ITPP: Learning Disentangled Event Dynamics in Marked Temporal Point Processes

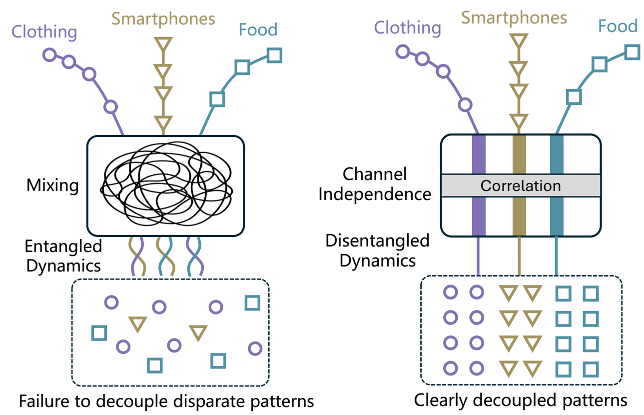
Wang-Tao Zhou[1], **Zhao Kang**[1], **Ke Yan**[1*], **Ling Tian**[12]

[1]University of Electronic Science and Technology of China
[2]Shenzhen Institute for Advanced Study, UESTC
wtzhou@std.uestc.edu.cn, zkang@uestc.edu.cn, kyan@uestc.edu.cn, lingtian@uestc.edu.cn

Abstract

Marked Temporal Point Processes (MTPPs) provide a principled framework for modeling asynchronous event sequences by conditioning on the history of past events. However, most existing MTPP models rely on channel-mixing strategies that encode information from different event types into a single, fixed-size latent representation. This entanglement can obscure type-specific dynamics, leading to performance degradation and increased risk of overfitting. In this work, we introduce ITPP, a novel channel-independent architecture for MTPP modeling that decouples event type information using an encoder-decoder framework with an ODE-based backbone. Central to ITPP is a type-aware inverted self-attention mechanism, designed to explicitly model inter-channel correlations among heterogeneous event types. This architecture enhances effectiveness and robustness while reducing overfitting. Comprehensive experiments on multiple real-world and synthetic datasets demonstrate that ITPP consistently outperforms state-of-the-art MTPP models in both predictive accuracy and generalization.

Code — https://github.com/AnthonyChouGit/ITPP



(a) Channel mixing (b) Channel independence

Figure 1: An illustration of channel-mixing versus channel-independent architectures applied to user browsing history from the Taobao e-commerce dataset. The channel-independent approach avoids the pattern confusion inherent in channel-mixing.

Introduction

The ability to accurately predict future events is of paramount importance across numerous real-world applications, such as healthcare (Tariq et al. 2025), finance (Yang et al. 2019), traffic (Hong et al. 2024), etc. While traditional predictive models often focus on whether an event will occur within a discrete time window, many critical systems operate in continuous time, where the precise timing of an event is as crucial as its occurrence. Modeling such fine-grained temporal event dynamics requires a framework that can capture the stochastic evolution of event times and their associated characteristics. Marked Temporal Point Processes (MTPP) have emerged as a powerful and principled tool for this purpose, providing a mathematical foundation for modeling the joint probability distribution over the time and mark of the next event, conditioned on the history of past events.

MTPP models have garnered considerable research attention in recent years. A substantial portion of these models leverage variants of Recurrent Neural Networks (RNNs) (Du et al. 2016; Mei and Eisner 2017; Shchur, Biloš, and Günnemann 2019; Zhou et al. 2023) or self-attention mechanisms (Zhang et al. 2020; Zuo et al. 2020; Yang, Mei, and Eisner 2022) to encode the sequential patterns within event sequences. More recently, Neural Ordinary Differential Equations (ODEs) have gained traction for modeling temporal irregularities. Following this trend, several ODE-based MTPP models have been proposed (De Brouwer et al. 2019; Jia and Benson 2019; Chen, Amos, and Nickel 2021; Zhou et al. 2025a), which focus on designing sophisticated drift and jump networks to simulate the infinitesimal patterns of state evolution over time.

Existing MTPP models predominantly adopt a channel-mixing approach, encoding the entire history of diverse event types into a unified context embedding. Fig. 1a illustrates the problem of channel-mixing approaches. By forcing a single representation to encapsulate the complex and often disparate dynamics of all event types, e.g., browsing history of irrelevant products, the model's ability to capture the unique, type-specific temporal patterns crucial for accu-

*Corresponding author
Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

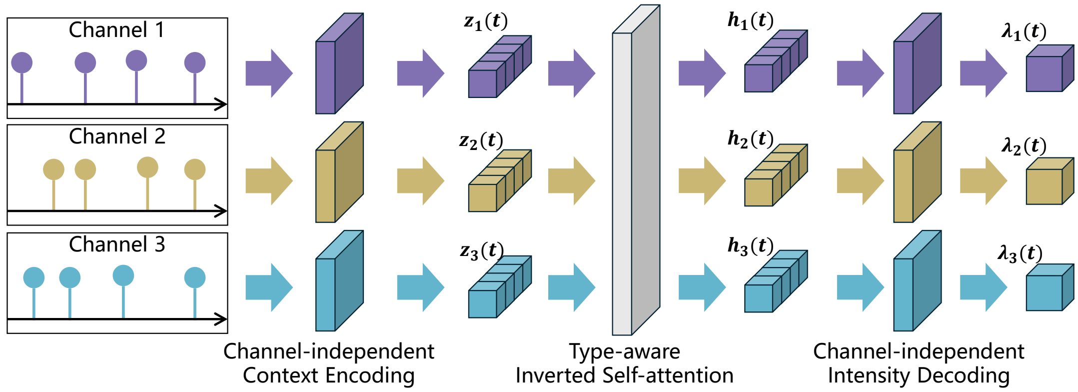


Figure 2: The general architecture of the proposed ITPP model. The framework has a channel-independent encoder-decoder architecture, with a type-aware inverted self-attention layer lying in the middle, which explicitly captures the latent correlations between different channels.

rate prediction can be diluted. This entanglement of information may introduce noise and interference. Consequently, the model may struggle to distinguish and learn the distinct generative processes of each event type, leading to a suboptimal understanding of the underlying dynamics and ultimately, a compromised predictive performance. For instance, when the dynamics of user browsing history for disparate categories like clothing, smartphones, and food are combined into an entangled representation, the model may struggle to isolate category-specific patterns. As a result, predictions for one category can be polluted by noise from another, causing a decline in overall accuracy.

To address this, we propose a strategy that first decoupling event dynamics and then explicitly models inter-channel dependencies. This channel-independent strategy is illustrated in Fig. 1b. We begin by decomposing event sequences into distinct channels, each containing the dynamics regarding a specific event type. This parallel simulation of each channel's state evolution prevents interference between heterogeneous patterns. Recognizing that inter-type correlations are critical for accurate prediction, we then introduce a correlation layer, which is designed specifically to capture the reciprocal effects among channels, e.g., one event type may excite or inhibit another. This channel-independent strategy guarantees a set of decoupled representations for disparate dynamic patterns of different event types. We argue that the proposed architecture enables the model to learn a complete and robust representation of the system's dynamics.

This paper introduces the channel-**I**ndependent marked **T**emporal **P**oint **P**rocess (ITPP), a model designed to enhance fine-grained event prediction by disentangling type-specific event dynamics while explicitly modeling inter-type correlations. As illustrated in Fig. 2, ITPP employs a novel 'encoding-correlation-decoding' architecture. The encoding and decoding stages operate independently on each channel, while an interposed inverted self-attention layer captures cross-channel dependencies. Extensive experiments demonstrate the superiority of our model over state-of-the-art methods. The contributions of this paper can be summarized as follows:

- We propose a novel channel-indepedent architecture for MTPP modelling, named ITPP, which adopts an encoder-decoder architecture with an ODE-based backbone. This framework disentangles the heterogeneous dynamics of different event types, which we argue helps improve the robustness of the model.
- We devise a type-aware inverted self-attention module that captures the correlations between different event channels. A set of type-aware biases is integrated with the vanilla attention mechanism to preserve the inherent connections between different event types while incorporating state-based correlations.
- Extensive experiments are conducted to validate the superiority of the proposed ITPP framework over state-of-the-art MTPP models.

Related Works

Marked Temporal Point Processes

Most Marked Temporal Point Process (MTPP) models use an encoder-decoder architecture to handle asynchronous event data. In this framework, an encoder summarizes the event history into a fixed-size embedding, which a decoder then uses to predict the distribution of the next event. The majority of prior research has concentrated on designing more powerful encoders. (Du et al. 2016; Mei and Eisner 2017) extends RNNs to a sophisticated continuous-time version to jointly encode event marks and irregular time intervals. (Zhang et al. 2020; Zuo et al. 2020; Yang, Mei, and Eisner 2022; Yu et al. 2025) propose modified versions of self-attention modules integrated with elaborately designed temporal embeddings, to replace RNNs as the sequential encoding backbone. Other architectures like CNNs (Zhou et al. 2023) and MLPs (Omi, Aihara et al. 2019) have also been proved effective in MTPP encoding. Recently, neural ordinary differential equations (Chen et al. 2018; De Brouwer et al. 2019) have become an effective tool for modelling infinitesimal dynamic evolution. (Jia and Benson 2019; Chen, Amos, and Nickel 2021; Zhou et al. 2025a; Zhang et al. 2024) adopt ODE-based architectures for encoders, where latent states are formulated as fine-grained trajectories that evolve over time. Some other works devise different decoding architectures other than intensity fitting. (Shchur, Biloš, and Günnemann 2019) propose to use a mixture of log normal distributions to fit the target distribution. Denoising diffusion models have also been applied to MTPP in more complex scenarios such as long-term event prediction (Zhou et al. 2025b), spatio-temporal event modelling (Yuan et al. 2023) and high-dimensional event prediction (Dong, Fan, and Zhu 2025). Existing MTPP models generally use a channel-mixing approach, in which information from different event types is mixed into a single encoding vector. We argue that this entanglement of information undermines the model's robustness and potentially lead to overfitting.

Channel Independence

The concept of channel independence has been explored in multivariate time series forecasting. (Nie et al. 2023) propose to model each variable of a multivariate time series independently with Transformers. This approach stands in contrast to traditional channel-mixing models, which immediately project all variables into a shared latent space. They argue that the most valuable predictive information is contained within the history of a channel itself, and that premature mixing can corrupt these signals. This idea is pushed further by (Chen et al. 2023; Zeng et al. 2023; Yi et al. 2024; Ye et al. 2024), which demonstrate the power of this principle under different architectures. Recently, iTransformer (Liu et al. 2024) has attracted a great deal of attention by combining a channel-independent encoding architecture and an attentive correlation learning module. It treats the entire time series of individual variates as tokens and applies attention across the different independent channels. Despite the demonstrated success of channel independence in time series forecasting, to our knowledge, this principle has not yet been extended to MTPP modeling, a domain where information disentanglement is arguably even more critical. MTPP modelling differs greatly from multivaraite time series forecasting in terms of asynchronization and semantic discretization, and therefore existing frameworks for channel-independence are not applicable in this scenario. In this work, we reformulate the idea of channel-independence for MTPP modelling to decouple disaprate dynamics of different event types.

Preliminary: Marked Temporal Point Process

A Marked Temporal Point Process (MTPP) models a sequence of events occurring in continuous time, where each event i is a time-mark pair (t_i, m_i). The events are ordered by their arrival times $0 < t_1 \leq t_2 \leq \cdots \leq t_L < T$, and each mark m_i belongs to a discrete set of K event types. The behavior of an MTPP is fully characterized by its conditional intensity function, $\lambda_m(t|\mathcal{H}_t)$, where $\mathcal{H}_t = \{(t_j, m_j)|t_j < t\}$ is the history of past events. For clarity, the history condition $\mathcal{H}_t$ is omitted from our notation for the remainder of this paper. The intensity function represents the instantaneous rate of an event occurrence, defined as the expectation of the number of event occurrences per unit of time:

$$\lambda(t) = \lim_{\Delta t \to 0} \frac{\mathbb{E}[N(t + \Delta t) - N(t)]}{\Delta t} \tag{1}$$

where $N(t)$ is a counting measure. A key advantage of the conditional intensity function is that it only needs to be non-negative, unlike a probability density function (PDF) which must integrate to one. This modeling flexibility has led most MTPP research to focus on directly parameterizing the intensity function.

Methodology

In this section, we introduce the proposed ITPP framework, which employs channel-independent encoding and decoding to preserve type-specific dynamic patterns, while adopting a type-aware inverted self-attention mechanism to capture the correlation between different event channels. The general architecture of the proposed model is shown in Fig. 2.

Model Architecture

As shown in Fig. 2, the ITPP model can be divided into three components, namely channel-independent context encoding, type-aware inverted self-attention and channel-independent intensity decoding. Inspired by recent works like (Nie et al. 2023), we propose to treat event occurrences of different types as separate time series and group them into separate channels. Each channel is then encoded independently into a time-variable context embedding $\boldsymbol{z}_k(t) \in \mathbb{R}^d$, where $k \in \{1, ..., K\}$ is the channel index and t is the underlying timestamp. Then, an inverted self-attention layer is applied to exploit the explicit correlation between different channels. Unlike the vanilla Transformers (Vaswani et al. 2017) or iTransformers (Liu et al. 2024), where mapping parameters are generally shared among different entries, our type-aware inverted self-attention uses channel-specific parameters in order to preserve natural connections between event types rather than only state-dependent correlations. The attention output $\boldsymbol{h}_k(t) \in \mathbb{R}^d$ is finally independently decoded to get the type-specific intensity value $\lambda_k(t)$.

Channel-Independent Context Encoding

Unlike state-of-the-art MTPP models, which adopt channel-mixing approaches to encode the history event sequence into a single hidden state, we separate the sequence into multiple channels, each representing an event type. We adopt neural ODEs with jumps to model the fine-grained dynamics of the temporal context. Different channels are simulated simultaneously with the same model formulation and shared parameters. The architecture of our proposed channel-independent context encoding is shown in Fig. 3. The channel-independent state evolution can be divided into two types, namely extrapolations and jumps. The extrapolation process is used to model the smooth state transition within event intervals, while a jump simulates the abrupt state change induced by an event occurrence. Specifically, given the posterior hidden state of the i-th event in the k-th channel, denoted as $\boldsymbol{z}_k(t_{k,i}^+)$, the extrapolation process can be defined as the following ODE:

$$d\boldsymbol{z}_k(t) = \boldsymbol{f}_{\boldsymbol{\theta}_f}(\boldsymbol{z}_k(t), t)dt \tag{2}$$

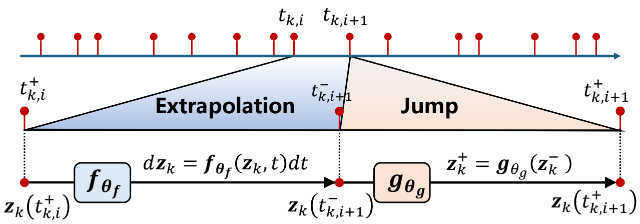


Figure 3: The architecture of channel-independent context encoding. The encoder simulates two types of state transition, namely extrapolation and jump.

where $\boldsymbol{f}_{\boldsymbol{\theta}_f}$ is the drift function, parameterised by $\boldsymbol{\theta}_f$. The hidden state of any time before the next event occurrence can be computed by solving an initial value problem:

$$\boldsymbol{z}_k(t) = \boldsymbol{z}_k(t^+_{k,i}) + \int_{t^+_{k,i}}^{t} \boldsymbol{f}_{\boldsymbol{\theta}_f}(\boldsymbol{z}_k(\tau), \tau)d\tau \tag{3}$$

where $t^+_{k,i} \leq t \leq t^+_{k,i+1}$. A jump is parametrised with a state update neural network. Given the prior state of the i-th event in the k-th channel, denoted as $\boldsymbol{z}_k(t^-_{k,i})$, the jump is defined as:

$$\boldsymbol{z}_k(t^+_{k,i}) = \boldsymbol{g}_{\boldsymbol{\theta}_g}(\boldsymbol{z}_k(t^-_{k,i})) \tag{4}$$

where $\boldsymbol{g}_{\boldsymbol{\theta}_g}$ is the jump function, parameterised by $\boldsymbol{\theta}_g$. The encoded context hidden state $\boldsymbol{z}_k(t)$ is then fed to the type-aware inverted self-attention layer to capture inter-channel correlations, which will be introduced in the next subsection.

Type-Aware Inverted Self-Attention

The type-aware inverted self-attention layer, parameterised by $\boldsymbol{\theta}_c$, is used to capture the latent correlations between different event channels. The general architecture of this module is shown in Fig. 4. This layer adopts a similar structure as the vanilla Transformer. However, unlike other application areas such as Natural Language Processing (NLP), Computer Vision (CV), and time series forecasting, where entries (word tokens, pixels, time steps, etc.) are homogeneous items with positional relations, the channels in our scenario have distinct nature from each other–different event types should not be considered equal entries in the attention process. Thus, we use different parameterisations of the linear mapping layer for different channels. Specifically, the query, key and value computation of channel k is defined as follows:

$$\boldsymbol{q}_k(t) = \boldsymbol{\mathcal{M}}^{\mathcal{Q}}_k(\boldsymbol{z}_k(t)) = \boldsymbol{W}^{\mathcal{Q}}\boldsymbol{z}_k(t) + \boldsymbol{b}^{\mathcal{Q}}_k \tag{5}$$

$$\boldsymbol{k}_k(t) = \boldsymbol{\mathcal{M}}^{\mathcal{K}}_k(\boldsymbol{z}_k(t)) = \boldsymbol{W}^{\mathcal{K}}\boldsymbol{z}_k(t) + \boldsymbol{b}^{\mathcal{K}}_k \tag{6}$$

$$\boldsymbol{v}_k(t) = \boldsymbol{\mathcal{M}}^{\mathcal{V}}_k(\boldsymbol{z}_k(t)) = \boldsymbol{W}^{\mathcal{V}}\boldsymbol{z}_k(t) + \boldsymbol{b}^{\mathcal{V}}_k \tag{7}$$

where $\boldsymbol{W}^{\mathcal{Q}}$, $\boldsymbol{W}^{\mathcal{K}}$ and $\boldsymbol{W}^{\mathcal{V}}$ are the mapping matrices shared across different channels, while $\boldsymbol{b}^{\mathcal{Q}}_k$, $\boldsymbol{b}^{\mathcal{K}}_k$ and $\boldsymbol{b}^{\mathcal{V}}_k$ are channel-specific biases that preserves inherent latent features of event types. The cross-channel self-attention is then performed as:

$$\mathrm{ATTN}(\boldsymbol{Q}(t), \boldsymbol{K}(t), \boldsymbol{V}(t)) = \mathrm{softmax}(\frac{\boldsymbol{Q}(t)\boldsymbol{K}^T(t)}{\sqrt{d_k}})\boldsymbol{V}(t) \tag{8}$$

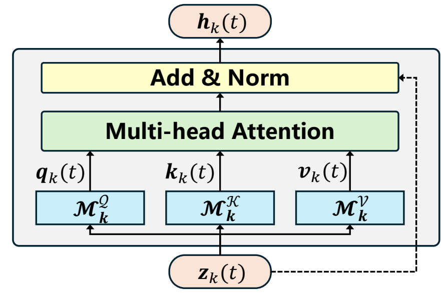


Figure 4: The architecture of type-aware inverted self-attention. The property of type-awareness is achieved by channel-specific mapping functions.

where $\boldsymbol{Q}(t) \in \mathbb{R}^{K \times d_k}$, $\boldsymbol{K}(t) \in \mathbb{R}^{K \times d_k}$ and $\boldsymbol{V}(t) \in \mathbb{R}^{K \times d_v}$ are the query, key and value matrices whose rows are stacked by $\boldsymbol{q}_k(t)$, $\boldsymbol{k}_k(t)$ and $\boldsymbol{v}_k(t)$ respectively. Following the design of the vanilla Transformer, residual connection and layer normalisation are also adopted for stable performance.

Channel-Independent Intensity Decoding

The intensity decoding layer computes the intensity value $\lambda_k(t)$ from the hidden state $\boldsymbol{h}_k(t)$. The decoding process is defined as follows:

$$\lambda_k(t) = \boldsymbol{r}_{\boldsymbol{\theta}_r}(\boldsymbol{h}_k(t)) \tag{9}$$

where $\boldsymbol{r}_{\boldsymbol{\theta}_r}$ is the channel-independent decoding network with shared parameters $\boldsymbol{\theta}_r$ across channels. The joint probability of the next event can be calculated as:

$$f(t,k) = \lambda_k(t)\exp(-\int_{\bar{t}}^{t} \lambda(\tau)d\tau) \tag{10}$$

where $\bar{t}$ is the arrival time of the last known event and $\lambda(t) = \sum_k \lambda_k(t)$ is the total intensity of all event types.

Training

We train our model using the commonly adopted Maximum Log-likelihood Estimation (MLE) approach. The training loss is the Negative Log-Likelihood (NLL) of the event sequence $\mathcal{S} = \{(t_i, m_i)\}_{i=1}^{L}$ in a rolling prediction manner as follows:

$$\mathcal{L}_{\boldsymbol{\theta}}(\mathcal{S}) = -\sum_{i=1}^{L} \log \lambda_{m_i}(t_i) + \int_0^T \lambda(\tau)d\tau \tag{11}$$

where $\boldsymbol{\theta} = \{\boldsymbol{\theta}_f, \boldsymbol{\theta}_g, \boldsymbol{\theta}_c, \boldsymbol{\theta}_r\}$ are the trainable model parameters. The gradients of the model parameters are computed by backpropagation with the help of the adjoint sensitivity method (Chen et al. 2018), which runs the ODEs backward in time.

Experiments

To comprehensively evaluate our proposed model, ITPP, we conduct four sets of experiments, namely probabilistic evaluation, prediction evaluation, intensity recovery and ablation study. Collectively, these experiments validate the effectiveness and robustness of ITPP.

	StackOverflow			MIMIC			Taobao			Earthquake		
Methods	TM-NLL	T-NLL	M-NLL	TM-NLL	T-NLL	M-NLL	TM-NLL	T-NLL	M-NLL	TM-NLL	T-NLL	M-NLL
RMTPP	2.303	0.724	1.580	1.801	0.673	1.128	-0.113	-1.510	1.407	1.819	0.474	1.346
NHP	2.233	0.703	*1.531*	1.204	*0.451*	0.753	-1.003	-2.500	1.497	1.724	0.382	1.342
LogNormMix	2.198	**0.626**	1.572	*1.191*	0.430	0.761	-1.176	-2.539	1.363	1.706	0.367	**1.339**
THP	2.325	0.772	1.553	1.600	0.601	0.999	0.260	-1.337	1.597	1.906	0.561	1.345
SAHP	2.268	0.705	1.562	1.278	0.495	0.783	-0.999	-2.495	1.497	1.740	0.389	1.351
NeuralODE	2.251	0.657	1.594	1.352	0.477	0.874	-1.039	-2.496	1.457	1.722	*0.371*	1.351
ODE-GRU	*2.228*	0.715	1.514	1.340	0.463	0.877	-0.888	-2.376	1.488	1.731	0.388	1.343
CTPP	2.229	*0.651*	1.578	1.176	0.464	0.713	*-1.158*	**-2.570**	*1.412*	*1.715*	0.373	1.342
AttNHP	2.267	0.735	1.532	1.438	0.595	0.842	-0.969	-2.401	1.432	1.717	0.374	1.344
NJDTPP	2.398	0.772	1.626	1.374	0.643	*0.731*	-0.439	-2.004	1.565	1.802	0.451	1.351
ITPP	**2.106**	0.636	**1.470**	**1.054**	**0.427**	**0.627**	**-1.189**	*-2.538*	**1.349**	**1.692**	**0.347**	1.344

Table 1: Probabilistic evaluation results, with the first, second and third best ones shown in bold, underlined and italic styles, respectively.

Datasets	**# events**	**Avg len.**	**# seqs**	**# types**
StackOverflow	142,777	65	2,203	22
MIMIC	2,419	4	650	75
Taobao	115,397	58	2,000	17
Earthquake	70,723	16	4,300	7
Poisson	39,893	80	500	3
Hawkes	21,166	42	500	3

Table 2: Dataset statistics

	SO		MIMIC		Earth.	
Methods	RMSE	F1	RMSE	F1	RMSE	F1
RMTPP	1.021	0.306	0.888	0.823	1.249	0.317
NHP	1.017	0.299	0.863	0.850	1.231	0.303
SAHP	1.019	0.299	0.842	0.834	1.252	0.309
NeuralODE	1.020	0.293	0.884	0.847	1.238	0.328
ODE-GRU	1.009	**0.313**	0.873	0.836	1.252	0.307
NJDTPP	1.030	0.321	0.957	0.841	1.255	**0.320**
ITPP	**1.007**	**0.313**	**0.831**	**0.856**	**1.226**	0.319

Table 3: Prediction evaluation results

Datasets

We evaluate our model on six datasets, comprising four real-world benchmarks—StackOverflow (Leskovec and Krevl 2014), MIMIC (Johnson et al. 2016), Taobao (Xue et al. 2022), and Earthquake (Xue et al. 2024)—and two synthetic datasets generated from a Poisson and a Hawkes process, respectively. These datasets vary in scale, sequence length, and number of event types. The statistics of the datasets are given in Table 2. This heterogeneity allows for a rigorous assessment of our model's robustness and its ability to accommodate across varied data characteristics, which we will compare against state-of-the-art methods. Please refer to the Appendix for more information regarding these datasets.

Baselines

For a comprehensive comparison, we evaluate our model against 10 baselines. These methods span several architectural categories, including RNN-based (RMTPP (Du et al. 2016), NHP (Mei and Eisner 2017), LogNormMix (Shchur, Biloš, and Günnemann 2019)), self-attentive (THP (Zuo et al. 2020), SAHP (Zhang et al. 2020), AttNHP (Yang, Mei, and Eisner 2022)), convolutional (CTPP (Zhou et al. 2023)), and differential equation-based models (NeuralODE (Chen et al. 2018), ODE-GRU (De Brouwer et al. 2019), NJDTPP (Zhang et al. 2024)). A brief introduction of each baseline is given in the Appendix.

Probabilistic Evaluation

We evaluate ITPP's probabilistic performance against state-of-the-art models on four real-world datasets, using the average negative log-likelihood (NLL) per event. The primary metric, the joint NLL of arrival times and marks (TM-NLL), quantifies the models' overall fitting capability. As shown in Table 1, ITPP achieves the best TM-NLL results on all datasets, surpassing the second-best models by a significant margin in every case.

A comparison across different model architectures yields some interesting insights. First, contrary to expectations, self-attentive models (THP, SAHP, and AttNHP) do not outperform strong RNN-based counterparts like LogNormMix and NHP, even on datasets with long sequences (StackOverflow and Taobao) where their ability to capture long-range dependencies should be most advantageous. We hypothesize that this is because self-attention mechanisms inherently lack a strong representation of sequential and temporal information, a factor that is more fundamental to event prediction tasks than to typical NLP applications. Second, the performance of ODE-based models like NeuralODE and ODE-GRU is inconsistent, showing a competitive probabilistic fit only on the StackOverflow dataset, which contains the largest number of events (see Table 2). This suggests

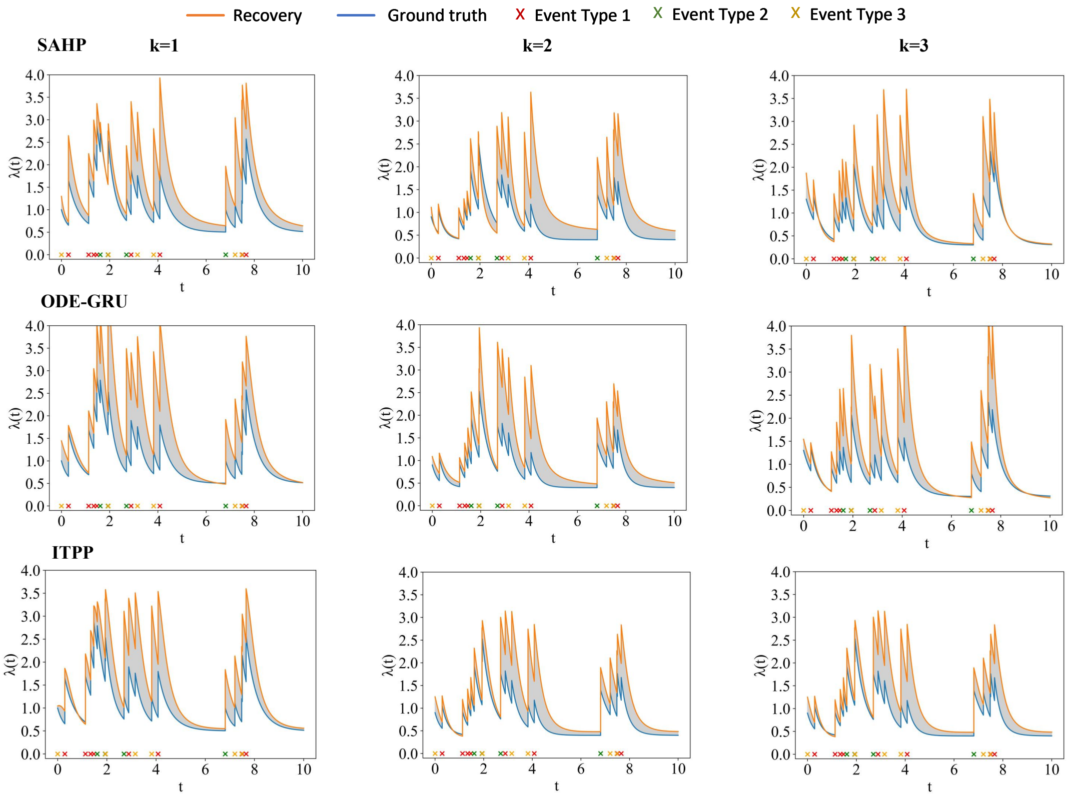


Figure 5: Visualisation of intensity recovery. The grey areas indicate the mass of recovery loss.

a vulnerability to overfitting, as their continuous dynamics may struggle to disentangle information from multiple event types without a large amount of data. However, ITPP resolves this problem with a channel-independent strategy, and achieves a significant performance boost, proving the effectiveness of ODE-based architecture in MTPP modelling.

The time (T-NLL) and mark (M-NLL) components reveal a notable trade-off in the models' focus. On datasets with a relatively large number of event types (StackOverflow, MIMIC, and Taobao, see Table 2), ITPP exhibits significant performance improvements in M-NLL, at the expense of a slight decrease in T-NLL performance. Conversely, on the Earthquake dataset, which has few event types, it focuses more on time loss. We attribute this behavior to ITPP's channel-independent design, which disentangles the dynamics of different event types. This separation enables the model to better distinguish between marks, making it much easier to reduce the mark loss. This effect becomes more pronounced as the number of event types increases, thereby shifting the model's focus to the mark component.

Prediction Evaluation

We now assess the models' capabilities on prediction tasks using three real-world datasets. The Root Mean Square Error (RMSE) and F1 score are used as the metrics for the evaluation of time and mark prediction, respectively. The prediction evaluation results are shown in Table 3. Overall, ITPP dominates the prediction tasks, securing top performance on five of the six indicators and falling only marginally short on the F1 score for the Earthquake dataset. This robust performance contrasts with the inconsistency of the channel-mixing ODE-GRU, which—mirroring the probabilistic results—performs well only on the extensive StackOverflow dataset. This pattern reinforces the conclusion that channel-mixing ODE architectures struggle with generalization and are susceptible to overfitting without large-scale data.

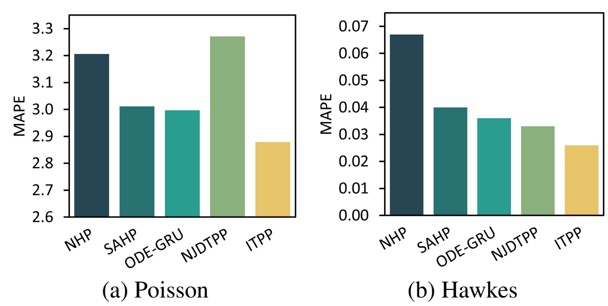
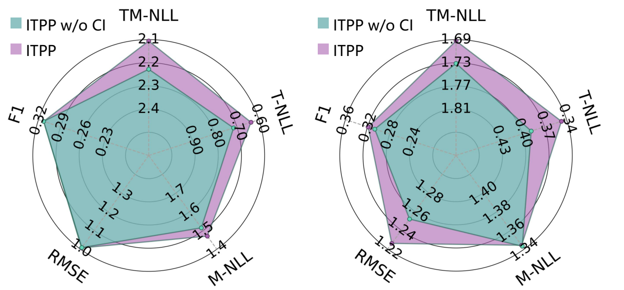


Figure 6: Mean absolute percentage error of intensity recovery on synthetic datasets.

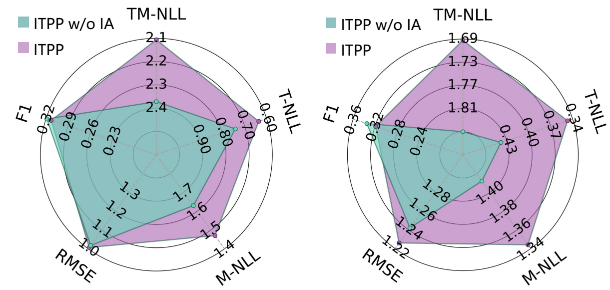
Intensity Recovery

This section evaluates the models' ability to recover the underlying conditional intensity function. We use two synthetic



(a) Stackoverflow (b) Earthquake

Figure 7: Ablation study results of channel independence.



(a) Stackoverflow (b) Earthquake

Figure 8: Ablation study results of inverted self-attention.

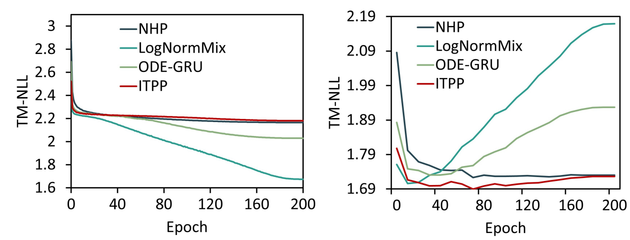
datasets, Poisson and Hawkes, for this analysis, as their ground-truth intensity functions are analytically known. The Mean Absolute Percentage Error (MAPE) is used to measure performance, with the results presented in Figure 6. ITPP significantly outperforms existing intensity-based MTPP models in intensity recovery on both synthetic datasets. This provides a clear rationale for ITPP's superior performance in the probabilistic fitting and point prediction tasks discussed previously. For a qualitative view, Figure 5 visualizes the recovered intensities from the top three models on the Hawkes dataset. The plot confirms the quantitative results, illustrating that ITPP's predicted intensity more closely tracks the ground-truth for all three event types, achieving a visibly tighter fit.

Ablation Study

We now conduct an ablation study to quantify the specific contributions of our two core design choices: channel independence and inverted self-attention. The following experiments are designed to demonstrate that both concepts are crucial to the model's superior performance.

Channel Independence. To quantify the impact of our channel-independent design, we compare the full ITPP model against ITPP w/o CI, a variant that reverts to a conventional channel-mixing approach. The evaluation is conducted across the five metrics used in our previous probabilistic and prediction analyses. Fig. 7 shows the comparison results on StackOverflow and Earthquake datasets. The results demonstrate that the channel-independent design provides a significant performance improvement. This effect is particularly pronounced on smaller-scale datasets, such as Earthquake, as shown in the figure.

Inverted Self-Attention. Next, we evaluate the contribution of the inverted self-attention module, which is designed to capture correlations between different event channels. To achieve this, we compare the full ITPP model against a variant that lacks this module, with the results presented in Fig. 8. We find that removing the inverted self-attention module causes a substantial drop in model performance, which suggests that different event types in the datasets are highly correlated and that failing to capture these dependencies undermines the model's predictive power.



(a) Training loss (b) Testing loss

Figure 9: The changes of training and testing loss as the training progresses.

Overfitting

Overfitting is a significant challenge when training MTPP models, particularly on smaller datasets. Fig. 9 visualizes the training and testing loss curves for ITPP and three baseline models on the Earthquake dataset. The figure reveals severe overfitting in LogNormMix and ODE-GRU, where their training losses decrease steadily while the testing loss starts to surge after some point. This behavior complicates training and compromises the models' robustness for real-world applications. The proposed ITPP, however, by enforcing information disentanglement and explicit correlation capturing, demonstrates stronger resistance to overfitting.

Conclusion

This paper introduces a novel channel-independent framework for MTPP modeling, addressing the limitations of conventional channel-mixing approaches. By leveraging a multi-channel, ODE-based architecture, our model effectively disentangles and simulates the distinct temporal dynamics of each event type. A type-aware inverted self-attention mechanism is proposed to explicitly model inter-type dependencies, enhancing the model's expressiveness. Extensive empirical evaluations demonstrate that our approach improves robustness and generalization and sets a new benchmark in MTPP modeling performance.

Acknowledgments

This research work was partly supported by National Natural Science Foundation of China under grant No. 62376055.

References

Chen, R. T.; Amos, B.; and Nickel, M. 2021. Neural Spatio-Temporal Point Processes. In *International Conference on Learning Representations*.

Chen, R. T.; Rubanova, Y.; Bettencourt, J.; and Duvenaud, D. K. 2018. Neural ordinary differential equations. *Advances in neural information processing systems*, 31.

Chen, S.-A.; Li, C.-L.; Arik, S. O.; Yoder, N. C.; and Pfister, T. 2023. TSMixer: An All-MLP Architecture for Time Series Forecast-ing. *Transactions on Machine Learning Research*.

De Brouwer, E.; Simm, J.; Arany, A.; and Moreau, Y. 2019. GRU-ODE-Bayes: Continuous modeling of sporadically-observed time series. *Advances in neural information processing systems*, 32.

Dong, Z.; Fan, Z.; and Zhu, S. 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.

Du, N.; Dai, H.; Trivedi, R.; Upadhyay, U.; Gomez-Rodriguez, M.; and Song, L. 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.

Hong, Y.; Zhu, H.; Shou, T.; Wang, Z.; Chen, L.; Wang, L.; Wang, C.; and Chen, L. 2024. Storm: A spatio-temporal context-aware model for predicting event-triggered abnormal crowd traffic. *IEEE Transactions on Intelligent Transportation Systems*.

Jia, J.; and Benson, A. R. 2019. Neural jump stochastic differential equations. In *Advances in Neural Information Processing Systems*, 9847–9858.

Johnson, A. E.; Pollard, T. J.; Shen, L.; Lehman, L.-w. H.; Feng, M.; Ghassemi, M.; Moody, B.; Szolovits, P.; Anthony Celi, L.; and Mark, R. G. 2016. MIMIC-III, a freely accessible critical care database. *Scientific data*, 3(1): 1–9.

Leskovec, J.; and Krevl, A. 2014. SNAP Datasets: Stanford Large Network Dataset Collection. http://snap.stanford.edu/data.

Liu, Y.; Hu, T.; Zhang, H.; Wu, H.; Wang, S.; Ma, L.; and Long, M. 2024. iTransformer: Inverted Transformers Are Effective for Time Series Forecasting. In *The Twelfth International Conference on Learning Representations*.

Mei, H.; and Eisner, J. M. 2017. The neural hawkes process: A neurally self-modulating multivariate point process. *Advances in neural information processing systems*, 30.

Nie, Y.; H. Nguyen, N.; Sinthong, P.; and Kalagnanam, J. 2023. A Time Series is Worth 64 Words: Long-term Forecasting with Transformers. In *International Conference on Learning Representations*.

Omi, T.; Aihara, K.; et al. 2019. Fully neural network based model for general temporal point processes. *Advances in neural information processing systems*, 32.

Shchur, O.; Biloš, M.; and Günnemann, S. 2019. Intensity-Free Learning of Temporal Point Processes. In *International Conference on Learning Representations*.

Tariq, A.; Kaur, G.; Su, L.; Gichoya, J.; Patel, B.; and Banerjee, I. 2025. Adaptable graph neural networks design to support generalizability for clinical event prediction. *Journal of biomedical informatics*, 163: 104794.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L. u.; and Polosukhin, I. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Xue, S.; Shi, X.; Chu, Z.; Wang, Y.; Hao, H.; Zhou, F.; Jiang, C.; Pan, C.; Zhang, J. Y.; Wen, Q.; Zhou, J.; and Mei, H. 2024. EasyTPP: Towards Open Benchmarking Temporal Point Processes. In *International Conference on Learning Representations (ICLR)*.

Xue, S.; Shi, X.; Zhang, J.; and Mei, H. 2022. HYPRO: A Hybridly Normalized Probabilistic Model for Long-Horizon Prediction of Event Sequences. *Advances in Neural Information Processing Systems*, 35: 34641–34650.

Yang, C.; Mei, H.; and Eisner, J. 2022. Transformer Embeddings of Irregularly Spaced Events and Their Participants. In *Proceedings of the Tenth International Conference on Learning Representations (ICLR)*.

Yang, Y.; Wei, Z.; Chen, Q.; and Wu, L. 2019. Using external knowledge for financial event prediction based on graph neural networks. In *Proceedings of the 28th ACM international conference on information and knowledge management*, 2161–2164.

Ye, W.; Deng, S.; Zou, Q.; and Gui, N. 2024. Frequency Adaptive Normalization For Non-stationary Time Series Forecasting. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Yi, K.; Fei, J.; Zhang, Q.; He, H.; Hao, S.; Lian, D.; and Fan, W. 2024. FilterNet: Harnessing Frequency Filters for Time Series Forecasting. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Yu, S.; Guo, D.; Fu, Y.; and Jin, W. 2025. EventFormer: a hierarchical neural point process framework for spatio-temporal clustering events prediction. *Journal of Big Data*, 12(1): 162.

Yuan, Y.; Ding, J.; Shao, C.; Jin, D.; and Li, Y. 2023. Spatio-temporal diffusion point processes. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 3173–3184.

Zeng, A.; Chen, M.; Zhang, L.; and Xu, Q. 2023. Are transformers effective for time series forecasting? In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, 11121–11128.

Zhang, Q.; Lipani, A.; Kirnap, O.; and Yilmaz, E. 2020. Self-attentive Hawkes process. In *International conference on machine learning*, 11183–11193. PMLR.

Zhang, S.; Zhou, C.; Liu, Y.; Zhang, P.; Lin, X.; and Ma, Z.-M. 2024. Neural jump-diffusion temporal point processes. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Zhou, W.-T.; Kang, Z.; Liu, S.; Zhang, L.; and Tian, L. 2025a. *Fine-grained Spatio-temporal Event Prediction with Self-adaptive Anchor Graph*, 558–567.

Zhou, W.-T.; Kang, Z.; Tian, L.; and Su, Y. 2023. Intensity-free Convolutional Temporal Point Process: Incorporating Local and Global Event Contexts. *Information Sciences*, 119318.

Zhou, W.-T.; Kang, Z.; Tian, L.; Zhang, J.; and Liu, Y. 2025b. Non-autoregressive diffusion-based temporal point processes for continuous-time long-term event prediction. *Expert Systems with Applications*, 267: 126210.

Zuo, S.; Jiang, H.; Li, Z.; Zhao, T.; and Zha, H. 2020. Transformer hawkes process. In *International conference on machine learning*, 11692–11702. PMLR.