Beyond Models: Forecasting Complex Network Processes Directly from Data

Bruno Ribeiro
Carnegie Mellon University
Pittsburgh, PA, USA
ribeiro@cs.cmu.edu

Minh X. Hoang
University of California,
Santa Barbara, CA, USA
mhoang@cs.ucsb.edu

Ambuj K. Singh
University of California,
Santa Barbara, CA, USA
ambuj@cs.ucsb.edu

ABSTRACT

Complex network phenomena – such as information cascades in online social networks – are hard to fully observe, model, and forecast. In forecasting, a recent trend has been to forgo the use of parsimonious models in favor of models with increasingly large degrees of freedom that are trained to learn the behavior of a process from historical data. Extrapolating this trend into the future, eventually we would renounce models all together. *But is it possible to forecast the evolution of a complex stochastic process directly from the data without a model?* In this work we show that model-free forecasting is possible. We present SED, an algorithm that forecasts process statistics based on relationships of statistical equivalence using two general axioms and historical data. To the best of our knowledge, SED is the first method that can perform axiomatic, model-free forecasts of complex stochastic processes. Our simulations using simple and complex evolving processes and tests performed on a large real-world dataset show promising results.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database applications—*Data mining*

General Terms

Algorithms; Measurement

Keywords

Model-free Forecasting; Cascade Forecast

1. INTRODUCTION

Complex networked processes – such as information cascades and the spread of influence and viruses over online social networks – are hard to fully observe, model, and forecast. Self-reinforcing and latent effects can drive random and deterministic amplifications that are hard to predict without aggregate, large-population models. For instance, the way people react to social and monetary incentives can affect how information cascades spread over online social networks [12]; individuals also have evolving social interests [19, 24].

In our work we focus on forecasting statistics of complex network processes, such as the distribution of sizes of an epidemic process over a network. Recently, the availability of large-sample historical data – that details the evolution of similar processes in the past – has started a trend of forgoing parsimonious models in favor of models with increasingly large degrees of freedom that are trained over this historical data; these include latent Markovian infection models [24], auto-regressive models [20, 21], distance-based models for time series [14, 37], and classification-based approaches such as logistic regression [4] and naïve Bayes classifiers [13]. Hidden Markov models have also been proposed to replace parsimonious population models in ecology research [26].

Extrapolating this trend into the future, eventually we could renounce models all together. But is this possible? That is, *can we forecast statistics of complex network processes directly from the data without the help of models?* Such forecasting algorithm would be the *ultimate data-driven method.*

Contributions

In this work, we propose SED (Statistical Equivalence Digraph algorithm), an algorithm to forecast statistics of complex networked processes using general axioms that do not entail a model. Our algorithm works by extracting statistical information contained in the historical data through two axioms that define relationships of statistical equivalence between stochastic processes. To the best of our knowledge, our algorithm is the first that can perform (true) model-free forecast of network processes. Using simulation results and also a large real-world dataset, we show that SED is able to accurately forecast a variety of metrics under complex scenarios. We also show that accurate unbiased forecasting is limited to time horizons less than twice of that of the training data for a class of seemly simple infection processes. This proof further motivates our quest for forecasting directly from the data.

An important property of SED is its ability to adjust predictions according to the amount of evidence. Consider forecasting the distribution of sizes of hashtag cascades #A and #B on Twitter:

- We observe 20 hashtag seeds[1] of #A: ten seeds get cascades of size one and ten get cascades of size at least ten.

[1]Seeds are network nodes that are infected independently from other already infected nodes.

WWW 2015, May 18–22, 2015, Florence, Italy.
ACM 978-1-4503-3469-3/15/05.
http://dx.doi.org/10.1145/2736277.2741677.

- We also observe two seeds of #B generating cascades of size one and of size greater than ten, respectively.

Should we make the same forecast for #A and #B? As #B has less samples than #A we are more uncertain about #B's behavior. The lack of evidence prompts SED to assign greater uncertainty when forecasting metrics of #B, automatically adjusting forecasts to the amount of evidence.

Outline

Section 2 provides some definitions used throughout the paper. Section 3 explains the SED algorithm using the prediction of cascade statistics as an example. Section 4 presents the theory behind our approach. Section 5 tests the accuracy of SED using simulated data and shows an application of SED to forecast complex cascades over a large online social network dataset. Finally, Section 6 discusses the related work and Section 7 presents our conclusions.

2. DEFINITIONS

We first give a few definitions used throughout this work. In this paper we also often simplify our exposition by describing what are really general stochastic processes as an infection process on a network, a.k.a., a *cascade process*.

DEFINITION 2.1. *[Cascade Process]* $\Psi^{(i)} \in \Omega$ *is a cascade process with ID* i, $i = 1, \ldots, |\Omega|$. *For each cascade process* $\Psi^{(i)} \in \Omega$ *we observe a set of sample paths. The random variable* $X^{(i)}_{\Delta t}$ *is a measure over the sample paths of* $\Psi^{(i)}$ *over relative time interval* $[0, \Delta t]$, *where time zero is the time relative to the beginning of the cascade process. For instance,* $X^{(i)}_{\Delta t}$ *may be a random variable that gives the size of a cascade of* $\Psi^{(i)}$ *over time window* $[0, \Delta t]$.

The definition of $X^{(i)}_{\Delta t}$ allows us to be conservative and "mistakenly" merge two or more independent cascades of process $\Psi^{(i)}$ into a single larger cascade. This property is particularly useful when there is ambiguity to which independent cascade seed is responsible to infect a set of nodes.

DEFINITION 2.2. *[Forecasting] The process to forecast* $\Psi^{(0)}$, *has* k_0 *observations* $x^{(0,1)}_{\Delta t_1}, \ldots, x^{(0,k_0)}_{\Delta t_1} \sim X^{(0)}_{\Delta t_1}$ *over time window* $[0, \Delta t_1]$, $\Delta t_1 > 0$, *where time zero represents the time that the process starts. A forecast is an estimate of an statistic of* $X^{(0)}_{\Delta t_2}$ *where* $\Delta t_2 > \Delta t_1$. *A forecast assumes that all processes in* Ω *have been observed for at least* Δt_2 *time.*

For instance, if $\Psi^{(0)}$ is a viral marketing campaign we may want to forecast, using the observations of the first day of the campaign, what will be the average number of infected nodes per infected seed after a week, a statistic that predicts the performance of the campaign in the first week per user exposure.

DEFINITION 2.3. *[Historical Data] The historical data of process* $\Psi^{(i)}$, $i = 0, \ldots$, *defined as* $\mathcal{X}^{(i)}_{\Delta t} = \{x^{(i,1)}_{\Delta t}, \ldots, x^{(i,k_i)}_{\Delta t} \sim X^{(i)}_{\Delta t}\}$, *is the set of all available observations of process* $\Psi^{(i)}$ *over time interval length* Δt.

In what follows we illustrate the SED algorithm. The theory behind the algorithm is explained in Section 4.

3. ALGORITHM

In this section, we present the SED Algorithm. Here we focus on the "how-to" solution. Our description of SED focuses on cascade processes to simplify our exposition; the algorithm for general networked processes is presented in Section 4, along with the justifications and the theory behind SED.

Algorithm 1 The SED Algorithm

Input: Historical data $\mathcal{X}^{(i)}_{\Delta t}$, $i = 1, \ldots, |\Omega|$, $\Delta t = \Delta t_1, \Delta t_2$ and $\mathcal{X}^{(0)}_{\Delta t_1}$;
α: Probability amplifier parameter;
n: Bootstrap resample size;
m: Number of bootstrap resamples;
Stat: Statistic of interest of $X^{(0)}_{\Delta t_2}$.

Output: Set of predicted observations of $\text{Stat}(X^{(0)}_{\Delta t_2})$

1: **for** $i := 0$ to $|\Omega|$ **do**
2: $\quad p_i \Leftarrow \frac{\sum_{j=0, j\neq i}^{|\Omega|} P_{\text{Kuiper}}(\mathcal{X}^{(i)}_{\Delta t_1}, \mathcal{X}^{(j)}_{\Delta t_1})}{|\Omega|}$
3: $\quad w(i \rhd 0) \Leftarrow \frac{P_{\text{Kuiper}}(\mathcal{X}^{(i)}_{\Delta t_1}, \mathcal{X}^{(0)}_{\Delta t_1})}{p_i}$
4: **end for**
5: $\vec{W} \Leftarrow [w(1 \rhd 0)^\alpha, ..., w(|\Omega| \rhd 0)^\alpha]$
6: $\vec{W} \Leftarrow \frac{1}{\|\vec{W}\|_1} \vec{W}$
7: $\hat{Y} \Leftarrow [\,]$
8: **for** $j = 1$ to m **do**
9: $\quad$ Sample i with probability $\vec{W}_i$, $i \in \{1, \ldots, |\Omega|\}$.
10: $\quad \hat{\mathcal{X}}^{(0)}_{\Delta t_2} \Leftarrow \text{BoostrapSampling}(\text{src} = \mathcal{X}^{(i)}_{\Delta t_2}, \text{size} = n)$
11: $\quad \hat{Y}[j] \Leftarrow \text{Stat}(\hat{\mathcal{X}}^{(0)}_{\Delta t_2})$
12: **end for**
13: **return** $\hat{Y}$

The SED algorithm (Algorithm 1) takes as input the set of observations of each process $\Psi^{(i)}$, over time interval lengths $\Delta t_1 > 0$ and $\Delta t_2 > \Delta t_1$ for $i = 1, \ldots, |\Omega|$ and over interval length Δt_1 for the process we want to forecast $i = 0$. The goal is to forecast a statistic of $\Psi^{(0)}$ over time interval length Δt_2. The algorithm also requires a parameter α that later we describe how to automatically optimize. For now it is enough to know that a large value of α indicates that the process to forecast behaves like an outlier in our dataset. Two other parameters are related to bootstrap sampling, n and m, where $n = m = 100$ should suffice for most applications but larger values always give more accurate results. The final input is the statistic we wish to forecast.

Step 2 of Algorithm 1 applies a two-sample Kuiper's test [16] to get the probabilities that the observations of two processes i and j, $i \neq j$, come from the same underlying distribution. Step 3 of Algorithm 1 computes the equivalence score $w(i \rhd 0)$ between process $\Psi^{(i)}$ and the process to forecast $\Psi^{(0)}$. We call this equivalence score the *Equivalence Odds Ratio* (EOR), described in details in Section. 4.

With the EOR computed we build a weight vector $\vec{W}$ at step 5 of Algorithm 1, and normalize it using its $L1$-norm at step 6. The probability amplifier α (step 5) is a real number that is chosen automatically by another algorithm that we describe later. The next steps (9-11) forecast the statistic of interest using a two-step bootstrapping process with m and n resamples. These statistics may include, but are not limited to, the Complementary Cumulative Distribution Function (CCDF), mean, standard deviation and second moment of cascade sizes. We start by randomly sampling a process index i according to the weight in $\vec{W}$ (step 9). At step 10 we perform bootstrap sampling from the long-term observation $\mathcal{X}^{(i)}_{\Delta t_2}$ to obtain an estimate of the bootstrap resample

of $\mathcal{X}^{(0)}_{\Delta t_2}$, where $|\hat{\mathcal{X}}^{(0)}_{\Delta t_2}| = n$. Finally, the statistics of the estimated $\hat{\mathcal{X}}^{(0)}_{\Delta t_2}$, e.g., the mean of $\hat{\mathcal{X}}^{(0)}_{\Delta t_2}$, are computed and stored in the vector $\hat{Y}$ (step 11). By repeating steps 9 to 11 m times we obtain the bootstrapped statistics, which provide the forecast of the statistic of interest over $X^{(0)}_{\Delta t_2}$.

3.1 Choosing Amplifier α and Sample Size n

To complete the algorithm, we also need to choose the amplification factor α and the sample size n. Larger values of n make the forecast more accurate, thus n is only limited by computational constraints. In our forecasts, however, we would like to compare our prediction with the ground truth of $X^{(0)}_{\Delta t_2}$, thus, we set $n = |X^{(0)}_{\Delta t_2}|$ to make the comparison easy. Finally, we determine α heuristically by minimizing the squared error in SED's prediction of the value of $\text{Stat}(X^{(0)}_{\Delta t_1})$, which we can easily get an estimate with $\text{Stat}(\mathcal{H}^{(0)}_{\Delta t_1})$.

4. THEORY

The framework of classical statistics, as developed by pioneers like R.A. Fisher [6], describes stochastic processes using models as building blocks. For instance, a process can be a Galton-Watson process, Markovian, auto-regressive, or a hierarchical combination of processes and non-parametric distributions. In 1933 Kolmogorov [15] laid the modern axiomatic foundations of probability theory that eliminated the necessity of models, although models are still useful in Kolmogorov's framework. To the best of our knowledge, our SED method is the first method able to perform forecasting using axioms rather than models.

Roadmap: In what follows (Section 4.1) we detail our axiomatic forecast framework and our SED method. Section 4.2 takes an in-depth look at the problem providing forecast accuracy bounds for a class of stochastic processes, proving that in some scenarios even for the best model cannot make accurate long-term forecasts of seemly simple processes.

4.1 SED: Axiomatic Forecasting

In what follows, we describe our SED method. SED performs forecasts using the following two axioms in lieu of a model. For the sake of conciseness and clarity, the definition of dynamic networked processes is given later in the text in Definition 4.1.

AXIOM 4.1. *The state of any dynamic networked process is unique at any time $t \geq 0$.*

Axiom 4.1 guarantees that at any given time t there is no ambiguity about the true state of the process, even if this true state is not observable.

AXIOM 4.2. *Let Ω be a set of processes with historical data and let $\Psi^{(0)}$ be the process that we wish to forecast. Each process $\Psi' \in \Omega$, where $\Omega' = \Omega \cup \{\Psi^{(0)}\}$, with the exception of at most one process, is equivalent to one and only one other process in Ω' besides itself according to a measure $\lambda : \mathcal{S} \to \mathbb{R}$, where $\mathcal{S}$ is the appropriate σ-algebra.*

Note that Axioms 4.1 and 4.2 do not entail a model. To illustrate the use of Axiom 4.2 in our method consider the following example. Ω' is the set of epidemic processes over the same graph $G = (V, E)$. We wish to forecast another epidemic $\Psi^{(0)}$ using multiple independent sample paths of the processes in Ω'. Let $\lambda(\sigma)$ as defined in Axiom 4.2 count the number of infected nodes in a sample path $\sigma \in \mathcal{S}$. The example also works if λ is another non-trivial metric, such as the cumulative number of infected nodes, the Wiener index, the reproduction number at time $t > 0$. Axiom 4.2 guarantees that the set of all observed infection processes Ω' is such that for any process $\Psi' \in \Omega'$ there is one and only one other process $\Psi'' \in \Omega'$ for which the distribution of the number of infected nodes is exactly the same.

Unless stated otherwise in what follows we simplify our exposition assuming that Ω' has an even number of elements. Our method also can be readily extended to support $\lambda : \mathcal{S} \to \mathbb{R}^n$, $n > 1$, to incorporate complex features of the process. As we see later, the scenario with $n > 1$ requires the use of kernel two-sample tests [10] for the forecast. In what follows we consider $\lambda : \mathcal{S} \to \mathbb{R}$ unless stated otherwise.

Intuition: Axiom 4.2 goes at the heart of what it means to forecast the evolution of a process using historical data. We can only predict the future if, somehow, the present mimics what has happened in the past. When aided by models, there is a notion of behavior distance (e.g. likelihood function), which helps us find which processes in the past look similar to the process we are trying to forecast and use them to predict the future. Because models often have trouble assigning measures of similarity between two significantly different processes, many modern methods only consider "close enough distances", creating a manifold that is then used in the forecast. One of the challenges in manifold learning is determining a close enough distance where the model empirically works without being fooled by the data. Axiom 4.2 makes this distance precisely zero, which brings some advantages. First, the forecast does not need to learn parameters, often a computationally intensive task. Second, in a complex scenario it is probably better to state ignorance about process behavior than being very wrong. Network processes can be hard to forecast. In Section 4.2 we show that the statistical information used in the forecast of a seemly simple process cannot be extrapolated by models beyond the time horizon of the dataset (Δt_2).

Choice of λ: The measure λ can be thought of as a "fingerprint" of process behavior. With a bad choice of λ, too many "unrelated" processes have similar (but not equivalent) statistics, which in turn makes it hard to empirically distinguish them. In these cases SED assigns similar weights to most processes, and forecasting is performed conservatively (large confidence intervals). If the process to forecast has too few observations, SED also acts conservatively to reflect the uncertainty in the limited number of observations.

Violating Axiom 4.2: If no process in the historical data is equivalent to the process we wish to forecast (a reality in some scenarios), then one of two things can happen. If the process to forecast and most processes in the historical data have similar processes but not an equal process, then Axiom 4.2 provides a robust metric of process similarity. It is robust because it does not assume how processes behave and cannot be fooled by data. This shows well in our experiments with a real dataset. If most processes in the historical data are equal and the process to forecast is an outlier, SED will forecast in an uninformative way with large confidence intervals. This outlier status is easy to spot and stating the lack of confidence in the prediction is highly desirable.

Formal Definitions: In what follows, we use Random Marked Point Processes (RMPP) to formalize the networked process. Note that RMPPs allow us to avoid describing how the process evolves and, thus, no model is described. Let $G = (V, E)$ be a directed graph with node set V and edge set $E \subseteq V \times V$. Let $\mathcal{L}_V$ and $\mathcal{L}_E$ denote an arbitrary set of node and edge labels that will be used to define the state of the process at any time. As an example, in an epidemic process $\mathcal{L}_V = \{\text{infected}, \text{susceptible}\}$ and $\mathcal{L}_E = \emptyset$. The following generalization takes into account the complexity of real-world cascading processes.

DEFINITION 4.1. *[Cascade Process] Let $Z_k^{(i)} \subseteq V \times \mathcal{L}_V \times E \times \mathcal{L}_E$ denote the state of the i-th process after $k > 0$ events. Let $W_k^{(i)} > 0$ be the time between the k-th and the $k+1$-st events. A dynamic networked process $\Psi^{(i)}$ is a simple Random Marked Point Process (RMPP) $\Psi^{(i)} = \{(Z_k^{(i)}, W_k^{(i)})\}_{k \in \mathbb{Z}^\star}$ operating over $V \times \mathcal{L}_V \times E \times \mathcal{L}_E$, where $\mathbb{Z}^\star$ is the set of non-negative integers. By convention,*

$$0 = T_0^{(i)} < T_1^{(i)} < \cdots$$

are the successive times at which the process evolves with $W_k^{(i)} := T_{k+1}^{(i)} - T_k^{(i)}$, $k \geq 0$.

Note that the dynamic networked process *need not follow the edges* of G. RMPPs are versatile stochastic models that allow arbitrary spatial and temporal correlations in process evolution. Also note that in real life we only have a finite number of realizations of the processes and, thus, we may not be able to distinguish some of them. In what follows, we define the measure λ precisely and introduce the notion of λ-stochastic equivalence.

DEFINITION 4.2 (λ-STOCHASTIC EQUIVALENCE). *We define two stochastic processes $\Psi^{(a)}$ and $\Psi^{(b)}$ as equivalent according to a Lebesgue measure $\lambda : \mathcal{S} \to \mathbb{R}$ at time window $[0, \Delta t]$, $\Delta t \in (0, \infty)$, if $X_{\Delta t}^{(a)} \stackrel{d}{=} X_{\Delta t}^{(b)}$, where $X_{\Delta t}^{(k)} \equiv \lambda(\sigma_k^{(\Delta t)})$ is a random variable, $\sigma_k = \{(Z_i^{(k)}, W_i^{(k)}) : 0 \leq T_i^{(k)} \leq \Delta t\}$ is a sample path of the stochastic process $\Psi^{(k)}$, $k \in \{a, b\}$, $\mathcal{S}$ is the appropriate σ-algebra, and the operator $\stackrel{d}{=}$ defines the random variables are equal in distribution.*

In what follows, we use Axioms 4.1 and 4.2 and the definition of λ-stochastic equivalence to build an equivalence digraph representing the equivalence relationships between all processes in Ω'.

4.1.1 Stochastic Equivalence Digraph

Axioms 4.1 and 4.2 state that the cascade process $\Psi \in \Omega'$ – e.g., the process of spreading a Twitter hashtagover the Twitter network – has one and only one λ-stochastic equivalent process $\Psi' \in \Omega'$. To create our forecast, all we need to do is to identify this process. This idea yields a simple statement wherein lies the answer to our practical forecasting problem:

THEOREM 4.1. *Any two-sample hypothesis test can be used to determine whether two processes $\Psi^{(i)}, \Psi^{(j)} \in \Omega'$ are λ-stochastic equivalent at time interval $\Delta t > 0$.*

PROOF. Axiom 4.1 states that sample paths are simple and thus admit a stochastic equivalence measure λ. From Axiom 4.2, we know that all processes in Ω' have a λ-stochastic equivalent process (except at most one process if $|\Omega'|$ is odd). The set of sample paths of $\Psi^{(i)}$ and $\Psi^{(j)}$ entail a set of independent observations of $X_{\Delta t}^{(i)}$ and $X_{\Delta t}^{(i)}$). Thus, λ-stochastic equivalence yields $X_{\Delta t}^{(i)} \stackrel{d}{=} X_{\Delta t}^{(i)}$. A two-sample hypothesis test assesses whether independent observations of the random variables $X_{\Delta t}^{(i)}$ and $X_{\Delta t}^{(i)}$ are drawings of the same underlying distribution. Hence, a two-sample hypothesis test is a tool to determine whether $\Psi^{(i)}$ and $\Psi^{(j)}$ are λ-stochastic equivalent at time $\Delta t > 0$. $\square$

Interestingly, a test with low statistical power forces us to make conservative forecasts with large confidence intervals. The forecast uncertainty reflects the lack of statistical information. In such case, we can identify the problem as the test says that all processes in Ω' have a similar p-value. An interesting consequence of Axiom 4.2 and Theorem 4.1 is our ability to create a complete digraph whose nodes are the processes in Ω' and the weights represent the probability that they are λ-stochastic equivalent. We call the weights of the edges in this digraph the Equivalence Odds Ratio (EOR), defined as follows.

DEFINITION 4.3. *[Equivalence Odds Ratio (EOR)] The EOR, denoted $w_t(i \rhd j)$, $i \neq j$, under n observations $\mathcal{X}_{\Delta t}^{(i)} = \{x_{\Delta t}^{(i,1)}, \ldots, x_{\Delta t}^{(i,n)}\}$ of $X_{\Delta t}^{(i)}$ and m observations $\mathcal{X}_{\Delta t}^{(j)} = \{x_{\Delta t}^{(j,1)}, \ldots, x_{\Delta t}^{(j,m)}\}$ of $X_{\Delta t}^{(j)}$ is*

$$w_t(i \rhd j) = |\Omega'| \frac{P_{Kuiper}(\mathcal{X}_{\Delta t}^{(j)}, \mathcal{X}_{\Delta t}^{(i)})}{\sum_{\Psi^{(h)} \in \Omega' \setminus \{\Psi^{(j)}\}} P_{Kuiper}(\mathcal{X}_{\Delta t}^{(i)}, \mathcal{X}_{\Delta t}^{(h)})} \quad (1)$$

where $P_{Kuiper}(\cdot, \cdot)$ is the p-value of a two-sample Kuiper's test [16, 32], $\mathcal{X}^{(h)}$ are the observations of process $\Psi^{(h)}$, and with Ω' is as defined in Axiom 4.2 and $X_{\Delta t}^{(i)}$, $X_{\Delta t}^{(j)}$ as in Definition 4.2.

The Equivalence Odds Ratio (EOR) $w_t(i \rhd j)$ is the ratio between $p(\mathcal{X}_{\Delta t}^{(i)}, \mathcal{X}_{\Delta t}^{(j)})$ and the average p-value when i is compared to all other cascades in $\Omega' \setminus \{\Psi^{(j)}\}$. The EOR is not to be confused with a likelihood-ratio test, which compares the likelihood of two models given the data. In our framework there are no models.

DEFINITION 4.4 (STOCHASTIC EQUIVALENCE DIGRAPH). *Let $\mathcal{G}_{X_{\Delta t}}$ be a simple (no loops) complete digraph that connects all processes in Ω' with directed edge weights. For any two cascade processes $\Psi^{(i)}, \Psi^{(j)} \in \Omega'$, $i \neq j$, the weight of edge $i \to j$ is the EOR $w_t(i \rhd j)$.*

An EOR $w_t(i \rhd j) > 1$ suggests that process $\Psi^{(i)}$ is $w_t(i \rhd j)$ times more likely to be λ-stochastic equivalent to $\Psi^{(j)}$ than a random process selected from $\Omega' \setminus \{\Psi^{(j)}\}$. Note that the equivalence odds ratio is not symmetric, i.e., there can be processes $\Psi^{(i)}$ and $\Psi^{(j)}$ such that $w_t(i \rhd j) \neq w_t(j \rhd i)$. Note that Definition 4.4 does not impose the one-to-one equivalence required by Axiom 4.2. This is because the one-to-one equivalence is unnecessary and computationally expensive if Ω' is large, thus we assume we can safely approximate it through probabilistic matching. The Kuiper test in P_{Kuiper} can be also replaced by any two-sample hypothesis test such as Kolmogorov-Smirnov's test or extend our method to include P_{kernel} [10] that allows the use of a multi-dimensional λ through kernel two-sample tests. In one dimension, we choose Kuiper's test over Kolmogorov-Smirnov's test because the former gives equal importance to all domain values while the latter tends to be most sensitive around the median value [16, 32].

Next, we show how to use the *stochastic equivalence digraph* $\mathcal{G}_{X_{\Delta t}}$ to make predictions of cascade characteristics.

4.1.2 Forecasting Procedure in Time

Using the stochastic equivalence digraph $\mathcal{G}_{X_{\Delta t_1}}$ we can forecast $\Psi^{(i)}$ using an of estimate $P[X^{(i)}_{\Delta t_2} > x]$, $0 < \Delta t_1 < \Delta t_2$, obtained by the mixture

$$P[X^{(i)}_{\Delta t_2} > x] = \frac{1}{C} \sum_{\Psi^{(j)} \in \Omega' \setminus \{\Psi^{(i)}\}} w_{\Delta t_1}(j \rhd i) P[X^{(j)}_{\Delta t_2} > x],$$

where $C = \sum_{j \in \Omega'} w_{\Delta t_1}(j \rhd i)$ is a normalization constant. The above equation gives the first iteration of Sinkhorn's algorithm that finds probabilistic pairwise matches in a weighted graph [31]. For computational reasons we limit our matching to a single iteration. Sinkhorn's algorithm converges to a doubly stochastic matrix that defines the probability of pairwise matchings in a weighted graph, widely used in soft matching problems [40]. If the SED adjacency matrix is irreducible then the probabilistic matching is unique [31]. To increase estimation accuracy when Ω' is large, we can add an exponential amplification factor:

$$P[X^{(i)}_{\Delta t_2} > x] = \frac{1}{C_\alpha} \sum_{\Psi^{(j)} \in \Omega' \setminus \{\Psi^{(i)}\}} w_{\Delta t_1}(j \rhd i)^\alpha P[X^{(j)}_{\Delta t_2} > x],$$

where $C_\alpha = \sum_{j \in \Omega'} w_{\Delta t_1}(j \rhd i)^\alpha$ and α is chosen as to minimize a regret function over the estimated distribution $P[X^{(i)}_{\Delta t'_2} > x]$ for $\Delta t'_2 \approx \Delta t_1$. Note that even for historical cascades $\Psi^{(j)} \in \Omega' \setminus \{\Psi^{(i)}\}$, we do not have the true function $P[X^{(j)}_{\Delta t_2} > x]$ that is used inside the sum. Therefore, we estimate $P[X^{(b)}_{\Delta t_2} > x]$ by bootstrapping the observations of $X^{(j)}_{\Delta t_2}$ in the historical cascades. In Section 5.2 we show that the estimates have good accuracy in practice.

4.2 A Big Data Forecast Paradox

In what follows we present a seemly simple forecasting problem and prove that under conditions common to large social networks, no model or procedure is capable of extrapolating accurate unbiased cascade statistics beyond the time horizons of the historical data. This inability of models to extrapolate cascade statistics beyond what is already in the historical (training) data indicate that model-free forecasts can do as well as forecasts with perfect models.

Moreover, we also show the following paradox: As a the size of a power law network scales and increases both the historical data and the maximum cascade sizes, forecasts beyond the historical data horizon get more inaccurate while forecasts within the horizon get more accurate. This paradox poses a great challenge for big data analytics, where short-term forecasts can get increasingly better as the system and the historical data grow while long-term forecasts get worse.

This paradox happens because the present is a biased view of the future. More precisely, in the historical data we are more likely to see at least one infection from a cascade that will be larger in the future than a cascade that will be smaller. This bias is related to the inspection paradox [36] and depends on the observation window. The bias is so hard to correct for power law distributions that it denies us the ability to forecast beyond historical data horizons, where we have not yet recorded the bias. Moreover, our results are not confined to power laws; we consider all distributions (Type I, II, and III). The results apply to statistics such as the unbiased average of cascade sizes and the cascade size distribution.

Model: We collect data during time interval $[0, T]$ and seek to forecast statistics for interval $[0, cT]$, $c > 1$. Consider n independent cascades. Cascade infections arrive according to a constant-rate Poisson process with possibly distinct rates in the interval $[0, cT]$. The size of a cascade at time cT has distribution $\Lambda(cT) \sim \theta$, where θ is a distribution with support $\{1, \ldots, W\}$. Our initial goal is to estimate an unbiased average of the cascade sizes over the interval $[0, cT]$, $c > 1$. At first inspection the forecasting problem looks like a simple task and the impossibility results stated above seem surprising. We choose the Poisson process in our illustration precisely because its simplicity allows us to clearly understand why the forecasting problem is hard.

Our use of the Poisson process is also of interest because it connects the axiomatic framework with the classic model-based one. Poisson processes are arguably among the simplest and most widely used stochastic processes. Let N_t be the number of events in the time interval $[0, t]$. A Poisson process $\{\Lambda(\Delta t)\}$ with constant rate β is defined as an arrival process that satisfies the following three axioms:

A1. $\lim_{\epsilon \to 0} P[\Lambda(\Delta t) - \Lambda(\Delta t + \epsilon) = 1] = \beta\epsilon + o(\epsilon)$, and $\lim_{\Delta t \to 0} P[N_t - N_{t+\Delta t} > 1] = o(\Delta t)$; that is, no two events can happen at the same time;

A2. for all t, $s > 0$, $N_{t+s} - N_t$ is independent of the history up to t, $\{N_u, u \leq t\}$;

A3. for all t, $s > 0$, $N_{t+s} - N_t$ is independent of t.

An equivalent constructive way to define the same process is: $P[N_t \geq k] = P[\sum_{i=1}^{k} X_i < t]$, where $\{X_1, \ldots, X_k\}$ are independent and identically distributed (i.i.d.) exponential random variables with parameter β.

Forecast Problem Definition. Let $\mathcal{O}_T$ denote the set of indices of the cascades in the historical data observed during interval $[0, T]$. The forecasting problem is defined as follows: We observe the cascades during time window $[0, T]$ and wish to forecast the average number of events we will observe in the window $[0, cT]$, $m_{cT} = \sum_{i=1}^{n} N^{(i)}_{cT}/n$.

4.2.1 Forecast Accuracy Bounds

In the following theorem, we prove that forecasting the average number of events in our mixture Poisson process is a hard problem.

THEOREM 4.2. *Let $\{N^{(i)}_{cT}\}_{i=1}^{n}$, $N^{(i)}_{cT} \sim (\theta_1, \ldots, \theta_W)$, be a set of observed events of a mixture of n Poisson processes during time interval $[0, cT]$. Let $|\mathcal{O}_T|$ denote the number of processes with at least one arrival in the interval $[0, T]$. Assume we are given the best unbiased forecast function $m_{cT|T}$ of the average number of events m_{cT}, where $m_{cT|T}$ takes as input the observations over time interval $[0, T]$. Then, the following conditions regarding the mean squared error $MSE(m_{cT|T}) = E[(m_{cT|T} - m_{cT})^2]$ hold:*

1. *If θ_W decreases faster than exponentially in W, i.e., $-\log \theta_W = \omega(W)$, then $MSE(m_{cT|T}) = \Omega(1/|\mathcal{O}_T|)$.*

2. *If θ_W decreases exponentially in W, i.e., $\log \theta_W = W \log b + o(W)$ for some $0 < b < 1$, then*

   (a) *$\log[MSE(m_{cT|T})] = \Omega(W - \log|\mathcal{O}_T|)$, provided $c > 1 + 1/b$,*

(b) $MSE(m_{cT|T}) = \Omega(W/|\mathcal{O}_T|)$, *provided* $c = 1 + 1/b$,
(c) $MSE(m_{cT|T}) = \Omega(1/|\mathcal{O}_T|)$, *provided* $c < 1 + 1/b$.

3. *If* θ_W *decreases more slowly than exponential, i.e.,* $-\log \theta_W = o(W)$, *then*
   (a) $\log[MSE(m_{cT|T})] = \Omega(W - \log|\mathcal{O}_T|)$, *provided* $c > 2$,
   (b) $MSE(m_{cT|T}) = \omega(1/|\mathcal{O}_T|)$, *provided* $c = 2$ *and* $\sum_{j=1}^{W} j^2\theta_j = \omega(1)$,
   (c) $MSE(m_{cT|T}) = \Omega(1/|\mathcal{O}_T|)$, *provided either* $c < 2$ *or* $c = 2$ *and* $\sum_{j=1}^{W} j^2\theta_j = O(1)$.

Proof of Theorem 4.2. Our proof of Theorem 4.2 shows that the forecasting mixtures of Poisson processes can be mapped into the set size distribution estimation problem; we then use our results on the hardness of estimating set size distributions from sampling [23] to prove the theorem. We start by showing how to map the forecasting mixtures of Poisson processes into the set size distribution estimation problem. Let $P[N_T^{(i)} = k|N_{cT}^{(i)}]$ be the probability that k events happen in the time window $[0, T]$ given the number of events at time cT is $N_{cT}^{(i)}$, $c > 1$. The distribution of $P[N_T^{(i)} = k|N_{cT}^{(i)}]$ is given in the following lemma.

LEMMA 4.1. *If* N_{cT} *is the number of events of a Poisson process at time* cT *and* N'_T *is the number of events that happened between* $[0, T]$, $c > 1$, *then* $P[N'_T = k|N_{cT}] = \binom{N_{cT}}{k} c^{-k}(1 - 1/c)^{N_{cT}-k}$.

PROOF. A Poisson process is defined as an arrival process that satisfies the three axioms listed above. Axiom A3 states that $\forall t$, $s > 0$, $N_{t+s} - N_t$ is independent of t. Thus, the counts of the number of events at a time interval $[0, T]$ is independent of the number of events at time interval $(T, cT]$ (axiom A2). Our Poisson process is time-homogeneous and thus the probability that an event lands in $[0, T]$ is proportional to $T/(cT) = 1/c$ (axiom A1). As the number of events between $[0, T]$ and $(T, cT]$ are independent, we have $P[N'_T = k|N_{cT}] = \binom{N_{cT}}{k} c^{-k}(1 - 1/c)^{N_{cT}-k}$. □

In what follows, we show that the timestamps of the past events in the interval $[0, T]$ are not relevant in the forecast.

LEMMA 4.2. *The event counts in the historical data provide all the statistical information collected in the interval* $[0, T]$ *w.r.t. the counts* $\{N_{cT}^{(i)}\}_{i=1}^{n}$ *in the interval* $[0, cT]$.

PROOF. Axiom A2 states that the exact timestamps of the events in the interval $[0, T]$ gives no statistical information about future events in $(T, cT]$. □

We are now ready for theorem that connects the forecast with an estimation problem that is known to be hard.

THEOREM 4.3. *The problem of forecasting any function* $g(\{N_{cT}^{(i)}\}_{i=1}^{n})$ *of the mixture Poisson sample paths described in Section 4.2 is equivalent to the set size estimation problem [23], where elements are randomly sampled from a collection of non-overlapping sets and we seek to recover the original set size distribution from the samples.*

PROOF. We prove the theorem by mapping the forecasting problem into the set size estimation problem. Let $\{N_{cT}^{(i)}\}_{i=1}^{n}$ be the sizes of n sets, whose elements are sampled independently with probability $1/c > 0$, leading to sampled set sizes that are binomially distributed as in Lemma 4.1. Finally, Lemma 4.2 shows that these non-zero sampled set sizes are the only information available to forecast the process. □

Theorem 4.3 shows that the present is a biased view of the future, as the set size problem suffers from the *inspection paradox* [23]. In what follows we prove our main theorem.

PROOF OF THEOREM 4.2. Using Theorem 4.3 we can construct a perfect map between the forecasting problem and the set size distribution problem. Theorem 4.2 follows from using Theorem 4.3 of our previous work (Murai et al. [23]) to this mapping, where we analytically compute the inverse of the Fisher information matrix of the data, giving rise to the error bounds in the theorem (through the Cramér-Rao bound). □

Theorem 4.2 shows clear limitations of forecasting with models. In realistic scenarios, where event sizes are large and have heavier-than-exponential distributions, no algorithm can obtain accurate unbiased forecasts of the average number of events beyond time interval $[T, 2T]$ if the dataset is limited to time horizon $[0, T]$. Fortunately, SED is not impacted by this problem as it limits its forecasts to the time horizons in the historical data. These results inspire the following apparent paradox.

4.2.2 A Big Network Data Paradox

When More Data Means Worse Forecasts. Theorem 4.2 creates an apparent paradox. The forecasting can become more inaccurate as the dataset $\mathcal{O}_T$ grows. The forecasting is not just more inaccurate, the forecast simply *breaks down*; as $\mathcal{O}_T$ grows, and the data time horizon remains the same $[0, T]$, there is nearly zero statistical (Fisher) information to forecast beyond horizon $[T, 2T]$.

To showcase the impact of this paradox, consider a growing network where the distribution of the size of events (cascade sizes) is Pareto with parameter $\beta > 1$. Our results hold generally but for illustration we use a scenario where the number of seeds increases proportionally with network size. In this scenario the relationship between the maximum number of events per seed W and the number of seeds $|\mathcal{O}_T| \gg 1$ is $E[W] \approx \sqrt[\beta]{|\mathcal{O}_T|}\,\Gamma(\beta - 1)$, a known result from extreme value theory [28]. Theorem 4.2 case 3(a) shows that the MSE error is lower bounded by $\log[\text{MSE}(m_{cT|T})] = \Omega(W - \log|\mathcal{O}_T|)$. As $E[W] \propto \sqrt[\beta]{|\mathcal{O}_T|}$ and $|\mathcal{O}_T|$ increases, so does W. We can think of the MSE lower bound growing roughly as $\exp(\sqrt[\beta]{|\mathcal{O}_T|})/|\mathcal{O}_T|$.

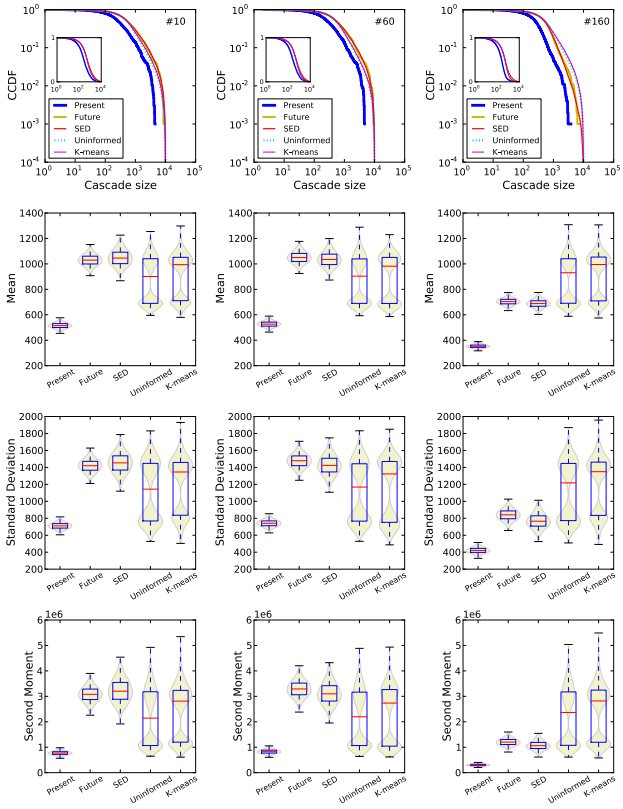
Thus, if the training data $\mathcal{O}_T$ has time horizon $[0, T]$ and event (cascade) sizes have a power law distribution, Theorem 4.2 cases 3(b-c) show that as the network grows the estimate in the interval $[T, 2T]$ gets more accurate. However, Theorem 4.2 case 3(a) shows that as the network grows the estimation error in the interval $(2T, cT]$, $c > 2$, grows exponentially with network size.

5. RESULTS

In this section we present our results. We start with our simulation results in Section 5.1. Section 5.2 shows our results over a large Twitter dataset.

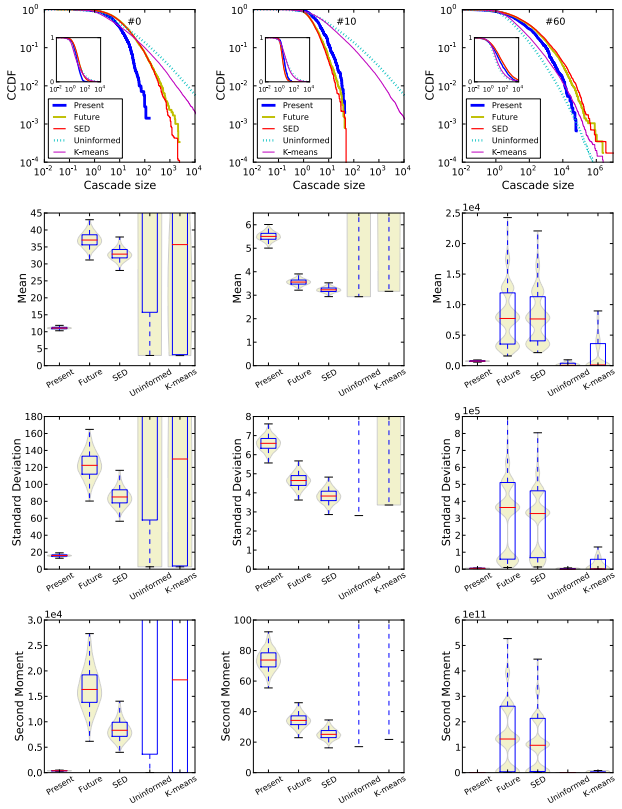
5.1 Forecasting Using Simulated Data

In this section, we test the forecasts of SED on synthetic datasets. We test the performance of SED over two distinct models: the Poisson process and the Galton-Watson process. Recall that $\Omega' = \Omega \cup \{\Psi^{(0)}\}$ is the union of the historical processes Ω and the process to forecast $\Psi^{(0)}$. SED



(a) ID 10 (b) ID 60 (c) ID 160

Figure 1: Prediction for mixture Poisson processes (Scenario 1) of the future CCDF, average, standard deviation, and second moment metrics. SED predictions match well the true future distributions and clearly outperform uninformed predictions.



(a) ID 0 (b) ID 10 (c) ID 60

Figure 2: Prediction for Galton-Watson processes (Scenario 2) of the future CCDF, average, standard deviation, and second moment metrics. SED predictions again match well the future distributions and are clearly superior to uninformed predictions.

forecasts metrics of $\Psi^{(0)}$ at time interval Δt_2 using data from time interval Δt_1, $\Delta t_1 < \Delta t_2$ In addition to SED, we also have two baseline naive predictors: *uniformed prediction*, and *K-means prediction*. Uninformed prediction returns the bootstrapped statistics at time interval Δt_2 of a randomly selected process in Ω. Whereas, K-means prediction first clusters the set of historical cascades using K-means algorithm and Euclidean distances between the CCDFs of cascade metrics (probability density functions, PDFs, give similar results). The number of cluster k is chosen using the elbow method, without making the clusters too small (k equals 10 for the two synthetic datasets, and 20 for the Twitter dataset). After that, given a new cascade $\Psi^{(0)}$, K-means prediction finds the cluster it belongs to, and returns the bootstrapped statistics at time interval Δt_2 of a randomly selected process in that cluster.

For evaluation, we use violin plots to compare the predicted statistics with the bootstrap statistics from the ground-truth empirical distribution of $\Psi^{(0)}$ at Δt_2. A violin plot is a box plot combined with a kernel density estimate of the probability density function, giving a more detailed view of the data's variance. The box extends from the first quartile Q_1 to the third quartile Q_3 of the data, with a line at the median. The whiskers extend from the box to $Q_1 - 1.5 * \text{IQR}$ and $Q_3 + 1.5 * \text{IQR}$, where $\text{IQR} = Q_3 - Q_1$ is the inter-quartile range. The violin plots show the spread.

5.1.1 *Scenario 1: Mixture Poisson Processes*

Our mixture Poisson process is a Poisson process with random rate β that has support $(0, W]$, $W > 0$.

Data Generation: The simulated historical data $\mathcal{H}$ is generated through simulations from time zero until $\Delta t_2 = 1$ month from two sets of mixture Poisson processes. The first set has $n_1 = 100$ processes with rate distribution $\gamma_1 \propto \beta^{-2}$ and the second set has $n_2 = 100$ processes with rate distribution $\gamma_2 \propto \beta^{-3}$. Poisson rates are measured in number of events per month. Each process has $1,000$ infection seeds. The process to forecast $\Psi^{(0)}$ belongs to one of two sets and was observed from time zero until $\Delta t_1 = \Delta t_2/4$, approximately one week. In what follows we test how well SED can forecast $\Psi^{(0)}$. In Section 4.2 we show that estimating the Poisson rates is hard due to observation bias.

Results: For conciseness, Figure 1 shows only the results for three processes, one per column: processes 10, 60 (type 1), and 160 (type 2). The remaining results can be found in our technical report [29]. Four forecasts are evaluated: forecasting the CCDFs, mean, standard deviation, and second moment. For comparison baselines we use uninformed prediction – uniform matching weights – and matching using K-means with Euclidean distances, as the latter have been extensively and successfully used in mining and classifying time series [37]. For each process we present the true statistics at Δt_1 (Present) and Δt_2 (Future), and the predicted statistics at Δt_2, including SED, uninformed, and

K-means predictions. The Figure 1 shows that SED is the method that better represents, consistently, the true statistics at time Δt_2 under all scenarios (Future).

5.1.2 Scenario 2: Galton-Watson Process

The Galton-Watson process is a branching process $\{X_n\}$ such that $X_{n+1} = \sum_{j=1}^{X_n} \psi_j^{(n)}$, where $X_0 = 1$ and $\{\psi_j^{(n)} \sim Poisson(\lambda) : n, j \in \mathbb{N}\}$ is a set of i.i.d. natural number-valued random variables [22]. Intuitively, $\psi_i^{(n)}$ is the number of male children of the j-th descendants, and X_n is the number of descendants in the n-th generation. The total number of people with the considered family name at time t would be $\sum_{i=0}^{t} X_t$. A simple epidemic cascading on a network can be modeled as a Galton-Watson process.

To make this model harder to forecast, we replace the Poisson distribution of the number of children by a log-normal distribution. The purpose of the log-normal distribution is to skew the obtained distribution of the number of children, mimicking real life situation where the degree of a node in a network follows a heavy tail distribution. We call the values generated by the log-normal distribution the descendant effect of a node. Here is the model we use $X_{n+1} = \sum_{j=1}^{\lfloor X_n \rfloor} \psi_j^{(n)}$, where $X_0 = 1$ and $\{\psi_j^{(n)} \sim \text{log-normal}(\mu, \sigma^2) : n, j \in \mathbb{N}\}$. The operation $\lfloor X_n \rfloor$ is to get an integer-valued number of descendants.

Cascade Process Evolution: For each information cascade, we use a birth rate γ to generate new seeds over time. Given a single seed node ($X_0 = 1$), the total descendant effect up to time t in a Galton-Watson process, i.e., $\sum_{i=0}^{t} X_i$, will be used as the cascade size at time t for this seed. As a result, we obtain a sample of the cascade size per seed. To add more complexity to the model, we allow the parameters to change over time: $\mu_t = \mu_0 + t * \Delta_\mu$, $\sigma_t = \sigma_0 + t * \Delta_\sigma$, $\gamma_t = \gamma_0 + t * \Delta_\gamma$, where μ_t, σ_t, γ_t are the parameter values at time point t, and Δ_μ, Δ_σ, Δ_γ are the amount of change at each time step.

Table 1 shows our parameter settings. We create 9 different cascade types, each one containing 10 cascades spanning a period of seven time units. These cascade types can be grouped into three big groups – small, medium and large – based on their mean μ in the log-normal distribution. Larger cascades have a larger variation (larger σ) and higher birth rate γ. For each of these big groups, we generate three subgroups with different evolutionary trends: increasing in size (inc), decreasing in size (dec), and constant size (const). Finally, we set $\Delta t_1 = 2$ time units and $\Delta t_2 = 7$ time units.

Results: Figure 2 shows the results of three processes: 0 (Small-inc), 10 (Small-dec) and 60 (Large-inc). We get similar results in the other processes. Note that the statistics of Present and Future are significantly different. Figure 2 shows that the uninformed and k-means forecasts are far from the truth that the confidence intervals do not fit in our plot. Over the same scenario SED accurately forecasts the mean, standard deviation and second moment as shown in the last three rows of Figure 2.

5.2 Forecasting Twitter Cascades

We study the Twitter dataset collected by Yang et al. [39][2], which contains 467 million Twitter posts from 20 million users covering a 7-month period from June 1st to December 31st, 2009. The underlying Twitter follower-followee network is obtained from Kwak et al. [18]. The fact that this Twitter dataset is only a sample of all tweets does not invalidate the dataset to test SED. SED is designed to make predictions over general dissemination processes, including sampled cascading processes such as our Twitter dataset.

[2] `http://snap.stanford.edu/data/twitter7.html`

		Parameters			Temporal delta		
Process Type	IDs	μ_0	σ_0	γ_0	Δ_μ	Δ_σ	Δ_γ
Small-inc	0..9				+0.2	+0.1	+30
Small-dec	10..19	1.5	1	300	-0.2	-0.1	-30
Small-const	20..29				0	0	0
Medium-inc	30..39				+0.2	+0.1	+30
Medium-dec	40..49	2	2	500	-0.2	-0.1	-30
Medium-const	50..59				0	0	0
Large-inc	60..69				+0.2	+0.1	+30
Large-dec	70..79	3	2.5	700	-0.2	-0.1	-30
Large-const	80..89				0	0	0

Table 1: Galton-Watson process simulation parameters.

Hashtag cascades

We are interested in the diffusion of a hashtag over the Twitter online social network. When an individual uses a hashtag in her tweet at a given time, if neither she nor any of her followees have tweeted the same hashtag for at least a week prior to the post timestamp, we consider her as a seed. The timestamp of the first tweet is the timestamp of a node hashtag "infection". Once the seed of a cascade has been identified, we start tracking the cascade over time. Note that our definition of seed is stringent and we may merge almost unrelated cascades into a single cascade. From our RMPP definition, metrics over merged cascade processes are allowed and, thus, we can be conservative to define independent cascades in our dataset.

If a hashtag was not been used for more than a week, we assume the cascade has ended. The first occurrence of the same hashtag after that would mark the beginning of a new cascade. Due to this one-week time window, hashtags appearing for the first time in the dataset during the first and last weeks of the monitored time frame are ignored. The time difference between the first and the last posts of an information cascade is defined as the duration of the cascade.

Cascade metric

In our case study, we are interested in the distribution of the number of infected descendants of a seed node for each hashtag in Twitter. In particular, given an infected seed node, we quantify how many of its descendants are "infected" by a hashtag complying with temporal infection order (infected children in the branching process must be infected after their parents). These quantities are obtained from the induced subgraph of the infected nodes over the original follower-followee network. Note that a user may have several followees who posted the same hashtag before she did. As a result, simply counting the number of infected descendants of an infected user does not guarantee the independence between samples required by our theory. To reduce the effect of these independence violations, we assume that two infected parents of an infected node are evenly responsible for the infection. Under this assumption we propose a normalized metric as follows.

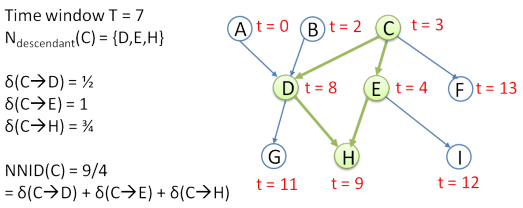


Figure 3: An example induced subgraphs of infected nodes and cascade metric. Node infection timestamps are shown in red.

DEFINITION 5.1. *Given a time window length T, the number of normalized infected descendants (NNID) of a node v are defined as:*

$$NNID(v) = \sum_{w \in N_{des}^{T}(v)} \delta(v \to w)$$

where $N_{in}^{T}(v)$ and $N_{des}^{T}(v)$ are the set of infected parents, and infected descendants of node v within the time window T respectively; $\delta(v \to v) = 1$; $\delta(v \to w)$ is the amount of infection spread from node v to w through all possible time-ordering paths, and is defined recursively as follows:

$$\delta(v \to w) = \frac{\sum_{u \in N_{in}^{T}(w) \cap N_{des}^{T}(v)} \delta(v \to u)}{|N_{in}^{T}(w)|}$$

We assume that if a node v is infected at time point t, then after time point $t+T$, its effect on other nodes will diminish to zero. Thus, the set of infected descendants of node v do not contains nodes infected after time $t+T$. To get a distribution of NNIDs for a cascade c at a given time point t, we first extract the induced subgraph $G_{ind} = (V_{ind}, E_{ind})$, where V_{ind} is the set of infected nodes of c up to time t. In addition, given two infected nodes u and v, if the edge $(u \to v) \in E$ and $t_{infect}(v) - t_{infect}(v) \leq T$, then $(u \to v) \in E_{ind}$, where $t_{infect}(v)$ is the timestamp at which v is infected in c. Next, for each seed node of the cascade, i.e., nodes with no infected parents within a time window T, we compute its NNID in G_{ind} using Definition. 5.1. In the end, we obtain a sample of NNID values, each for a seed in c, giving us an empirical distribution of NNIDs of c.

Example 1. *An induced subgraph of infected nodes is shown in Fig. 3. Given a time window $T = 7$, the descendant set of C is $\{D, E, H\}$; the infected parent sets of D and E are $\{B, C\}$ and $\{C\}$ respectively. Thus, $\delta(C \to D) = \frac{1}{2}$, and $\delta(C \to E) = 1$. The infection from C can spread to H following two different paths: $C \to D \to H$, and $C \to E \to H$. Thus, $\delta(C \to H) = \frac{\delta(C \to D) + \delta(C \to E)}{2} = \frac{3}{4}$. Finally, the number of normalized descendants of C is $NNID(C) = \delta(C \to D) + \delta(C \to E) + \delta(C \to H) = \frac{1}{2} + 1 + \frac{3}{4} = \frac{9}{4}$.*

Similarly, the set of seed nodes with no infected parents within a time window $T = 7$ is $\{A, B, C, F, I\}$, which gives a sample of NNIDs $\{1, \frac{3}{4}, \frac{9}{4}, 1, 1\}$ accordingly.

Twitter Results

For our tests, we identify the Twitter hashtag cascades that last at least 8 weeks. Using this dataset, we predict distribution of NNID (time window $T = 1$ week) at week 8 (Δt_2) using data from the week 2 (Δt_1). To guarantee that enough data is available for prediction, we only consider cascades that contain at least 300 infected users after 8 weeks and at least 30 infected users after one week. In the end, we are left with 1050 cascades split among the different hashtags. Similar to the simulations, besides the CCDF plots, we use violin plots to compare the predicted statistics (SED prediction and uninformed prediction) with the bootstrap statistics from the ground-truth sample at Δt_2. It is worth noting that the uninformed prediction simply returns the average statistics of all cascades in Ω at time Δt_2.

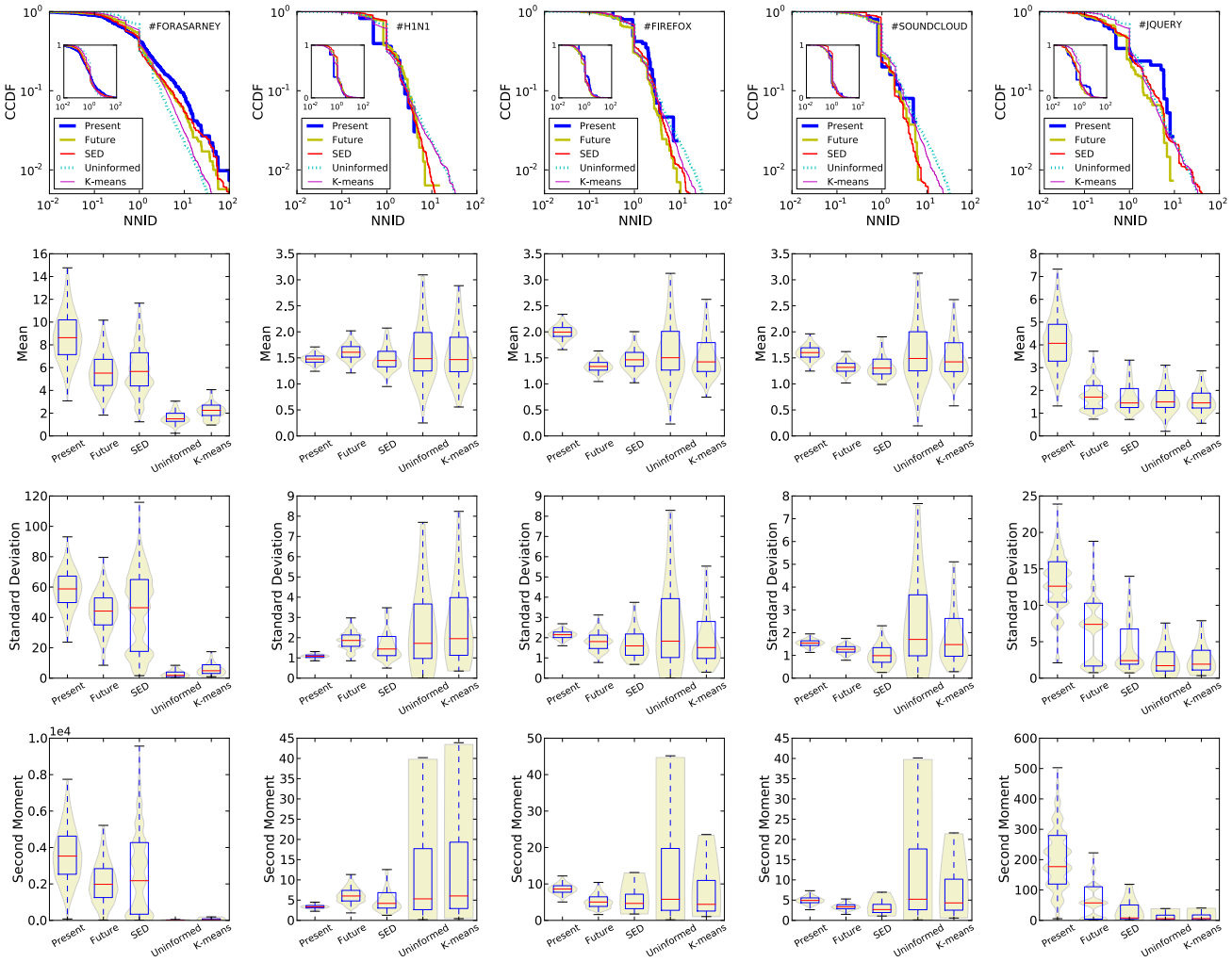
The forecasting results for five Twitter hashtags are shown in Figure 4. Our technical report shows all other hashtag forecasts left out of this paper due to space constraints [29]. SED forecasts frequently outperform uninformed and k-means predictions. For example, all #FORASARNEY statistics differ considerably from those of uninformed and k-means predictions. Whereas SED accurately forecasts the average CCDFs and all three statistics, as observed by contrasting SED forecast violin plots against those of the ground truth (Future). For #H1N1, #FIERFOX and #SOUNDCLOUD, the uninformed forecasts are rather uncertain of the values of the mean, the standard deviation, and the second moment, leading to significantly lengthened violin plots. On the other hand, the SED forecasts are, once again, close to the ground truth. In the case of forecasting the mean of #JQUERY (last column, second row), the true mean (Future) is so close to the future average statistics (uninformed prediction), that SED shows no clear improvement over naive approaches. We include this #JQUERY example to show that in hard-to-forecast cases SED does no worse than uninformed and k-means. Note, however, that for standard deviation and the second moment of #JQUERY show SED superior to uninformed and k-means.

It must be noted that the hashtag Twitter dataset that we are using is noisy (cascades are not fully independent), making our forecasting task very difficult. Moreover, for many of the cascades in this dataset, the distribution at week 8 (Δt_2) is very close to the average prediction (uninformed prediction). Thus, more challenging datasets and further research are needed to further validate practical aspects of our approach.

6. RELATED WORK

Analyzing and predicting network processes in general, and information cascades in particular, has attracted much attention in recent years. Multiple works focus on building theoretical models (e.g., [8, 9, 38]) of dissemination processes. These models capture information diffusion at nodes through network topology as well as user interests and the information content [7, 27, 38]. Wang et al. [35] uses partial differential equations to predict information diffusion over both temporal and spatial dimensions. One challenge in this line of research is the gap between empirical data and theory. Building parsimonious models is a complex task and the resulting models are often not vetted against real data.

Most relevant to this paper are the works that predict cascade popularity. Some of these works predict the volume of aggregate activity, such as the number of votes on Digg stories [33], or Twitter hashtag usage [20]. A few works examine how information cascades grow in size depending on its content [19, 30] and user interests [2]. Other works make prediction using observations of a cascade during a given fixed time interval [17, 20, 34]. The cascade size prediction task is seen as a regression problem in a variety of works [3, 17, 33, 34]. Matsubara et al. [21] proposes a parsimonious model that captures the rise and fall patterns in information diffusion. In lieu of predicting the exact cascade sizes, many studies bin the cascades sizes and solve a binary



(a) FORASARNEY (b) H1N1 (c) FIREFOX (d) SOUNDCLOUD (e) JQUERY

Figure 4: Predictions of statistics of NNID for five example hashtags in Twitter.

classification problem of whether or not a cascade becomes viral [11, 13, 17] or if it doubles in size [4]. Instead of modeling the diffusion process, Najar et al. [25] directly predicts the final propagation state of the information given its initial state. Outside cascade sizes, Cheng et al. [4] also predicts structural features of the cascade sample paths. Our goal is different as we do not perform binary classification of the evolution of a sample path, or just predict cascade sizes. We propose a general axiomatic forecasting framework that is not tied to a specific set of cascade features or process and can be used off-the-shelf in any similar forecasting scenario.

Since we predict cascade statistics, our work also relates to research on fitting empirical data to parsimonious statistical models [1, 5]. While the empirical data can be readily fitted to many known parsimonious models such as power laws, log-normal, or exponential, there is no guarantee that the fitted model can be used to predict the tail of the distribution or how the distribution changes with the observation window. Indeed, our experimental results using Twitter data show that cascade statistics can significantly change over time. Thus, the need to go beyond parsimonious models to design a data-driven statistical approach using axiomatic forecasting.

7. CONCLUSIONS

In this work we propose SED, a new algorithm for forecasting statistics of complex networked processes. SED is the first of its kind, a (true) model-free approach that uses axioms rather than models to extract statistical information from the data, pointing to a promising new direction of axiomatic forecasting. More importantly, we provide the underlying theory behind SED's model-free axiomatic forecasting approach. We test SED on two synthetic datasets and a large Twitter dataset, showing that SED can forecast accurately a variety of statistics under complex scenarios.

Acknowledgements

We thank C. Faloutsos and D. Towsley for their helpful comments. This work was partially supported by NSF grants CNS-1065133, IIS-1219254, and ARL Cooperative Agreement W911NF-09-2-0053. The views and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied of the NSF, ARL, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

8. REFERENCES

[1] Lada A Adamic. Zipf, power-laws, and pareto-a ranking tutorial. *Xerox Palo Alto Research Center*, 2000.

[2] Lars Backstrom, Dan Huttenlocher, Jon Kleinberg, and Xiangyang Lan. Group formation in large social networks: membership, growth, and evolution. In *Proc. SIGKDD*, 2006.

[3] Eytan Bakshy, Brian Karrer, and Lada A Adamic. Social influence and the diffusion of user-created content. In *EC*, 2009.

[4] Justin Cheng, Lada Adamic, P Alex Dow, Jon Michael Kleinberg, and Jure Leskovec. Can cascades be predicted? In *Proc. WWW*, 2014.

[5] Aaron Clauset, Cosma Rohilla Shalizi, and Mark EJ Newman. Power-law distributions in empirical data. *SIAM review*, 51, 2009.

[6] R. A. Fisher. *The design of experiments*. Oliver & Boyd, 1935.

[7] W Galuba, K Aberer, and D Chakraborty. Outtweeting the twitterers-predicting information cascades in microblogs. In *Proc. WOSN*, 2010.

[8] Michaela Goetz, Jure Leskovec, Mary McGlohon, and Christos Faloutsos. Modeling blog dynamics. In *ICWSM*, 2009.

[9] Amit Goyal, Francesco Bonchi, and Laks VS Lakshmanan. Learning influence probabilities in social networks. In *Proc. WSDM*, 2010.

[10] Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *JMLR*, 13(1):723–773, March 2012.

[11] Liangjie Hong, Ovidiu Dan, and Brian D Davison. Predicting popular messages in twitter. In *Proc. WWW*, 2011.

[12] Ting-Kai Huang, Bruno Ribeiro, Harsha V Madhyastha, and Michalis Faloutsos. The socio-monetary incentives of online social network malware campaigns. In *Proc. COSN*, pages 259–270. ACM, 2014.

[13] Maximilian Jenders, Gjergji Kasneci, and Felix Naumann. Analyzing and predicting viral tweets. In *Proc. WWW*, 2013.

[14] Eamonn Keogh, Stefano Lonardi, and CA Ratanamahatana. Towards parameter-free data mining. *Proc. SIGKDD*, pages 206–215, 2004.

[15] A. N. Kolmogorov. *Foundations of the Theory of Probability*. Chelsea Publishing Co., 1950.

[16] N H Kuiper. *Tests concerning random points on a circle*. Proceedings of the Koninklijke Nederlandse Akademie, 1960.

[17] Andrey Kupavskii, Liudmila Ostroumova, Alexey Umnov, Svyatoslav Usachev, Pavel Serdyukov, Gleb Gusev, and Andrey Kustarev. Prediction of retweet cascade size over time. In *CIKM*, 2012.

[18] Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon. What is Twitter, a social network or a news media? In *Proc. WWW*, 2010.

[19] Jure Leskovec, Lars Backstrom, and Jon Kleinberg. Meme-tracking and the dynamics of the news cycle. In *Proc. SIGKDD*, 2009.

[20] Zongyang Ma, Aixin Sun, and Gao Cong. On predicting the popularity of newly emerging hashtags in twitter. *JASIST*, 64, 2013.

[21] Yasuko Matsubara, Yasushi Sakurai, B Aditya Prakash, Lei Li, and Christos Faloutsos. Rise and fall patterns of information diffusion: model and implications. In *Proc. SIGKDD*, pages 6–14, 2012.

[22] Fabricio Murai, Bruno Ribeiro, Don Towsley, and Krista Gile. Characterizing Branching Processes from Sampled Data. In *Proc. WWW Companion*, pages 805–811, 2013.

[23] Fabricio Murai, Bruno Ribeiro, Don Towsley, and Pinghui Wang. On Set Size Distribution Estimation and the Characterization of Large Networks via Sampling. *IEEE Journal on Selected Areas in Communications*, 31(6):1017–1025, 2013.

[24] Seth A Myers, Chenguang Zhu, and Jure Leskovec. Information diffusion and external influence in networks. In *Proc. SIGKDD*, 2012.

[25] Anis Najar, Ludovic Denoyer, and Patrick Gallinari. Predicting information diffusion on social networks with partial knowledge. In *Proc. WWW Companion*, pages 1197–1204. ACM, 2012.

[26] Charles T Perretti, Stephan B Munch, and George Sugihara. Model-free forecasting outperforms the correct mechanistic model for simulated and experimental data. *PNAS*, 110(13):5253–7, March 2013.

[27] Daniel Ramage, Susan T Dumais, and Daniel J Liebling. Characterizing microblogs with topic models. *ICWSM*, 10, 2010.

[28] Sidney I Resnick. *Extreme values, regular variation, and point processes*. Springer Science & Business Media, 2007.

[29] Bruno Ribeiro, Minh X. Hoang, and Ambuj K. Singh. Beyond models: Forecasting complex network processes directly from data. `http://www.cs.cmu.edu/~ribeiro/pdf/Ribeiro_etal_BeyondTR15.pdf`.

[30] Daniel M Romero, Chenhao Tan, and Johan Ugander. On the interplay between social and topical structure. In *ICWSM*, 2013.

[31] Richard Sinkhorn and Paul Knopp. Concerning nonnegative matrices and doubly stochastic matrices. *Pacific Journal of Mathematics*, 21(2):343–348, 1967.

[32] M A Stephens. Use of the Kolmogorov-Smirnov, Cramer–Von-Mises and related statistics without extensive tables. *J. of the Royal Statistical Society Series B*, 32, 1970.

[33] Gabor Szabo and Bernardo A Huberman. Predicting the popularity of online content. *Communications of the ACM*, 53, 2010.

[34] Oren Tsur and Ari Rappoport. What's in a hashtag?: content based prediction of the spread of ideas in microblogging communities. In *Proc. WSDM*, 2012.

[35] Feng Wang, Haiyan Wang, and Kuai Xu. Diffusive Logistic Model Towards Predicting Information Diffusion in Online Social Networks. *WINE*, cs.SI(June), August 2011.

[36] James R. Wilson. The inspection paradox in renewal-reward processes. *Operations Research Letters*, 2(1):27–30, April 1983.

[37] Zhengzheng Xing, Jian Pei, and Eamonn Keogh. A brief survey on sequence classification. *ACM SIGKDD Explorations Newsletter*, 12(1):40, November 2010.

[38] Jaewon Yang and Jure Leskovec. Modeling information diffusion in implicit networks. In *ICDM*, 2010.

[39] Jaewon Yang and Jure Leskovec. Patterns of temporal variation in online media. In *Proc. WSDM*, 2011.

[40] Ron Zass and Amnon Shashua. Probabilistic graph and hypergraph matching. In *CVPR*, pages 1–8. IEEE, 2008.