STOCHASTIC MESSAGE PASSING FOR EPIDEMICS MODELING ON SOCIAL NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

The behavior of stochastic social phenomena such as epidemics or crowd movements is often unknown and rarely easy to model. This work presents a framework for stochastic graph neural network (SGNN) training designed to learn complex stochastic behavior from graph data. Unlike standard graph neural networks, the proposed architecture introduces stochastic message propagation through a continuous relaxation of Bernoulli transmission events. Repeated rollouts from the same graph and initial condition generate multiple possible trajectories, from which node-level marginal state distributions are estimated. The model evolves recurrently over time where at each time step, stochastic messages are propagated along eligible edges, aggregated at the receiving nodes, and combined with the current node states through a multilayer perceptron. We evaluate the model on synthetic graphs and real social networks using the susceptible-infected (SI) epidemic process as a case study. The evaluation examines both the model's ability to reproduce the temporal evolution of node states and its ability to recover the underlying transmission probability. We further analyze the sensitivity of parameter recovery and predictive performance of two factors: the number of model simulations and the initial value of the learnable transmission probability. This work represents the first step towards graph neural networks that can act as stochastic simulators of graph-based phenomena.

1 INTRODUCTION

Understanding the world often requires reasoning about possible outcomes and their probabilities. Many phenomena arise from interactions between entities whose evolution is naturally described stochastically. An infectious contact may or may not transmit a disease, while thermal fluctuations allow a protein to follow different microscopic pathways as it folds. In both cases, local interactions shape the evolution of the system, and similar initial conditions can lead to different trajectories. Understanding these phenomena therefore requires accounting for both the structure of the interactions and their stochastic nature (Schuler et al., 2002).

The effective rules governing such dynamics are often only partially known. Mechanistic models express these rules explicitly through assumptions informed by observations and domain knowledge. Machine learning offers a complementary approach by learning complex dependencies directly from data. For interacting systems represented as graphs, message passing neural networks (MPNNs) provide a natural starting point (Micheli, 2009; Scarselli et al., 2009; Gilmer et al., 2017a; Bacciu et al., 2020), allowing each entity to update its state using information received from its neighbors. Extending this approach to stochastic dynamics requires learning how random interactions influence the evolution of node states.

This task involves more than fitting a recurrent network to observed sequences. A deterministic recurrent model produces the same trajectory whenever it receives the same initial condition, even when the underlying process admits several possible outcomes. Although such a model can predict average behavior or state probabilities, generating different realizations requires an explicit source of randomness. On graphs, this challenge is compounded by the dependence between neighboring entities. A random event at one edge can change subsequent interactions and propagate its effects across the network. The placement of stochasticity within the model therefore matters for representing how local events give rise to collective dynamics.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

In this work, we investigate a stochastic graph neural network (SGNN) that introduces randomness directly into message passing. At each time step, the model samples whether messages are transmitted along edges according to a learnable transmission probability. The receiving nodes combine these sampled messages with their current states through a learned update rule. Repeating this procedure produces different model trajectories from the same graph and initial condition. We train the model by matching the node-state probabilities averaged over these trajectories to those estimated from observations. We show that this training objective is equivalent to minimizing the total variation distance between the predicted and true epidemic state distributions at each node and time step, giving the loss a precise distributional interpretation. A differentiable relaxation of the sampling operation enables the transmission probability and the neural update rule to be learned jointly.

We focus on epidemic modeling, where stochastic compartmental models on contact networks provide a controlled setting for evaluating this approach. Specifically, we study the susceptible–infected (SI) process, in which infected individuals can transmit infection to susceptible neighbors and remain infected thereafter. Its explicit transmission mechanism lets us assess two distinct capabilities of the learned model, reproducing the evolution of infection probabilities and recovering the probability governing individual transmission events. This setting connects stochastic message passing to an interpretable reference process whose parameters are known during evaluation.

Our experiments use simulated SI dynamics on synthetic graphs and real social-network topologies. We compare predicted node-state probabilities with estimates from repeated SI simulations and evaluate the error in the learned transmission probability. We also vary the number of model simulations and the initial value of the transmission probability to examine their effects on prediction and parameter recovery. Together, these experiments assess whether learning stochastic message passing can reproduce epidemic dynamics while retaining information about the local transmission mechanism, and when accurate predictions are accompanied by accurate parameter estimates.

In summary, the contributions of this work are as follows:

- We formulate an SGNN with learnable stochastic message transmission for modeling random interaction dynamics on epidemic graphs;
- We develop a training approach that jointly learns the transmission probability and the neural state update rule from empirical node-state probabilities and the total variation distance, using repeated model simulations and differentiable sampling;
- We evaluate the approach through SI epidemics on synthetic and real network topologies, examining predictive accuracy, transmission-probability recovery, and sensitivity to the number of simulations and parameter initialization.

With this work, our hope is to advance the state of the art on graph neural networks as learnable stochastic simulators, which can capture how random local interactions shape the evolution of complex systems.

2 Related work

Classical compartmental models provide a mathematical foundation for studying epidemic dynamics (Kermack & McKendrick, 1927; Murray, 2002), while network-based extensions explicitly account for interactions between individuals and the influence of contact structure on disease propagation (Pastor-Satorras et al., 2015). Dynamic message-passing methods further provide probabilistic descriptions of infection pathways on networks and support tasks such as epidemic-source inference (Karrer & Newman, 2010; Lokhov et al., 2015; 2014).

For the past thirty years, deep neural networks have been extended to integrate the message passing algorithm (MPNNs). Several graph neural network architectures have shown great success to learn node and edge representation features (Micheli, 2009; Scarselli et al., 2009; Bacciu et al., 2020; Kipf & Welling, 2016; Gilmer et al., 2017b; Veličković et al., 2017). However, some of the limitations of MPNNs have motivated other approaches based on rewiring the graph. Qian et al. (2024) propose probabilistically rewired MPNNs (PR-MPNNs), which learn a distribution over candidate edges and sample sparse adjacency matrices via differentiable k-subset sampling, letting the model add or remove edges to mitigate over-squashing and under-reaching. Like our approach, PR-MPNNs rely on a continuous relaxation of a discrete sampling operation to keep the pipeline end-to-end differ-

entiable, and both works show that principled stochastic relaxations can be trained jointly with a downstream predictor. While stochasticity in our setting models the recurrent transmission process adn enables recovery of the interpretable epidemic parameter, PR-MPNNs instead use randomness to modify a static graph for prediction.

A complementary, theory-driven perspective is given by Alimohammadi et al. (2021), who study independent-cascade/SIR spreading and ask how much of an outbreak's size can be predicted from purely local graph information. They show that for graphs with a local weak limit that are large-set expanders (e.g., configuration and preferential-attachment models), sampling the constant-size neighborhoods of a few random nodes suffices for a $(1-\varepsilon)$-accurate estimate of outbreak probability and final size, with no need to estimate the network or fit a generative model. This is a useful counterpoint rather than a competitor: under strong structural assumptions and a known transmission probability, it gives asymptotic guarantees for aggregate outbreak statistics from local features alone, whereas our SGNN targets node-level temporal marginals and an unknown α, learned jointly from data on finite, non-expander topologies (e.g., Sandia, Coauthorship) where such guarantees do not apply. At the application level, the COVID-19 Forecast Hub evaluation of Cramer et al. (2022) underscores the value of well-calibrated probabilistic forecasting: comparing 27 individual models against a multi-model ensemble on US COVID-19 mortality, they find the ensemble consistently the most accurate and best-calibrated across locations and horizons. This motivates two choices in our work: reporting predictions as full distributions rather than point estimates, and using repeated stochastic rollouts, analogous to ensembling, to obtain calibrated marginals. Their study, however, operates on aggregated, population-scale time series with largely non-network compartmental and statistical models, while ours operates at the individual/node level on an explicit contact graph.

3 BACKGROUND

Preliminaries. Let $G = (\mathcal{V}, \mathcal{E})$ denote a graph of size N with adjacency matrix $A \in \{0,1\}^{N\times N}$ such that $A_{uv} = 1$ for all edges $(u, v) \in \mathcal{E}$ and $A_{uv} = 0$ otherwise, and $|\mathcal{V}| = N$ with $\mathcal{V}$ representing the set of nodes. The set of edges $\mathcal{E} \subseteq \{(u, v)\colon u, v \in \mathcal{V},\ u \neq v\}$ represents the possible connections between nodes. We denote by $\mathcal{N}(u) = \{v\colon (u, v) \in \mathcal{E}\}$ the set of neighbors of node u. The feature matrix is $X = [\mathrm{x}_1, \mathrm{x}_2, \ldots, \mathrm{x}_N]^\top \in \mathbb{R}^{N\times D}$ where x_u is the feature vector associated with node u and D is the feature dimension. In addition, we assume in this paper G to be static, undirected and unweighted. The graph topology remains fixed over the time horizon $t = 0, 1, \ldots, T$; hence, its adjacency matrix satisfies $A^{(t)} = A$.

Epidemic dynamics on graphs. Given G, each node represents an individual and each edge represents a possible transmission pathway. Though several epidemic dynamics could be simulated on G, in this work we are interested in the SI model. We formulate this epidemic process as a two-class dynamic problem with classes $\mathcal{C} = \{0, 1\}$ where S stands for class 0 and I for class 1, and they denote the susceptible and infected states, respectively. At the initial time $t = 0$, one node is selected at random as the infected node, and the remaining $N - 1$ nodes are susceptible. At each subsequent time step, an infected node attempts to transmit the infection to each susceptible neighbor with probability $\alpha \in (0, 1)$.

Marginal distributions. Let $X_{u,r}^{(t)} \in \mathcal{C}$ denote the state of node u at time t in empirical Monte Carlo simulation r. Given R independent empirical simulations, the node marginal probability is estimated as

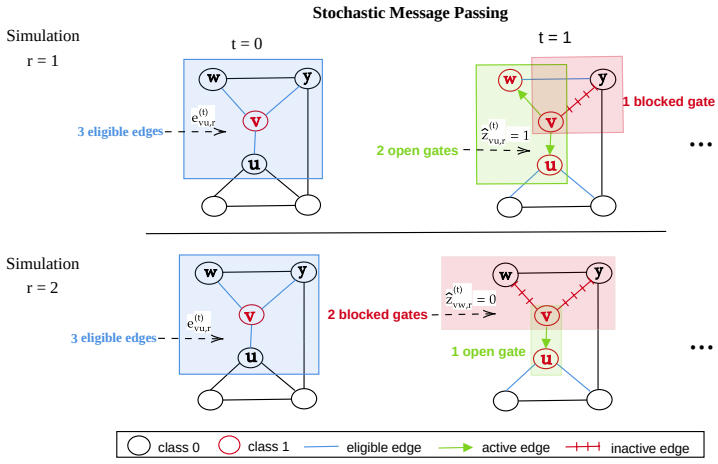
$$p_u^{(t)}(c) = \frac{1}{R}\sum_{r=1}^{R} \mathbb{I}\{X_{u,r}^{(t)} = c\};\ \ c \in \mathcal{C},$$

where $\mathbb{I}$ is the indicator function. The resulting probability vector

$$p_u^{(t)} = [p_u^{(t)}(0), p_u^{(t)}(1)]^\top, \quad p_u^{(t)}(0) + p_u^{(t)}(1) = 1$$

describes the empirical state distribution of node u at time t. For the model simulation r, the MPNN predicted state representation of node u at time t is

$$\hat{p}_{u,r}^{(t)} = [\hat{p}_{u,r}^{(t)}(0), \hat{p}_{u,r}^{(t)}(1)]^\top. \tag{1}$$



At time $t = 0$, node v is in class 1 and has three neighbors in class 0, making all three incident edges eligible for transmission. An independent stochastic gate is sampled for each eligible edge. In simulation $r = 1$, two gates are active receiving the signal $\hat{z}^{(t)}_{vu,r} = 1$, allowing a message to propagate from v to u and from v to w; whereas in $r = 2$, two gates are inactive receiving the signal $\hat{z}^{(t)}_{vu,r} = 0$, and the eligible messages are blocked.

Figure 1: Illustration of the stochastic message propagation framework presented in this work, demonstrating how identical initial states can produce different outcomes across multiple simulations.

Learning the parameters. Let $\omega = (\theta, a)$ denote the joint model learnable parameters, where θ represents the weights and biases of the neural state update function f which determines how node states at $t + 1$ are updated from state at t and incoming transmission signals; a is an unconstrained scalar, $a \in \mathbb{R}$, introduced to learn the transmission probability $\hat{\alpha}$ parametrized as the stochastic edge-level transmission mechanism defined as $\hat{\alpha} = \sigma(a)$ with σ the sigmoid function.

4 METHOD

Given G, the initial node states, and the empirical marginal distribution $p^{(t)}_u$, our goal is to learn a model that can generate multiple stochastic epidemic realizations under the same graph and initial condition. Averaging these realizations should reproduce the estimated empirical state marginal distribution. In addition, the model should estimate the underlying transmission probability α. A standard MPNN with deterministic operations produces the same prediction when repeatedly evaluated using the same graph, initial state, and model parameters. To represent stochastic epidemic transmission, we therefore introduce edge-level Bernoulli transmission events into the message passing mechanism, and use a differentiable relaxation to enable end-to-end learning.

4.1 STOCHASTIC MESSAGE PASSING

A standard MPNN, as defined in (Gilmer et al., 2017b), updates node representations by computing edge-level messages, aggregating incoming information and applying a node update function. At time t, this operation is defined as

$$m^{(t)}_{vu} = \text{Message}_\theta(h^{(t)}_v, h^{(t)}_u), \qquad h^{(t+1)}_u = \text{Update}_\theta\Big(h^{(t)}_u, \text{Aggregate}(\{m^{(t)}_{vu} \colon v \in \mathcal{N}(u)\})\Big).$$

We introduce stochasticity at the edge-level by restricting message propagation to eligible edges and modulating each eligible message with a stochastic transmission gate.

Edge eligibility. Transmission along the edge (v, u) is possible only when v is in class 1 and u in class 0. We represent this condition using the differentiable eligibility score

$$e^{(t)}_{vu,r} = p^{(t)}_{v,r}(1) \cdot p^{(t)}_{u,r}(0).$$

Stochastic transmission gate. Let $\pi(\hat{\alpha}) = [1 - \hat{\alpha}, \hat{\alpha}]^\top$ denote the probability of transmission failure and success. For each eligible edge, simulation r, and time t, we generate a transmission gate

$$\hat{z}^{(t)}_{vu,r} = \Big[\mathcal{G}_\tau\big(\pi(\hat{\alpha}); \xi^{(t)}_{vu,r}\big)\Big]_1$$

where $\mathcal{G}_\tau(\pi(\hat{\alpha});\xi)$ denotes the binary straight-through Gumbel-Softmax relaxation of $\pi(\hat{\alpha})$ driven by Gumbel noise ξ, returning a two-dimensional one-hot vector, $\xi^{(t)}_{vu,r}$ contains independent Gumbel noise, $\tau > 0$ is the relaxation temperature, and $[\cdot]_1$ selects the success entry, the sample of class 1. More details regarding this technique are discussed in (Jang et al., 2016; Maddison et al., 2016). The forward pass represents a discrete transmission outcome, whereas the backward pass differentiates through the continuous relaxation. The full details are provided in Appendix A.1.

Stochastic message propagation. The active message transmitted along edge (v, u) is

$$m^{(t)}_{vu,r} = e^{(t)}_{vu,r}\hat{z}^{(t)}_{vu,r}.$$

Incoming messages at node u are aggregated, and then concatenated $||$ with the current state of u before being passed through a multilayer perceptron f_θ to produce the next node state representation

$$h^{(t)}_{u,r} = \hat{p}^{(t)}_{u,r}|| \sum_{v\in\mathcal{N}(u)} m^{(t)}_{vu,r}, \qquad \hat{p}^{(t+1)}_{u,r} = \mathrm{Softmax}\Big(f_\theta(h^{(t)}_u)\Big).$$

Recurrent rollout. The model is formulated to learn a multi-step epidemic evolution recurrently. Starting from the initial one-hot node state $X^{(0)}$, the full stochastic message passing mechanism is applied recurrently over the simulation horizon

$$X^{(0)} \to \hat{P}^{(1)}_r \to \ldots \to \hat{P}^{(T)}_r, \quad \hat{P}^{(t)}_r \in [0,1]^{N\times 2},$$

containing the state representations (see equation 1) of all nodes in simulation r.

4.2 Marginal learning

Let R_m denote the number of model simulations. The marginal distribution for node u, time t, and class c is estimated by averaging the simulation predictions such that each individual simulation r represents one stochastic realization. Explicitly,

$$\bar{p}^{(t+1)}_u(c;\omega) = \frac{1}{R_m}\sum_{r=1}^{R_m} \hat{p}^{(t+1)}_{u,r}(c;\omega,\xi_r), \qquad \sum_{c\in\mathcal{C}} \bar{p}^{(t+1)}_u(c;\omega) = 1.$$

Training objective. Starting from the observed state at $t = 0$, the model produces predictions for $t = 1,\ldots,T$. The initial state is only used as input, not in the loss. We train the model by minimizing the aggregated absolute value of the elementwise distances between the predicted and empirical marginals

$$\mathcal{L}(\omega) = \frac{1}{2T|\mathcal{V}|}\sum_{t=0}^{T-1}\sum_{u\in\mathcal{V}}\sum_{c\in\{S,I\}} |\bar{p}^{(t+1)}_u(c;\omega) - p^{(t+1)}_u(c)| \tag{2}$$

leading to solve the following problem $\omega^* = \arg\min_\omega \mathcal{L}(\omega)$ s.t. $\omega^* = (\theta^*, a^*)$ and $\hat{\alpha}^* = \sigma(a^*)$.

This choice of loss is not an arbitrary elementwise error. Since each node's state at a given time is itself a two-class distribution, susceptible or infected, the sum of absolute differences in equation 2 has an exact interpretation in terms of a standard distance between distributions as described in the following proposition.

Proposition 1 (Loss as total variation)**.** *For each node u and time t, the predicted and empirical class distributions $\bar{p}^{(t+1)}_u(\cdot;\omega)$ and $p^{(t+1)}_u(\cdot)$ are supported on $\mathcal{C} = \{S, I\}$. The total variation distance between them is*

$$d_{TV}\big(\bar{p}^{(t+1)}_u(\cdot;\omega), p^{(t+1)}_u(\cdot)\big) = \frac{1}{2}\sum_{c\in\{S,I\}} \big|\bar{p}^{(t+1)}_u(c;\omega) - p^{(t+1)}_u(c)\big|.$$

Because $\bar{p}^{(t+1)}_u(S;\omega) + \bar{p}^{(t+1)}_u(I;\omega) = 1$ and $p^{(t+1)}_u(S) + p^{(t+1)}_u(I) = 1$, the two summands above are equal, and

$$d_{TV}\big(\bar{p}^{(t+1)}_u(\cdot;\omega), p^{(t+1)}_u(\cdot)\big) = \big|\bar{p}^{(t+1)}_u(I;\omega) - p^{(t+1)}_u(I)\big|.$$

Summing over $u \in \mathcal{V}$ and $t = 0, \ldots, T-1$ and matching the normalization in $\mathcal{L}(\omega)$ gives

$$\mathcal{L}(\omega) = \frac{1}{T|\mathcal{V}|} \sum_{t=0}^{T-1} \sum_{u \in \mathcal{V}} d_{TV}\left(\bar{p}_u^{(t+1)}(\cdot;\omega), p_u^{(t+1)}(\cdot)\right).$$

The training objective is therefore the average total variation distance between the predicted and empirical state distributions over all nodes and time steps.

Proof. Fix u and t, and write $\bar{p} = \bar{p}_u^{(t+1)}(I;\omega)$, $p = p_u^{(t+1)}(I)$. The normalization constraint gives $\bar{p}_u^{(t+1)}(S;\omega) = 1 - \bar{p}$ and $p_u^{(t+1)}(S) = 1 - p$, so

$$\left|\bar{p}_u^{(t+1)}(S;\omega) - p_u^{(t+1)}(S)\right| = |(1-\bar{p}) - (1-p)| = |p - \bar{p}| = \left|\bar{p}_u^{(t+1)}(I;\omega) - p_u^{(t+1)}(I)\right|.$$

The two terms under $\sum_{c \in \{S,I\}}$ are equal, so their sum is $2|\bar{p} - p|$ and the factor $\frac{1}{2}$ in the definition of d_{TV} cancels it, leaving $d_{TV} = |\bar{p} - p|$. Summing over u and t recovers $\mathcal{L}(\omega)$ up to the stated normalization. □

This equivalence justifies the loss in equation 2 on principled grounds, as minimizing the element-wise absolute error across nodes and time steps is equivalent to minimizing the average total variation distance between the predicted and true epidemic state distributions.

5 Experiments

Data generation. We evaluated our approach on five datasets, including three synthetic graphs generated from random graph models: Barabasi-Albert, Erdős–Rényi, Watts-Strogatz, and two real social networks: Sandia collaboration and Coauthorship (Rossi & Ahmed, 2016). See more details on the graph statistics, the preprocessing steps and a description of how the empirical data was generated in the AppendixA.2.

Experimental setup. To assess the performance of our method and to keep computational cost during hyperparameter selection manageable, we search over the learning rate $\{10^{-2}, 5 \cdot 10^{-3}\}$, the weight decay $\{0, 10^{-4}\}$ and the temperature $\tau \in \{0.1, 1\}$ while the number of model simulations R_m and the initial value of alpha α_0 remain fixed, 50 and 0.1, respectively. The top three configurations are retained and repeatedly retrained five times. We then select the configuration with the lowest mean validation loss. After selecting the best configuration for each dataset, we ran a one factor at a time sensitivity analysis based on R_m and α_0. Finally, we report the test loss and alpha absolute error which demonstrate how R_m and α_0 could affect both the prediction accuracy evaluated on the test dataset, and the parameter recovery accuracy based on the true α. Each result is reported as the mean and standard deviation over five independent runs.

5.1 Sensitivity Analysis

To further investigate the robustness of the proposed method, we performed a sensitivity analysis using the best model configuration selected for each dataset. Two factors are considered: the number of model simulations and the initial value of the parameter α.

5.1.1 Number of model simulations

For this analysis, we fixed $\alpha = 0.1$ while varying $R_m \in \{1, 10, 50, 100\}$. Table 1 and Table 2 report the mean test loss $\mathcal{L}_{\text{test}}$ and α estimation error $|\alpha - \hat{\alpha}|$, respectively. It is worth to note that the test loss here is related to the model with the best validation loss. The method explored in this work develops a training approach that jointly performs two tasks: node state marginals prediction and the transmission probability. Table 1 reports the mean and standard deviation of the node prediction losses on the test datasets for different numbers of model simulations over five independent runs. We could observe that the effect of the number of simulations is not consistent across experiments. Some cases show performance improvement as the number of simulations increases, for instance the experiments for Watts, Sandia with $\alpha = 0.6$ and Coauthorship with $\alpha = 0.2$.

Table 1: Test loss under different numbers of simulations.

		NUMBER OF SIMULATIONS			
		1	10	50	100
DATASET	α	$\mathcal{L}_{\text{test}}$	$\mathcal{L}_{\text{test}}$	$\mathcal{L}_{\text{test}}$	$\mathcal{L}_{\text{test}}$
Barabási–Albert	0.2	0.026 ± 0.013	0.037 ± 0.003	0.002 ± 0.005	0.082 ± 0.102
	0.6	0.010 ± 0.002	0.021 ± 0.002	0.012 ± 0.002	0.011 ± 0.001
Watts–Strogatz	0.2	0.049 ± 0.015	0.027 ± 0.003	0.013 ± 0.001	0.012 ± 0.001
	0.6	0.023 ± 0.006	0.022 ± 0.003	0.011 ± 0.001	0.010 ± 0.002
Erdős–Rényi	0.2	0.025 ± 0.007	0.032 ± 0.006	0.030 ± 0.004	0.168 ± 0.315
	0.6	0.012 ± 0.006	0.025 ± 0.003	0.014 ± 0.001	0.013 ± 0.005
Sandia collaboration	0.2	0.207 ± 0.100	0.076 ± 0.047	0.110 ± 0.112	0.160 ± 0.128
	0.6	0.068 ± 0.011	0.035 ± 0.007	0.028 ± 0.005	0.025 ± 0.005
Coauthorship	0.2	0.314 ± 0.341	0.132 ± 0.084	0.126 ± 0.078	0.121 ± 0.073
	0.6	0.100 ± 0.022	0.141 ± 0.090	0.152 ± 0.086	0.105 ± 0.007

We could also notice in some cases that additional simulations, more specifically the ones with small transmission probability, are associated with a lot of variability and do not consistently improve the performance, this is the case for Barabási–Albert, Erdős–Rényi and Sandia.

Table 2: Estimated infection probability error under different numbers of simulations.

		NUMBER OF SIMULATIONS			
		1	10	50	100
DATASET	α	$\lvert\alpha - \hat{\alpha}\rvert$	$\lvert\alpha - \hat{\alpha}\rvert$	$\lvert\alpha - \hat{\alpha}\rvert$	$\lvert\alpha - \hat{\alpha}\rvert$
Barabási–Albert	0.2	0.619 ± 0.136	0.015 ± 0.028	0.035 ± 0.032	0.051 ± 0.036
	0.6	0.363 ± 0.008	0.023 ± 0.021	0.008 ± 0.002	0.022 ± 0.002
Watts–Strogatz	0.2	0.219 ± 0.149	0.006 ± 0.004	0.002 ± 0.002	0.006 ± 0.001
	0.6	0.266 ± 0.018	0.010 ± 0.006	0.010 ± 0.003	0.011 ± 0.003
Erdős–Rényi	0.2	0.339 ± 0.080	0.061 ± 0.043	0.096 ± 0.040	0.111 ± 0.031
	0.6	0.366 ± 0.009	0.013 ± 0.012	0.013 ± 0.006	0.017 ± 0.009
Sandia collaboration	0.2	0.054 ± 0.027	0.053 ± 0.028	0.039 ± 0.018	0.042 ± 0.022
	0.6	0.226 ± 0.050	0.031 ± 0.024	0.044 ± 0.006	0.048 ± 0.005
Coauthorship	0.2	0.079 ± 0.025	0.080 ± 0.041	0.069 ± 0.031	0.085 ± 0.037
	0.6	0.475 ± 0.020	0.468 ± 0.030	0.473 ± 0.022	0.404 ± 0.057

While Table 1 discusses how well the stochastic model performs the node state prediction task, Table 2 is looking at the performance results related to the task of learning α. It shows how R_m plays a crucial role to accurately recover α. More precisely, we could see that models with only one simulation appears to be worse in several settings; it shows more variations from the independent runs and large differences between the true and learned values of the parameter α. However, we could especially see in the current experiment for Sandia collaboration and Coauthorship datasets with true $\alpha = 0.2$ that the behavior is totally different. The difference in the error magnitude is quite small across the number of simulations. An additional note we could observe from this table is that increasing the number of simulations from 10 to 50 is beneficial, while 100 does not consistently improve the performance. Thus, there is no optimal R_m that works best for all datasets, however the best choice depends on the graph structure and statistics as well as the underlying infection parameter which requires further investigation. In the next section, we will examine the sensitivity to the initial value α_0

5.1.2 INITIALIZATION

Similarly to the previous analysis, we fixed $R_m = 50$ and vary $\alpha_0 \in \{0.1, 0.5, 0.9\}$. We report $\mathcal{L}_{\text{test}}$ and $|\alpha - \hat{\alpha}|$ respectively in Table 3 and Table 4. We analyze the sensitivity to α_0 for the predictive

Table 3: Test loss under different initialization values of α.

		INITIALIZATION OF ALPHA (α_0)		
		0.1	0.5	0.9
DATASET	α	$\mathcal{L}_{\text{test}}$	$\mathcal{L}_{\text{test}}$	$\mathcal{L}_{\text{test}}$
Barabási–Albert	0.2	0.022 ± 0.006	0.163 ± 0.310	0.012 ± 0.003
	0.6	0.012 ± 0.002	0.012 ± 0.001	0.013 ± 0.004
Watts–Strogatz	0.2	0.022 ± 0.009	0.297 ± 0.386	0.016 ± 0.012
	0.6	0.011 ± 0.001	0.011 ± 0.001	0.010 ± 0.004
Erdős–Rényi	0.2	0.030 ± 0.004	0.151 ± 0.293	0.011 ± 0.001
	0.6	0.014 ± 0.001	0.012 ± 0.001	0.011 ± 0.001
Sandia collaboration	0.2	0.110 ± 0.112	0.342 ± 0.314	0.162 ± 0.296
	0.6	0.028 ± 0.005	0.019 ± 0.002	0.019 ± 0.001
Coauthorship	0.2	0.126 ± 0.078	0.314 ± 0.407	0.226 ± 0.321
	0.6	0.152 ± 0.085	0.010 ± 0.002	0.013 ± 0.003

performance of the proposed method in Table 3. The results are indicating that irrespectively from where we start, the experiments with $\alpha = 0.6$ yields comparable results over the initial values α_0, except Coauthorship network which appears to be very sensitive to $\alpha_0 = 0.1$. Synthetic networks show that $\alpha_0 = 0.5$ is unstable for $\alpha = 0.2$, while for Sandia and Coauthorship, we can observe large mean losses and standard deviations for both $\alpha_0 = 0.5$ and $\alpha_0 = 0.9$. From Table 4, we can observe

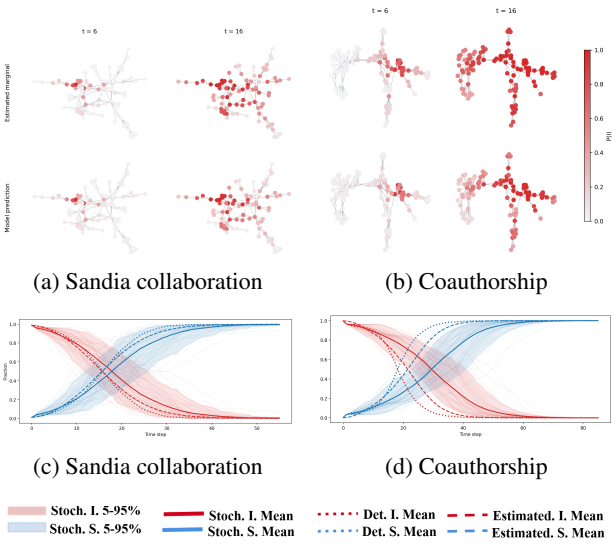
Table 4: Estimated infection probability under different initial values of α.

		INITIALIZATION OF ALPHA (α_0)		
		0.1	0.5	0.9
DATASET	α	$\lvert\alpha - \hat{\alpha}\rvert$	$\lvert\alpha - \hat{\alpha}\rvert$	$\lvert\alpha - \hat{\alpha}\rvert$
Barabási–Albert	0.2	0.035 ± 0.032	0.152 ± 0.137	0.707 ± 0.009
	0.6	0.008 ± 0.002	0.004 ± 0.002	0.112 ± 0.097
Watts–Strogatz	0.2	0.034 ± 0.052	0.162 ± 0.097	0.692 ± 0.014
	0.6	0.010 ± 0.003	0.011 ± 0.010	0.236 ± 0.101
Erdős–Rényi	0.2	0.096 ± 0.040	0.161 ± 0.081	0.715 ± 0.007
	0.6	0.013 ± 0.007	0.005 ± 0.004	0.145 ± 0.088
Sandia collaboration	0.2	0.039 ± 0.018	0.235 ± 0.043	0.501 ± 0.100
	0.6	0.044 ± 0.006	0.017 ± 0.007	0.160 ± 0.042
Coauthorship	0.2	0.069 ± 0.031	0.227 ± 0.096	0.618 ± 0.090
	0.6	0.473 ± 0.022	0.012 ± 0.008	0.088 ± 0.011

that starting at $\alpha_0 = 0.1$ consistently produces the lowest mean absolute error, especially when the true transmission probability $\alpha = 0.2$, while $\alpha_0 = 0.5$ produces large errors and $\alpha_0 = 0.9$ results in the largest errors across all networks. A similar behavior could also be observed for $\alpha = 0.6$ when starting at $\alpha_0 = 0.5$, although the errors produced by $\alpha_0 = 0.9$ are smaller compared to the case of $\alpha = 0.2$. Although $\alpha_0 = 0.1$ yields acceptable results for $\alpha = 0.6$ in these experiments, a particular behavior can be observed for Coauthorship network demonstrating that accurately recovering α requires good initialization.

5.2 Qualitative analysis for predictions

In the previous section, we discussed the node state predictive performance as well as the transmission parameter recovery accuracy. Now, we want to provide a sample example using the Sandia and Coauthorship networks that qualitatively demonstrates how the proposed model can reproduce the overall transition from state 0 to state 1.



(a) Sandia collaboration (b) Coauthorship

(c) Sandia collaboration (d) Coauthorship

The top row compares the empirical marginals with the predictions from the stochastic model, while the bottom row compares the mean empirical marginals with the mean predictions from the stochastic and deterministic models. For the Sandia dataset, we can see that the predicted deterministic mean marginals are closer to the empirical ones and their trajectories reach about 50% of infected nodes around the same time $t = 16$, while it occurs later around $t = 18$ for the stochastic model. In contrast to the Coauthorship dataset, we can still observe a smaller gap between the deterministic and empirical cases compared to the stochastic one. However, the gap is much more larger compared to Sandia and the 50% occurs at different times: earlier around $t = 18$ for the deterministic model, around $t = 22$ for the empirical mean and later around $t = 30$ for the stochastic models.

Figure 2: Epidemic dynamics on the Sandia collaboration network and the Coauthorship network for the infection probability $\alpha = 0.2$. (a) and (b) show node's probabilities of being infected at some specific time steps. While (c) and (d) show mean marginals of being in states susceptible and infected over time. The solid and dotted lines represent the mean predictions from the stochastic and deterministic models, respectively. The dashed lines represent the empirical marginals and the shaded regions indicate the 5% –95% ranges across simulations.

6 Conclusion

This work introduced a stochastic graph neural network for learning epidemic dynamics through stochastic message passing. In this approach, the transmission probability and the neural state update rule are learned jointly from empirical node-state marginals via differentiable sampling. We showed that SGNN can learn SI infection probabilities on synthetic and real social-network topologies, with the SGNN matching the behavior of the true stochastic simulator. Sensitivity analysis provides insight into how initialization and the number of model simulations influence predictive performance and transmission-probability recovery. These findings offer practical guidance for training stochastic graph models and motivate further investigation into the relationship between local transmission mechanisms and collective dynamics. Our contribution opens the way for new studies on the usefulness of GNNs as stochastic simulators, extending them to more complex interaction dynamics.

7 Use of AI statement

During the preparation of this work, the authors used AI tools (ChatGPT 5.5 and Claude) to improve the clarity and writing of some paragraphs. The authors reviewed the content as necessary, taking full responsibility for the final content of this work.

ACKNOWLEDGEMENT

The authors gratefully acknowledge support from the U.S. Department of Energy (DOE grant DE-SC0025801).

REFERENCES

Yeganeh Alimohammadi, Christian Borgs, and Amin Saberi. Algorithms using local graph features to predict epidemics. *https://arxiv.org/abs/2110.08961*, 2021.

Davide Bacciu, Federico Errica, Alessio Micheli, and Marco Podda. A gentle introduction to deep learning for graphs. *Neural Networks*, 129:203–221, 2020.

Estee Y Cramer, Evan L Ray, Velma K Lopez, Johannes Bracher, Andrea Brennen, Alvaro J Castro Rivadeneira, Aaron Gerding, Tilmann Gneiting, Katie H House, Yuxin Huang, et al. Evaluation of individual and ensemble probabilistic forecasts of covid-19 mortality in the united states. *Proceedings of the National Academy of Sciences*, 119(15):e2113561119, 2022.

Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pp. 1263–1272, 2017a.

Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International Conference on Machine Learning*, pp. 1263–1272. Pmlr, 2017b.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

Brian Karrer and Mark EJ Newman. Message passing approach for general epidemic models. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, 82(1):016101, 2010.

William Ogilvy Kermack and Anderson G McKendrick. A contribution to the mathematical theory of epidemics. *Proceedings of the royal society of london. Series A, Containing papers of a mathematical and physical character*, 115(772):700–721, 1927.

Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Andrey Y Lokhov, Marc Mézard, Hiroki Ohta, and Lenka Zdeborová. Inferring the origin of an epidemic with a dynamic message-passing algorithm. *Physical Review E*, 90(1):012801, 2014.

Andrey Y Lokhov, Marc Mézard, and Lenka Zdeborová. Dynamic message-passing equations for models with unidirectional dynamics. *Physical Review E*, 91(1):012811, 2015.

Chris J Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.

Alessio Micheli. Neural network for graphs: A contextual constructive approach. *IEEE Transactions on Neural Networks*, 20(3):498–511, 2009.

James D Murray. Mathematical biology: I. an introduction. interdisciplinary applied mathematics. *Mathematical Biology, Springer*, 17, 2002.

Romualdo Pastor-Satorras, Claudio Castellano, Piet Van Mieghem, and Alessandro Vespignani. Epidemic processes in complex networks. *Reviews of modern physics*, 87(3):925–979, 2015.

Chendi Qian, Andrei Manolache, Kareem Ahmed, Zhe Zeng, Guy Van den Broeck, Mathias Niepert, and Christopher Morris. Probabilistically rewired message-passing neural networks. In *International Conference on Learning Representations*, volume 2024, pp. 32051–32076, 2024.

Ryan A Rossi and Nesreen K Ahmed. An interactive data repository with visual analytics. *ACM SIGKDD Explorations Newsletter*, 17(2):37–41, 2016.

Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80, 2009.

Benjamin Schuler, Everett A Lipman, and William A Eaton. Probing the free-energy surface for protein folding with single-molecule fluorescence spectroscopy. *Nature*, 419(6908):743–747, 2002.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

A Appendix

A.1 Stochastic message passing

Let $\alpha \in (0, 1)$ denote the probability that infection is transmitted across an eligible edge e_{vu}. The main goal is to recover the underlying transmission probability from state observations. Accordingly, we introduce α as a learnable scalar parameter denoted by $\hat{\alpha}$. Thus, the edge-level transmission event across an eligible edge is parametrized as

$$z_{vu}^{(t)} \sim \text{Bernoulli}(\hat{\alpha}),$$

where $\hat{\alpha} \in (0, 1)$. The Bernoulli distribution is represented by a two-class categorical distribution

$$\pi_0 = 1 - \hat{\alpha}, \quad \pi_1 = \hat{\alpha}$$

where subscript 0 corresponds to failure (no possible transmission) and 1 to success (effective transmission). However, because Bernoulli sampling is discrete, and therefore not differentiable, preventing gradient propagation through the transmission event, we use the binary continuous relaxation formulation of the Gumbel-Softmax from Jang et al. (2016); Maddison et al. (2016) as follows

$$\xi_k = -\log(-\log U_k), \qquad U_k \sim \mathcal{U}(0, 1), \qquad k \in \{0, 1\},$$

denote independent Gumbel noise variables. The relaxed categorical sample is defined as

$$y_k^{(t)} = \frac{\exp\left((\log \pi_k + \xi_k)/\tau\right)}{\sum_{\ell=0}^{1} \exp\left((\log \pi_\ell + \xi_\ell)/\tau\right)}, \qquad k \in \{0, 1\},$$

where $\tau > 0$ is the Gumbel parameter, called temperature. The stochastic transmission gate is then taken as the samples of class 1

$$\tilde{z}_{vu}^{(t)} = y_1^{(t)}.$$

A.2 Data generation

We used these graphs for the experiments: Barabasi-Albert, Erdős–Rényi, Watts-Strogatz, and two real social networks: Sandia collaboration and Coauthorship (Rossi & Ahmed, 2016). We preprocessed each graph by retaining only its largest connected component and then converting it to a static, undirected, and unweighted network. These preprocessing steps ensure that all datasets satisfy the assumptions made in this work.

Table 5: Graph statistics for the datasets used in the experiments.

	GRAPH STATISTICS						
GRAPH	**NODES**	**EDGES**	$\langle k \rangle$	$k_{\max}$	**CLUSTERING**	**AVG. PATH**	**DIAMETER**
Barabási–Albert	300	894	5.96	50	0.072	3.07	5
Watts–Strogatz	250	750	6.00	9	0.228	3.95	7
Erdős–Rényi	200	600	6.00	13	0.029	3.14	6
Sandia collaboration	86	123	2.86	12	0.390	4.84	11
Coauthorship	379	913	4.82	34	0.736	6.05	17

We generated the empirical SI stochastic epidemic dynamics on graphs from 10^3 Monte Carlo simulations parametrized by the infection probability α, and the time horizon determining the length of the simulation T defined as d/α where d is the diameter of the graph.