Optimal Timelines for Network Processes

Daniel J. DiTursi*†
dditursi@siena.edu

Carolyn S. Kaminski*
ckaminski@albany.edu

Petko Bogdanov*
pbogdanov@albany.edu

***Abstract*—Structural models for network dynamics typically assume a discrete timeline of network events (node activation or link creation) and a stochastic generative process giving rise to new events based on the event history and the network structure. In order to employ these models for prediction, observational data is often aggregated at a fixed temporal resolution (e.g., minutes or days). However, the underlying network processes may "speed up" or "slow down" at different points in time, rendering observations unlikely and predictions incorrect. The challenge is to optimize the timescale for the analysis of network event data, which in turn is based on structural models of the underlying network processes.**

We introduce the general problem of inferring the optimal temporal resolution for network event data. The goal is to map observed network events to discrete time steps by aggregation and/or disaggregation of their original timeline such that they are collectively well-explained by structural dynamics models. We unify network growth and information diffusion models and differentiate between short- and long-memory processes. We demonstrate that while optimal temporal aggregation can be performed in polynomial time, disaggregation—and thus, the general timescale inference problem—is NP-hard. We propose scalable heuristics for the problem, some with approximation guarantees, and employ them for missing event recovery and temporal link prediction, demonstrating significant improvements (absolute increase of 10% in F_1 measure for event recovery and of 5% in AUC for link prediction) compared to employing the same algorithms on the default timescale of data collection.

***Index Terms*—Temporal networks, Network clock, Timeline reconstruction, Structural network models**

I. Introduction

Abundant socio-behavioral event data is continuously generated online: posts in social media, creation of follower and friendship links, reactions to posts in the form of likes and comments, and more. Individual events are not independent of each other. For example, friends-of-friends are more likely to become friends than arbitrary pairs [1] [2], and a user is more likely to post a meme if their friends have recently posted it [3] [4]. A number of structural models have been proposed for such dependencies, inspired both by empirical evidence and longstanding theory from sociology and economics [5] [6]. Such models and data have been employed to study and improve product marketing [7], increase participation in the political process [8], improve social recommendations [9], study the financial markets [10], and detect insurgent networks [11], among others.

A key challenge in employing such models is the need to select a discrete timescale, a common assumption in many of the models [10]. A typical approach is to define a fixed window (e.g. minutes, hours, days) and aggregate events into temporal snapshots before employing structural dynamics models. It has been realized, however, that in-network process rates may vary across time [12] [13] [14], thus limiting the accuracy of trained models [15] [16] [17]. *How can we infer an optimal temporal resolution for network event data that accounts for the varying rate of the underlying processes?*

* State University of New York at Albany, Albany, N.Y., U.S.A.
† Siena College, Loudonville, N.Y., U.S.A.
This work is in part supported by NSF Smart and Connected Communities (SC&C) grant CMMI-1831547.

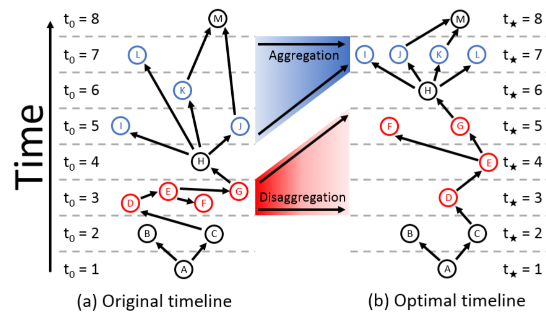

Fig. 1. (a) An information cascade. (b) The same cascade mapped onto a new timeline, producing a higher likelihood according to the Independent Cascade model [18]. Original timestep 3 has been disaggregated into three timesteps, while timesteps 5, 6, 7 have been aggregated into a single timestep.

An example information cascade is presented in Fig. 1. Under the Independent Cascade (IC) model [18], a node activated at time t has a chance to activate neighbors at time $t+1$ but not later. The original timeline (Fig. 1(a)) includes some periods that have low likelihood under IC. At $t_0 = 3$ there are four activations which would have higher likelihood if occurring in consecutive time steps. Meanwhile, the activations at $t_0 = \{5, 6, 7\}$ could all be explained as a consequence of the activation of node H at $t_0 = 4$ *if* they were simultaneous. Intuitively, the cascade "sped up" during the third time period and "slowed down" during $t_0 = \{5, 6, 7\}$. Such effects could be observed at different time scales in real-world data due to cooperation or competition of interacting cascades [19] or due to committed nodes in opinion dynamics [20]. An optimal timeline (Fig. 1(b)) should (i) increase the resolution at $t_0 = 3$ by *disaggregation*, i.e. spreading observed events over multiple time steps; and (ii) decrease the resolution at $t_0 = \{5, 6, 7\}$ by *aggregation* of all events into a single time step.

The example from Fig. 1 focuses on a diffusion process under a short-memory model (IC). Similar sub-optimal timescale challenges are also present in models for network growth (e.g. [21]), especially where acceleration and deceleration patterns have been observed [22] [23] [24]. In addition, the dependency structure among events that the model imposes could be less rigid than the IC; that is, diffusion activation or link creation may depend on events older than a single time step.

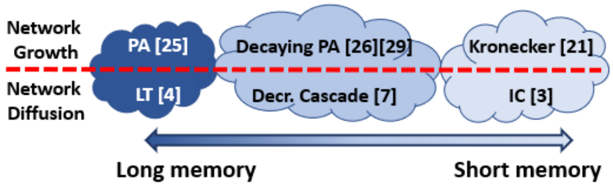

Fig. 2. Network models on the continuum of memory, i.e. the temporal range of dependencies among events. Event age has a higher impact on likelihood of new events in short-memory models compared to long-memory ones.

In this paper we formalize the problem of inferring the *optimal temporal resolution (OTR)* for network event data. We unify network growth and diffusion processes and corresponding models, considering both long- and short-memory dependencies among events. We prove that OTR is NP-hard and propose scalable heuristics for the problem, some with theoretical approximation guarantees.

Our contributions in this paper are as follows:

• **Generality:** We propose the novel problem of optimal temporal resolution for network event data, study its complexity and unify both network growth and diffusion processes within the same optimization framework.

• **Accuracy:** Our solutions for OTR exhibit good accuracy in reconstructing the order of events both in terms of likelihood maximization and comparison to ground truth.

• **Applicability:** The optimal resolution learned by our algorithms enables significant quality improvements in temporal link prediction (5% **increase in AUC**) and cascade reconstruction (10% **improvement in** F_1).

II. Preliminaries: Network Process Models

In all cases, we examine a set of timestamped events $\mathbb{X}|\Delta_o$ that each have a discrete timestamp according to a timeline (or clock) $\Delta_o = [1, T]$; the events occur on a graph $G(V, E)$. Let $\mathbb{M}(\Phi, \mathbb{X}, \Delta)$ be a probabilistic model, parametrized by a set of parameters Φ (e.g., diffusion probabilities or edge creation probabilities), which quantifies the likelihood of observed events $x \in \mathbb{X}$ timestamped according to a timeline $\Delta = [1, T]$. The likelihood of observations $\mathcal{L}_{\mathbb{M}}(\mathbb{X}|\Delta)$ depends critically on the timeline Δ (see Fig. 1), and finding an optimal timeline $\Delta_\star$ for a given model is the main objective of our work.

This general setting encompasses a variety of network process models in which the likelihood of events can be expressed as a function of pasts events and their timeline. One distinctive feature of relevant models from the literature is the degree to which the *age* of prior events impacts future ones. In *long-memory* models such as Barabási and Albert's classic preferential attachment (PA) model [25], age is irrelevant; all past edge creation events contribute equally to future edges. In *short-memory* models like IC [3], an event's influence lasts only for a single timestep. In between are models such as the one introduced by Sun et al. [26] in which events' influence decays as a function of time.

Figure 2 maps some existing models on the continuum between short- and long-memory. While our work on optimal temporal resolution is applicable to all of them, we concentrate our discussion on several example models which are commonly adopted in the literature and also span two main groups of network processes: *network growth*, or the process of edge creation from which the evolving network structure arises [25], and *network diffusion*, modeling the spread of information or other contagions from node to node along the static network structure [3], [4]. We focus on demonstrating the importance and applicability of optimal timelines for several representative models, laying the foundations for a more general and exhaustive evaluation of our methods as future work. A comprehensive discussion of additional model variants is available in several detailed surveys on network process models: [2], [6], [27].

Network Diffusion Models. In this setting, $\mathbb{X}$ represents adoption/infection events. In terms of models $\mathbb{M}$, we restrict our discussion to *linear threshold (LT)* [4] and *independent cascade (IC)* [18]. Both are progressive in that once a node becomes active, it remains in this state.

LT is a long-memory diffusion model. Let A represent the set of neighbors of v activated before time t. Then the likelihood of a node v activating at time t is given by $\mathcal{L}(v, t) = p_n$ if $|A| \geq \theta_v$. Since it does not matter how long ago the neighbors were activated, LT is a long-memory model.

On the other hand, IC is a short-memory model because only the neighbors activated in the immediately preceding timestep have any impact on a node's activation. Let A_{t-1} represent the neighbors of v activated precisely at time $t-1$. The likelihood of activation under IC is then given by $\mathcal{L}(v, t) = 1 - (1 - p_n)^{|A_{t-1}|}$ if $|A_{t-1}| > 0$.

Both models include a (small) probability p_e of a *spontaneous* node activation to avoid assigning a zero probability to observed events. WLOG, we keep the parameters constant across the entire network.

Network Growth Models. Events $\mathbb{X}$ in this case are edges $e \in E$ stamped by the time of their creation, $\tau(e)$. The model $\mathbb{M}$ we consider for experiments is a generalized version of *preferential attachment (PA)* [28] which further allows a decaying contribution of an edge to adjacent nodes' effective degrees similar to the decaying-relevance from [29]. The probability of an edge forming is given by:

$$\mathcal{L}(v_1, v_2, t) = p_n \left(\frac{d_\lambda(v_1, t) d_\lambda(v_2, t)}{(d_\lambda(V, t))^2}\right)^\alpha. \quad (1)$$

$d_\lambda(v, t)$ tracks the effective (i.e. time-decayed) degree of v at time t; the parameter λ controls the decay. When $\lambda = 0$ we get standard PA (a long-memory model); large values of λ produce a short-memory model analogous to IC, where only recently-formed edges drive the creation of new ones.

III. Problem Formulation and Analysis

A. Optimal temporal resolution (OTR)

Consider again Figure 1 which depicts a cascade following the short-memory IC model. In the original timeline, the activation of node E is a low-probability spontaneous event: $\mathcal{L}(E, 3) = p_e$. The same is true of nodes F, G, K, L, and M: none have a neighbor activated in the timestep before their own activation. Thus, the overall likelihood $\mathcal{L}(\mathbb{X}|\Delta_0)$ of the cascade under IC in the original timeline is relatively low.

There are two kinds of opposite "derangements" driving this low likelihood. Activations D through G (shown in red) form an IC-explainable chain of events, but they appear simultaneous in the original timeline. To maximize their likelihood, we need to *disaggregate* this "accelerated" period of the timeline by recovering a high-likelihood ordering of the events. Conversely, activations I through L (shown in blue) could all have likely been influenced by H; however they are dispersed in time and in this case *aggregating* this "decelerated" period will improve their likelihood. Overall $\mathcal{L}(\mathbb{X}|\Delta_\star)$, the likelihood of the timeline in Fig. 1b, is much higher than $\mathcal{L}(\mathbb{X}|\Delta_0)$, since the only spontaneous activation is that of node A; $\Delta_\star$ is the optimal timeline for this cascade.

Based on the above observations, we formalize the general timeline optimization problem as follows:

Definition 1. *Optimal Temporal Resolution (OTR)*: Find the optimal timeline $\Delta_\star \in \mathbb{D}$ which maximizes the likelihood $\mathcal{L}(\mathbb{X}|\Delta_\star)$ of $\mathbb{X}$ according to model $\mathbb{M}$. $\Delta_\star$ is one of all possible timelines $\mathbb{D}$ obtained by recursively aggregating or disaggregating periods of the original timeline Δ_0.

Our analysis examines the two core subproblems of OTR: (a) *OD*, the separation of simultaneous events into two or more timesteps, and (b) *OA*, the combination of two or more consecutive timesteps into a single one.

B. Optimal disaggregation (OD)

We begin by formulating disaggregation as a decision problem to facilitate analysis of hardness:

Definition 2. *Optimal Disaggregation (OD)*: Given G, $\mathbb{X}|\Delta_o$, a model $\mathbb{M}$, and a target timeslice $t \in \Delta_o$, find a disaggregation $\Delta_o \to \Delta_n$ that maps t onto $\{t_{(1)}, \ldots, t_{(j)}\}$ such that $\mathcal{L}(\mathbb{X}|\Delta_n) \geq \mathcal{L}^\star$.

An example disaggregation is depicted in Fig. 1, where the (red) activations $D - F$ from $t_0 = 3$ in the original timeline are disaggregated into $t_\star = \{3, 4, 5\}$ in the optimal timeline. Note that the events are mapped to specific new time steps in a way that will maximize their likelihood.

Theorem 1. OD, and thus general OTR, is NP-hard for all settings considered herein. *(Proof available in [30])*

Given the above, we must turn to approximations in our solutions. Fortunately, the process of splitting a timestep into two (hereinafter called *2-OD*) is submodular. Let X_t be the set of events to be disaggregated. Let $L(Y)$, where $Y \subseteq X_t$, represent our likelihood function when the events in Y are assigned to $t_{(1)}$ and $X_t \setminus Y$ are assigned to $t_{(2)}$. Then:

Theorem 2. $L(Y)$ is a submodular set function. *(Proof available in [30])*

This process is also non-monotonic, since reassigning an event from $t_{(2)}$ to $t_{(1)}$ may reduce the overall likelihood.

Corollary 1. A probabilistic greedy maximization of the likelihood of 2-OD will yield a $\frac{1}{2}$-approximation algorithm to the optimal solution. *(Due to [31])*

C. Optimal Aggregation (OA)

The goal in aggregation is to maximize the likelihood of events in consecutive timesteps by combining them, effectively rendering events in them simultaneous. An example of such aggregation is depicted in Fig. 1, where the original steps $t_0 = 4 \ldots 7$ are aggregated into a single new step $t_\star = 7$. A recent work [17] focused on aggregation of IC diffusion events and demonstrated that optimal aggregation can be performed via dynamic programming in $O(|\mathbb{X}|^4)$ time. In Section IV, we discuss a faster solution for long-memory processes and generalize to network growth in addition to diffusion.

IV. Solutions

A. Disaggregation

Here we assume that the input data is over-aggregated, i.e. the observed events are grouped in large windows in which the expected ordering by the process model is partially lost. While optimal disaggregation of even a single timestep (2-OD) is NP-hard, the submodularity property discussed above allows us to employ the optimization framework from [31] for 2-OD to obtain a randomized greedy heuristic with a $\frac{1}{2}$-approximation ratio. To disaggregate the whole timeline, we apply the greedy 2-OD approximation recursively on steps of the original timeline in chronological order. The chronological order is driven by the observation that events are only affected by preceding events; therefore, it makes sense to disaggregate the timeline in order from earliest to latest.

Our proposed disaggregation scheme is presented in Alg. 1. We first initialize $\Delta_\star$ with the default timeline. Then we perform a forward sweep in time attempting to disaggregate earlier time steps first (Steps 3-20). For a time step t, we first create a candidate split into two time steps $t_{(1)}$ and $t_{(2)}$ and save the events currently mapped to this time step in X (Steps 4, 5). Next we apply the randomized greedy heuristic for submodular functions [31] on the likelihood $L()$ of events from X being assigned to the candidate time t' (Steps 6-16). This function was introduced for the 2-OD problem in Sec. III-B—$L(A)$ is the likelihood of the timeline if events in A are mapped to the first of the new candidate steps $t_{(1)}$ and the complement $X \setminus A$ to $t_{(2)}$.

The greedy heuristic initializes temporary sets A and B as empty and the full set of events as X respectively (Step 6). It proceeds by evaluating the likelihood gains of adding consecutive members of $x \in X$ to A: ∇_A, or removing x from B: ∇_B (Steps 8, 9). One of the respective actions, add x to A or remove it from B, is performed with probability proportional to its likelihood gain (Steps 10-13). When all events $x \in X$ are processed, A and B are equivalent, and the proposed split places events from A into $t_{(1)}$ and the complement into $t_{(2)}$ (Steps 15, 16). If the likelihood of the new candidate timeline is better than the current one, the new timeline is kept and t remains at the same position (i.e. an attempt to further disaggregate $t_{(1)}$ will be made in the next iteration), otherwise the t-th step is kept *as is* and t is incremented (Steps 17-19). In the worst case, this process is

Algorithm 1 Forward greedy disaggregation

Require: Graph G, model $\mathbb{M}$, set of events on an input timeline $\mathbb{X}|\Delta_n$
Ensure: Disaggregated timeline $\mathbb{X}|\Delta_\star$
1: $\Delta_\star = \Delta_n$
2: $t = 1$
3: **while** $t < len(\Delta_\star)$ **do**
4: Let $\Delta_\star^{(t)}$ be $\Delta_\star$ with time step t split in two: $t_{(1)}$ and $t_{(2)}$
5: Let X be the events at time t in $\Delta_\star$
6: $A = \emptyset, B = X$
7: **for** Events $x \in X$ **do**
8: $\nabla_A = L(A \cup x) - L(A)$
9: $\nabla_B = L(B \setminus x) - L(B)$
10: $p = \frac{\nabla_A}{\nabla_A + \nabla_B}$
11: **if** $rand(0,1) \leq p$ **then** $A = A \cup x$
12: **else** $B = B \setminus x$
13: **end if**
14: **end for**
15: Map events from A to step $t_{(1)}$ in $\Delta_\star^{(t)}$
16: Map events from $\{X \setminus A\}$ to step $t_{(2)}$ in $\Delta_\star^{(t)}$
17: **if** $\mathcal{L}(\mathbb{X}|\Delta_\star^{(t)}) > \mathcal{L}(\mathbb{X}|\Delta_\star)$ **then** $\Delta_\star = \Delta_\star^{(t)}$
18: **else** $t = t + 1$
19: **end if**
20: **end while**
21: **return** $\mathbb{X}|\Delta_\star$

$O(|\mathbb{X}|^2)$, as the process could conceivably start with a single timestep containing all of the events of $\mathbb{X}$ and end with every event in its own timestep; this would require $|\mathbb{X}|$ iterations of the loop in Step 3, and iterations would, on average, have to consider a number of events equal to $c|\mathbb{X}|$ for some constant fraction c.

B. Aggregation

Some portions of the timeline may be at too high a resolution—that is, events are overly dispersed in time. (See timesteps 5-7 in Fig. 1.) In this case, we must *aggregate* the timeline, merging two or more adjacent timesteps to make their events simultaneous. This problem was addressed in [17] for the specific setting of IC diffusion. We extend that dynamic-programming approach to network growth and long-memory models; details are available in [30].

The DP algorithm, however, is not scalable to real-world datasets, and so we fall back upon a greedy approach. Starting from a fully aggregated timeline, we add back the divisions which have the most explanatory power according to $\mathbb{M}$; i.e., those that produce the largest increase in likelihood. The intuition is that if two related events are simultaneous, then neither can help explain the other; we ideally would like one (the "cause") to precede the other (the "effect"). However, if the first event precedes the second one by too much, we will also lose explanatory power. (The precise definition of "too much," of course, is controlled by the memory length of our model $\mathbb{M}$.)

C. Full algorithm for OTD: disaggregate, then aggregate

Our complete procedure, *Full*, applies both aggregation and disaggregation, similar to our motivating example from Fig. 1. We first disaggregate the whole timeline as needed according to Alg. 1, and then aggregate using either the DP algorithm or the greedy heuristic. Applying disaggregation first ensures that we never reverse any pair of ordered events in the input data. We believe there are opportunities for speedup by carefully interspersing these actions; this is a direction for future work.

An important note is that the same implementation of the algorithm can be used for both diffusion and growth problems, as network growth problems can be *transformed* into a diffusion-like form by using the line graph of G; details of this process are omitted due to space constraints.

A summary of solutions for each (sub)problem and setting is presented in Table I.

Problem	Memory	Optimal	Greedy	APX
OD	Short	NP-hard	$O(\|\mathbb{X}\|^2)$	1/2
	Long	NP-hard	$O(\|\mathbb{X}\|^2)$	1/2
OA	Short [17]	DP: $O(\|\mathbb{X}\|^4)$	$O(\|\mathbb{X}\|log\|\mathbb{X}\|)$	-
	Long	DP: $O(\|\mathbb{X}\|^3)$	$O(\|\mathbb{X}\|log\|\mathbb{X}\|)$	-
OTR	Both	NP-hard	$O(\|\mathbb{X}\|^2)$	-

TABLE I
SUMMARY OF SOLUTIONS FOR *OTR* AND SPECIAL CASES *OD* AND *OA*, COMPLEXITY AND APPROXIMATIONS FOR *2-OD*.

V. Experimental Evaluation

A. Data

We produce synthetic PA networks and, for diffusion, we produce cascades according to the IC model. We then remove divisions from the ground truth timeline and add the same number of new divisions uniformly at random at different positions. The *distortion factor* is the ratio of the number of changed divisions to the timeline length.

We also use a pair of real-world datasets to evaluate our methods. In the diffusion setting, we employ the Flickr data published in [32], using the friendship graph as our static network and photos marked as favorites as the cascades. For network growth, we selected a connected subgraph of the MPI Facebook friendship network published in [33]. In each case, we first apply a regular aggregation (roughly hourly for Flickr and weekly for Facebook) that produces a timeline of approximately 100 timesteps. We then apply the same type of distortion (with compression and expansion) described above.

B. Direct measurements of timeline reconstruction

As a first-cut measure of effectiveness, we count how many pairs of events have been deranged—either simultaneous events that have been separated in time or ordered events that now appear simultaneous. We calculate three decision rates in relation to these: the fraction of derangements fixed correctly, the fraction "fixed" incorrectly (that is, our algorithm changes the order in a way that does not match the original timeline), and the net rate, which is simply the difference between these.

Figure 3 shows our results for these measures. We distort the timeline by artificially expanding and compressing it and then see how well we can recover the deranged pairs. Interestingly, in synthetic data the amount of distortion has little effect on this measure - we correctly identify over 20% (growth) or 30% (diffusion) of the pairs with very low error rates. Furthermore, we compare favorably with the baseline: the aggregation-only approach proposed in [17].

The bottom two figures show the results on our two real-world datasets. For diffusion, the distortion factor again has

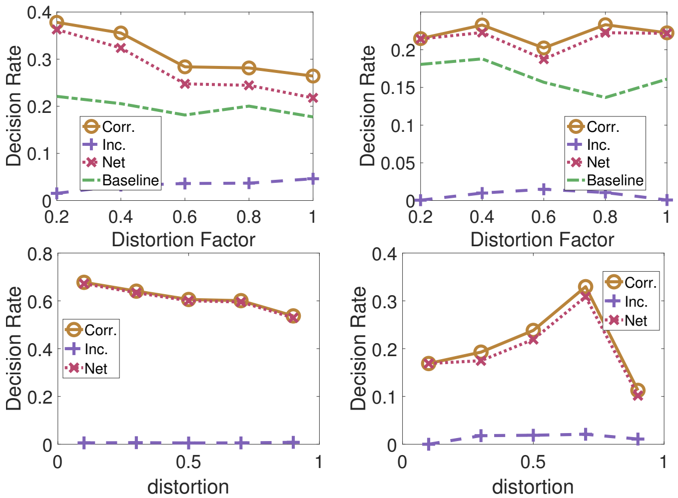

Fig. 3. Decision rates in both the diffusion (left) and network growth (right) settings, with synthetic data at the top and real-world data below. The baseline in the top two figures is [17]. In all cases, we maintain a very low incorrect decision rate.

very little impact. In the network growth data, larger distortion rates lead to an increase in opportunities to fix the timeline. Our algorithm takes increasing advantage of these through $df = 0.7$; beyond this, too much temporal information has been removed and the problem becomes too difficult.

C. Cascade completion

The application we use to test our method in the information diffusion setting is the prediction of missing cascade activations, using the algorithm of Zong et al. [34]. In both synthetic and real datasets, we generate test cases by simply removing some of the activations; the dropped activations are chosen uniformly at random excluding those in the first or last time step, as those cannot be predicted by Zong's algorithm. We use F_1 as our quality metric; results are shown in Figure 4. In all cases, the full OTR algorithm using both aggregation and disaggregation outperforms using only one or the other.

The first three plots show three different ways of increasing problem difficulty on synthetic data. The full algorithm maintains its advantage over both the default timeline *(None)* and the all-aggregation baseline of [17] across all settings. The final two plots show results on the real-world Flickr data. Notably, as shown in the final plot, our approach works best precisely when the default timeline is useless.

D. Link prediction

We test our method using a tensor-factorization link predictor from Dunlavy et al. [35] whose output is a matrix of similarity scores, reflecting the likelihood of a link in the future. We use AUC (area under the ROC curve) as our metric, with results shown in Fig. 5.

By varying α in the synthetic dataset we found that a greater simulated preferential attachment made link prediction more difficult across all timelines. When all edges are linking to the same popular node, it is likely increasingly difficult to disentangle which nodes linked to said popular node first. In contrast, when nodes have variety in degree, more information can be gleaned from the patterns of attachment. Thus we limited α to a lower range of values.

For the full OTR algorithm, the level of improvement was approximately 0.1 (12-15%) for all parameter settings on synthetic data and 0.05 (7%) or more on real-world data. Given the inherent difficulty of the link prediction task, these gains represent a substantial improvement over the default timeline.

VI. Related Work

The authors of references [15], [36] were among the first to demonstrate the critical importance of using the correct temporal resolution for various predictive tasks on networks. Variations in the speed of network process was also demonstrated empirically in several prior works [13], [14]. Despite these observations, very little work has been done that explores the idea of learning a variable temporal resolution. Once such work is [37], but it considers continous-time processes as opposed to discrete time processes which are predominantly studied in the network context.

The closest to this work is [17] which is limited to short memory diffusion and does not consider disaggregation. While we use the above as a baseline in some experiments for severely disaggregated data, it cannot handle over-aggregation or network growth processes directly. We generalize [17] by considering both long-memory and short-memory processes as well as network growth in addition to diffusion.

VII. Conclusion

We have presented a method for optimizing a timeline of observed network events in order to maximize their overall likelihood. This optimized timeline can be used to recover the order of events when the data collection process has distorted the temporal information; it can also be used to enhance any process that heavily depends upon the temporal nature of network event data. We have shown that an optimized timeline can produce significant improvements in the solution of two very common tasks: link prediction and detection of missing cascade activations.

Our theoretical contributions included a very general problem framework that can be applied to any probabilistic model of temporal network event creation. We proved the NP-hardness of the overall problem and provided an optimal algorithm for one key subproblem and an approximation guarantee for another one.

Perhaps most importantly, we have laid the groundwork for future investigation into optimal inhomogeneous-resolution timelines. Directions for further exploration include improvements of the algorithms presented herein, especially exploring the possibility of alternating aggregation and disaggregation actions, and application of the OTR framework to other tasks using data from temporal network processes.

References

[1] G. Kossinets and D. J. Watts, "Empirical analysis of an evolving social network," *science*, vol. 311, no. 5757, pp. 88–90, 2006.

[2] C. Aggarwal and K. Subbian, "Evolutionary network analysis: A survey," *ACM Computing Surveys (CSUR)*, vol. 47, no. 1, p. 10, 2014.

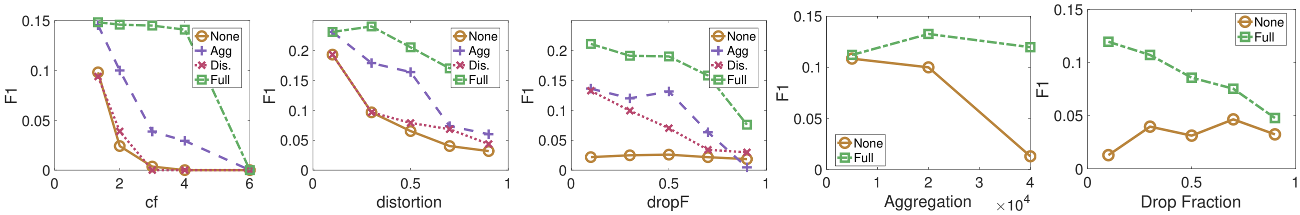

Fig. 4. F_1 measure for the cascade completion experiment. The first three plots are for synthetic data, and the last two are on the Flickr data set from [32]. In all plots, None is the input (distorted) timeline, Agg is the timeline resulting from applying only the aggregation algorithm from [17], Dis is the timeline resulting from applying only our disaggregation algorithm (Alg.1), and Full is the timeline resulting from applying our complete OTR algorithm.

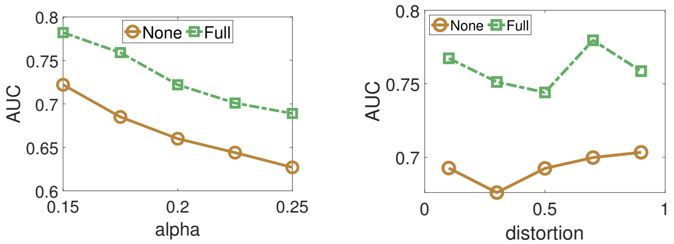

Fig. 5. Effect of timeline on the TF-based link prediction algorithm of Dunlavy et al. [35], measured via AUC. Synthetic data appears on the left; parameter settings of $k = 5$ TF components and $\tau = 3$ time steps of "lookback" were used. On the right are results with a subset of the MPI Facebook [33] dataset; parameter values were $k = 5$ and $\tau = 6$.

[3] J. Goldenberg, B. Libai, and E. Muller, "Using complex systems analysis to advance marketing theory development: Modeling heterogeneity effects on new product growth through stochastic cellular automata," *Academy of Marketing Science Review*, vol. 2001, p. 1, 2001.

[4] M. Granovetter, "Threshold models of collective behavior," *American journal of sociology*, pp. 1420–1443, 1978.

[5] A. Guille, H. Hacid, C. Favre, and D. A. Zighed, "Information diffusion in online social networks: A survey," *ACM Sigmod Record*, vol. 42, no. 2, pp. 17–28, 2013.

[6] A. Goldenberg, A. X. Zheng, S. E. Fienberg, E. M. Airoldi *et al.*, "A survey of statistical network models," *Foundations and Trends® in Machine Learning*, vol. 2, no. 2, pp. 129–233, 2010.

[7] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2003, pp. 137–146.

[8] R. M. Bond, C. J. Fariss, J. J. Jones, A. D. Kramer, C. Marlow, J. E. Settle, and J. H. Fowler, "A 61-million-person experiment in social influence and political mobilization," *Nature*, vol. 489, no. 7415, p. 295, 2012.

[9] J. Bao, Y. Zheng, D. Wilkie, and M. Mokbel, "Recommendations in location-based social networks: a survey," *GeoInformatica*, vol. 19, no. 3, pp. 525–565, 2015.

[10] D. Easley, J. Kleinberg *et al.*, *Networks, crowds, and markets*. Cambridge university press Cambridge, 2010, vol. 8.

[11] S. T. Smith, K. D. Senne, S. Philips, E. K. Kao, and G. Bernstein, "Covert network detection," *Lincoln Laboratory Journal*, vol. 20, no. 1, pp. 3–1, 2013.

[12] K. Macropol, P. Bogdanov, A. Singh, L. Petzold, and X. Yan, "I act, therefore i judge: Network sentiment dynamics based on user activity change," in *ASONAM*, 2013.

[13] A. Tagarelli and R. Interdonato, "Time-aware analysis and ranking of lurkers in social networks," *SNAM*, vol. 5, no. 1, p. 46, 2015.

[14] J. Yang, J. McAuley, J. Leskovec, P. LePendu, and N. Shah, "Finding progression stages in time-evolving event sequences," in *WWW*, 2014.

[15] G. Krings, M. Karsai, S. Bernhardsson, V. D. Blondel, and J. Saramäki, "Effects of time window size and placement on the structure of an aggregated communication network," *EPJ Data Science*, vol. 1, no. 4, pp. 1–16, 2012.

[16] R. S. Caceres, T. Berger-Wolf, and R. Grossman, "Temporal scale of processes in dynamic networks," in *ICDM Workshop*, 2011.

[17] D. J. DiTursi, G. A. Katsios, and P. Bogdanov, "Network clocks: Detecting the temporal scale of information diffusion," in *IEEE International Conference on Data Mining*. IEEE, 2017, pp. 841–846.

[18] J. Goldenberg, B. Libai, and E. Muller, "Talk of the network: A complex systems look at the underlying process of word-of-mouth," *Marketing letters*, vol. 12, no. 3, pp. 211–223, 2001.

[19] Q. Li, Z. Wang, B. Wu, and Y. Xiao, "Competition and cooperation: Dynamical interplay diffusion between social topic multiple messages in multiplex networks," *IEEE Transactions on Computational Social Systems*, 2019.

[20] V. Amelkin, P. Bogdanov, and A. Singh, "A distance measure for the analysis of polar opinion dynamics in social networks," in *Proceedings of the IEEE International Conference on Data Engineering (ICDE)*, 2017. [Online]. Available: http://www.cs.albany.edu/ petko/lab/papers/absicde17.pdf

[21] J. Leskovec, D. Chakrabarti, J. Kleinberg, C. Faloutsos, and Z. Ghahramani, "Kronecker graphs: An approach to modeling networks," *Journal of Machine Learning Research*, vol. 11, no. Feb, pp. 985–1042, 2010.

[22] S. N. Dorogovtsev and J. F. F. Mendes, "Effect of the accelerating growth of communications networks on their structure," *Physical Review E*, vol. 63, no. 2, p. 025101, 2001.

[23] D. M. Smith, J.-P. Onnela, and N. F. Johnson, "Accelerating networks," *New Journal of Physics*, vol. 9, no. 6, p. 181, 2007.

[24] M. J. Gagen and J. Mattick, "Accelerating, hyperaccelerating, and decelerating networks," *Physical Review E*, vol. 72, no. 1, 2005.

[25] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *science*, vol. 286, no. 5439, pp. 509–512, 1999.

[26] J. Sun, S. Staab, and F. Karimi, "Decay of relevance in exponentially growing networks," in *Proceedings of the 10th ACM Conference on Web Science*. ACM, 2018, pp. 343–351.

[27] M. Li, X. Wang, K. Gao, and S. Zhang, "A survey on information diffusion in online social networks: Models and methods," *Information*, vol. 8, no. 4, p. 118, 2017.

[28] A.-L. Barabâsi, H. Jeong, Z. Néda, E. Ravasz, A. Schubert, and T. Vicsek, "Evolution of the social network of scientific collaborations," *Physica A: Statistical mechanics and its applications*, vol. 311, no. 3-4, pp. 590–614, 2002.

[29] M. Medo, G. Cimini, and S. Gualdi, "Temporal effects in the growth of networks," *Physical review letters*, vol. 107, no. 23, p. 238701, 2011.

[30] D. J. DiTursi, C. S. Kaminski, and P. Bogdanov, "Set your clock: Optimal timescales for network processes (full version)." [Online]. Available: http://www.cs.albany.edu/∼petko/lab/papers/dkbicdm2019.pdf

[31] N. Buchbinder, M. Feldman, J. Seffi, and R. Schwartz, "A tight linear time (1/2)-approximation for unconstrained submodular maximization," *SIAM Journal on Computing*, vol. 44, no. 5, pp. 1384–1402, 2015.

[32] M. Cha, A. Mislove, and K. P. Gummadi, "A measurement-driven analysis of information propagation in the flickr social network," in *WWW*, 2009.

[33] B. Viswanath, A. Mislove, M. Cha, and K. P. Gummadi, "On the evolution of user interaction in facebook," in *Proceedings of the 2nd ACM SIGCOMM Workshop on Social Networks*, August 2009.

[34] B. Zong, Y. Wu, A. K. Singh, and X. Yan, "Inferring the underlying structure of information cascades," in *ICDM*, 2012.

[35] D. M. Dunlavy, T. G. Kolda, and E. Acar, "Temporal link prediction using matrix and tensor factorizations," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 5, no. 2, p. 10, 2011.

[36] M. Budka, K. Musial, and K. Juszczyszyn, "Predicting the evolution of social networks: optimal time window size for increased accuracy," in *Proc. PASSAT SocialCom*, 2012.

[37] M. G. Rodriguez, D. Balduzzi, and B. Schölkopf, "Uncovering the temporal dynamics of diffusion networks," *arXiv preprint arXiv:1105.0697*, 2011.