Identifying Complicated Contagion Scenarios from Cascade Data

Galen Harrison
gh7vp@virginia.edu
University of Virginia
USA

Amro Alabsi Aljundi
nmm2uy@virginia.edu
University of Virginia
USA

Jiangzhuo Chen
chenj@virginia.edu
University of Virginia
USA

S. S. Ravi
ssravi0@gmail.com
University of Virginia
USA

Anil Kumar Vullikanti
vsakumar@virginia.edu
University of Virginia
USA

Madhav V. Marathe
marathe@virginia.edu
University of Virginia
USA

Abhijin Adiga
abhijin@virginia.edu
University of Virginia
USA

ABSTRACT

We consider the setting of cascades that result from contagion dynamics on large realistic contact networks. We address the question of whether the structural properties of a (partially) observed cascade can characterize the contagion scenario and identify the interventions that might be in effect. Using epidemic spread as a concrete example, we study how social interventions such as compliance in social distancing, extent (and efficacy) of vaccination, and the transmissibility of disease can be inferred. The techniques developed are more generally applicable to other contagions as well.

Our approach involves the use of large realistic social contact networks of certain regions of USA and an agent-based model (ABM) to simulate spread under two interventions, namely vaccination and generic social distancing (GSD). Through a machine learning approach, coupled with parameter significance analysis, our experimental results show that subgraph counts of the graph induced by the cascade can be used effectively to characterize the contagion scenario even during the initial stages of the epidemic, when traditional information such as case counts alone are not adequate for this task. Further, we show that our approach performs well even for partially observed cascades. These results demonstrate that cascade data collected from digital tracing applications under poor digital penetration and privacy constraints can provide valuable information about the contagion scenario.

CCS CONCEPTS

• **Computing methodologies** → **Machine learning**; **Simulation evaluation**.

KDD '23, August 6–10, 2023, Long Beach, CA, USA

ACM ISBN 979-8-4007-0103-0/23/08.
https://doi.org/10.1145/3580305.3599841

KEYWORDS

Epidemic model, network diffusion, simulation, cascade, subgraph enumeration, digital contact tracing

ACM Reference Format:
Galen Harrison, Amro Alabsi Aljundi, Jiangzhuo Chen, S. S. Ravi, Anil Kumar Vullikanti, Madhav V. Marathe, and Abhijin Adiga. 2023. Identifying Complicated Contagion Scenarios from Cascade Data. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '23), August 6–10, 2023, Long Beach, CA, USA.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3580305.3599841

1 INTRODUCTION

1.1 Background and Motivation

In the context of contagion processes on networks, the evolution of a contagion in a contact graph can be represented as a who-infected-who graph [40], which we refer to as a *cascade graph* or simply a cascade. Various cascade sensing or tracing methods are employed to collect fine-grained cascade data. For example, in the context of infectious diseases (human or livestock), manual and digital tracing have been applied at varied spatial scales ranging from individual buildings (such as hospitals) [55] to the regional scale [6, 23, 41]. The ongoing COVID-19 pandemic has seen the emergence of mobile tracing technologies (see e.g., [2, 4, 6, 12, 14, 16, 20, 44, 49, 53, 56]). Delimitation surveys in the context of invasive species [19] and reconstruction of diffusion trees in social contagions [1, 7] are other examples of cascade tracing. While there are many challenges associated with data collection [44] (e.g., low penetration levels and adoption, privacy concerns), cascade tracing can provide valuable contact-network-level information. This helps in understanding the heterogeneous individual-level and location-level interactions that result from complex activity patterns of the population and its response (or the lack thereof) to an ongoing epidemic. A natural question is how data generated from cascade tracing can be used effectively to inform public health decisions for ongoing and future disease outbreaks. The application context considered in this paper is that of infectious disease spread in a human population, but it can be readily extended to study other types of contagions.

The evolution of an epidemic is influenced not only by the characteristics of the disease but also by those of the population, such as its size, immunity levels, nature of interactions, and the response to the disease [11, 46, 54]. In recent years, there have been several papers on using very high-resolution agent-based models to represent and study these complex contagion scenarios [3, 13, 20, 25, 27]. These models use digital twins of real-world populations and infrastructure, and capture individual-level, spatial and temporal heterogeneity by using large node- and edge–attributed networks in the context of disease spread. Such simulation systems are an ideal platform for exploring and evaluating the utility of cascade tracing datasets. However, simulation analysis is a challenging task. Even simple contagion scenarios can lead to a distribution of large attributed cascade graphs that are not amenable to analysis.

In this setting, we consider the *scenario identification problem*, where given features of a (partially) observed cascade graph, the objective is to identify the disease properties, and behavioral aspects of the population. Accounting for these dynamics in modeling efforts is critical in predicting the future course of the disease and making informed public health decisions [9, 22, 54]. Statistics such as case counts over time are important and are usually adequate to characterize the disease under the assumption of simple and homogeneous systems that account only for disease characteristics and population size. However, drawing conclusions based solely on these aggregated measures can turn out to be misleading in complex scenarios, where behavioral aspects shape individual-level interactions. The COVID-19 pandemic has highlighted how shifts in control policies and public response can shape the evolution of a disease [31].

We view the scenario identification problem as a learning problem in the context of a simulation system used to study contagion dynamics. Cascades generated from various scenarios are used to train a machine learning algorithm for the following multiclass classification problem: given the features of a cascade, identify the scenario(s) that generated it. Through subsequent interpretability and parameter significance analysis, one can identify how different model parameters affect the cascade structure. Such approaches are being considered in the analysis of complex simulation systems using machine learning techniques [5, 21, 30].

1.2 Our contributions

A learning-based approach in an adversarial setting. In this work, we consider cascades resulting from complex disease dynamics that include pharmaceutical and non-pharmaceutical interventions in large realistic contact networks. Our objective is to study how the different aspects of disease scenarios manifest in the graph structure of the cascade, and if these relationships can be effectively utilized in tasks such as forecasting and counterfactual scenario analyses. To this end, we address the scenario identification problem in an adversarial setting where we construct multiple scenarios differing in levels of transmissibility, vaccination, and generic social distancing (GSD), in such a way that they cannot be distinguished from one another based on infection counts at the early stages of the disease (denoted by *time horizon* T). The input feature vector corresponds to the counts of structural features of the cascade as described below. The objective of the learner is to classify a given cascade into a scenario given its features.

Network features. We consider features of cascade graphs consistent with data that can be collected from cascade tracing protocols such as number of infected nodes, number of interactions, the average number of neighbors infected by nodes in the cascade (out degree), sequence of interactions (unlabeled and labeled path motifs), properties of uninfected boundary nodes (i.e., uninfected nodes adjacent to at least one infected node), etc. In many of these cases, we also use features where the count is a function of time. To the best of our knowledge, previous works in the context of contact tracing have not considered these features even though such information is potentially available [49, 56]. Also, all the features considered are aggregate counts of subgraph structures. Some of them can be computed using *private* methods (see e.g., [48]).

Novel simulation analytics. We consider realistic scenarios of disease propagation in a population with complex interactions and multiple types of interventions in place. To this end, we use a digital twin of two regions from the contiguous United States [13, 18] that have node-level and edge-level attributes embedded in them representative of the populations of the regions. We use an agent-based model (ABM) to simulate disease spread under vaccination and GSD interventions.

Partial observation model. Motivated by the fact that not all individuals in a population can be traced, we consider a partial observation model, where only a subset of nodes are observed, and the features are available for only the subgraph of the cascade induced by this subset. The size of the observed node set is determined by the *coverage* parameter $\kappa \in (0, 1]$, where $\kappa = 1$ corresponds to complete information about the cascade.

Experimental results. We performed extensive experimental analysis on the two networks for different values of the time horizon T and coverage κ for four disease scenarios with varying levels of transmissibility and interventions. We used three machine learning algorithms for the scenario prediction problem: Support Vector Machines (SVM), Random Forest (RF), and Logistic Regression (LR). The k-fold cross-validation technique was used for hyperparameter tuning and model selection. For interpretability, we used Shapley value analysis and ablation analysis. Our results are as follows.

- Learning algorithms fare poorly when trained with features corresponding to case counts alone (accuracy $\leq$ 0.3), but when network structural information is included in the feature set, the performance increases greatly (accuracy $\approx$ 0.8).
- Labeled path motifs played a significant role in distinguishing different scenarios, while unlabeled structural characteristics did not. The edge labels on the paths encode the activity types of the end points such as essential–essential, essential–non-essential, etc. Some of this can be attributed to the behavioral model in our simulations; the greater the level of GSD, the larger is the number of non-essential interactions removed from the network.
- For decreasing coverage levels, the performance of the learner degraded gracefully, showing that these statistics can be applied in practical situations where the digital penetration or public acceptance is low.

- Our results show that even during very early stages of the diffusion process (small time horizon), the prediction performance is good.

2 DISEASE CASCADES AND THEIR STRUCTURAL FEATURES

2.1 Preliminaries

For an integer $k > 0$, let $[k]$ denote the set $\{1, 2, \cdots, k\}$. We consider both undirected and directed graphs. In an undirected simple graph $G(V, E)$ with node set V and edge set E, let $N(v)$ denote the set of neighbors of v and $d(v) = |N(v)|$ denote its degree. A graph $H(V, E)$ with node set V and edge set E is a *directed acyclic graph* (DAG), if it is directed and does not have any directed cycles. Following [15], a node v of a DAG is a *source* node if v has no in-neighbors (i.e., nodes from which v has an incoming edge). Likewise, a node v of a DAG without any out-neighbors (i.e., nodes to which v has outgoing edges) is called a *sink* node. When $H(V, E)$ is restricted to be a directed tree, level(v) denotes the distance of v from the unique source node from which v is reachable. Let $N_{\text{out}}(v)$ and $N_{\text{in}}(v)$ denote respectively the set of out- and in-neighbors of v. Thus, $v' \in N_{\text{out}}(v) \Leftrightarrow (v, v') \in E(H)$ and $v' \in N_{\text{in}}(v) \Leftrightarrow (v', v) \in E(H)$.

Contact graph. Let $G(V, E)$ be a contact graph with node set $V(G)$ and edge set $E(G)$ on which the contagion occurs. It is undirected, with each edge having a label and weight associated with it. Each edge label indicates the nature of interaction between the two end points and can be of the following types: (i) `essential` (like interactions at home or work); (ii) `non-essential` (like shopping); or (iii) `mixed` (one end point performing essential activity, while the other a non-essential activity). In addition, each edge weight (w_e for an edge e) corresponds to the time *duration* of interaction.

Diffusion model. Our work is applicable to a broad class of diffusion models on networks [17]. In this work, we consider a simple class of discrete-time network-based *Susceptible-Exposed-Infectious-Recovered* (SEIR) models [35], where a susceptible v (node state **S**) is infected by an infectious neighbor u (node state **I**) probabilistically. Transmission is modeled by the Direct Gillespie Method. The probability that node v is infected depends on the *propensity* ρ for each edge e between v and each u of the infectious neighbors of v: $\rho(v, u, e) = w_e \tau \sigma_v \iota_u$, where weight w_e of edge e is the duration of the contact, transmissibility τ is a global parameter representing a rate proportional to the likelihood of transmission per unit of time, σ_v is the susceptibility of v, and ι_u is the infectivity of u. Nodes in **E** state transition to the infectious (**I**) state after a dwell time (mean 1.7 days). Nodes in **I** state transition to the recovered (**R**) state after a dwell time (mean 4.1 days).

2.2 Cascade graph

Supposing that a contagion process is observed for time steps $0, \ldots, T$ on the contact graph $G(V, E)$. We refer to T as the *time horizon*. The evolution of the diffusion process can be captured through a *cascade graph* $C(\mathcal{V}_C, \mathcal{E}_C)$ whose node set $\mathcal{V}_C \subseteq V \times [T]$ is the *time-expanded* version of the V, where each node $(v, t) \in \mathcal{V}_C$ denotes that v was exposed at time t. Its edge set E_C corresponds to the set of directed edges $\big((v, t), (v', t')\big)$, with $t < t'$, denoting that v infected v' at time t'. Let $\mathcal{C}$ denote the set of all valid cascade graphs. Note that each cascade graph is a DAG. In the case of SEIR model, a node is infected at most once. Hence, in some instances, we use a simpler version of the node and edge set by stripping the time information. Let $V_C = \{v \mid (v, t) \in \mathcal{V}_C\}$ denote the set of nodes in V that were infected in C and $E_C = \{(v, v') \mid \big((v, t), (v', t')\big) \in \mathcal{E}_C\}$ denote the set of edges on which transmissions occurred. An example cascade is illustrated in Figure 1.

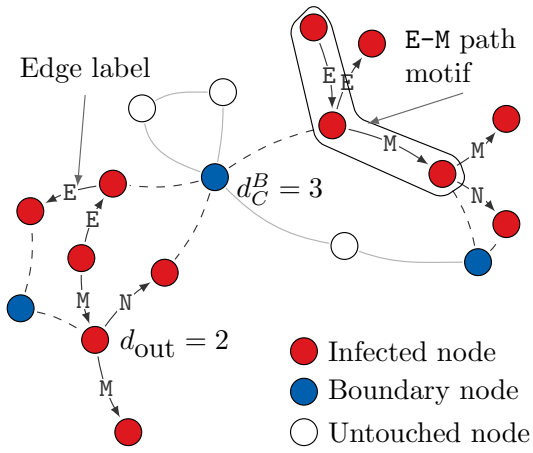


Figure 1: An example of a cascade graph with boundary nodes and edge labels.

2.3 Structural features of cascade graphs

Directed labeled path motifs. We consider both labeled and unlabeled directed path subgraphs or motifs. Let $n_P(k)$ denote the number of k-length unlabeled directed motifs in the cascade graph C. We will also consider path count as a function of time in which case, $n_P(k, t)$ denotes the number of k-length paths where the t is the time at which the first node in the path was exposed.

In the labeled case, we assume that each directed edge has a label from a given set $\mathcal{L}$. A path motif is highlighted in Figure 1. For any $k \geq 1$, given a label sequence $L = \langle \ell_1, \ell_2, \ldots, \ell_k \rangle$ with k labels each from $\mathcal{L}$ and a directed simple path $P = (e_1, e_2, \ldots, e_k)$ with k directed edges (i.e., a k-length directed path motif), we say that P is **consistent** with L if the label of edge $e_i = \ell_i$ for $1 \leq i \leq k$. Let $n_P(k, L)$ denote the number of k-length paths that are consistent with a label sequence L. Given a DAG, integer k and a label sequence L, we will present a polynomial time algorithm to compute $n_P(k, L)$ in Section 2.4.

Out-degree or directed star motifs. The out-degree of a node v in the cascade, denoted by $d_{\text{out}}(v)$, corresponds to the number of neighbors of v that were successfully infected by v in the cascade. Since out-degree counts correspond to star motifs, we will use the notation $n_S(k)$ to denote the number of directed star graph motifs in the cascade with k edges (and therefore, $k + 1$ nodes). Similarly, $n_S(k, t)$ corresponds to number of star motifs of size k with the center node infected at time t.

Boundary of the cascade graph. Given a cascade graph $C(V_C, E_C)$ and the underlying contact graph $G(V, E)$, the node boundary δV_C

is the set of nodes outside V_C that have at least one neighbor in V_C, i.e., $\delta V_C = \{v \mid v \in V \setminus V_C,\ N(v) \cap V_C \neq \emptyset\}$. Similarly δE_C is the set of edges for which one end point is in V_C and the other in $V \setminus V_C$. For each boundary node v, let its **boundary degree** $d_C^B(v)$ be the number of its neighbors in the cascade graph C, i.e., $d_C^B(v) = |N(v) \cap V_C|$. See Figure 1 for an example. Let $n_B(k)$ denote the number of boundary nodes with boundary degree equal to k, which we will refer to as boundary-degree counts.

2.4 Computing consistent labeled path counts

Let $H(V_H, E_H)$ denote the given DAG, where $|V_H| = n$ and $|E_H| = m$. For some $k \geq 1$, let $L = \langle \ell_1, \ell_2, \ldots, \ell_k \rangle$ denote a label sequence of length k. Our algorithm for computing $n_P(k, L)$, that is, the number of k-length directed paths that are consistent with L, is based on dynamic programming (DP). We recall that the nodes of any DAG can be arranged along a line in a **topologically sorted** order [15] so that every directed edge has a left to right orientation. For $1 \leq j \leq k$, we use L_j to denote the subsequence of L containing the first j entries of L. Our DP approach maintains an array $q_i[1:k]$ of size k with each node v_i. Specifically, for $1 \leq j \leq k$, $q_i[j]$ will store the number of j-length directed paths that satisfy the following two conditions: (i) they are consistent with L_j and (ii) they *end* at v_i. Once we have the quantities $q_i[k]$ for $1 \leq i \leq n$, the quantity $n_P(k, L)$ is equal to $\sum_{i=1}^{n} q_i[k]$. Thus, we now focus on computing the quantities $q_i[j]$, $1 \leq j \leq k$, for each node v_i. We will do this in the chosen topological order.

Note that $q_1[j] = 0$ for $1 \leq j \leq k$, since v_1 is a source node (which has no incoming edges) in the topological order. For each node q_i, with $i \geq 2$, we have

$$q_i[1] \ = \ \text{No. of incoming edges to } v_i \text{ with label } \ell_1.$$

For $2 \leq j \leq k$, let $V_i[j]$ denote the subset of $\{v_1, v_2, \ldots, v_{i-1}\}$ such that each $v_r \in V_i[j]$ is an in-neighbor of v_i and the label on the directed edge (v_r, v_i) is ℓ_j. Then, we have

$$q_i[j] \ = \ \sum_{v_r \in V_i[j]} q_r[j-1] \, .$$

Since nodes are processed in topologically sorted order, when we need to compute $q_i[j]$ for some j, all the required values would have already been computed.

The pseudo code for the above computation appears as Algorithm 1. We can estimate the running time of the algorithm as follows. Recall that n and m denote the number of nodes and edges in the given DAG H. A topological sort of H can be obtained in $O(m+n)$ time [15]. For any node v_i and any j, $1 \leq j \leq k$, the time used to compute $q_i[j]$ is $O(\text{Indegree}(v_i))$ since only the incoming edges of v_i are used in the computation of $q_i[j]$. Thus, for any j, the total time used to compute the entry $q_i[j]$ for all the nodes of H is $O(\sum_i \text{Indegree}(v_i)) = O(m)$. Consequently, the time used to compute $q_i[k]$ for all the nodes of H is $O(km)$. Hence, the overall running time to compute the quantity $n_P(k, L)$ is $O(km + n)$.

The following theorem summarizes the above discussion.

Theorem 2.1. *Given a DAG $H(V_H, E_H)$, where $|V_H| = n$, $|E_H| = m$, and a label sequence L of length $k \geq 1$, Algorithm 1 finds the number of k-length directed paths in H that are consistent with L in $O(km + n)$ time.*

Algorithm 1: Counting the number of k-length paths in a DAG that are consistent with a given label sequence of length k.

Input : Directed acyclic graph $H(V_H, E_H)$ and a label sequence $L = \langle \ell_1, \ldots, \ell_k \rangle$ of length k.
Output : The value $n_P(k, L)$, i.e., the number of k-length directed paths in H that are consistent with L.

Let $L = \langle \ell_1, \ell_2, \ldots, \ell_k \rangle$ denote the given label sequence of length k. For $1 \leq j \leq k$, let $L_j = \langle \ell_1, \ell_2, \ldots, \ell_j \rangle$ denote the j-length subsequence of L.
Let $\langle v_1, v_2, \ldots, v_n \rangle$ denote a topologically sorted order of the nodes of H.
For all $v, i \in V$, let $q_i[j]$, $1 \leq j \leq k$, denote the number of j-length paths which are consistent with L_j and which end at v_i.
Set $q_1[j] = 0$, for $1 \leq j \leq k$.
for $i = 2, \ldots, k$ **do**
  Set $q_i[1]$ = No. of incoming edges to v_i with label ℓ_1.
end
for $j = 2, \ldots, k$ **do**
  for $i = 2, \ldots, n$ **do**
    Let $V_i[j]$ denote the subset of $\{v_1, \ldots, v_{i-1}\}$ such that each $v_r \in V_i[j]$ is an in-neighbor of v_i and the label of the directed edge (v_r, v_i) is ℓ_j.
    Set $q_i[j] = \sum_{v_r \in V_i[j]} q_r[j-1]$.
  end
end
Return $n_P(k, L) = \sum_{i=1}^{n} q_i[k]$.

3 SCENARIOS OF DISEASE DYNAMICS

A *scenario* S corresponds to the model parameters that generated the cascade graphs, which could include disease or diffusion model parameters, seeding, and agent behavior capturing various responses to the disease spread. In this work, the transmissibility τ determined the infectiousness of the disease in each scenario.

3.1 Interventions

Interventions include pharmaceutical interventions (PIs), such as vaccination, which change node susceptibility and/or infectivity; and non-pharmaceutical interventions (NPIs), such as social distancing, which change node behavior, and as a result, change edges of the contact graph. An intervention is triggered by either time or a threshold, and has a target set. For example, we can apply vaccines to senior people (65+) from the beginning of September; or we can quarantine people who have close contacts with an individual once we confirm that this individual is infected. People in the targeted set may or may not comply.

Vaccination (VAX). This intervention reduces the susceptibility σ of a compliant node by a fraction, which is called the vaccine efficacy (VE). This intervention reduces the probability that the node will be infected by other nodes.

Generic social distancing (GSD). This intervention removes all non-essential activities of a compliant node. The *essential* activities

include home, work and school, while the *nonessential* activities include shop and other. GSD removes each edge in the contact graph that satisfies the following two properties: (i) the edge is incident on compliant nodes and (ii) the edge label has at least one non-essential activity. Since every edge is associated with two activities, one for each end point, we combine the two activities in the following manner to generate a label for each edge: essential–essential (E), non-essential–non-essential (N) and essential–non-essential (M). All type-N edges incident with the compliant nodes and those type-M edges where the non-essential side belongs to a compliant node will be removed from the contact graph.

3.2 Partial observation model

To model low penetration levels and adoption, we define *coverage* κ as the fraction of nodes that are observed in a cascade. We consider a simple partial observation model where a random subset V_{obs} of $\kappa \cdot |V|$ nodes from the contact graph $G(V, E)$ are chosen as the observed nodes. For any cascade $C(V_C, E_C)$, the structural features are extracted from the subgraph induced by $V_{\text{obs}} \cap V_C$. Also, we only consider the boundary nodes corresponding to $V_{\text{obs}} \cap V_C$. When $\kappa = 1$, all nodes in the contact graph are observable, and therefore, all the structural features are computed for the entire cascade.

4 PROBLEM FORMULATIONS

To understand how structural features of the cascade can characterize the scenario that generated it, we consider a learning-based framework. Using well-established techniques to assess parameter significance and interpretability, we will then be able to quantify the importance of each class of features (such as number of infections, motif counts, etc.). Further, this framework will also provide the first steps for incorporating these features in learning tasks like forecasting or predicting the time of peak infection.

We define the *scenario identification* (SI) problem as follows. Let $\mathcal{S} = \{S_1, S_2, \cdots, S_m\}$ denote a set of contagion scenarios and $f^{(T)} : C \to \mathbb{Z}^k$ correspond to the feature vector of a cascade graph observed up to a time horizon T consisting of motif counts, number of nodes, edges, epidemic characteristics, etc. Let $\mathcal{F}^{(T)}$ be the set of all possible feature vectors, which will be till some time horizon. Let $C_\ell = \{(f_1, l_1), (f_2, l_2), \cdots\}$ denote a set of labeled feature vectors (f_i, ℓ_i) where $f_i = f^{(T)}(C)$ is a feature vector corresponding to a cascade graph C for the scenario $\ell_i \in \mathcal{S}$. The objective of the full information horizon T SI problem is to learn a classifier $g : \mathcal{F}^{(T)} \to \mathcal{S}$ that, given a feature vector, classifies it as being generated by a particular scenario. We get different variations by changing T—these correspond to varying amounts of available information. We also consider different classes of features $\mathcal{F}^{(T)}$, e.g., those using epidemic features, or graph structural features, and explore how well such classifiers can be learned. It is possible that (i) the scenarios are not mutually exclusive, and (ii) the same cascade graph can be generated by two different scenarios.

Given this framework, we consider an adversarial setting where the scenarios are chosen in such a manner that the distribution of cascades observed until time-horizon T are indistinguishable by simply observing the case counts or number of infected nodes at every time step $t \le T$.

5 EXPERIMENT DESIGN

Cascade data sets. Our data sets include cascade graphs generated by simulations of SEIR disease spread on synthetic contact networks for two states in USA, namely Tennessee (TN) and Virginia (VA). Structural parameters of these two networks appear in Table 1. The simulation takes a scenario, which specifies the parameters of the disease and interventions, seeds 20 randomly chosen nodes, lets the disease propagate through the contact network for 300 days, and generates cascade data as output. We design four scenarios, with different transmissibility, vaccination, and generic social distancing levels, for each state, and run 1000 replicates per scenario. Thus, each of the TN and VA data sets includes 4000 cascade graphs. We choose the parameter values carefully so that it is not easy to distinguish between scenarios by aggregate measures of the cascade, such as the overall size or the daily difference in cascade size (aka daily infection incidence). Both VAX and GSD interventions are applied to compliant nodes, which are randomly chosen with a probability equal to the specified compliance, at the beginning (day 0) and remain effective until the end (day 300). The vaccine reduces the susceptibility of the node that receives it by 80%. The disease transmissibility τ, vaccine coverage VAX, and generic social distancing compliance GSD for simulations that generate TN cascade data are shown in Table 2. Those for VA cascade data are similar.

Graph	|V|	|E|	Max deg.	Avg deg.	Dia.
TN	6,041,517	62,149,441	461	20.57	14
VA	7,602,717	83,162,927	543	21.88	14

Table 1: General structural information about the synthetic contact networks used to generate the cascade graphs.

Scenario	τ	VAX	GSD
No vax + Low GSD	0.09	None	25%
No vax + High GSD	0.09	None	70%
Vax + Low GSD	0.16^*	50%	25%
Vax + High GSD	0.16^*	50%	$65\%^\dagger$

Table 2: Disease and intervention parameters for different scenarios. *Scenarios on the VA network used $\tau = 0.155$. †Scenario on the VA network used GSD = 60%.

Partial observation. The generated cascades were further sampled based on the coverage κ. We used $\kappa = 0.6, 0.7, \ldots, 1$. In each cascade, a subset of nodes was chosen depending on κ. We considered five such sets of sampling instances for each κ value.

Learning methodology. We considered several machine learning algorithms for this purpose. But we have provided results corresponding to three of them based on superior performance: random forest classifiers (RF), support vector machines (SVM), and logistic regression (LR). We considered both one-versus-one and one-versus-all approaches. We explored different approaches to feature engineering, in particular with respect to the epicurve related features. This was in part to ensure that there was very low likelihood that an epicurve-based approach could meaningfully distinguish

Name	Description
Infection progress (epicurve)	Number of infections that occurred within evenly spaced time periods. The periods used are 1, 5, and 10 days.
Out-degree values	Number of infected nodes in the cascade with out-degrees that fall within equally sized bins. The bin sizes are 1, 2, and 10.
Boundary degree bins	Number of boundary nodes with boundary degrees falling within equally sized periods. The bin sizes used are 1,2 and 10.
Counts of labeled paths of length-1 and 2	Number of occurrences of each length-1 and length-2 labeled motif.
Normalized counts of labeled length-1 and length-2 path counts	Number of occurrences of each labeled path normalized by the total number of paths with the same length.
Unlabeled path counts	Number of occurrences of unlabeled paths of lengths 1, 2, 3, 4.

Table 3: Groups of features of cascade graphs considered in the paper.

different scenarios. The feature groups used in this exercise are shown in Table 3.

Though our data in some sense already represents a true distribution, and thus mitigates much of the risk of overfitting, we split our data into training and evaluation sets. We used stratified sampling, and used 25% of our data as the evaluation set. We selected hyperparameters by stratified 5-fold cross validation within the training set. The hyperparameters we selected were C, the regularization factor for SVM and LR approaches, and the maximum number of features considered by each tree in the RF approach.

Our objective for feature importance studies was to evaluate the importance of groups of features (described in Table 3), not that of an individual feature (such as out degree count for one particular value of out degree). We used ablation analysis and Shapley additive explanations (SHAP) [33, 34]. We note that the SHAP value is a local method which quantifies the importance of a feature for a data instance. The mean absolute SHAP value across data instances is indicative of the global importance of the target feature. Here, since we are evaluating groups of features, we adapt the SHAP method to our purpose in the following manner. In our problem, given a data instance x_j, feature i and scenario S, let $\phi_{x_j,i,S}$ denote the SHAP value of feature i for deciding whether x_j belongs to scenario S or not. To assess the importance of groups of features, we report the distribution of the absolute value of SHAP value across data instances x_j and across features belonging to one group. We take a similar approach with ablation analysis, where we assess the importance of a feature group by comparing the performance of the model trained with all features with the model trained with all features minus the target feature group. We used the `SHAP` [52] estimator in Python [45] and associated packages to do this analysis. Our implementation and analysis code may be found at https://github.com/NSSAC/cascade_analytics_public.

6 RESULTS

Cascade graph structure under different scenarios. The values of various groups of features were computed for each scenario (Table 2), with 100 replicates per scenario. The results are in Figures 2, 3 and 4. Figure 2 shows the distribution of the number of infections per day or the epicurves. Due to space constraints, we show results only for the TN network in Figures 3 and 4. We found similar results for the VA network. These results are included in the full version of this work [24].

As the disease progresses, the epicurves can be distinguished from one another, especially around the peak of the infection spread; pharmaceutical and nonpharmaceutical interventions lead to smaller peaks in the infection curve. We observe a similar trend with other feature groups as well except for the labeled path motifs. In Figure 3, the scenario-specific counts of path frequency, out-degree, boundary degree, and cascade size counts are shown. From this, we can conclude that from the structure of the unlabeled cascade graph, it seems to be hard to distinguish between scenarios at the early stages ($T = 70$) of the cascade (left column), while in the long-term ($T = 300$), the counts are very different and the cascades can be easily distinguished using any feature group (row).

On the other hand, labeled motif counts, shown in Figure 4, exhibit distinct patterns that remain consistent as time progresses. We observe that the proportion of non-`E` edges is less in the high GSD case as compared to the low GSD case when the vaccination aspect is kept the same. This is because in the high GSD case, due to a large number of nodes selected for GSD, and subsequent removal of all their non-`E` edges, their proportion is reduced in the residual graph. However, we also note that for the same level of GSD, the proportion of `E` edges is less in the high VAX case when compared to low VAX case; this is an unexpected outcome of the simulation analysis process. We attribute this to the non-uniform distribution of essential and non-essential edges among nodes. We observed similar trends with longer labeled paths.

Scenario identification under complete observation. The results of the performance of the different machine learning approaches are summarized in Table 4. We first note that models trained with the feature group of infected node counts (epicurve) alone did not do well. This is a reflection of the way the scenarios were chosen (see Figure 2 and the related discussion). However, we observe that the addition of structural features greatly improves the performance of all the algorithms consistently across networks. For LR, the one-versus-all approach performed better, while for SVM, the one-versus-one approach with linear kernel did better. We found that a linear kernel resulted in the best performance for the SVM.

Significance of different network measures. Our results for the TN experiment using SHAP are provided in Figure 5 for the different machine learning algorithms. We observe that in all three algorithms, the labeled path counts play a dominant role in decision-making. Particularly in the case of LR, the labeled edges have very high average absolute SHAP values compared to others, while in the case of SVM, the labeled path counts for longer paths play a more prominent role. In the case of SVM, unlike the other methods, there are several outliers suggesting that individual features corresponding to almost all the groups play a prominent role. The column ablation results in the full version [24] also show very similar results. In summary, we found that the group of features that achieves the best performance varies depending on the model, but

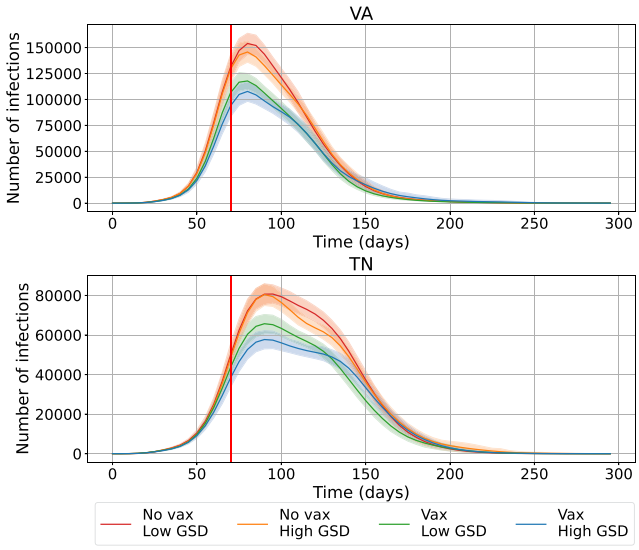


Figure 2: Distribution of the number of infections per day on the TN and VA networks over 400 cascades. Each colored line represents the mean value of all the cascades of the corresponding scenario, and the shaded area around the line is the confidence interval of the data. The red vertical line marks day 70 on the time horizon T.

	TN		VA	
Model	All	Epi-only	All	Epi-only
SVM	0.82 ± 0.032	0.30 ± 0.040	0.87 ± 0.037	0.25 ± 0.017
RF	0.80 ± 0.041	0.26 ± 0.041	0.82 ± 0.029	0.29 ± 0.034
LR	0.79 ± 0.020	0.26 ± 0.024	0.80 ± 0.036	0.25 ± 0.038

Table 4: The accuracy values of the different models on the test set and their standard deviations derived by repeating the training and evaluation steps 10 times.

all the most accurate models included either the labeled edge count (path length 1) or the labeled 2-length path counts.

Coverage Effects. We looked at the effect of coverage, or how much of the network was necessary to observe in order to derive usable information under the observation model defined in Section 5. The proportion of edge motifs of each type by scenario remained relatively stable. These results, available in the full version [24] are also reflected in the model performance. The performance for the TN network is shown in Figure 7a. We observed minimal differences between model accuracy at 60% coverage as opposed to 90% coverage, with SVM performing the best. The epicurve-only-based models did not perform significantly above random guessing and therefore were unaffected by the coverage. We observed similar patterns for the VA network (see Figure 7b). That there was so little change with less coverage is encouraging, as it suggests that even partial efforts at contact tracing may yield useful insights.

Time horizon. We investigated whether the predictability was impacted by the time at which the observations were taken. That is, how early in a cascade can we predict the scenario? As mentioned

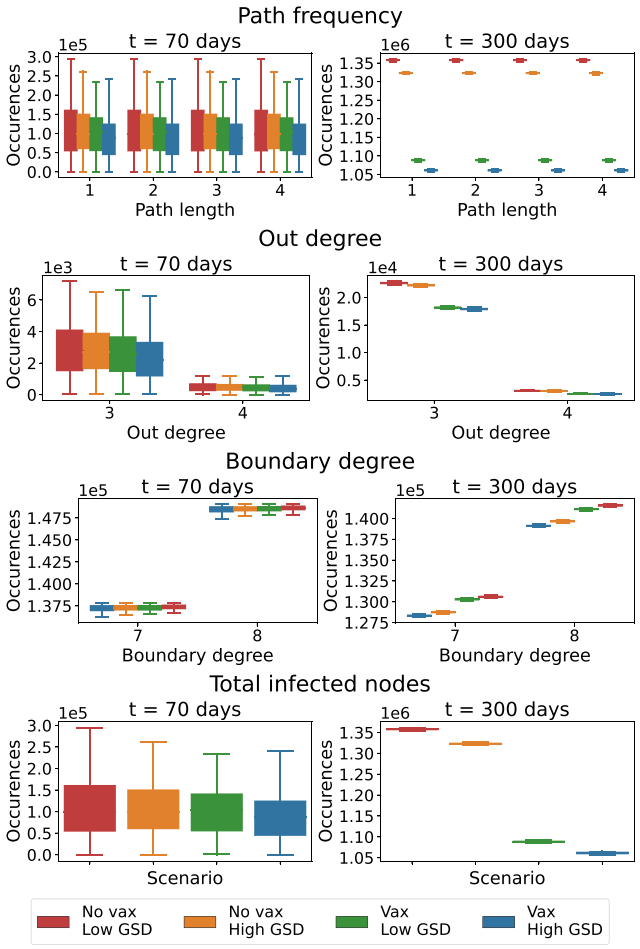


Figure 3: Network and node-level feature distributions at two points in time on TN cascades. Within each figure, a box represents the distribution over 100 cascades. The distributions on the left column are measured at time $t = 70$, and the ones on the right are measured at time $t = 300$. The plots show the frequencies of unlabeled paths of different lengths, frequencies of nodes with the corresponding out-degrees (i.e., star motif counts), frequencies of nodes with the corresponding boundary degrees, and the total number of infections at time t. Due to the wide range of their values, the out-degree and boundary degree plots show distributions for two elements only. However, more results in the full version [24] demonstrate that the same trend can be seen across other values of boundary degrees and out-degrees. We provide corresponding visualizations for the VA network in the full version [24].

in the discussion for Figure 2, the likely trajectories of infections overlap early in the cascade. However, it is similarly clear that there is a point at which scenarios diverge. In Figure 3, the total number of infected nodes at $t = 70$ is not informative as to the scenario. However, at $t = 300$ (the end of the cascade), there is a clear separation. A similar pattern occurs with path frequency,

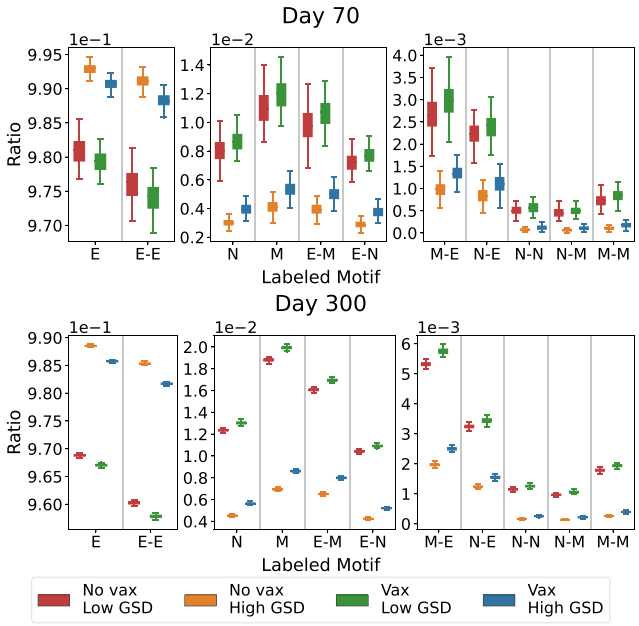


Figure 4: The ratio of the occurrences of every labeled motif with respect to the total number of labeled motifs with the same length. Values were measured at two different time points on TN cascades. Each box represents value distributions over 100 cascades. Three figures were used for each time point as the scales differ significantly. Corresponding plots for the VA dataset are presented in the full version [24].

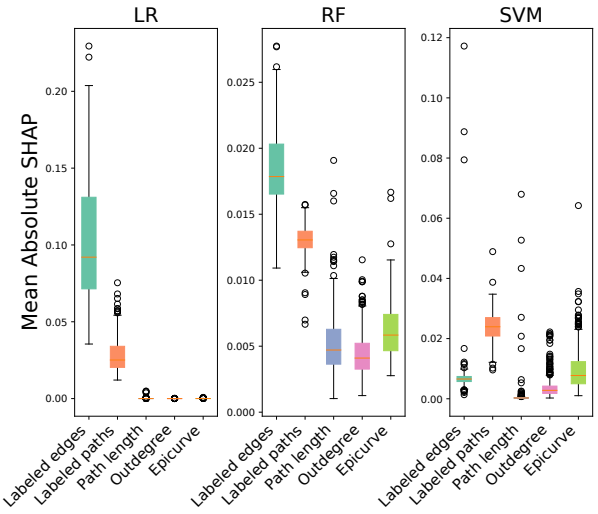


Figure 5: The absolute SHAP value for each model, averaged over each set of features considered. The results are for cascades over the TN network. Distributions were computed over the entire training data set.

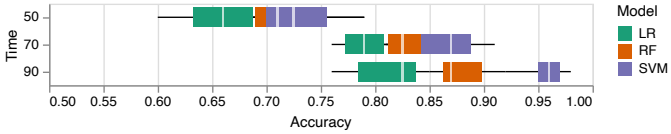


Figure 6: The accuracy of models trained on Epicurve features. Bounds were generated by 10 repetitions of train/test splits.

though out degree and boundary degree remain relatively similar at both time points. Labeled motifs (Figure 4) similarly vary in how informative they are depending on the time step. While they overlap somewhat at $t = 70$, they are totally separate at $t = 300$.

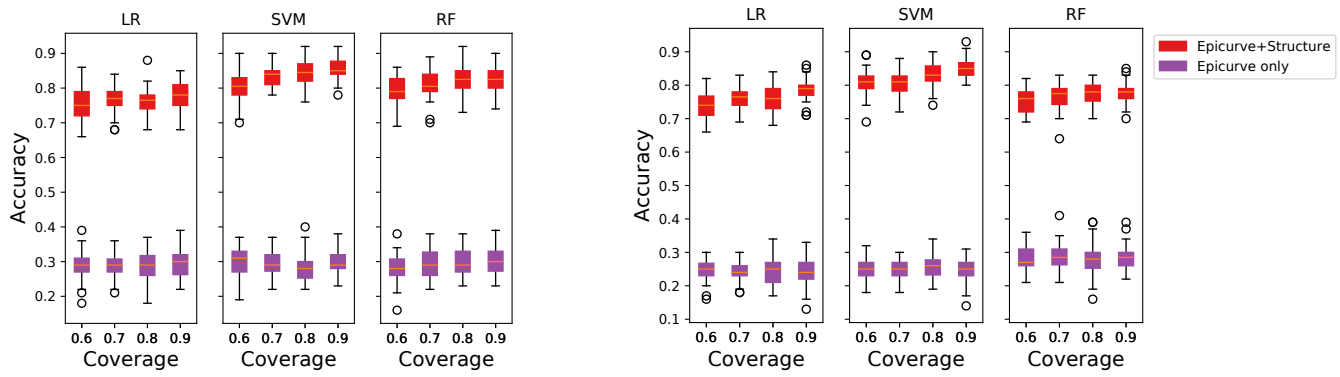
To investigate how sensitive our learning models were to the time of observation, we reran the training and testing using data derived from observations up to $t = 50, 70, 90$. The results of these experiments are shown in Figure 6. We observed a moderate decrease in accuracy–the models at time 50 had lower accuracy than those at time 70 and 90. At the same time, even at $t = 50$, the performance is not particularly bad, with a range between 0.6 and 0.8 depending on the model class. We have results in the full version [24] to show how the importance of parameters varies over time. We see that for SVM, some out degree features start becoming important as T increases, while for LR and RF, the labeled paths are consistently the dominant feature group.

7 DISCUSSION AND RELATED WORK

The COVID-19 pandemic has underlined the importance of considering the characteristics of interactions (e.g., interactions in assisted living facilities, super-spreader events, etc.) [4]. However, recent studies (see e.g., [3, 27–29]) that have modeled contact tracing do not assume the availability of such interaction-level information. Emerging technologies are enabling us to collect richer data on individual-level interactions and study their effect on disease spread. Our work is motivated by the need to investigate how such information can be utilized to better understand a contagion scenario, and therefore, inform public policy.

Finding and enumerating motifs in graphs is an important topic in areas such as computational biology and graph mining (see e.g., [26, 43]). Other researchers have presented techniques for finding motifs in temporal graphs [10, 42]. Many of the proposed methods are for unlabeled networks. Some researchers have studied problems related to identifying subgraphs in labeled networks. For example, a scalable framework for finding dense subgraphs that contain specified labeled motifs (i.e., smaller subgraphs) is presented in [50]. Our work does not use dense subgraphs; it relies on simple labeled directed paths. Another study [36] focuses on finding motifs in undirected graphs with node labels. Our work uses counts of subgraphs determined by edge labels.

Machine learning approaches are being used to understand the phase space of complex models. Lamperti et al. [30] used gradient boosted trees for calibrating a complex agent-based model and subsequently performing parameter importance analysis. Fox et al. [21] provide an overview of methods by which simulation systems can be coupled with machine learning based approaches to understand complex systems. Angione et al. [5] analyze and evaluate a number of machine learning surrogates for an agent-based model.



(a) TN network

(b) VA network.

Figure 7: Performance of models with access to structural features vs. Epicurve only models. Bounds were computed over 5 different random selections of nodes, and over 10 train/test repetitions for each of the random selections.

A number of papers have presented methods to analyze observed cascades and determine various aspects of an epidemic process. For example, Lokhov [32] considers the problem of identifying the disease parameters of a spreading epidemic from partially observed cascades. Raisi et al. [47] present optimization and deep learning based methods for predicting the future course of a disease. Shah & Zaman [51] and Zhu & Ying [57] address the problem of identifying the initially infected nodes under specific propagation models such as SIR. Problems related to the inference of influence functions at the nodes of a network are addressed in [39]. Mishra et al. [37] present an approach based on maximum likelihood estimate (MLE) to reconstruct an epidemic cascade from partial observations. There are many differences between our work and the ones mentioned above. We consider more complex scenarios with interventions and our goal is to characterize a given cascade in terms of measures such as compliance to social distancing and efficacy of vaccines.

8 CONCLUSION

In order to characterize the utility of potential data from real-world cascade tracing and to understand how structural graph parameters influence network-based disease simulations, we introduced the scenario identification problem. We performed a series of experiments using realistic networks and cascades. These experiments demonstrate that even in a semi-adversarial environment, effective classification is possible if simple structural measures are available.

This work can be considered as a first step towards understanding the importance of subgraph features of cascade graphs. For the intervention scenarios considered in this work, we observe that the types of interactions between pairs or chains of individuals (i.e., labeled path counts) are by far the most significant set of features that can be used to characterize the scenarios. However, this could be scenario dependent. For example, in a more dynamic intervention scenario such as contact tracing followed by quarantining, it is possible that other features are prominent. Also, in this work, we have considered a simple SEIR model. Therefore, more complex scenarios may require characterizations through more complex subgraphs of cascade graphs. The general framework is relevant for other kinds of complex contagion phenomena, which can be modeled as graph dynamical systems [8, 38].

9 ACKNOWLEDGMENTS

We thank the members of the Network Systems Science and Advanced Computing (NSSAC) Division and UVA Research computing. This work was supported in part by the following grants: University of Virginia Strategic Investment Fund (Award Number SIF160), National Science Foundation Grants CCF-1918656 (Expeditions), OAC-1916805 (CINES), IIS-1955797, VDH Grant PV-BII VDH COVID-19 Modeling Program VDH-21-501-0135, DTRA subcontract/ARA S-D00189-15-TO-01-UVA, NIH 2R01GM109718-07, CDC MIND cooperative agreement U01CK000589, USDA-NIFA Awards No. 2021-67021-35344 (AgAID AI Institute) and No. 2019-67021-29933 (Network Models of Food Systems and their Application to Invasive Species Spread).

REFERENCES

[1] Eytan Adar and Lada A Adamic. Tracking information epidemics in blogspace. In *The 2005 IEEE/WIC/ACM International Conference on Web Intelligence (WI'05)*, pages 207–214. IEEE, 2005.

[2] Nadeem Ahmed, Regio A Michelin, Wanli Xue, Sushmita Ruj, Robert Malaney, Salil S Kanhere, Aruna Seneviratne, Wen Hu, Helge Janicke, and Sanjay K Jha. A Survey of COVID-19 Contact Tracing Apps. *IEEE access*, 8:134577–134601, 2020.

[3] Alberto Aleta, David Martin-Corral, Ana Pastore y Piontti, Marco Ajelli, Maria Litvinova, Matteo Chinazzi, Natalie E Dean, M Elizabeth Halloran, Ira M Longini Jr, Stefano Merler, et al. Modelling the impact of testing, contact tracing and household quarantine on second waves of COVID-19. *Nature Human Behaviour*, 4(9):964–971, 2020.

[4] Uzoma Rita Alo, Friday Onwe Nkwo, Henry Friday Nweke, Ifeanyi Isaiah Achi, and Henry Anayo Okemiri. Non-Pharmaceutical Interventions against COVID-19 Pandemic: Review of Contact Tracing and Social Distancing Technologies, Protocols, Apps, Security and Open Research Directions. *Sensors*, 22(1):280, 2022.

[5] Claudio Angione, Eric Silverman, and Elisabeth Yaneske. Using machine learning as a surrogate model for agent-based simulations. *PloS one*, 17(2):e0263150, 2022.

[6] Andrew Anglemyer, Theresa HM Moore, Lisa Parker, Timothy Chambers, Alice Grady, Kellia Chiu, Matthew Parry, Magdalena Wilczynska, Ella Flemyng, and Lisa Bero. Digital contact tracing technologies in epidemics: a rapid review. *Cochrane Database of Systematic Reviews*, 8, 2020.

[7] Eytan Bakshy, Jake M Hofman, Winter A Mason, and Duncan J Watts. Everyone's an influencer: quantifying influence on twitter. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 65–74, 2011.

[8] C. L. Barrett, S. Eubank, and M. V. Marathe. An interaction based approach to computational epidemics. In *AAAI' 08: Proceedings of the Annual Conference of AAAI*, Chicago USA, 2008. AAAI Press.

[9] Jamie Bedson, Laura A Skrip, Danielle Pedi, Sharon Abramowitz, Simone Carter, Mohamed F Jalloh, Sebastian Funk, Nina Gobat, Tamara Giles-Vernick, Gerardo Chowell, et al. A review and agenda for integrated disease models including social and behavioural factors. *Nature human behaviour*, 5(7):834–846, 2021.

[10] Austin R Benson, David F Gleich, and Jure Leskovec. Higher-order organization of complex networks. *Science*, 353(6295):163–166, 2016.

[11] Caroline Buckee, Abdisalan Noor, and Lisa Sattenspiel. Thinking clearly about social aspects of infectious disease transmission. *Nature*, 595(7866):205–213, 2021.

[12] Justin Chan, Dean Foster, Shyam Gollakota, Eric Horvitz, Joseph Jaeger, Sham Kakade, Tadayoshi Kohno, John Langford, Jonathan Larson, Puneet Sharma, et al. PACT: Privacy Sensitive Protocols and Mechanisms for Mobile Contact Tracing. *arXiv preprint arXiv:2004.03544*, 2020.

[13] Jiangzhuo Chen, Stefan Hoops, Achla Marathe, Henning Mortveit, Bryan Lewis, Srinivasan Venkatramanan, Arash Haddadan, Parantapa Bhattacharya, Abhijin Adiga, Anil Vullikanti, et al. Effective Social Network-Based Allocation of COVID-19 Vaccines. *Proceedings of the KDD Health Day*, 2022.

[14] Vittoria Colizza, Eva Grill, Rafael Mikolajczyk, Ciro Cattuto, Adam Kucharski, Steven Riley, Michelle Kendall, Katrina Lythgoe, David Bonsall, Chris Wymant, Lucie Abeler-Dörner, Luca Ferretti, and Christophe Fraser. Time to evaluate COVID-19 contact-tracing apps. *Nature Medicine*, 27:361–362, 2021.

[15] Thomas H. Cormen, Charles Eric Leiserson, Ronald L Rivest, and Clifford Stein. *Introduction to Algorithms*. MIT Press and McGraw-Hill, Cambridge, MA, Second edition, 2009.

[16] Michael D Dzandu. Antecedent, behaviour, and consequence (a-b-c) of deploying the contact tracing app in response to COVID-19: Evidence from Europe. *Technological Forecasting and Social Change*, 187:122217, 2023.

[17] D. Easley and J. Kleinberg. *Networks, Crowds and Markets: Reasoning About a Highly Connected World*. Cambridge University Press, New York, NY, 2010.

[18] Stephen Eubank, Hasan Guclu, VS Anil Kumar, Madhav V Marathe, Aravind Srinivasan, Zoltan Toroczkai, and Nan Wang. Modelling disease outbreaks in realistic urban social networks. *Nature*, 429(6988):180–184, 2004.

[19] Hui Fang, Barney P Caton, Nicholas C Manoukis, and Godshen R Pallipparambil. Simulation-based evaluation of two insect trapping grids for delimitation surveys. *Scientific Reports*, 12(1):1–12, 2022.

[20] Luca Ferretti, Chris Wymant, Michelle Kendall, Lele Zhao, Anel Nurtay, Lucie Abeler-Dörner, Michael Parker, David Bonsall, and Christophe Fraser. Quantifying SARS-CoV-2 transmission suggests epidemic control with digital contact tracing. *Science*, 368(6491):eabb6936, 2020.

[21] Geoffrey Fox, James A Glazier, JCS Kadupitiya, Vikram Jadhao, Minje Kim, Judy Qiu, James P Sluka, Endre Somogyi, Madhav Marathe, Abhijin Adiga, et al. Learning Everywhere: Pervasive Machine Learning for Effective High-Performance Computation. In *2019 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*, pages 422–429. IEEE, 2019.

[22] Sebastian Funk, Marcel Salathé, and Vincent AA Jansen. Modelling the influence of human behaviour on the spread of infectious diseases: a review. *Journal of the Royal Society Interface*, 7(50):1247–1256, 2010.

[23] George Grekousis and Ye Liu. Digital contact tracing, community uptake, and proximity awareness technology to fight COVID-19: a systematic review. *Sustainable cities and society*, 71:102995, 2021.

[24] Galen Harrison, Amro Alabsi Aljundi, Jiangzhuo Chen, S S Ravi, Anil K Vullikanti, Madhav V Marathe, and Abhijin Adiga. Full version: Identifying complicated contagion scenarios from cascade data. https://biocomplexity.virginia.edu/our-research/institute-publications/identifying-complex-disease-scenarios-cascade-data, 2023.

[25] Stefan Hoops, Jiangzhuo Chen, Abhijin Adiga, Bryan Lewis, Henning Mortveit, Hannah Baek, Mandy Wilson, Dawen Xie, Samarth Swarup, Srinivasan Venkatramanan, et al. High Performance Agent-Based Modeling to Study Realistic Contact Tracing Protocols. In *2021 Winter Simulation Conference (WSC)*, pages 1–12. IEEE, 2021.

[26] Ali Jazayeri and Christopher C Yang. Motif discovery algorithms in static and temporal networks: A survey. *Journal of Complex Networks*, 8(4):1–38, 12 2020.

[27] Cliff C Kerr, Robyn M Stuart, Dina Mistry, Romesh G Abeysuriya, Katherine Rosenfeld, Gregory R Hart, Rafael C Núñez, Jamie A Cohen, Prashanth Selvaraj, Brittany Hagedorn, et al. Covasim: An agent-based model of COVID-19 dynamics and interventions. *PLOS Computational Biology*, 17(7):e1009149, 2021.

[28] Mirjam E Kretzschmar, Ganna Rozhnova, Martin CJ Bootsma, Michiel van Boven, Janneke HHM van de Wijgert, and Marc JM Bonten. Impact of delays on effectiveness of contact tracing strategies for COVID-19: a modelling study. *The Lancet Public Health*, 5(8):e452–e459, 2020.

[29] Adam J Kucharski, Petra Klepac, Andrew JK Conlan, Stephen M Kissler, Maria L Tang, Hannah Fry, Julia R Gog, W John Edmunds, Jon C Emery, Graham Medley, et al. Effectiveness of isolation, testing, contact tracing, and physical distancing on reducing transmission of SARS-CoV-2 in different settings: a mathematical modelling study. *The Lancet Infectious Diseases*, 20(10):1151–1160, 2020.

[30] Francesco Lamperti, Andrea Roventini, and Amir Sani. Agent-based model calibration using machine learning surrogates. *Journal of Economic Dynamics and Control*, 90:366–389, 2018.

[31] You Li, Harry Campbell, Durga Kulkarni, Alice Harpur, Madhurima Nundy, Xin Wang, and Harish Nair. The temporal association of introducing and lifting non-pharmaceutical interventions with the time-varying reproduction number (R) of SARS-CoV-2: a modelling study across 131 countries. *The Lancet Infectious Diseases*, 21(2):193–202, 2021.

[32] A. Lokhov. Reconstructing Parameters of Spreading Models from Partial Observations. In *Advances in Neural Information Processing Systems*, pages 3467–3475, 2016.

[33] Scott M Lundberg, Gabriel Erion, Hugh Chen, Alex DeGrave, Jordan M Prutkin, Bala Nair, Ronit Katz, Jonathan Himmelfarb, Nisha Bansal, and Su-In Lee. From local explanations to global understanding with explainable AI for trees. *Nature machine intelligence*, 2(1):56–67, 2020.

[34] Scott M Lundberg and Su-In Lee. A Unified Approach to Interpreting Model Predictions. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[35] Madhav Marathe and Anil Kumar S Vullikanti. Computational epidemiology. *Communications of the ACM*, 56(7):88–96, 2013.

[36] Giovanni Micale, Rosalba Giugno, Alfredo Ferro, Misael Mongiovì, Dennis Shasha, and Alfredo Pulvirenti. Fast analytical methods for finding significant labeled graph motifs. *Data Mining and Knowledge Discovery*, 32:504–531, 2018.

[37] Ritwick Mishra, Jack Heavey, Gursharn Kaur, Abhijin Adiga, and Anil Vullikanti. Reconstructing an Epidemic Outbreak Using Steiner Connectivity. To appear in Proc. AAAI, 2023.

[38] Henning Mortveit and Christian Reidys. *An introduction to sequential dynamical systems*. Springer Science & Business Media, 2007.

[39] H. Narasimhan, D. C. Parkes, and Y. Singer. Learnability of Influence in Networks. In *Advances in Neural Information Processing Systems*, pages 3186–3194, 2015.

[40] Mark EJ Newman. The Structure and Function of Complex Networks. *SIAM review*, 45(2):167–256, 2003.

[41] Maria Nöremark and Stefan Widgren. EpiContactTrace: an R-package for contact tracing during livestock disease outbreaks and for risk-based surveillance. *BMC veterinary research*, 10(1):1–9, 2014.

[42] Ashwin Paranjape, Austin R Benson, and Jure Leskovec. Motifs in Temporal Networks. In *Proceedings of the tenth ACM international conference on web search and data mining*, pages 601–610, 2017.

[43] Sabyasachi Patra and Anjali Mohapatra. Review of tools and algorithms for network motif discovery in biological networks. *IET systems biology*, 14(4):171–189, 2020.

[44] Ashish Viswanath Prakash and Saini Das. Explaining citizens' resistance to use digital contact tracing apps: A mixed-methods study. *International Journal of Information Management*, 63:102468, 2022.

[45] https://docs.python.org/3/library/. Access date: Feb. 1, 2023.

[46] Zirou Qiu, Baltazar Espinoza, Vitor V Vasconcelos, Chen Chen, Sara M Constantino, Stefani A Crabtree, Luojun Yang, Anil Vullikanti, Jiangzhuo Chen, Jörgen Weibull, et al. Understanding the coevolution of mask wearing and epidemics: A network perspective. *Proceedings of the National Academy of Sciences*, 119(26):e2123355119, 2022.

[47] Maziar Raissi, Niloofar Ramezani, and Padmanabhan Seshaiyer. On parameter estimation approaches for predicting disease transmission through optimization, deep learning and statistical inference methods. Letters in Biomathematics, 2019.

[48] Sofya Raskhodnikova and Adam Smith. Differentially Private Analysis of Graphs. In *Encyclopedia of Algorithms*, pages 543–547. Springer, 2016.

[49] Pablo Rodríguez, Santiago Graña, Eva Elisa Alvarez-León, Manuela Battaglini, Francisco Javier Darias, Miguel A Hernán, Raquel López, Paloma Llaneza, Maria Cristina Martín, RadarCovidPilot Group, et al. A population-based controlled experiment assessing the epidemiological impact of digital contact tracing. *Nature communications*, 12(1):587, 2021.

[50] Ahmet Erdem Sarıyüce. Motif-driven Dense Subgraph Discovery in Directed and Labeled Networks. In *Proceedings of the Web Conference 2021*, pages 379–390, 2021.

[51] D. Shah and T. Zaman. Rumors in a Network: Who's the Culprit? *IEEE Trans. Information Theory*, 57(8):5163–5181, 2011.

[52] https://shap.readthedocs.io/en/latest/. Access date: Feb. 1, 2023.

[53] Damyanka Tsvyatkova, Jim Buckley, Sarah Beecham, Muslim Chochlov, Ian R O'Keeffe, Abdul Razzaq, Kaavya Rekanar, Ita Richardson, Thomas Welsh, Cristiano Storni, et al. Digital Contact Tracing Apps for COVID-19: Development of a Citizen-Centered Evaluation Framework. *JMIR mHealth and uHealth*, 10(3):e30691, 2022.

[54] Frederik Verelst, Lander Willem, and Philippe Beutels. Behavioural change models for infectious disease transmission: a systematic review (2010–2015). *Journal of The Royal Society Interface*, 13(125):20160820, 2016.

[55] Gerald Wilmink, Ilyssa Summer, David Marsyla, Subhashree Sukhu, Jeffrey Grote, Gregory Zobel, Howard Fillit, Satish Movva, et al. Real-Time Digital Contact Tracing: Development of a System to Control COVID-19 Outbreaks in Nursing Homes and Long-Term Care Facilities. *JMIR public health and surveillance*, 6(3):e20828, 2020.

[56] Chris Wymant, Luca Ferretti, Daphne Tsallis, Marcos Charalambides, Lucie Abeler-Dörner, David Bonsall, Robert Hinch, Michelle Kendall, Luke Milsom, Matthew Ayres, et al. The epidemiological impact of the NHS COVID-19 app. *Nature*, 594(7863):408–412, 2021.

[57] K. Zhu and L. Ying. Information Source Detection in the SIR Model: A Sample-Path-Based Approach. *IEEE/ACM Transactions on Networking*, 24(1):408–421, 2014.