[27] Hongyuan Mei and Jason Eisner. 2017. The Neural Hawkes Process: A Neurally Self-Modulating Multivariate Point Process. In *Advances in Neural Information Processing Systems*, Vol. 30.
[28] Yosihiko Ogata. 1988. Statistical models for earthquake occurrences and residual analysis for point processes. *Journal of the American Statistical association* 83, 401 (1988), 9–27.
[29] Maziar Raissi, Paris Perdikaris, and George E. Karniadakis. 2019. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J. Comput. Phys.* 378 (2019), 686–707.
[30] Alex Reinhart. 2018. A review of self-exciting spatio-temporal point processes and their applications. *Statist. Sci.* 33, 3 (2018), 299–318.
[31] Abhishek Sharma, Aritra Ghosh, and Madalina Fiterau. 2019. Generative sequential stochastic model for marked point processes. In *ICML Time Series Workshop*.
[32] Oleksandr Shchur, Ali Caner Türkmen, Tim Januschowski, and Stephan Günnemann. 2021. Neural Temporal Point Processes: A Review. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence Organization, 4585–4593.
[33] Tao Wang, Kean Chen, Weiyao Lin, John See, Zenghui Zhang, Qian Xu, and Xiaobo Jia. 2020. Spatio-temporal point process for multiple object tracking. *IEEE Transactions on Neural Networks and Learning Systems* 34, 4 (2020), 1777–1788.
[34] Shuai Xiao, Hongteng Xu, Junchi Yan, Mehrdad Farajtabar, Xiaokang Yang, Le Song, and Hongyuan Zha. 2018. Learning conditional generative models for temporal point processes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.
[35] Chenghao Yang, Hongyuan Mei, and Jason Eisner. 2022. Transformer Embeddings of Irregularly Spaced Events and Their Participants. In *Proceedings of the 10th International Conference on Learning Representations*.
[36] Bing Yu, Haoteng Yin, and Zhanxing Zhu. 2018. Spatio-Temporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*. 3634–3640.
[37] Peiyuan Zhang, Guangtao Zeng, Tianduo Wang, and Wei Lu. 2024. TinyLlama: An Open-Source Small Language Model. *arXiv preprint arXiv:2401.02385* (2024).
[38] Qiang Zhang, Aldo Lipani, Omer Kirnap, and Emine Yilmaz. 2020. Self-Attentive Hawkes Process. In *Proceedings of the 37th International Conference on Machine Learning*. 11183–11193.
[39] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. 2021. Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 11106–11115.
[40] Shixiang Zhu, Shuang Li, Zhigang Peng, and Yao Xie. 2021. Imitation learning of neural spatio-temporal point processes. *IEEE Transactions on Knowledge and Data Engineering* 34, 11 (2021), 5391–5402.
[41] Shixiang Zhu, Haoyun Wang, Zheng Dong, Xiuyuan Cheng, and Yao Xie. 2022. Neural Spectral Marked Point Processes. In *International Conference on Learning Representations*.
[42] Shixiang Zhu and Yao Xie. 2022. Spatiotemporal-textual point processes for crime linkage detection. *The Annals of Applied Statistics* 16, 2 (2022), 1151–1170.
[43] Simiao Zuo, Haoming Jiang, Zichong Li, Tuo Zhao, and Hongyuan Zha. 2020. Transformer Hawkes Process. In *Proceedings of the 37th International Conference on Machine Learning*. 11692–11702.

A Experimental Setup

Dataset details. We construct event sequences from three publicly available sources, each exhibiting distinct characteristics in terms of temporal regularity, spatial structure, and textual diversity.

- *Amazon Reviews* consists of product review sequences grouped by individual reviewers. Each event corresponds to a review posted by a user, with the timestamp recording the posting date and the text containing the full review body. We select users with at least 10 reviews and construct sequences ordered chronologically. This dataset is temporal-only (no spatial coordinates) and exhibits irregular inter-event times ranging from hours to months, with review text that varies substantially in length and topic across product categories.
- *GitHub Events* captures repository activity streams from public GitHub repositories. Each event records an action (e.g., push, pull request, issue comment) with a timestamp and an automatically generated or user-written description. Sequences are grouped by repository, producing temporal-only event streams with relatively regular activity patterns. The text in this dataset tends to be concise and repository-specific, making it lexically distinctive across sequences.
- *Chicago Crime* contains crime incident reports from the City of Chicago open data portal. Each event records the timestamp, geographic coordinates (latitude and longitude), and a textual description combining the primary crime type with a detailed narrative of the incident. Sequences are constructed by grouping incidents within geographic regions. This is the

only spatio-temporal dataset in our evaluation, and it features relatively homogeneous text across incidents of similar type.

All datasets are split into train, validation, and test sets with an 80/10/10 ratio. Sequences shorter than 5 events are discarded. Event texts are truncated to a maximum of 32 tokens to maintain tractable sequence lengths during training.

Training configuration. We fine-tune the LLM backbone using LoRA with rank 16, scaling factor $\alpha = 16$, and dropout 0.05, applied to the query, key, value, and output projections of each attention layer. The auxiliary loss weights are set to $\beta_{\text{time}} = 1.0$ and $\beta_{\text{loc}} = 1.0$. The spatial GMM uses $K = 5$ mixture components. We train all models for up to 20 epochs using the Adam optimizer with a learning rate of 5×10^{-4} and apply early stopping based on validation loss with a patience of 5 epochs. Gradient norms are clipped at 1.0 to stabilize training. All experiments are conducted on a single NVIDIA A100 GPU with gradient checkpointing enabled to accommodate the memory requirements of the LLM backbone.

Text generation evaluation protocol. For each test event, we construct a candidate pool of K=50 texts: the ground-truth next event text plus 49 distractors randomly sampled without replacement from all unique texts in the test split. Each model scores all candidates by computing the conditional log-likelihood of each text given the event history and true timestamp/location, then ranks them in descending order. We report three metrics: Mean Reciprocal Rank (MRR = $\frac{1}{|Q|}\sum_{q\in Q}\frac{1}{\text{rank}_q}$), Hits@1 (fraction of queries where the correct text is ranked first), and Hits@5 (fraction where the correct text appears in the top 5). The Random baseline assigns uniform scores to all candidates, yielding Hits@$k = k/K$ and MRR $= H_K/K$ (where $H_K = \sum_{j=1}^{K} 1/j$ is the K-th harmonic number), producing identical performance across datasets. The difficulty of the retrieval task varies by dataset due to differences in text pool diversity: datasets with highly distinctive per-sequence patterns (e.g., GitHub repository-specific terminology) are easier to rank correctly than datasets with more homogeneous text (e.g., Chicago crime descriptions that share similar phrasing across incidents).

B Hyperparameter Sensitivity Analysis

We investigate the sensitivity of LA-STPP to key hyperparameters on the Chicago Crime dataset, varying one parameter at a time while holding all others at their default values. Results are reported in Table 4.

LoRA rank. The LoRA rank r controls the expressiveness of the low-rank adaptation applied to the LLM backbone. Performance improves substantially as rank increases from 4 to 16, reflecting the model's growing capacity to encode domain-specific event patterns into its hidden representations. However, increasing the rank beyond 16 yields slightly worse results, suggesting that excessive adaptation capacity leads to overfitting on the training sequences. The default rank of 16 provides a favorable balance between adaptation expressiveness and generalization.

Table 4: Hyperparameter sensitivity analysis on Chicago Crime. Each row varies one parameter from its default (marked with *). We report testing log-likelihood, time MAE, location MAE, and text retrieval MRR.

Parameter	Value	Testing ℓ	Time MAE	Loc. MAE
LoRA rank (r)	4	-4.819	8.71	0.84
	8	-3.540	7.29	0.75
	16*	-3.172	6.81	0.68
	32	-3.284	6.92	0.70
GMM components (K)	1	-3.183	6.86	0.98
	3	-3.177	6.83	0.78
	5*	-3.172	6.81	0.68
	7	-3.175	6.80	0.69
β_{time}	0.0	-3.166	7.98	0.63
	0.5	-3.170	7.23	0.67
	1.0*	-3.172	6.81	0.68
	2.0	-3.389	6.49	0.82
β_{loc}	0.0	-3.168	6.76	0.85
	0.5	-3.171	6.79	0.74
	1.0*	-3.172	6.81	0.68
	2.0	-3.179	6.82	0.66

GMM components. The number of Gaussian mixture components K governs the flexibility of the spatial density model. A single component produces notably higher location error, as it cannot capture the multi-modal spatial distribution of crime incidents across different neighborhoods. Increasing K from 1 to 5 steadily improves spatial prediction, but further increases to 7 yield diminishing returns with comparable performance. The log-likelihood and time MAE remain largely unaffected by K, confirming that the spatial head operates independently of the temporal intensity component.

Loss weights. The auxiliary loss weights β_{time} and β_{loc} exhibit complementary trade-offs. For β_{time}, removing auxiliary time supervision entirely preserves log-likelihood but degrades time prediction accuracy, while over-weighting it begins to interfere with the intensity-based log-likelihood optimization. For β_{loc}, the location MAE improves monotonically with increasing weight, but aggressive weighting marginally degrades the log-likelihood. The default setting of $\beta_{\text{time}} = \beta_{\text{loc}} = 1.0$ achieves a balanced trade-off across all metrics without substantially compromising any single objective.