A. Additional Related Work

Event Sequence Modeling With the increasing availability of multi-type event sequences, there has been considerable interest in modeling such data for both prediction and inference. The majority of prior research in this direction has been based on the theory of point processes (Daley & Vere-Jones, 2003), a particular class of stochastic processes that characterize the distribution of random points on the real line. Notably, Hawkes process (Hawkes, 1971a;b), a special class of point process, has been widely investigated, partly due to its ability to capture mutual excitation among events and its mathematical tractability. Hawkes process-based methods (Eichler et al., 2017; Xu et al., 2016; Hall & Willett, 2016; Yang et al., 2017; Achab et al., 2018), however, assume that past events can only independently and additively influence the occurrence of future events, and that influence can only be excitative; these inherent limitations have restricted their modeling flexibility and render them unable to capture complex event interaction in real-world data.

As such, other more flexible models have been proposed, including the *piecewise-constant conditional intensity model* (PCIM) (Gunawardana et al., 2011) and its variants (Weiss & Page, 2013; Bhattacharjya et al., 2018), and more recently a class of models loosely referred to as *neural point processes* (NPPs) (Du et al., 2016; Xiao et al., 2017; Mei & Eisner, 2017; Xiao et al., 2019; Zhang et al., 2020). These models, particularly NPPs, generally enjoy better predictive performance than parametric point processes, since they use more expressive machine learning models (e.g., decision trees, random forests, or recurrent neural networks) to sequentially compute the conditional intensity until next event is generated. A significant weakness of these models, however, is that they are generally uninterpretable and thus unable to provide summary statistics for determining the Granger causality among event types.

Granger Causality Discovery In his seminal paper, Granger (1969) first proposed the concept of Granger causality for time series data. Many approaches have been proposed for uncovering Granger causality for multivariate time series, including the Hiemstra-Jones test (Hiemstra & Jones, 1994) and its improved variant (Diks & Panchenko, 2006), Lasso-Granger method (Arnold et al., 2007), and approaches based on information-theoretic measures (Hlavackova-Schindler et al., 2007). However, as these methods are designed for the synchronous multivariate time series, they are not directly applicable to asynchronous multi-type event sequence data, since otherwise one has to discretize the continuous observation window.

Didelez (2008) first established the Granger causality for event types in event sequences under the framework of marked point processes. Later, Eichler et al. (2017) shows that Granger causality for Hawkes processes is entirely encoded in the excitation kernel functions (also called impact function). To our best knowledge, existing research for Granger causality discovery from event sequences appears to be limited to the case of Hawkes process (Eichler et al., 2017; Xu et al., 2016; Achab et al., 2018), possibly because of this direct link between the process parameterization and Granger causality.

Prediction Attribution for Black-Box Models Prediction attribution, the task of assigning to each input feature a score for representing the feature's contribution to model prediction, has been attracting considerable interest in the field due to its ability to provide insight into predictions made by black-box models such as neural networks. While various approaches have been proposed, there are two prominent groups of approaches: perturbation-based and gradient-based approaches. Perturbation approaches (Zeiler & Fergus, 2014) typically comprise, first, removing, masking, or altering a feature, and then measuring the attribution score of that feature by the change of the model's output. While perturbation-based methods are simple, intuitive, and applicable to almost all black-box models, the quality of the resultant scores is often sensitive to how the perturbation is performed. Moreover, as these methods scale linearly with the number of input features, they become computationally unaffordable for high-dimensional inputs.

In contrast, backpropagation-based methods construct the attribution scores based on the estimation of local gradients of the model around the input instance with backpropagation. The ordinary gradients, however, could suffer from a "saturation" problem for neural networks with activation functions that contain constant-valued regions (e.g., rectifier linear unit (ReLUs)); that is, the gradient coming into a ReLU during the backward pass is zero'd out if the input to the ReLU during the forward pass is in a constant region. One valid solution to this issue is to replace gradients with discrete gradients and use a modified form of backpropagation to compose discrete gradients into attributions, such as layer-wise relevance propagation (LRP) (Bach et al., 2015) and DeepLIFT (Shrikumar et al., 2017). Another solution, proposed by Integrated Gradient (IG) (Sundararajan et al., 2017), is to use the line integral of the gradients along the path from the input to a chosen baseline. Sundararajan et al. (2017) show that IG satisfies many desirable properties, as detailed in Proposition 1.

It is worth mentioning that much existing work often uses the intermediate results, produced by certain intelligible neural network architecture, as the attribution scores for an input. A most notable example of such an idea is the use of attention

weights induced by some attention mechanism as the importance of the input (Bahdanau et al., 2015; Xu et al., 2015). Recently, however, there are growing concerns on the validity of attention weights being used as the explanation of neural networks (Jain & Wallace, 2019; Serrano & Smith, 2019). In particular, Jain & Wallace (2019) show that across a variety of NLP tasks, the learned attention weights are frequently uncorrelated with feature importance produced by gradient-based prediction attribution methods, and random permutation of attention weights can nonetheless yield equivalent predictions.

B. Additional Technical Details

B.1. Proof of Proposition 1

Proof. That both IG and DeepLIFT satisfy A1–A4 has been established in (Sundararajan et al., 2017). P1 is straightforward from the definion of either method. Thus, we only prove that both methods satisfy batchability (P2) with $F(\mathbf{X}) \triangleq \sum_{i=1}^{n} f(\mathbf{x}_i)$.

To prove that IG satisfies batchability, we first rewrite the $\mathrm{IG}(F, \mathbf{X}, \underline{\mathbf{X}})$ as follows:

$$\begin{aligned}
\mathrm{IG}(F, \mathbf{X}, \underline{\mathbf{X}}) &= (\mathbf{X} - \underline{\mathbf{X}}) \odot \int_0^1 \nabla_{\mathbf{X}} F\left[\underline{\mathbf{X}} + \alpha(\mathbf{X} - \underline{\mathbf{X}})\right] d\alpha \\
&= (\mathbf{X} - \underline{\mathbf{X}}) \odot \int_0^1 \sum_{i=1}^{n} \nabla_{\mathbf{X}} f\left[\underline{\mathbf{x}}_i + \alpha(\mathbf{x} - \underline{\mathbf{x}}_i)\right] d\alpha \\
&= (\mathbf{X} - \underline{\mathbf{X}}) \odot \int_0^1 \sum_{i=1}^{n} \{\nabla_{\mathbf{x}_i} f\left[\underline{\mathbf{x}}_i + \alpha(\mathbf{x} - \underline{\mathbf{x}}_i)\right]\} \mathbf{e}_i^T d\alpha \\
&= (\mathbf{X} - \underline{\mathbf{X}}) \odot \left[\int_0^1 \nabla_{\mathbf{x}_i} f\left[\underline{\mathbf{x}}_i + \alpha(\mathbf{x} - \underline{\mathbf{x}}_i)\right] d\alpha\right]_{i=1,\ldots,m},
\end{aligned}$$

where the second step is due to that summation and gradients are swapable, and the third step is because the gradients of different terms are separable. Thus, we have

$$[\mathrm{IG}(F, \mathbf{X}, \underline{\mathbf{X}})]_{:,i} = (\mathbf{x}_i - \underline{\mathbf{x}}_i) \odot \int_0^1 \nabla_{\mathbf{x}_i} f\left[\underline{\mathbf{x}}_i + \alpha(\mathbf{x} - \underline{\mathbf{x}}_i)\right] d\alpha = \mathrm{IG}(f, \mathbf{x}_i, \underline{\mathbf{x}}_i), \tag{14}$$

which establishes the formula.

The proof of DeepLIFT satisfying batchability can be established in a similar way as IG. The key part, shown in the Proposition 2 of (Ancona et al., 2018), is that the attribution scores produced by DeepLIFT for a neural-network-like function f, an input $\underline{\mathbf{x}}$, and a baseline $\underline{\mathbf{x}}$, i.e., $\mathrm{DeepLIFT}(f, \mathbf{x}, \underline{\mathbf{x}})$ can be viewed as the Hadamard product between $\mathbf{x} - \underline{\mathbf{x}}$ and a modified gradient of f at all its internal nonlinear layers. Since the last layer of F is a simple linear addition of all $f(\mathbf{x}_i)$'s, the modified gradient of F for input $\mathbf{x}_i$ is the same as the one of f for $\mathbf{x}_i$. Thus, we have

$$[\mathrm{DeepLIFT}(F, \mathbf{X}, \underline{\mathbf{X}})]_{:,i} = \mathrm{DeepLIFT}(f, \mathbf{x}_i, \underline{\mathbf{x}}_i). \tag{15}$$

□

B.2. Proof of Proposition 2

We first briefly review Shapley values. Suppose there is a team of d players working together to earn a certain amount of value. The value that every coalition $U \subseteq [d]$ achieves is $v(U)$, where $v : 2^d \mapsto \mathbb{R}$ is a value function. Shapley values, proposed by Shapley (1953), provide a well-motivated way to decide how the total earning $v([d])$ should be distributed among such d players. Specifically, the Shapley value for each player $i \in [d]$ is defined as

$$\phi_v(i) = \sum_{U \subseteq [d] \setminus \{i\}} \frac{(|U|!(d - |U| - 1)!)}{d!} \left[v(U \cup \{i\}) - v(U)\right]. \tag{16}$$

For any target function $f \in \mathcal{F}_d$, input $\mathbf{x} \in \mathcal{X}$, and baseline $\underline{\mathbf{x}} \in \mathcal{X}$, we define a value function $v_{f,\mathbf{x},\underline{\mathbf{x}}}(U) \triangleq f(\mathbf{x}_U \sqcup \mathbf{x}_{\bar{U}})$ for any $U \in [d]$, where $\mathbf{x}_U \sqcup \mathbf{x}_{\bar{U}}$ is the spliced data point between $\mathbf{x}$ and $\underline{\mathbf{x}}$, defined in (3). Then the Shapley values $[\phi_{v_{f,\mathbf{x},\underline{\mathbf{x}}}}(i)]_{i \in [d]}$ can be viewed as an attribution method that provides the attribute scores for any f, $\mathbf{x}$, and $\underline{\mathbf{x}}$.

Now we prove that this attribution method based on Shapley valeus satisfies all four axioms (A1–A4) and the fidelity-to-control (P1), as stated in Proposition 2.

Proof. First, it is clear from the definition of Shapley values in (16) that $\phi_{v_{f,\mathbf{x},\underline{\mathbf{x}}}}(\cdot)$ satisfies linearity (A1) and implementation variance (A4). Since Shapley (1953) shows that for any value function v, the Shapley values $\phi_v(\cdot)$ satisfies that

$$\phi_v([d]) - \phi_v(\emptyset) = \sum_{i=1}^{d} \phi_v(i),$$

substituting our definition of the value function $\phi_{v_{f,\mathbf{x},\underline{\mathbf{x}}}}(\cdot)$ into the above equation yields

$$f(\mathbf{x}) - f(\underline{\mathbf{x}}) = \sum_{i=1}^{d} \phi_{v_{f,\mathbf{x},\underline{\mathbf{x}}}}(i),$$

which establishes the completeness (A2). For any $i \in [d]$ and $U \subseteq [d] \setminus \{i\}$, we have

$$v_{f,\mathbf{x},\underline{\mathbf{x}}}(U \cup \{i\}) - v_{f,\mathbf{x},\underline{\mathbf{x}}}(U) = f(\mathbf{x}_{U\cup\{i\}} \sqcup \mathbf{x}_{\bar{U}\setminus\{i\}}) - f(\mathbf{x}_U \sqcup \mathbf{x}_{\bar{U}})$$

Note that $\mathbf{x}_{U\cup\{i\}} \sqcup \mathbf{x}_{\bar{U}\setminus\{i\}}$ and $\mathbf{x}_U \sqcup \mathbf{x}_{\bar{U}}$ only potentially differ on the i-th dimension. If f does not depend on the i-th dimension of its input or $x_i = \underline{x}_i$ (which implies $\mathbf{x}_{U\cup\{i\}} \sqcup \mathbf{x}_{\bar{U}\setminus\{i\}} = \mathbf{x}_U \sqcup \mathbf{x}_{\bar{U}}$), then $f(\mathbf{x}_{U\cup\{i\}} \sqcup \mathbf{x}_{\bar{U}\setminus\{i\}}) = f(\mathbf{x}_U \sqcup \mathbf{x}_{\bar{U}})$ and thereby $\phi_{v_{f,\mathbf{x},\underline{\mathbf{x}}}}(i) = 0$. Thus, $\phi_{v_{f,\mathbf{x},\underline{\mathbf{x}}}}(\cdot)$ satisfies null player (A3) and fidelity-to-control (P1).

□

B.3. Proof of Proposition 3

Proof. We omit the index s in this proof for brevity. First, we rewrite $\tilde{Y}_{k,k'}$ as

$$\begin{aligned}
\tilde{Y}_{k,k'} &= \sum_{i=1}^{n}\sum_{j=1}^{i} \mathbb{I}(k_j = k') A_j(f_k, \mathbf{x}_i, \underline{\mathbf{x}}_i) \\
&= \sum_{j=1}^{n} \mathbb{I}(k_j = k') \left[\sum_{i=j}^{n} A_j(f_k, \mathbf{x}_i, \underline{\mathbf{x}}_i)\right] \\
&= \sum_{j=1}^{n} \mathbb{I}(k_j = k') \left[\sum_{i=j}^{n} A_j(F_{k,i}, \mathbf{x}_n, \underline{\mathbf{x}}_n)\right],
\end{aligned}$$

where in the step step, we replace f with F. Since $F_{k,i}$, i.e., $\int_{t_i}^{t_{i+1}} \lambda_k(t')dt$, does not depend on the events before the i-th event, with null player (A3), we have $A_j(F_{k,i}, \mathbf{x}_n, \underline{\mathbf{x}}_n) = 0$ for any $j < i$, which further implies that

$$\tilde{Y}_{k,k'} = \sum_{j=1}^{n} \mathbb{I}(k_j = k') \left[\sum_{i=1}^{n} A_j(F_{k,i}, \mathbf{x}_n, \underline{\mathbf{x}}_n)\right].$$

With linearity (A1), we have

$$\tilde{Y}_{k,k'} = \sum_{j=1}^{n} \mathbb{I}(k_j = k') A_j \left(\sum_{i=1}^{n} F_{k,i}, \mathbf{x}_n, \underline{\mathbf{x}}_n\right),$$

which establishes the formula.

□

C. Additional Experimental Details

C.1. The Settings for Synthetic and Real-World Datasets.

We describe below the setup and preprocessing details for the five datasets that we consider in this paper.

- **Excitation.** This dataset was generated by a multivariate Hawkes process, whose CIFs are of the form:

$$\lambda_k^*(t) = \mu_k + \sum_{i:t_i<t} \alpha_{k,k'}\beta_{k,k'} \exp[-\beta_{k,k'}(t-t_i)].$$

  We set $S = 1000$, $K = 10$, $n_s \sim \text{Poisson}(250)$, $\mu_k \sim \text{Uniform}(0, 0.01)$, and $\beta_{k,k} \sim \text{Exp}(0.05)$. To generate a sparse excitation weight matrix $\mathbf{A} \triangleq [\alpha_{k,k'}]_{k,k'\in[K]}$, we first selected all its diagonal entries and $M = 16$ random off-diagonal entries, then generated the values for these entries from $\text{Uniform}(0, 1)$, and finally scaled the matrix to have a spectral radius of 0.8.

- **Inhibition.** This dataset was generated by a multivariate self-correcting point process, whose CIFs take the form:

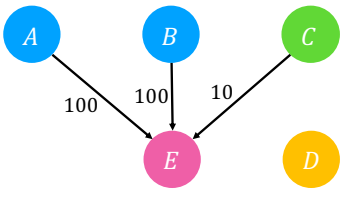
$$\lambda_k^*(t) = \exp(\alpha_k t + \sum_{i:t_i<t} w_{k,k_i}).$$

  We chose $S = 1000$, $K = 10$, $n_s \sim \text{Poisson}(250)$, and $\alpha_k \sim \text{Uniform}(0, 0.05)$. To generated a sparse weight matrix $\mathbf{W} = [w_{k,k'}]_{k,k'\in[K]}$, we first selected all its diagonal entries and $M = 16$ random off-diagonal entries and further generated the values for these entries from $\text{Uniform}(-0.5, 0)$.

- **Synergy.** This dataset was generated by a proximal graphical event model (PGEM) (Bhattacharjya et al., 2018). PGEM assumes that the CIF of an event type depends only on whether or not its parent event types (specified by a dependency graph) have occurred in the most recent history. We designed a local dependency structure that consists of five event types labeled as A–E. Among these event types, type E is the outcome and can be excited by the occurrence of type A, B, or C; type A and B, only when both occurred in the most recent history, would incur a large synergistic effect on type E; type C has an isolated excitative effect on type E and does not interacts with other event types; and finally, type D does not have any excitative effect and is introduced to complicate the learning task. The dependency graph, together with the corresponding time windows and intensity tables, illustrated in Figure 5. To add more complexity to this dataset, we further replicated this local structure for another copy, leading to a total of $K = 10$ event types. We generated $S = 1000$ event sequences with a maximum time span of $T = 1000$.

- **IPTV.** We obtained the dataset from[4]. We further normalized the timestamps into the days and splitted long event sequences so that the length of each sequence is smaller or equal to 1000.

- **MT.** We downloaded the raw MemeTracker phrase data from[5]. We filtered the phrase data that occurred from 2008-08-01 to 2008-09-30 and from the top-100 website domains. We further normalized the timestamps into hours and filtered out those event sequences (i.e., phrase cascades) whose lengths are not in between 3 and 500.



$\lambda_A = \lambda_B = 0.01$
$\lambda_C = \lambda_D = 0.1$
$\lambda_E = 0.01 + \lambda_E(x_A, x_B) + \lambda_E(x_C)$

$\lambda_E(x_A, x_B)$	$\neg x_A$	x_A
$\neg x_B$	0	0
x_B	0	0.1

	$\neg x_C$	x_C
$\lambda_E(x_C)$	0	0.03

Figure 5: The dependency graph, time windows, and intensity tables for the PGEM used in generating the Synergy dataset.

C.2. Implementation Details and Hyperparameter Configurations for Various Methods

For CAUSE, the $\text{Enc}(\cdot)$ and the $\boldsymbol{\alpha}(\cdot)$ were implemented by a single-layer GRU and a two-layer fully connected network with skip connections, respectively. The dimension of event type embeddings was fixed to 64, and the number of hidden

[4] https://github.com/HongtengXu/Hawkes-Process-Toolkit/tree/master/Data
[5] https://www.memetracker.org/data.html#raw

units for GRU was set to be 64 for synthetic datasets and 128 for real-world datasets. The number of basis functions R and the maximum mean L were chosen by a rule of thumb such that μ_2 and μ_R are of the same scale as the 50th and the 99th percentiles of the inter-event elapsed times, respectively. The optimization was conducted by Adam with an initial learning rate of 0.001. A hold-out validation set consisting of 10% of the sequences of the training set was used for model selection; the model snapshot that attains the smallest validation loss was chosen. As events sequence lengths vary greatly on two real-world datasets, in constructing mini-batches for both training and inference, we adopted the bucketing technique to reduce the amount of wasted computation caused by padding. Finally, the line integral of IG, defined in (13), was approximated by 20 steps for MT and 50 steps for other datasets; a smaller number of steps, although may result in certain lose of accuracy, allows for a larger batch size and thus shorter execution time for attribution.

For the Hawkes process-based baselines—HExp, HSG, and NHPC—their implementation was provided by the package `tick` (Bacry et al., 2017). The most relevant hyperparameters for each method were tuned by cross-validation; that is, the decay parameter for the kernel of HExp, the integration support for NPHC, and the maximum mean and the number of Gaussian bases for NPHC, as well as the penalty coefficient for all three methods.

As there is no publicly available codes for RPPN, we implemented it with our best effort. Its overall settings for architecture and optimization is similar to the ones for CAUSE.

C.3. Platform and Runtime

All experiments were conducted on a server with a 16-core CPU, 512G memory, and two Quadro P5000 GPUs. On the largest dataset, MT, the total runtime for CAUSE was less than 3 hours, including both training and computing the Granger causality statistic.

C.4. Qualitative Analysis on MT

Since there are too many event types in MT, instead of a heat map, we visualize the causality matrix as a graph and show in Figure 6a and Figure 6b the top-two communities of that graph, where the directed edges denote the estimated Granger causality between pairs of domains.[6] In Figure 6a, the domain `news.google.com` centers in the middle and is pointed by many sites, which is unsurprising because Google News aggregates articles from other publishers and websites. Our method also correctly identifies other major "information-consuming" domains such as `bogleheads.org`, an active forum for investment-related Q&A. In Figure 6b, the then very popular social networking website `blog.myspace.com` sits in the center of the community. Our method also identifies credible excitative relationships between the subdomains of `craigslist.org`, a mega-website that hosts classified, local advertisements.

D. A Primer on Measure and Probability Theory

In this section, we review some of the basic definitions of in measure theory, which may help the understanding of the definition of Granger causality for multivariate point processes. Most of the content in this section were adapted based on primarily based on the Chapter 1 of (Durrett, 2019) and the Appendix 3 of (Daley & Vere-Jones, 2003),

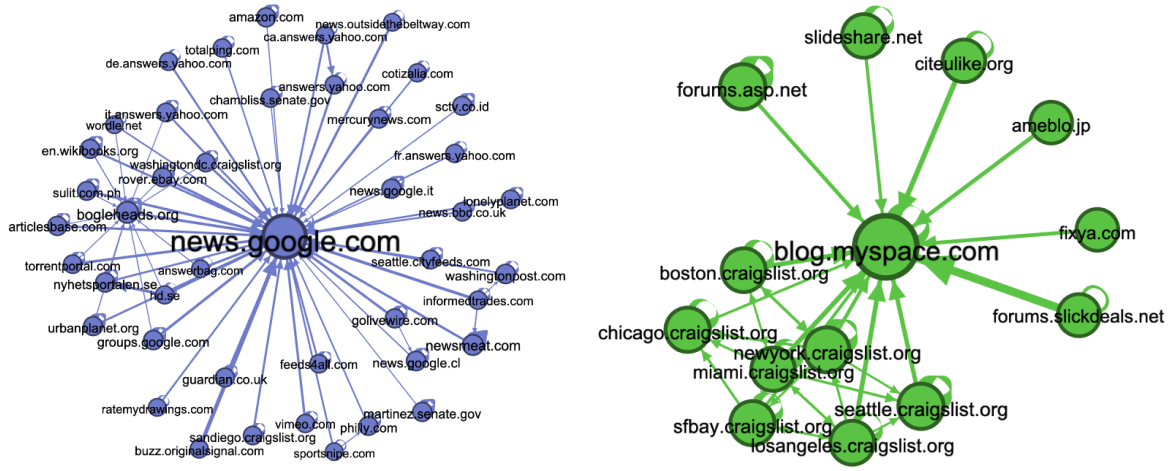
Let Ω be a set of "outcomes" and $\mathcal{F}$ a nonempty collection of subsets of Ω. The set $\mathcal{F}$ is **σ-algebra** of Ω, if it is closed under complement and countable unions; that is,

1. if $A \in \mathcal{F}$, then $\Omega \setminus A \in \mathcal{F}$, and
2. if $A_i \in \mathcal{F}$ is a countable sequence of sets, then $\cup_i A_i \in \mathcal{F}$.

With these two conditions, it's easy to see that σ-algebra is also closed under arbitrary (possibly uncountable) intersections. From this, it follows that given a nonempty set Ω and a collection of $\mathcal{A}$ of subsets of Ω, there is a smallest σ-algebra containing $\mathcal{A}$; we denote such smallest σ-algebra by $\sigma(\mathcal{A})$. One particular σ-algebra is of particular interst—**Borel σ-algebra**; that is, the smallest σ-algebra containing all open sets in $\mathbb{R}^d$, denoted by $\mathcal{R}^d$. Specifically, let $\mathcal{S}_d$ be the empty set plus all sets of the form $(a_1, b_1] \times \cdots (a_d, b_d] \subset \mathbb{R}^d$, where $-\infty \leq a_i < b_i \leq \infty$, then $\mathcal{R}^d = \sigma(\mathcal{S}^d)$. The superscript d is dropped when $d = 1$.

A pair $(\Omega, \mathcal{F})$, in which Ω is a set and $\mathcal{F}$ is a σ-algebra of Ω, is called a **measurable space**. A **measure** defined on $(\Omega, \mathcal{F})$ is a nonnegative countably additive set function; that is a function $\mu : \mathcal{F} \mapsto \mathbb{R}$ with

[6] The graph visualization and community detection were performed using the software `Gephi`.



(a) MT (Community 1)

(b) MT (Community 2)

Figure 6: The top-two communities of the estimated Granger causality statistic matrices on MT (Figure 6a & 6b). Better viewed on screen.

1. $\mu(A) \geq \mu(\emptyset) = 0$ for all $A \in \mathcal{F}$, and
2. if $A_i \in \mathcal{F}$ is a countable sequence of disjoint sets, then

$$\mu(\bigcup_i A_i) = \sum_i \mu(A_i)$$

If $\mu(\Omega) = 1$, we call such a μ a **probability measure**. The triplet $(\Omega, \mathcal{F}, \mu)$ is called a **measure space**, and a **probability space** if μ is a probability measure.

Given a probability space $(\Omega, \mathcal{F}, \mu)$, a real-valued function X defined on Ω is said to be a **random variable** if for every Borel set $B \in \mathcal{R}$ we have $X^{-1}(B) \triangleq \{\omega : X(\omega) \in B\} \in \mathcal{F}$; in another words, X is $\mathcal{F}$**-measuable**. A **stochastic process** is a collection of random variables $\{X_i\}_{i\in\mathcal{I}}$ defined on a common probability space and indexed by a **index set** $\mathcal{I}$. In most cases, the index set can be positive numbers $\mathbb{N}_+$, or real line $\mathbb{R}_+$. A **filtration** is a sequence of σ-algebras, denoted by $\{\mathcal{F}_i\}_{i\in\mathcal{I}}$, if $\mathcal{F}_j \subseteq \mathcal{F}_i$ if $j \leq i$ and $i, j \in \mathcal{I}$. Given a stochastic process $\{X_i\}_{i\in\mathcal{I}}$ defined on $(\Omega, \mathcal{F}, \mu)$, the **natural filtration** of $\mathcal{F}$ with respect to the process is given by

$$\mathcal{H}_i \triangleq \sigma\left(\{X_j^{-1}(B) | j \in \mathcal{I}, j \leq i, B \in \mathcal{R}\}\right). \tag{17}$$

It is in a sense that the simplest filtration filtration available for studying the given: all information concerning the process, and only that information, is avaiable in the natural filtration. Thus, the natural filtration $\mathcal{H}_i$ can be often be viewed as the "**history**" of the subprocess $\{X_j\}_{j\leq i, j\in I}$. Note that sometimes the definition in (17) is simply written as $\mathcal{H}_i \triangleq \sigma\left(\{X_j | j \in \mathcal{I}, j \leq i\}\right)$.

A **point process** $\{T_i\}_{i\geq 1}$ is a real-valued stochastic process indexed on N_+ such that $T_i \leq T_{i+1}$ almost surely. Each random variable is generally viewed as the arrival timestamp of an event. For each point process, one can define a continuously indexed stochastic process associated with it called **counting process**, as $N(t) \triangleq \sum_{i=1}^{\infty} \mathbb{1}(T_i \leq t)$. From this definition, it is easily seen that every realization of a counting process is a càdlàg (i.e. right continuous with left limits) step function, and that a counting process $N(t)$ equivalently defines a point process, as one can recover the event timestamp by $T_i = \inf\{t \geq 0 : N(t) = i\}$. Due to this equivalence, the phrases point process and counting process, as well as their notation, $\{T_i\}_{i\in\mathbb{N}_+}$ and $N(t)$, are often used interchangably in the literature. A K-dimensional **multivariate point process (MPP)** is a coupling of K point/counting process $\mathbf{N}(t) = [N_1(t), N_2(t), \ldots, N_K(t)]$. A realization of a multivariate point process is a multi-type event sequence, $\{(t_i, k_i)\}_{i\in\mathbb{N}_+}$, where t_i indicates the event timestmap of the i-th event, and the k_i indicates which dimension the i-th event comes from (often interpreted as event type).

The most common way to define an MPP is through a set of **conditional intensity functions (CIFs)**, one for each event type. Specifically, let $\mathcal{H}(t) \triangleq \sigma(\{N_k(s) | k \in [K], s < t\})$ for any t be the natural filtration of MPP and let $\mathcal{H}(t-) \triangleq \lim_{s\uparrow t} \mathcal{H}(s)$

the CIF for event type k is defined as the expected instantaneous event occurrence rate conditioned on natural filtration, i.e.,

$$\lambda_k^*(t) \triangleq \lim_{\Delta t \downarrow 0} \frac{\mathbb{E}[N_k(t+\Delta t) - N_k(t) | \mathcal{H}(t)]}{\Delta t},$$

where the use of the asterisk is a notational convention to emphasize that intensity $\lambda_k^*(t)$ must be $\mathcal{H}(t)$-measurable for every t.

Finally, for any $\mathcal{K} \subseteq [K]$, denote by $\mathcal{H}_{\mathcal{K}}(t)$ the natural filtration expanded by the sub-process $\{N_k(t)\}_{k \in \mathcal{K}}$, i.e., $\mathcal{H}_{\mathcal{K}}(t) = \sigma(\{N_k(s) | k \in \mathcal{K}, s < t\})$, and further write $\mathcal{H}_{-k}(t) = \mathcal{H}_{[K]\setminus\{k\}}(t)$ for any $k \in [K]$. For a K-dimensional MPP, event type k is **Granger non-causal** for event type k' if $\lambda_{k'}^*(t)$ is $\mathcal{H}_{-k}(t)$-measurable for all t. This definition amounts to saying that a type k is Granger non-causal for another type k' if, conditioned on the history of events other than type k, the future $\lambda_{k'}^*(t)$ does not depend on the historical events of type k at any time. Otherwise, type k is said to be *Granger causal* for type k.