Deep learning of stochastic contagion dynamics on complex networks

Charles Murphy

Universite Laval

Edward Laurence

Universite Laval

Antoine Allard (✉ antoine.allard@phy.ulaval.ca)

Universite Laval <https://orcid.org/0000-0002-8208-9920>

Article

Keywords: deep learning, contagion dynamics, complex networks

Posted Date: July 1st, 2020

DOI: <https://doi.org/10.21203/rs.3.rs-36564/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Version of Record: A version of this preprint was published at Nature Communications on August 5th, 2021. See the published version at <https://doi.org/10.1038/s41467-021-24732-2>.

Deep learning of stochastic contagion dynamics on complex networks

Charles Murphy, Edward Laurence, and Antoine Allard

Département de Physique, de Génie Physique, et d'Optique,

Université Laval, Québec (Québec), Canada G1V 0A6 and

Centre interdisciplinaire en modélisation mathématique,

Université Laval, Québec (Québec), Canada G1V 0A6

(Dated: June 19, 2020)

Forecasting the evolution of contagion dynamics is still an open problem to which mechanistic models only offer a partial answer. To remain mathematically and/or computationally tractable, these models must rely on simplifying assumptions, thereby limiting the quantitative accuracy of their predictions and the complexity of the dynamics they can model. Here, we propose a complementary approach based on deep learning where the effective local mechanisms governing a dynamic are learned automatically from time series data. Our graph neural network architecture makes very few assumptions about the dynamics, and we demonstrate its accuracy using stochastic contagion dynamics of increasing complexity on static and temporal networks. By allowing simulations on arbitrary network structures, our approach makes it possible to explore the properties of the learned dynamics beyond the training data. Our results demonstrate how deep learning offers a new and complementary perspective to build effective models of contagion dynamics on networks.

Our capacity to prevent or contain outbreaks of infectious diseases is directly linked to our ability to accurately model contagion dynamics. Since the seminal work of Kermack and McKendrick almost a century ago [1], a variety of models incorporating ever more sophisticated contagion mechanisms have been proposed, studied and used [2–5]. These mechanistic models have provided invaluable insights about how infectious diseases spread, and have thereby contributed to the design of better public health policies. However, several challenges remain unresolved, which call for contributions from new modeling approaches [6–8].

For instance, many complex contagion processes involve the nontrivial interaction of several pathogens [9–12], and some social contagion phenomena, like the spread of misinformation, require to go beyond pairwise interactions between individuals [13–15]. Also, while qualitatively informative, the forecasts of most mechanistic models lack quantitative accuracy. Indeed, most models are constructed from a handful of mechanisms which can hardly reproduce the intricacies of real complex contagion dynamics. One approach to these challenges is to complexify the models by adding more detailed and refined mechanisms. However, mechanistic models become rapidly intractable as new mechanisms are added. Moreover, more complex models require the specification of a larger number of parameters whose values can be difficult to infer from limited data.

On another front, the range of applications of deep learning has grown remarkably fast in recent years [16], from computer vision [17] to natural language processing [18] and time series forecasting [19]. More specifically, deep learning is now being applied to problems related to Network Science using tools like Graph Neural Networks (GNN) which have been designed to handle arbitrarily structured data [20–22]. Recent work showed great promise for applications in community detection and link prediction [23, 24], in the prediction of dynamical observables [25], in network reconstruction [26] and the detection of structural perturbations [27], as well as in the context of discovering new materials and drugs [28, 29]. In general, GNNs have been shown to adequately handle in-

tricate tasks, making them prime candidates to tackle several challenges of contagion dynamics modeling.

Here, we demonstrate how deep learning can be used to build effective models of stochastic contagion dynamics taking place on complex networks. Instead of constructing a model by specifying the mechanisms driving the dynamics, we consider learning these mechanisms directly from data. We start by posing the machine learning problem explicitly and propose a deep GNN architecture with a reliable protocol to train it. We demonstrate the validity of our approach using various dynamics on networks with increasing complexity on static and temporal networks. Finally, we show how our approach can provide predictions for previously unseen network structures, therefore allowing the exploration of the properties of the learned dynamics beyond the training data.

Our approach assumes that a hidden stochastic contagion dynamics takes place on a network $G = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of nodes and $\mathcal{E}$ is the set of links. As this hidden dynamical process evolves over time, it generates a multivariate time series $X = (X(t) : t \in [0, T])$, where $X(t)$ is the global state of the system at time t . This global state $X(t) = (x_i(t) : i \in \mathcal{V})$ consists of the state of every node i at this time, noted $x_i(t)$. We also denote the finite set of the possible node states by Ω , such that $x_i(t) \in \Omega$. We assume that this dynamical process can be entirely defined by its *local transition probabilities* (LTPs), denoted $p_{\sigma}^{\mu \rightarrow \nu}$. These correspond to the probability that a node of degree k in state μ at time t transitions to state ν at time $t + \Delta t$ given the states of its first neighbors σ , where $\mu, \nu \in \Omega$ and $\sigma \in \Omega^k$. By doing so, we assume that the hidden process is stationary, Markovian and discrete both in time and in terms of its available global states. Note that generalizing our approach to non-Markovian and/or continuous-state dynamics is straightforward; we choose to limit the presentation of our approach to these more restricting assumptions for the sake of clarity and conciseness.

We consider a trainable GNN model composed of a set of free and tunable parameters Θ that takes as inputs the current global state of the system, $X(t)$, and the structure of the

network at time t (e.g. its adjacency list), and outputs the predicted LTPs, denoted $\hat{p}_{\sigma}^{\mu \rightarrow \nu}(\Theta)$. Our objective is then to learn a set of parameters Θ^* such that

$$\hat{p}_{\sigma}^{\mu \rightarrow \nu}(\Theta^*) \approx p_{\sigma}^{\mu \rightarrow \nu}, \quad (1)$$

for all transitions $\mu, \nu \in \Omega$ and all possible neighborhoods σ . Note that there exist other techniques that can also perform this task. For instance, Bayesian approaches have been used for the parameters inference of simple and complex contagion under the assumption of a predefined model [30, 31]. In general, these approaches work well on synthetic data generated by the predefined model, in some cases even without the need of likelihood computations [31]. However, these simple models are only coarse representations of real systems. A more general approach is then to infer the complete Markov chains directly [32], where each LTP is considered as a parameter of the model. We adopt a similar strategy in that we use GNNs to parametrize these Markov chains with an underlying network structure. With their easily adaptable architecture, GNNs have the advantage of combining their parameters hierarchically to compute the LTPs as well as being universal estimators [33, 34], meaning that they can in principle estimate any set of LTPs. Furthermore, their parameters estimation is scalable and efficiently performed with stochastic gradient descent [35, 36].

In practice, the objective described by Eq. (1) must be encoded into a loss function, denoted $\mathcal{L}(\Theta)$. An appropriate choice is

$$\mathcal{L}(\Theta) = \sum_{k \in \mathcal{K}} \sum_{\mu \in \Omega} \sum_{\sigma \in \Omega^k} \frac{1}{Z} \left[- \sum_{\nu \in \Omega} p_{\sigma}^{\mu \rightarrow \nu} \log \hat{p}_{\sigma}^{\mu \rightarrow \nu} \right] \quad (2)$$

where $\mathcal{K}$ is the set of degree classes, Z is the total number of possible input configurations (μ, σ) , and the term between brackets corresponds to the cross entropy between the LTPs of the underlying process (the “ground truth”) and the LTPs predicted by the GNN, $\hat{p}_{\sigma}^{\mu \rightarrow \nu}$. Equation (2) quantifies the information difference [37] between the original LTPs and the ones predicted by the GNN, averaged over a uniform distribution of all possible inputs (μ, σ) .

Two obstacles prevent the exact evaluation of Eq. (2) in realistic settings. First, the LTPs $p_{\sigma}^{\mu \rightarrow \nu}$ that we want the model to predict—the “ground truth”—are a priori unknown. Second, finite datasets only explore a finite fraction of the configuration space. We propose to resolve both of these issues by approximating Eq. (2) from the dataset X directly. To do so, we consider the following approximate loss,

$$\mathcal{L}(\Theta) \simeq \sum_{t \in \mathcal{T}'} \sum_{i \in \mathcal{V}'(t)} \frac{\omega_i(t)}{|\mathcal{T}'| |\mathcal{V}'(t)|} \left[- \log \hat{p}_{x_{\mathcal{N}_i}(t)}^{x_i(t) \rightarrow x_i(t+\Delta t)} \right], \quad (3)$$

where $\omega_i(t)$ is a weight assigned to node i at time t , $x_{\mathcal{N}_i}(t)$ is the state of the neighbors of node i at time t , and where $\mathcal{V}' \subseteq \mathcal{V}$ and $\mathcal{T}' \subseteq [0, T]$ are subsets of the training dataset. Note that the complements of these subsets can also be used to validate the model during training (see Supplementary Material).

Two approximations are at play in Eq. (3). First, the first three sums of Eq. (2) are approximated using a sampling

FIG. 1. (color online) **Validation of the predictions of GNNs trained on a Barabási-Albert random network (BA) [38] for the (A) simple and (B) complex contagion dynamics.** The solid lines correspond to the LTPs of the dynamics used to generate the training data (labeled GT for “ground truth”), and the dashed lines show the predictions of the GNNs. Markers correspond to the maximum likelihood estimation (MLE) of the LTPs computed from the training dataset. LTPs corresponding to transmission events are shown in blue ($S \rightarrow I$), and the ones related to recovery events are shown in red ($I \rightarrow S$). Lines and markers were obtained by averaging the LTPs over every σ corresponding to a same value of ℓ . The standard deviation around these average values is shown using a colored area around the lines (typically narrower than the width of the lines) and using vertical bars for the markers. The training datasets were generated using a BA network composed of $|\mathcal{V}| = 1000$ nodes and with average degree $\langle k \rangle = 4$, and we used the parameters $(\tau, \gamma) = (0.04, 0.08)$ and $(\eta, \gamma) = (8, 0.06)$ for the simple and complex contagion dynamics, respectively. All technical details related to training are provided in Supplementary Material.

scheme where the weights $\omega_i(t)$ rebalance the importance to “overrepresented” inputs, i.e. inputs (μ, σ) whose frequency of occurrence in X , noted $\rho(\mu, \sigma)$, is high. Choosing $\omega_i(t) = \rho(x_i(t), x_{\mathcal{N}_i}(t))^{-\lambda}$ effectively evaluates Eq. (3) with a relaxed version of importance sampling [39], which reduces the importance of frequent inputs and increases that of the rare ones. This procedure is identical to standard importance sampling when $\lambda = 1$. Second, the average over ν is replaced by a Monte Carlo approximation since the “ground truth” LTPs, $p_{\sigma}^{\mu \rightarrow \nu}$, are a priori unknown. Note that, for any specific input (μ, σ) , this approximation converges to its expected value only if the corresponding number of transitions found in the subsets of the training dataset is sufficiently large. This second approximation will therefore be necessarily poor for rare inputs, and we set $\lambda < 1$ to limit their influence on the quality of the training. To isolate the effect of this second approximation, we consider a *semi-exact* training scheme where only the first three sums of Eq. (2) are approximated using the importance sampling scheme, and the sum on ν is computed using the “ground truth” LTPs—identically to Eq. (2). The models trained under the semi-exact scheme, denoted GNN*, represent a best case scenario of the models trained using Eq. (3), and will therefore allow us to assess the impact of the dataset quality on the accuracy of the predictions.

We propose the GNN architecture detailed in Supplementary Material. In a nutshell, the GNN is a nonlinear function $f(x_i(t), x_{\mathcal{N}_i}(t))$ that receives the state of a node i and the

FIG. 2. (color online) **Evaluation of the extrapolation capability of GNNs trained on Erdős-Rényi (ER, top row) and Barabási-Albert (BA, bottom row) random networks [38] for the (A, D) simple, (B, E) complex and (C, F) interacting contagion dynamics.** The dashed lines correspond to the GNN prediction error, i.e. the average Jensen-Shannon distance (JSD) between the LTPs of the dynamics used to generate the training datasets (labeled as GT for “ground truth”) and the predictions of the GNNs trained with these datasets. Similarly, the dotted lines correspond to the JSD error of GNN trained on the same datasets but using a semi-exact training scheme (denoted by GNN*). For comparison, the solid lines correspond to the JSD error of an uninformed baseline in which all local transitions are equiprobable, i.e. $p_{\sigma}^{\mu \rightarrow \nu} = |\Omega|^{-1}$ where $|\Omega|$ is the number of node states. The prediction error of the MLEs is shown with markers to clearly illustrate where data is lacking in the training datasets. Lines and markers were obtained by averaging the JSD errors over every σ corresponding to a same value of k . The standard deviation around these average values is shown with a colored area around the lines and with vertical bars for the markers. The training datasets are generated using a ER or BA network composed of $|\mathcal{V}| = 1000$ nodes and with average degree $\langle k \rangle = 4$. We used the same parameters as on Fig. 1 for the simple and complex contagion dynamics, and used $(\tau_1, \tau_2, \gamma_1, \gamma_2, \zeta) = (0.02, 0.01, 0.12, 0.13, 10)$ for the interacting contagion dynamics. All technical details related to training are provided in Supplementary Material.

states of its neighbors $\mathcal{N}_i$, and then returns the LTPs providing the probabilities for nodes i to be in each of the available states at $t + \Delta t$,

$$f(x_i(t), x_{\mathcal{N}_i}(t)) = \left(\hat{p}_{x_{\mathcal{N}_i}(t)}^{x_i(t) \rightarrow \nu} \right)_{\nu \in \Omega}. \quad (4)$$

This function is identical for all nodes and, in practice, the GNN is applied in parallel to every node. To allow the GNN to be applicable to any neighborhood size, the states of the neighbors are aggregated with an attention mechanism inspired by [40]. A notable advantage of this GNN architecture is its *inductive nature*: It learns how to combine the states of the neighbors locally, i.e. through the edges of the network. It is therefore independent from the global network structure, and can consequently be used on any network structure once it has been trained.

We illustrate our approach by applying it to three types of stochastic contagion dynamics whose behaviors heavily

depend on the structure of the underlying network [3, 12, 41]. We first consider a *simple contagion* dynamics: The discrete-time susceptible-infected-susceptible (SIS) dynamics in which nodes are either susceptible (S) or infected (I). At each time step, infected nodes transmit the disease to each of their susceptible neighbors with probability τ , and recover from the disease with probability γ , thus becoming susceptible again. The LTPs of this dynamics are

$$p_{\ell}^{S \rightarrow I} = 1 - p_{\ell}^{S \rightarrow S} = 1 - (1 - \tau)^{\ell} \quad (5a)$$

$$p_{\ell}^{I \rightarrow S} = 1 - p_{\ell}^{I \rightarrow I} = \gamma. \quad (5b)$$

Note that the state of the neighbors of a node, σ , is fully specified by the number of infected neighbors, ℓ . We stress that the GNN is not a priori designed to compute the LTPs as a function of ℓ , but rather computes them as a function of the complete state of the neighbors, σ . The GNN must therefore *learn* that the number of infected neighbor is sufficient to compute the LTPs.

Equation (5a) assumes that disease transmission events are independent. In other words, the probability for a susceptible node to be infected by any of its infectious neighbors does not depend on the state of its other neighbors. The second dynamical process we consider lifts this assumption by replacing Eq. (5a) with the “Planck-like” nonmonotonic infection probability

$$p_{\ell}^{S \rightarrow I} = \frac{1}{z(\eta)} \frac{\ell^3}{e^{\ell/\eta} - 1}, \quad (6)$$

where $z(\eta)$ is fixed such that $p_{\ell}^{S \rightarrow I}$ equals 1 at its maximum, the position of this maximum being controlled by the parameter $\eta > 0$. We refer to this second dynamics as *complex* [43] since transmission is now a nonlinear process that depends on the state of the other neighbors of a node. While unrealistic in the context of disease spreading, Eq. (6) has an interesting interpretation in the context of the propagation of a social behavior. The probability that an individual adopts a new behavior increases monotonously with its number of friends that have adopted the behavior if it appears new or scarce (low ℓ). However, the same individual will become reluctant to adopt this behavior if it has become mainstream and popular (high ℓ), resulting in monotonously decreasing probability of adoption.

The third contagion dynamics we consider is an extension of the SIS dynamics to two interacting infectious diseases in which nodes can be in four different states: Susceptible to both diseases ($S_1 S_2$), infected by one disease and susceptible to the other ($I_1 S_2$ or $S_1 I_2$), or infected by both diseases ($I_1 I_2$). Like the SIS dynamics, nodes infected by disease g will transmit it to a susceptible neighbor with probability τ_g and will recover from g with probability γ_g . The interaction between the two diseases affects the probability of transmission whenever either node, the infected or the susceptible one, is already infected by the other disease. A node infected by disease g will then infect its neighbor susceptible to g with probability $\zeta \tau_g$, where ζ controls the strength of the interaction between the two diseases. This *interacting contagion dynamics* is encoded in 12 LTPs similar to Eqs. (5) whose expressions are given in Supplementary Material.

FIG. 3. **Bifurcation diagrams of the simple, complex and interacting contagion dynamics on Poisson networks [38] composed of $|\mathcal{V}| = 2000$ nodes with different average degrees $\langle k \rangle$.** Prevalence is defined as the fraction of nodes infected by at least one disease. Panels (A–C) and (G–I) show the steady-state prevalence obtained by solving a set of mean-field equations adapted from Ref. [42] (see Supplementary Material). The solid vertical lines indicate the critical points where a steady state changes its stability; a colored area is used to indicate the bistable regime that exists when the two steady states do not change their stability simultaneously. Panels (D–F) and (J–L) show the steady-state prevalence obtained by standard stochastic numerical simulations; each marker is averaged over 100 simulations and the vertical bars show the standard deviation. Red lines, markers and areas are obtained by using the LTPs of the original dynamics (labeled as GT for “ground truth”) in the mean-field equations as well as in the numerical simulations. Blue lines, markers and areas (first row) are obtained by using the LTPs inferred with the GNN trained on a dataset generated with a Barabási-Albert network composed of $|\mathcal{V}| = 1000$ nodes and with average degree $\langle k \rangle = 4$ (value indicated by a vertical dotted black line for reference). Green lines, markers and areas (second row) are obtained by using the LTPs inferred with the GNN trained on the same dataset but using a semi-exact training scheme (denoted by GNN*). The parameters of the contagion dynamics are the same as for Figs. 1 and 2.

In Fig. 1, we show the LTPs predicted by the GNN and compare them with Eqs. (5)–(6) as well as with the maximum likelihood estimators (MLEs) of $p_{\sigma}^{\mu \rightarrow \nu}$. The MLEs are computed from the time series and correspond to the frequency at which nodes in state μ with neighborhood σ transition to ν at the next time step. We find that the GNN fits remarkably well the LTPs of the simple and complex contagion dynamics. In fact, we found the predictions of the GNN to be systematically smoother than the ones provided by the MLEs. This is because the MLEs estimate the LTPs for each individual pair (μ, σ) from disjoint subsets of the training dataset. This implies that a large number of samples of each pair (μ, σ) is needed for the MLEs to provide accurate estimates of every LTPs; a condition rarely met in realistic settings, especially for high degree nodes. This also means that the MLE cannot interpolate in between nor can it extrapolate beyond the pairs (μ, σ) present in the training dataset. In contrast, the GNN works in a conceptually different manner: It is differentiable by construction and all parameters are hierarchically involved in the estimation of the LTPs. Its predictions are therefore smoother, more consistent and able to inter- or extrapolate beyond the training dataset. It also means that the GNN benefits from any sample to improve its predictions for all LTPs.

Figure 2 addresses the accuracy of the predictions of the GNN for *unseen* local network structures, i.e. predictions of LTPs corresponding to pairs (μ, σ) that are not necessarily present from the training dataset. These predictions were extracted from the GNN by applying it to a star graph of $k + 1$ nodes— k nodes of degree 1 connected to a central node of

degree k . Because σ is typically multivariate, and thus cannot be summarized by a single scalar ℓ as for the simple and complex contagion dynamics presented in Fig. 1—it is three-dimensional for the interacting contagion dynamics—, we use the Jensen-Shannon distance (JSD) [37] to quantify the similarity between the LTPs predicted by the GNN and the “ground truth”. Figure 2 shows averages of the JSD, with standard deviations, over all possible neighborhoods σ and states μ for various values of k . Note that the JSD is similar to the loss function of Eq. (2), but it has the notable advantages of being symmetrical, of being more forgiving if a LTP is wrongly predicted to be equal to zero (something that happens with MLEs), and of allowing meaningful comparisons between more than two distributions at the same time (by virtue of being a metric).

Figure 2 confirms that the GNN provides more accurate predictions than MLEs in general. This is especially true in the case of the interacting contagion on BA networks, where MLEs are only marginally different from the uninformed baseline for high degree nodes. This is a consequence of how scarce the inputs are for this dynamics compared to both the simple and complex contagion dynamics for training datasets of the same size, and of how fast the size of the set of possible inputs scales, thereby quickly rendering MLEs completely ineffective for small training datasets. Figure 2(F) therefore provides a telling illustration of the challenge of inferring the parameters of slightly complex contagion models from limited data. The GNN, on the other hand, is less affected by the scarcity of the data, since any sample improves

FIG. 4. (color online) **Projection on real-world temporal networks of proximity contacts between individuals in conferences (Hypertext 2009, SFHH), workplaces (InVS13, InVS15), schools (Thiers13, HS13) and hospitals (LH10) [44–46].** (A) Steady-state fraction of susceptible nodes (in blue) and of infected nodes (in red) for the simple contagion dynamics on the aggregated version of these temporal networks. Triangles correspond numerical simulations averaged over 100 realizations obtained using the same LTPs as in Fig. 3(A, D). Circles show the results of numerical simulations performed with the “ground truth” LTPs of the original dynamics. (B–D) Numerical simulations of the (B) simple, (C) complex and (D) interacting contagion dynamics over time on the workplace InVS15 temporal network, averaged over 100 realizations. Solid lines used the “ground truth” LTPs, and dashed lines used the LTPs extracted from the GNN (the same LTPs as in Fig. 3). Colored areas around the lines indicate the standard deviation. Alternating white and blue zones indicates the time windows, each of which consisting of 10 time steps of the original dataset. Results for the other temporal networks are available in Supplementary Material.

the predictions for all LTPs, as discussed above. In fact, the GNN accuracy is often close to the best case scenario, illustrated by the GNN*, trained under similar conditions but with a semi-exact training scheme.

Figure 2 also suggests that the underlying network structure of the training dataset (mainly its degree distribution) plays a crucial role in the accuracy of the GNN’s predictions for unseen local structures. Indeed, our results show that the GNN provides more accurate predictions when interpolating in between observed values of k than when extrapolating for values of k higher than those present in the training dataset. In other words, it is *easier* for the GNN to interpolate in-between “landmarks” provided by the dataset than to extrapolate beyond the training dataset without any bearings whatsoever. Heterogeneous networks, like Barabási-Albert networks, are therefore expected to yield accurate predictions over a wider range of local structures (μ , σ) than homogeneous networks, even if the training datasets are similar in size. This is also supported by the fact that the best case scenario in terms of GNN accuracy (GNN*) is significantly poorer for homogeneous networks [Figs. 2(A–C)] than for heterogeneous networks [Figs. 2(D–F)]. The amount of *valuable information* contained in a given dataset could therefore be interpreted through the lens of the properties of the training dataset’s underlying network structure. Altogether, these findings suggest a wide applicability of our approach for real complex systems, whose underlying network structures recurrently exhibit a heterogeneous degree distribution [47].

To further illustrate the inductive nature of our GNN architecture, we use it to recover the bifurcation diagram of the three contagion dynamics and to project their behaviors on real-world networks. In the infinite-size limit $|\mathcal{V}| \rightarrow \infty$, these dynamics have two possible steady states: the absorbing state where all nodes are susceptible, and the endemic state in which a fraction of nodes remains infected over

time [3, 12, 41]. These steady states exchange stability during a phase transition which is continuous for the simple contagion and discontinuous for the complex and interacting contagion dynamics. The position of the phase transition depends both on the parameters of the dynamics and on the topology of the network on which they take place. Note that for the complex and interacting contagion dynamics, the stability of absorbing and endemic states do not change at the same point, giving rise to a bistable regime where both states are stable.

Figure 3 shows the bifurcation diagrams obtained by analyzing a mean field framework which uses the LTPs extracted from the GNN, as well as those obtained by performing numerical simulations with these same LTPs. In particular, the mean field perspective demonstrates the possibility to develop an approximate representation of the GNN predictions that, although not as accurate as the predictions obtained by numerical simulations, is amenable to mathematical analysis (i.e. position of fixed points, their stability and the critical thresholds of the phase transitions). The first row of Fig. 3 shows that the GNN correctly predicts whether the phase transition is continuous or discontinuous, as well as the existence of a bistable regime. Quantitatively, the predictions are also strikingly accurate—essentially perfectly accurate for the simple contagion dynamics—which is remarkable given that the bifurcation diagrams were obtained on networks the GNN had never seen before. The second row of Fig. 3 shows the same bifurcation diagrams but using GNN* models trained under the semi-exact training scheme. The almost perfect overlap and thresholds of the curves indicates that there is virtually no intrinsic limit to the accuracy the GNN’s predictions, and highlights once more the critical influence of the information contained in the training dataset. In other words, our GNN’s architecture is as accurate as its training data allows it to be.

Finally, similar levels of accuracy are obtained when simulating the three contagion dynamics on static [Fig. 4(A)] and

temporal [Figs. 4(B–D)] real-world complex networks. Recall that the GNN used to produce these results is the same as the one used in Fig. 3, and was trained on a synthetic static network of a different size. The projections obtained from the GNN remain in the same vicinity as those generated by the “ground truth” LTPs throughout the timespan of the dataset, and show similar patterns of population infections/recoveries when the configuration of the network changes.

In summary, we introduced a data-driven approach that learns the effective mechanisms governing the propagation of contagion dynamics on complex networks. We proposed a reliable training protocol, and we validated the projections of our GNN architecture on simple, complex and interacting contagion dynamics using synthetic as well as real-world static and temporal complex networks. Interestingly, we found that our approach performs better when trained on data whose underlying network structure is heterogeneous, which could prove useful in real-world applications of our method given the ubiquitousness of scale-free networks [47].

By recovering the bifurcation diagram of various dynamics, we illustrated how our approach can leverage time series from an unknown dynamical process to gain insights about

its properties—the existence of a phase transition and its order. Most importantly, we showed how to explicitly extract the LTPs inferred by the GNN architecture, which in turn can help uncover the underlying mechanisms governing the dynamics and help build effective mechanistic models. In a way, we see this approach as the equivalent of a numerical Petri dish—offering a new way to experiment and gain insights about an unknown contagion dynamics—that is complementary to traditional mechanistic modeling.

Although we focused the presentation of our method on contagion dynamics, its potential applicability reaches many other realms of complex systems modeling where intricate mechanisms are at play. We believe this work establishes solid foundations for the use of deep learning in the design of more effective realistic models of complex systems.

We are grateful to Emily Cyr and Guillaume St-Onge for helpful comments and to François Thibault, Patrick Desrosiers, Louis J. Dubé, Simon Hardy and Laurent Hébert-Dufresne for fruitful discussions. This work was supported by the Sentinelle Nord initiative from the Canada First Research Excellence Fund (CM, EL, AA), the Conseil de recherches en sciences naturelles et en génie du Canada (CM, AA) and the Fonds de recherche du Québec-Nature et technologie (EL).

-
- [1] W. O. Kermack and A. G. McKendrick, “A Contribution to the Mathematical Theory of Epidemics,” *Proc. R. Soc. A* **115**, 700–721 (1927).
 - [2] H. W. Hethcote, “The Mathematics of Infectious Diseases,” *SIAM Rev.* **42**, 599–653 (2000).
 - [3] I. Z. Kiss, J. C. Miller, and P. L. Simon, *Mathematics of Epidemics on Networks* (Springer, 2017) p. 598.
 - [4] F. Brauer, C. Castillo-Chavez, and Z. Feng, *Mathematical Models in Epidemiology* (Springer, 2019).
 - [5] C. I. Siettos and L. Russo, “Mathematical modeling of infectious disease dynamics,” *Virulence* **4**, 295–306 (2013).
 - [6] N. C. Grassly and C. Fraser, “Mathematical models of infectious disease transmission,” *Nat. Rev. Microbiol.* **6**, 477–487 (2008).
 - [7] A. Pastore y Piontti, N. Perrà, L. Rossi, N. Samay, and A. Vespignani, *Charting the Next Pandemic: Modeling Infectious Disease Spreading in the Data Science Age* (Springer, 2019).
 - [8] C. Viboud and A. Vespignani, “The future of influenza forecasts,” *Proc. Natl. Acad. Sci. U.S.A.* **116**, 2802–2804 (2019).
 - [9] L. Hébert-Dufresne, S. V. Scarpino, and J.-G. Young, “Macroscopic patterns of interacting contagions are indistinguishable from social reinforcement,” *Nat. Phys.* **16**, 426–431 (2020).
 - [10] S. Nickbakhsh, C. Mair, L. Matthews, R. Reeve, P. C. D. Johnson, F. Thorburn, B. von Wissmann, A. Reynolds, J. McMenamin, R. N. Gunson, and P. R. Murcia, “Virus–virus interactions impact the population dynamics of influenza and the common cold,” *Proc. Natl. Acad. Sci. U.S.A.* **116**, 27142–27150 (2019).
 - [11] D. M. Morens, J. K. Taubenberger, and A. S. Fauci, “Predominant Role of Bacterial Pneumonia as a Cause of Death in Pandemic Influenza: Implications for Pandemic Influenza Preparedness,” *J. Infect. Dis.* **198**, 962–970 (2008).
 - [12] J. Sanz, C.-Y. Xia, S. Meloni, and Y. Moreno, “Dynamics of Interacting Diseases,” *Phys. Rev. X* **4**, 041005 (2014).
 - [13] D. Centola, “The Spread of Behavior in an Online Social Network Experiment,” *Science* **329**, 1194–1197 (2010).
 - [14] S. Lehmann and Y.-Y. Ahn, eds., *Complex Spreading Phenomena in Social Systems*, Computational Social Sciences (Springer, 2018).
 - [15] I. Iacopini, G. Petri, A. Barrat, and V. Latora, “Simplicial models of social contagion,” *Nat. Commun.* **10**, 2485 (2019).
 - [16] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (MIT Press, 2016).
 - [17] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” in *NIPS Proc.* (2012) pp. 1097–1105.
 - [18] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention Is All You Need,” in *Adv. Neural Inf. Process. Syst. 30 (NIPS 2017)* (2017) pp. 5998–6008.
 - [19] M. Långkvist, L. Karlsson, and A. Loutfi, “A review of unsupervised feature learning and deep learning for time-series modeling,” *Pattern Recognit. Lett.* **42**, 11–24 (2014).
 - [20] Z. Zhang, P. Cui, and W. Zhu, “Deep Learning on Graphs: A Survey,” (2018), [arXiv:1812.04202](https://arxiv.org/abs/1812.04202).
 - [21] J. Zhou, G. Cui, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, “Graph Neural Networks: A Review of Methods and Applications,” (2018), [arXiv:1812.08434](https://arxiv.org/abs/1812.08434).
 - [22] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, “How Powerful are Graph Neural Networks?” (2019), [arXiv:1810.00826](https://arxiv.org/abs/1810.00826).
 - [23] W. L. Hamilton, R. Ying, and J. Leskovec, “Representation Learning on Graphs: Methods and Applications,” (2017), [arXiv:1709.05584](https://arxiv.org/abs/1709.05584).
 - [24] M. Balakrishnan and G. T. V, “A neural network framework for predicting dynamic variations in heterogeneous social networks,” *PLOS ONE* **15**, e0231842 (2020).
 - [25] F. A. Rodrigues, T. Peron, C. Connaughton, J. Kurths,

- and Y. Moreno, “A machine learning approach to predicting dynamical observables from network structure,” (2019), [arXiv:1910.00544](#).
- [26] Z. Zhang, Y. Zhao, J. Liu, S. Wang, R. Tao, R. Xin, and J. Zhang, “A general deep learning framework for network reconstruction and dynamics learning,” *Appl. Netw. Sci.* **4**, 110 (2019).
- [27] E. Laurence, C. Murphy, G. St-Onge, X. Roy-Pomerleau, and V. Thibeault, “Detecting structural perturbations from time series with deep learning,” (2020), [arXiv:2006.05232](#).
- [28] A. Fout, J. Byrd, B. Shariat, and A. Ben-Hur, “Protein Interface Prediction using Graph Convolutional Networks,” in *Adv. Neural Inf. Process. Syst.* **30** (2017) pp. 6530–6539.
- [29] M. Zitnik, M. Agrawal, and J. Leskovec, “Modeling polypharmacy side effects with graph convolutional networks,” in *Bioinformatics*, Vol. 34 (2018) pp. i457–i466.
- [30] F. Altarelli, A. Braunstein, L. Dall’Asta, A. Lage-Castellanos, R. Zecchina, L. Dall’Asta, A. Lage-Castellanos, and R. Zecchina, “Bayesian Inference of Epidemics on Networks via Belief Propagation,” *Phys. Rev. Lett.* **112**, 118701 (2014).
- [31] R. Dutta, A. Mira, and J.-P. Onnela, “Bayesian inference of spreading processes on networks,” *Proc. R. Soc. A* **474**, 20180129 (2018).
- [32] P. Eichelsbacher and A. Ganesh, “Bayesian Inference for Markov Chains,” *J. Appl. Probab.* **39**, 91–99 (2002).
- [33] G. Cybenko, “Approximation by superpositions of a sigmoidal function,” *Math. Control. Signals, Syst.* **2**, 303–314 (1989).
- [34] K. Hornik, “Approximation capabilities of multilayer feedforward networks,” *Neural Networks* **4**, 251–257 (1991).
- [35] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning representations by back-propagating errors,” *Nature* **323**, 533–536 (1986).
- [36] D. P. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization,” (2014), [arXiv:1412.6980](#).
- [37] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. (Wiley-Interscience, 2005) p. 776.
- [38] A.-L. Barabási, *Network Science* (Cambridge University Press, 2016) p. 474.
- [39] R. Y. Rubinstein and D. P. Kroese, *Simul. Monte Carlo Method*, 3rd ed. (Wiley, 2016) pp. 1–414.
- [40] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, “Graph attention networks,” in *International Conference on Learning Representations* (2018).
- [41] R. Pastor-Satorras, C. Castellano, P. Van Mieghem, and A. Vespignani, “Epidemic processes in complex networks,” *Rev. Mod. Phys.* **87**, 925–979 (2015).
- [42] P. G. Fennell and J. P. Gleeson, “Multistate Dynamical Processes on Networks: Analysis through Degree-Based Approximation Frameworks,” *SIAM Rev.* **61**, 92–118 (2019).
- [43] S. Lehmann and Y.-Y. Ahn, eds., *Complex Spreading Phenomena in Social Systems* (Springer, 2018) p. 368.
- [44] L. Isella, J. Stehlé, A. Barrat, C. Cattuto, J.-F. Pinton, and W. Van den Broeck, “What’s in a crowd? Analysis of face-to-face behavioral networks,” *J. Theor. Biol.* **271**, 166–180 (2011).
- [45] R. Mastrandrea, J. Fournet, and A. Barrat, “Contact Patterns in a High School: A Comparison between Data Collected Using Wearable Sensors, Contact Diaries and Friendship Surveys,” *PLOS ONE* **10**, e0136497 (2015).
- [46] M. Géniois and A. Barrat, “Can co-location be used as a proxy for face-to-face contacts?” *EPJ Data Sci.* **7**, 11 (2018).
- [47] I. Voitalov, P. van der Hoorn, R. van der Hofstad, and D. Krioukov, “Scale-free networks well done,” *Phys. Rev. Research* **1**, 033034 (2019).

Figures

Figure 1

Validation of the predictions of GNNs trained on a Barabasi-Albert random network (BA) [38] for the (A) simple and (B) complex contagion dynamics.

Figure 2

Evaluation of the extrapolation capability of GNNs trained on Erdos-Rényi (ER, top row) and Barabási-Albert (BA, bottom row) random networks [38] for the (A, D) simple, (B, E) complex and (C, F) interacting contagion dynamics.

Figure 3

Bifurcation diagrams of the simple, complex and interacting contagion dynamics on Poisson networks [38] composed of $|V| = 2000$ nodes with different average degrees $\langle k \rangle$.

Figure 4

Projection on real-world temporal networks of proximity contacts between individuals in conferences (Hypertext 2009, SFHH), workplaces (InVS13, InVS15), schools (Thiers13, HS13) and hospitals (LH10) [44–46].

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [supplementarymaterial.pdf](#)