

Structure-prior Informed Diffusion Model for Graph Source Localization with Limited Data

Hongyi Chen
Shenzhen International Graduate
School, Tsinghua University
Shenzhen, China

Jingtao Ding
Department of Electronic
Engineering, Tsinghua University
Beijing, China

Xiaojun Liang
Peng Cheng Laboratory
Shenzhen, China

Yong Li
Department of Electronic
Engineering, Tsinghua University
Beijing, China

Xiao-Ping Zhang
Shenzhen International Graduate
School, Tsinghua University
Shenzhen, China

Abstract

The source localization problem in graph information propagation is crucial for managing various network disruptions, from misinformation spread to infrastructure failures. While recent deep generative approaches have shown promise in this domain, their effectiveness is limited by the scarcity of real-world propagation data. This paper introduces SIDSL (Structure-prior Informed Diffusion model for Source Localization), a novel framework that addresses three key challenges in limited-data scenarios: unknown propagation patterns, complex topology-propagation relationships, and class imbalance between source and non-source nodes. SIDSL incorporates topology-aware priors through graph label propagation and employs a propagation-enhanced conditional denoiser with a GNN-parameterized label propagation module (GNN-LP). Additionally, we propose a structure-prior biased denoising scheme that initializes from structure-based source estimations rather than random noise, effectively countering class imbalance issues. Experimental results across four real-world datasets demonstrate SIDSL’s superior performance, achieving 7.5-13.3% improvements in F1 scores compared to state-of-the-art methods. Notably, when pretrained with simulation data of synthetic patterns, SIDSL maintains robust performance with only 10% of training data, surpassing baselines by more than 18.8%. These results highlight SIDSL’s effectiveness in real-world applications where labeled data is scarce.

1 Introduction

In today’s highly interconnected world, graph information propagation issues, such as misinformation spread, cyber threats, and infrastructure failures, have far-reaching consequences for society [4]. The ability to quickly identify the source of these disruptions is critical for mitigating their impact. By analyzing snapshots of affected networks, we can trace the origin of the spread, a process essential for managing crises like disease outbreaks [24], enhancing network security [15], and preventing further damage in scenarios such as power grid failures [2].

Early methods [18, 21, 22, 26, 42, 43] for source localization in graphs rely on metrics or heuristics derived from the graph’s topology, applicable only to specific propagation patterns like the Susceptible-Infected (SI) or Independent Cascade (IC) models. Notably, Wang et al. [34] overcome this limitation by introducing a label propagation algorithm based on the intuition of source centrality. Nevertheless, the approach has limitations in both expressiveness

and scalability, as they cannot effectively encode graph topological information. Data-driven methods [5, 10, 31] overcome these limitations as they directly learn graph neural networks (GNNs) to capture the propagation process exhibited in empirical data, but still neglect the indeterminacy of information propagation that corresponds to the uncertain nature of source localization [20]. Recently, deep generative models including variational autoencoders [20], normalization flows [36] and diffusion models [11, 37] have been adopted for solving the source localization problem. These probabilistic generative methods can quantify the indeterminacy in source localization by learning the empirical data distribution and advance the state-of-the-art performance.

However, collecting real-world propagation data for deep generative methods is difficult and costly, posing significant requirements on source localization models that can adapt to real-world environments with limited data. This brings up three main following challenges.

- **Firstly, real-world graphs typically exhibit unknown propagation patterns, which becomes far more challenging to characterize when data is limited.** In this regard, existing methods [5, 20, 31, 37] rely purely on data to gain an understanding of the propagation patterns, limiting their capability to generalize in unseen scenarios.
- **Secondly, complex interrelations between propagation patterns and graph topology are difficult to capture with limited data.** Existing deep learning methods rely on a large amount of labeled data from the target network (i.e., identified source nodes from historical propagation) to account for the impact of structural heterogeneity on propagation patterns. Due to this heavy dependence on labeled data, these models typically underperform when applied to networks with limited training samples, hindering their practical applicability.
- **Thirdly, the inherent class imbalance between source and non-source nodes becomes more harmful under data scarcity, compromising the accuracy and reliability of source identification.** In most propagation scenarios, source nodes naturally constitute a small minority of the total graph nodes [3]. Limited training data can further amplify this problem, leading to biased distribution models that tend to degenerate towards predicting all nodes as non-sources.

Therefore, in this paper, we propose a novel generative diffusion framework, namely **Structure-prior Informed Diffusion Model for Source Localization (SIDSL)**, which leverages topology-aware priors to achieve robust source localization with limited propagation data collected from real environments. Taking both graph topology and observed node states as input, SIDSL employs a denoising diffusion model guided by structural priors to predict source node distributions. To tackle the first challenge, We leverage graph structure-based source estimations, generated through graph label propagation, which identifies potential sources based on locally highest infection values. Integrated into the denoising network, these topology-aware priors provide stable guidance across different propagation patterns, enhancing generalization to unseen propagation scenarios. To address the second challenge, We design a propagation-enhanced conditional denoiser with a GNN-parameterized label propagation module (GNN-LP) that combines label propagation for tracing infection pathways with GNN for efficient topology-aware feature extraction, enabling effective learning of topology-propagation relationships from limited data. To tackle the third challenge, we develop a structure-prior biased denoising process that initializes denoising process from structure-based source estimations rather than random noise, creating a natural bias towards potential source nodes to prevent degeneration with limited, imbalanced data.

The above designs work together in our structure-prior informed diffusion framework to capture stable structural patterns independent of specific propagation dynamics, which allows our model to learn pattern-invariant features using synthetic propagation data simulated from established models [13, 14, 16]. By focusing on these invariant features rather than specific propagation patterns, our model can effectively transfer the learned knowledge to real-world scenarios through efficient few-shot or zero-shot learning, as shown in Figure 1.

Our contributions are summarized as follows:

- (1) We propose a structure-prior informed diffusion framework that effectively addresses the challenge of limited data in source localization by incorporating topology-aware priors, enabling robust generalization to unknown propagation patterns in real-world scenarios.
- (2) We introduce a series of innovative techniques, including a propagation-enhanced conditional denoiser with GNN-LP module and estimation-biased denoising, which work synergistically to handle structural heterogeneity and class imbalance issues.
- (3) We evaluate SIDSL’s performance across four real-world datasets and demonstrate its superior effectiveness in source identification tasks. Our method consistently outperforms state-of-the-art baselines by 7.5-13.3% in F1 scores. Through effective pretraining with simulation data of synthetic patterns, SIDSL maintains robust performance with limited empirical data, surpassing baselines by over 19% in few-shot (using only 10% training samples) and 40% in zero-shot settings, demonstrating strong generalization capability in real-world applications. Additional evaluations further validate SIDSL’s effective synthetic-to-real transfer capability in improving sample efficiency and reducing training time.

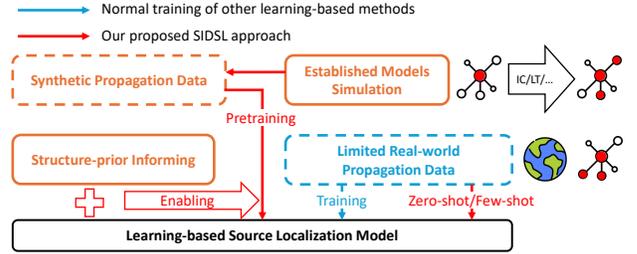


Figure 1: The proposed SIDSL approach.

2 Related Works

2.1 Source Localization

As the inverse problem of information propagation on networks, source localization refers to inferring the initial propagation sources given the current diffused observation, such as the states of the specified sensors or a snapshot of the whole network status [27]. It can be applied to tasks like rumor source identification and finding the origin of rolling blackouts in intelligent power grids [27]. Early approaches are rule-based and rely on metrics or heuristics derived from the network’s topology for source identification [25, 42, 43]. For example, Shah and Zaman [25] develop a rumor-centrality-based maximum likelihood estimator under the Susceptible-Infected (SI) [16] propagation pattern. This kind of method fails to effectively encode topology information. Later, deep learning-based methods devised for capturing the topological effect exhibited in empirical data have been proposed [5, 9, 18, 21, 31, 34]. However, most of them fail to model the uncertainty of the location of sources, as the forward propagation process is stochastic. To overcome this, deep generative models have been adopted [11, 20, 33, 36, 37]. SLVAE [20] utilizes the Variational Auto-Encoders (VAEs) backbone and optimizes the posterior for better prediction. However, it is difficult to converge when the propagation pattern is complicated due to the nature of VAEs. DDMSL [37] models the Susceptible-Infected-Recovered (SIR) [16] infection process into the discrete Diffusion Model (DM) [8], and design a reversible residual block based on Graph Convolutional Networks (GCNs) [17]. However, it requires additional intermediate propagation data and cannot be generalized to other propagation patterns. Our method demonstrates superior functionality and adaptability for real-world applications, requiring fewer input data while addressing existing limitations, thus offering greater practical value. We provide a comparison of typical source localization methods in the Appendix A.

2.2 Typical Propagation Models

Information propagation estimation models information spread in networks and explains propagation sources, with applications in event prediction [40], adverse event detection [32], and disease spread prediction [29]. Two main model categories exist: infection models and influence models. Infection models like Susceptible-Infected (SI) and Susceptible-Infected-Susceptible (SIS) manage transitions between susceptible and infected states in networks [13, 16]. In these models, infected nodes attempt to infect adjacent nodes with probability β at each iteration, while in SIS, infected nodes may revert to susceptible with probability λ . The Susceptible-Infected-Recovered (SIR) model extends this by adding a recovered state.

Independent Cascade (IC) and Linear Threshold (LT) [14] are influence models that examine influence spread in social and infrastructure networks. In the IC model, nodes are either active or inactive, starting with initial active nodes. Newly activated nodes get one chance to activate inactive neighbors, with activation probability based on edge weight. The LT model activates inactive nodes when accumulated neighbor influence exceeds a threshold.

3 Preliminaries

3.1 Problem Formulation

Our research problem is formulated as follows. Given an undirected social network $\mathcal{G} = (V, E)$ where V is the node set, E is the edge set, and $Y = \{Y_1, \dots, Y_{|V|}\}$ is an infection state of all nodes in \mathcal{G} , which describes that a subset of nodes in \mathcal{G} have been infected. Each $Y_i \in \{1, 0\}$ denotes the infection state of node $v_i \in V$, where $Y_i = 1$ indicates that v_i is infected and otherwise $Y_i = 0$ indicates it is uninfected. We aim to find the original propagation source \hat{X} from the propagated observation Y , so that the loss with the ground Truth source set $X^* \in \{1, 0\}^{|V| \times 1}$ is minimized, i.e. $\hat{X} = \operatorname{argmin}_X \|X - X^*\|_2^2$. To account for the uncertainty in source localization, we need to construct a probabilistic model $P(X|Y, \mathcal{G})$, which can be used to sample for the final prediction.

3.2 Label Propagation based Source Identification

In realistic situations, the intractable propagation process does not have an explicit prior, and it is also challenging to value appropriate parameters for the pre-selected underlying propagation model. To address this, Wang et al. [34] propose Label Propagation based Source Identification (LPSI). Since LPSI investigates the same problem as our research, we use it as a baseline to compare performance. LPSI captures source centrality characteristics in the method design. The centrality of sources shows that nodes far from the source are less likely to be infected than those near it [26], which can also be observed in the real-world data by our analysis in the Appendix B. Based on these ideas, they propose to perform label propagations on the observation state of the network. By setting $Y[Y = 0] = -1$ and $\mathcal{Z}^{t=0} \leftarrow Y$, the iteration of label propagation and the convergence states are as follows:

$$\mathcal{Z}_i^{t+1} = \alpha \sum_{j:j \in \mathcal{N}(i)} S_{ij} \mathcal{Z}_j^t + (1 - \alpha) Y_i. \quad (1)$$

\mathcal{Z} finally converges to: $\mathcal{Z}^* = (1 - \alpha)(I - \alpha S)^{-1} Y$, where $S = D^{-1/2} A D^{-1/2}$ is the normalized weight matrix of graph \mathcal{G} , α is the fraction of label information from neighbors, and $\mathcal{N}(i)$ stands for the neighbor set of the node i . After obtaining the converged label matrix \mathcal{Z}^* , one node can be identified as a source when its final label is larger than its neighbors. While node labels alone cannot fully capture complex structural information, this method still effectively identifies structural patterns related to source centrality (Appendix B). In our work, we leverage the idea of LPSI to generate structural guidance for efficient source identification and details can be found in Section 4.

4 SIDSL: the Proposed Method

4.1 Diffusion Model for Source Localization

To capture the indeterminacy of the ill-posed localization problem, it is essential to build a probabilistic model that can also leverage the topological information in the graph structure. We consider using the generative diffusion model (DM) framework to tackle this requirement by modifying it as a source predictor, which classifies each node into two categories: source or non-source. The diffusion model aims to learn source distributions by gradually adding Gaussian noise in samples in the forward process and learning to reverse this process using denoising networks conditioned on observations and graph structure.

The forward process of the diffusion model is to gradually corrupt the initial source labels ($X_0 = X$) by adding Gaussian noise over n timesteps, which can be formulated as:

$$p(X_{1:n}|X_0, Y, \mathcal{G}) = \prod_{t=1}^n p(X_t|X_{t-1}, Y, \mathcal{G}), \quad (2)$$

$$p(X_t|X_{t-1}, Y, \mathcal{G}) = \mathcal{N}(X_t; \sqrt{1 - \beta_t} X_{t-1}, \beta_t I),$$

where β_t is the noise schedule, X_0 represents the initial source labels, Y denotes the observations, \mathcal{G} represents the graph structure, and X_t denotes the noisy features at timestep t . At the end of forward process, X_n becomes pure Gaussian noise. While the forward process only requires source labels, the reverse process leverages observation Y and graph structure \mathcal{G} as conditional inputs to guide the denoising.

In the reverse process, it gradually transforms pure Gaussian noise into the original source labels, generating new samples $\hat{X} \in [0, 1]^{|V| \times 1}$ representing the probability of each node being a source. The final source predictions are obtained by thresholding these probabilities. To enhance stability across different propagation patterns, we leverage graph structure-based source estimations as conditional priors. These priors $X_{est} = \mathcal{Z}^*$ are obtained through label propagation (Section 3.2), which identifies potential sources by aggregating infection values through the graph's normalized weight matrix while capturing source centrality characteristics. The convergence result \mathcal{Z}^* reflects each node's structural importance relative to the observed infection pattern.

The reverse process can be formulated as:

$$q(X_{n-1:0}|X_n, Y, \mathcal{G}) = \prod_{t=n}^1 q(X_{t-1}|X_t, Y, \mathcal{G}), \quad (3)$$

$$q(X_{t-1}|X_t, Y, \mathcal{G}) = \mathcal{N}(X_{t-1}; \mu_\theta(X_t, t, Y, \mathcal{G}, X_{est}), \sigma_t^2 I),$$

where μ_θ is parameterized by a denoising network f_θ that predicts the mean of the Gaussian distribution and σ_t is the predicted variance. The diffusion framework, whose iterative diffusion process enables fine-grained integration of prior knowledge at multiple scales through its progressive denoising steps, allows for elegant integration of prior knowledge. By integrating topology-aware priors X_{est} as stable guidance across different propagation patterns, our diffusion framework combines structural knowledge with data-driven pattern learning, enhancing generalization to new propagation scenarios.

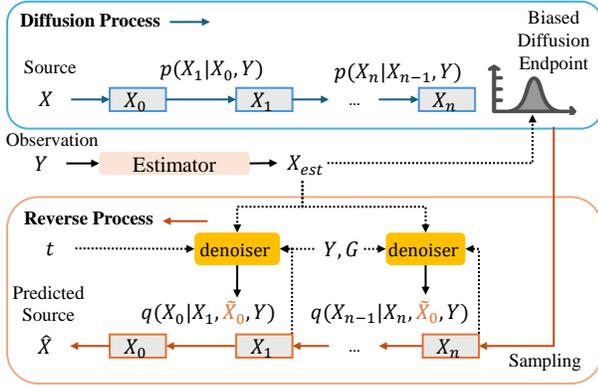


Figure 2: The framework of SIDL.

4.2 Structure-prior biased Denoising Process

In most propagation scenarios, source nodes constitute a small minority of total graph nodes [3]. With limited training data, this severe class imbalance often leads models to degenerate towards predicting all nodes as non-sources, as the model overfits to the dominant non-source class. To address this, inspired by Han et al. [7], we propose an structure-prior biased denoising scheme that set the prior of the denoising process to structure-based source estimations rather than random noise (illustrated in Figure 2). This design leverages two key insights: (1) structure-based estimations provide reliable initial source candidates by capturing inherent source structural properties like centrality, making them more informative than random noise, and (2) starting the denoising process from these estimations creates a natural bias in the trajectory space, encouraging the model to explore regions with higher likelihood of containing true sources. By incorporating these topology-aware priors into the initialization and denoising process, our approach effectively prevents model collapse while reducing the data requirements for learning accurate source patterns.

Specifically, we first modify the mean of the diffusion endpoint (or reverse starting point) as the graph structure-based source estimation $X_{est} = X_{est}(Y, \mathcal{G})$ instead of using standard Gaussian noise, i.e.:

$$p(X_n|Y, \mathcal{G}) = \mathcal{N}(X_{est}(Y, \mathcal{G}), I). \quad (4)$$

According to the original notation in Ho et al. [8], the Markov transition should be modified as:

$$p(X_t|X_{t-1}, Y, \mathcal{G}) = \mathcal{N}(\sqrt{1-\beta_t}X_{t-1} + (1-\sqrt{1-\beta_t})X_{est}, \beta_t I), \quad (5)$$

which derives the closed-form distribution with arbitrary t :

$$p(X_t|X_0, Y, \mathcal{G}) = \mathcal{N}(\sqrt{\alpha_t}X_0 + (1-\sqrt{\alpha_t})X_{est}(Y, \mathcal{G}), (1-\alpha_t)I), \quad (6)$$

where $\alpha_t := 1 - \beta_t$, $\bar{\alpha}_t := \prod_{i=1}^t \alpha_i$.

In the reverse denoising process, the reverse Markov denoiser $q(X_{t-1}|X_t, Y, \mathcal{G})$ recovers the original data. DM framework trains the parameterized denoiser to fit the ground truth forward process posterior:

$$\begin{aligned} p(X_{t-1}|X_t, X_0, Y, \mathcal{G}) &= p(X_{t-1}|X_t, X_0, X_{est}(Y, \mathcal{G})) \\ &= \mathcal{N}(\tilde{\mu}(X_t, X_0, X_{est}(Y, \mathcal{G})), \tilde{\beta}_t I), \end{aligned} \quad (7)$$

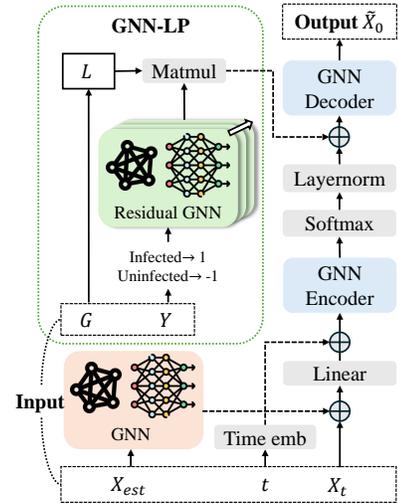


Figure 3: The architecture of the denoising network.

where

$$\begin{aligned} \tilde{\mu}(X_t, X_0, X_{est}(Y, \mathcal{G})) &:= \frac{\sqrt{\alpha_t-1}\beta_t}{1-\alpha_t} X_0 + \frac{(1-\alpha_t)\sqrt{\alpha_t}}{1-\alpha_t} X_t + \\ &\quad \left(1 + \frac{(\sqrt{\alpha_t}-1)(\sqrt{\alpha_t} + \sqrt{\alpha_t-1})}{1-\alpha_t}\right) X_{est}, \quad (8) \\ \tilde{\beta}_t &:= \frac{1-\alpha_t-1}{1-\alpha_t} \beta_t. \end{aligned}$$

The denoising network f_θ is set to estimate the ground truth source X (i.e. X_0 in Equation (8)), which we empirically find more effective. The denoise network outputs the estimated source vector $\hat{X}_0 := f_\theta(X_t, X_{est}, Y, \mathcal{G}, t)$ to calculate the posterior for step-by-step denoising. The denoising network f_θ can be trained by the simple L2 loss function [8]:

$$L(\theta) = \mathbb{E}_{X_0 \sim p(X_0|\cdot), t, \epsilon} \|X_0 - f_\theta(X_t, t, \cdot)\|_2^2, \quad (9)$$

where \cdot represents the conditional inputs.

4.3 Propagation-enhanced Conditional Denoiser

4.3.1 Denoising Network Architecture. The architecture of our denoising network is shown in Figure 3.

Encoding the noisy input and soft labels. The pre-estimated X_{est} is forwarded through a multi-layer GNN to capture the hidden message with graph structural information. Subsequently, it is added to the noisy input X_t and passed through a linear layer. The final input for the GNN encoder is $Z_e = \text{Linear}(\text{GNN}(X_{est}) \oplus X_t) \oplus \text{Emb}(t)$, where for the denoising step t , we use the classical sinusoidal embedding [30]. The \oplus indicates element-wise sum. Z_e is then passed through a GCN-based encoder and is smoothed through a softmax function and layer normalization:

$$Z_d = \text{LN}(\sigma(\text{GNN}(Z_e))). \quad (10)$$

Softmax and layer normalization operations are then used to improve the network's representational capacity and convergence performance, resulting in better performance and faster training [12].

Conditioning. Shown at the left part of the figure, a GCN-based module learns the encoding carrying the source prominence and

centrality from the infection state input Y , which will be elaborated on in the next section.

Decoder. Z_d and encoded condition h_{out} are decoded through a GCN-based module, resulting in the estimation for the uncorrupted sample X_0 (i.e. X):

$$\tilde{X}_0 = \text{GNN}(Z_d, h_{out}). \quad (11)$$

4.3.2 GNN-parameterized Label Propagation(GNN-LP). Our conditioning module takes the observed infection states as input, which contain crucial coupling information between spreading dynamics and network topology. To effectively extract this information, we introduce the GNN-LP module. We first employ label propagation [34], which first mark infected nodes as 1 and uninfected as -1 in the observation state Y , resulting in Y^* . The propagation follows the update rule: the label of a node in the next step is a combination of its original label and the sum of normalized labels from its neighbors. We can rewrite this iteration as:

$$Z_i^{t+1} = \hat{\alpha} Y_i^* + \sigma\left(\sum_{j:j \in \mathcal{N}(I)} \phi(Z_j^t, S_{ij})\right), \quad (12)$$

where we add non-linear transformations $h(\cdot)$ and $\sigma(\cdot)$ to enhance the expressiveness of the propagation process. As propagation continues, each node’s label value changes according to the graph structure. These changes reflect the node’s importance (centrality) and spreading influence in the network. By observing these changes, we can understand the relationship between propagation patterns and network structure. We then enhance this structural diffusion with GNNs, which learn topology-specific message passing rules to capture how different local structures influence propagation patterns. We can notice that the structure of the above equation exactly matches the form of the general Graph Neural Network (GNN) [6], and the parameterization can be achieved by using a residual block combined with a graph convolutional network(GCN, [17]):

$$\begin{aligned} g(h^{(l)}) &= \sigma(\tilde{D}^{-1/2} \tilde{A} \tilde{D}^{-1/2} \cdot h^{(l)} \cdot w), \\ h^{(0)} &= Y^* U^T, \quad h^{(l+1)} = h^{(0)} + g(h^{(l)}). \end{aligned} \quad (13)$$

Among them, $U \in \mathbb{R}^{C \times 1}$ is the linear transformation, σ is the activation operator PReLU, $h^{(l)}$ stands for the output hidden state of the l -th layer of the GCN, $\tilde{A} = A + I$ is the adjacency matrix with self-loops, and \tilde{D} is the degree matrix of \tilde{A} . The final layer’s output $h^{(l_f)}$ is projected back to dimension 1 and multiplied by the graph’s Laplacian matrix L , i.e. $h_{out} := L \cdot h^{(l_f)}$, which highlights the prominent nodes with higher propagation propagated labels among their neighbors.

Through this integration, the module transforms binary infection states Y into continuous representations h_{out} that simultaneously encode both the spreading process and network structure information, enabling effective learning of their complex interactions even with limited training samples.

4.4 Pretrain using Simulation data from Established Models

The proposed structure-prior informed diffusion framework effectively extracts topology-aware features that remain largely stable across different propagation dynamics. These designs enable the framework to learn diverse knowledge from different propagation

Algorithm 1 Pretraining on synthetic data and few-shot learning on real data.

Require: Graph $\mathcal{G} = (V, E)$, synthetic data generators \mathcal{M} s (e.g. SIS, IC), few-shot samples \mathcal{D}_{real}

- 1: // Phase 1: Learn from Diverse Propagation Patterns
- 2: Generate synthetic (X^*, Y) pairs using \mathcal{M} s on \mathcal{G}
- 3: Train diffusion model using Eq.9 with (X^*, Y, \mathcal{G})
- 4: // Phase 2: Adapt to Real Scenarios
- 5: **for** each (X^*, Y) in \mathcal{D}_{real} **do**
- 6: Fine-tune model using Eq.9 with (X^*, Y, \mathcal{G})
- 7: **end for**
- 8: // Inference on Real New Cases
- 9: **Input:** Observation Y_{new} on \mathcal{G}
- 10: Compute structure-based prior X_{est} from Y_{new} using Eq.1
- 11: Sample initial noise $X_n \sim \mathcal{N}(0, I)$
- 12: **return** $\hat{X} \leftarrow \text{denoise}(X_n)$ with $(X_{est}, Y_{new}, \mathcal{G})$

patterns while maintaining stable, pattern-invariant feature extraction capabilities centered on network structure. With this ability to capture generalizable source-topology relationships, we can effectively utilize synthetic data from established models (e.g. IC, LT, SIS) for pretraining, followed by efficient adaptation to real-world scenarios through few-shot learning, thus providing a practical solution for source localization in settings where real-world propagation training data is limited. We provide the illustration of the above process in Algorithm 1, and in Section 5.3, we evaluate SIDL’s ability to bridge simulation and reality by comparing two approaches: one using pretrained models with few-shot learning, and another trained exclusively on real-world data.

5 Experiments

5.1 Experiment Settings

The basic settings of our experiments are shown below.

Datasets. Following Ling et al. [20] and Huang et al. [11], we use four real-world propagation data to evaluate SIDL. The real-world datasets include *Digg*, *Twitter*, *Android* and *Christianity*, in which the real propagation cascades are available. For each cascade in all sets, we designate the infected nodes at the first 10% of the propagation time as source nodes and take the network’s infection status at 30% of the propagation time as observation input. In the context of real-world applications, we often can only collect sufficient data for analysis after some time has elapsed since the occurrence of the event. Therefore, attempting to predict what initially happened in the process when we have observed enough propagation patterns at a certain degree of infection time is very much in line with the needs of real-world operations. Please refer to the Appendix D for specific details of the datasets.

Implementation Details. For each dataset, the ratio of training, validation, and testing portion is 6:1:1. For the diffusion framework of SIDL, we use $T = 500$ maximum diffusion timestep and linear schedule for noise scheduling. In the denoising network, we leverage a 2-layer graph convolutional network (GCN) to forward the LPSI estimation X_{est} condition. The GNN encoder and decoder comprise 3-layer GCNs with a hidden dimension of 128. The residual GNN of the conditioning module is a 2-layer GCN, with a hidden

Table 1: Performance evaluation without pretraining under the real-world propagation. Best and second best (F1, RE, PR) are highlighted with bold and underlines respectively.

Datasets	Digg			Twitter			Android			Christianity		
Methods	F1	RE	PR									
Netsleuth	0.006	0.003	0.000	0.160	0.181	0.143	0.142	0.105	0.219	0.128	0.099	0.181
LPSI	<u>0.544</u>	0.516	0.575	0.487	0.495	0.479	0.348	0.517	0.268	0.221	0.282	0.198
GCNSI	0.458	0.411	0.517	0.374	0.352	0.399	0.383	0.474	0.321	0.343	0.321	0.370
TGASI	0.472	0.406	0.564	0.362	0.327	0.405	0.388	0.462	0.335	0.377	0.339	<u>0.423</u>
SLVAE	0.479	0.565	0.416	0.353	0.424	0.302	<u>0.467</u>	0.588	0.387	<u>0.458</u>	<u>0.662</u>	0.351
DDMSL	0.517	<u>0.592</u>	0.459	<u>0.492</u>	<u>0.504</u>	<u>0.481</u>	0.448	0.540	<u>0.432</u>	0.417	0.481	0.368
SIDSL(Ours)	0.585	0.605	<u>0.566</u>	0.546	0.516	0.580	0.522	0.702	0.439	0.519	0.747	0.436
Δ	+7.5%	+2.2%	-1.6%	+11.0%	+2.4%	+20.6%	+11.8%	+48.3%	+1.6%	+13.3%	+12.8%	+3.1%

Table 2: Few-shot learning performance evaluation under the real-world propagation with pretraining (P) and without pretraining (NP). Results show performance using simulation data(IC and LT) for pretraining, and using limited real-world data (10%) for few-shot learning. The rule-based methods are omitted as they do not support pretraining. Best and second best are highlighted with bold and underline respectively.

Dataset		Digg			Twitter			Android			Christianity		
Methods	Config	F1	RE	PR									
GCNSI	P	0.382	0.398	0.367	0.291	0.251	0.346	0.194	0.145	0.293	0.159	0.174	0.146
	NP	0.162	0.201	0.136	0.003	0.015	0.002	<u>0.012</u>	0.029	0.008	<u>0.176</u>	<u>0.179</u>	0.178
TGASI	P	<u>0.406</u>	0.407	<u>0.405</u>	<u>0.292</u>	0.249	<u>0.353</u>	<u>0.212</u>	<u>0.184</u>	0.251	<u>0.206</u>	0.193	<u>0.221</u>
	NP	<u>0.297</u>	<u>0.303</u>	<u>0.291</u>	0.003	0.015	0.001	0.012	<u>0.062</u>	0.006	0.036	0.038	0.035
SLVAE	P	0.315	0.441	0.246	0.195	0.325	0.139	0.036	0.132	0.021	0.171	<u>0.215</u>	0.142
	NP	0.103	0.188	0.071	0.030	0.042	0.023	0.009	0.010	0.008	0.008	<u>0.016</u>	0.005
DDMSL	P	0.348	<u>0.504</u>	0.266	0.250	<u>0.359</u>	0.192	0.109	0.114	0.104	0.165	0.145	0.191
	NP	0.005	0.003	0.015	<u>0.101</u>	<u>0.095</u>	<u>0.108</u>	0.010	0.015	<u>0.009</u>	0.009	0.005	0.045
SIDSL(Ours)	P	0.571	0.623	0.527	0.518	0.503	0.547	0.547	0.567	0.529	0.483	1.000	0.319
	NP	0.384	0.409	0.362	0.120	0.118	0.143	0.018	0.253	0.015	0.432	0.757	0.309
Δ	P	+41%	+24%	+30%	+77%	+40%	+55%	+158%	+208%	+81%	+134%	+365%	+44%
	NP	+29%	+35%	+24%	+19%	+24%	+32%	+50%	+308%	+67%	+145%	+323%	+74%

dimension of 8. The learning rate is searched from 0.01, 0.005, 0.001, and the maximum number of training epochs is set to 500 for all datasets. We train our model using Adam optimizer and a learning rate scheduler with a linear decay. Our model is trained on a single NVIDIA GeForce RTX 2080 Ti. The code implementation can be found at <https://anonymous.4open.science/r/ASLDiff-4FE0>.

Baselines. Following previous works [20, 37], we selected two representative heuristic methods, *i.e.*, Netsleuth [22] and LPSI [34], and deep learning methods, *i.e.*, GCNSI [5], TGASI [10] and recent generative deep learning methods SLVAE [20], DDMSL [37]. These baselines are all state-of-the-art (SOTA) multi-source localization methods in their domains. Please refer to the Appendix E for specific information of baselines and our method.

Metrics. Following previous works [31], we adopt three metrics: 1) F1-score (F1): The harmonic mean of recall and precision, emphasizing the balance between precision and recall; 2) Recall (RE): The proportion of positive cases (source nodes) that are correctly identified, focusing on the model’s ability to detect all relevant instances; 3) Precision (PR): The proportion of actual positive cases among the samples judged as positive, highlighting the model’s ability to avoid false positives. We do not use accuracy since in highly imbalanced datasets with rare positive cases, accuracy is a misleading metric as it can achieve deceptively high values by simply predicting the

majority (negative) class, thus failing to capture the model’s true discriminative capability for the minority class (source) of interest.

5.2 Overall Performance on Real-world Datasets

To evaluate the real-world performance of our proposed method against baselines, we conduct direct training and testing on four real-world datasets for all methods. The experimental results are presented in Table 1. Our proposed SIDSL outperforms all baselines across nearly all metrics on all datasets. Specifically, SIDSL achieves F1 scores that exceed the second-best baseline by 7.5%, 11.0%, 11.8%, and 13.3% on Digg, Twitter, Android, Christianity, respectively, which demonstrates SIDSL’s superior ability to predict source nodes despite their sparsity.

Further, we have the following findings. **First**, compared to other generative methods like DDMSL and SLVAE which rely purely on data-driven approaches, SIDSL demonstrates superior performance, validating the effectiveness of incorporating structural prior information on top of purely data-driven distribution learning. **Second**, compared to rule-based and pure learning-based methods, generative methods (SIDSL, DDMSL and SLVAE) show relatively strong performance, highlighting the advantages of effective source distribution modeling for source localization accuracy. LPSI and other

deep learning-based methods perform comparably well, as they successfully capture structural characteristics of source nodes through either rules or learning. Netsleuth demonstrates the weakest performance, likely due to its design for SI propagation patterns [13, 22], which limits its generalization to more complex patterns and larger network scales in real-world data. **Finally**, comparing the performance across different datasets, the most substantial improvement of SIDSLS is observed on the Christianity dataset (13.3% in F1), likely because its relatively smaller network scale (in terms of nodes and edges) makes topological features more easily capturable by all methods, thereby amplifying the benefits of structural prior knowledge injection in SIDSLS.

5.3 Performance in Low Data Regime

To assess how deep learning methods leverage simulation data pretraining for few-shot and zero-shot learning in real-world source localization, we generate pretraining data using standard IC and LT propagation models (1:1 ratio) across four networks. These models effectively capture social network dynamics by representing real-world interaction randomness, cumulative influence, and topology-based spread patterns. For few-shot learning (P), we fine-tune the pretrained models with limited real data (10%). We also compare the performance of all the methods trained on equivalent data volumes but without pretraining (NP). For zero-shot learning, all the methods are directly tested after being pretrained.

Few-shot analysis. The result of few-shot learning is shown in Tabel 2. Rule-based methods (Netsleuth and LPSI) are not shown here because they are training-free. SIDSLS demonstrates superior performance compared to all baseline methods across all datasets, both with and without pretraining, achieving improvements of 24%~365% across all metrics. Further, we have the following two findings. **First**, SIDSLS demonstrates enhanced performance advantages over baselines when leveraging pretraining, as evidenced by the improvement percentages across datasets. Among the improvements (Δ) across different datasets and metrics, 8 out of 12 cases show larger gains with P compared to NP. This consistent pattern of larger improvements with pretraining demonstrates SIDSLS’s superior ability to effectively leverage information from simulated data to real-world scenarios in limited-data settings. **Second**, the baseline methods exhibit varying capabilities in utilizing pretraining. While the non-generative method TGASI shows consistent improvements with pretraining across datasets, generative approaches like SLVAE and DDMSL demonstrate limited and unstable gains (e.g., SLVAE’s F1 only increases from 0.009 to 0.036 while DDMSL’s F1 only increases from 0.010 to 0.109 on Android). This contrast highlights the generative models’ sensitivity to distribution shifts between simulated and real-world data, whereas our method achieves robust and significant improvements through effective incorporation of pattern-invariant structural prior information.

Zero-shot analysis. The results of zero-shot learning are shown in Table 3. SIDSLS significantly outperforms all baseline methods across all datasets, achieving improvements of 85%~364% in F1 scores. The performance gap is particularly notable on Android and Christianity datasets, where SIDSLS achieves over 0.9 recall while maintaining reasonable precision. In contrast, baseline methods show limited zero-shot transfer capability with F1 scores mostly below 0.3.

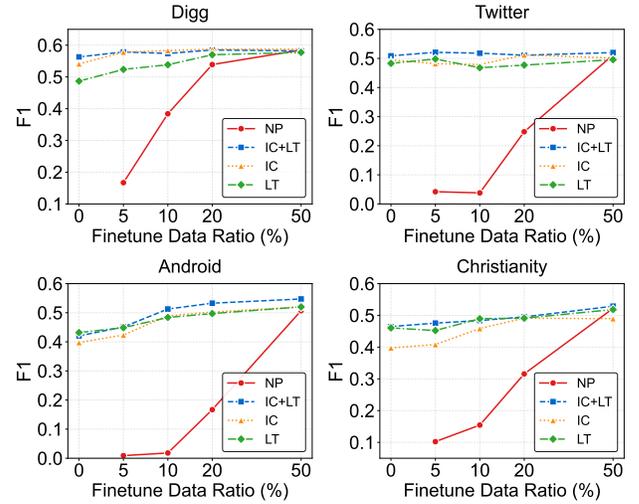


Figure 4: Few-shot performance of SIDSLS with different pre-training dataset compositions (IC+LT, IC, LT), followed by finetuning on real-world propagation data. "NP" denotes "non-pretrained".

Compared to rule-based methods in Table 1, zero-shot SIDSLS still outperforms them in most metrics. These findings demonstrate SIDSLS’s superior ability to leverage knowledge from simulation data without fine-tuning.

5.4 Analysis of Pretraining on Simulation Data

To investigate how pretraining on simulation data affects model’s few-shot/zero-shot performance under different data volume, we evaluate SIDSLS’s F1 scores across pretraining dataset with different propagation model combinations (IC, LT, and IC+LT with ratio 1:1) and compare it with non-pretrained (NP) SIDSLS. We then test them under different ratios of the finetune dataset of real-world propagation data. The results of four datasets are shown in Figure 4.

Finetune data volume analysis. Across all four datasets, models with pretraining consistently outperformed non-pretrained (NP) models in low-data scenarios. Under IC+LT combination, the pretrained models achieved optimal F1 scores on Digg and Twitter using just 5% of training data, while NP models required 50%. Similarly, on Android and Christianity datasets, pretrained models reached near-peak performance (within 0.033 and 0.018 F1) with only 20% of the data, compared to NP models which needed 50%. These findings suggest that aligning pretraining patterns with specific propagation characteristics benefits model performance.

Pattern analysis. On the Digg dataset, SIDSLS pretrained with IC+LT and IC configurations outperform that using LT, indicating that IC patterns better align with Digg’s propagation characteristics, as also shown in [39]. For Android and Christianity Q&A datasets, IC+LT and LT configurations performed better in few-shot scenarios. This is likely because user participation in these platforms’ information cascades (e.g., comments) is driven by cumulative influence from multiple sources (LT) rather than single independent triggers (IC). On Twitter, IC+LT configuration performs the best, demonstrating that pattern diversity enhances pretraining effectiveness on this dataset.

Table 3: Zero-shot performance evaluation with pretraining under the real-world propagation. Results show performance without finetuning when using simulation data(IC and LT) for pretraining. The rule-based methods are omitted as they do not support pretraining. Best and second best are highlighted with bold and underline respectively.

Dataset	Digg			Twitter			Android			Christianity		
	F1	RE	PR									
GCNSI	0.195	0.273	0.152	<u>0.203</u>	0.162	<u>0.271</u>	0.103	0.079	<u>0.148</u>	0.030	0.060	0.020
TGASI	0.189	0.281	0.142	0.199	0.172	0.236	<u>0.109</u>	<u>0.107</u>	0.110	0.059	0.067	0.052
SLVAE	0.278	0.236	<u>0.337</u>	0.172	0.205	0.148	0.004	0.025	0.002	0.080	<u>0.140</u>	0.056
DDMSL	<u>0.291</u>	<u>0.339</u>	0.264	0.203	<u>0.272</u>	0.162	0.019	0.013	0.038	<u>0.133</u>	0.127	<u>0.147</u>
SIDSL(Ours)	0.539	0.629	0.472	0.509	0.569	0.460	0.506	0.914	0.355	0.465	1.000	0.305
Δ	+85%	+86%	+40%	+151%	+109%	+70%	+364%	+754%	+140%	+250%	+614%	+108%

Table 4: Ablation study. The relative performance change of each ablation against SIDSL is reported.

Dataset		Digg			Christianity		
Data	Ablation	F1	RE	PR	F1	RE	PR
100%	SIDSL	100%	100%	100%	100%	100%	100%
	SIDSL w/o SBD	-34%	-47%	-16%	-19%	-49%	+2%
	SIDSL w/o GNN-LP	-7%	-9%	-5%	-4%	-6%	-5%
10%	SIDSL	100%	100%	100%	100%	100%	100%
	SIDSL w/o SBD	-46%	-54%	-36%	-35%	-55%	-24%
	SIDSL w/o GNN-LP	-17%	-16%	-17%	-21%	-34%	-17%

5.5 Ablation Study

We conduct ablation studies to investigate the significance of key components in SIDSL. For the first ablation model, SIDSL w/o SBD (structure-prior biased diffusion), instead of computing estimation through Equation 1, we directly substitute it with the node having the highest closeness centrality in the infection subgraph. For the second ablation, SIDSL w/o GNN-LP, we replace the observation state encoding with a standalone GNN while maintaining an equivalent parameter count.

Table 4 presents the ablation results on Digg and Christianity datasets under both full (100%) and limited (10%) training data conditions. Removing either component leads to decreased overall performance in both 100% and 10% data conditions. There are two more findings: **First**, removing SBD leads to more severe performance degradation compared to removing GNN-LP, with the impact most pronounced in recall metrics. The substantial decline in recall (TP/(TP+FN)) indicates a reduction in true positive predictions (as TP+FN remains constant for the same test set). Notably, the decline in recall exceeds that of precision (TP/(TP+FP)), which also implies a decrease in positive predictions (TP+FP). This pattern suggests that removing SBD causes the model to predict fewer positive samples overall, demonstrating SBD’s role in maintaining the model’s predictive balance. **Second**, the impact of removing GNN-LP is more pronounced under limited data conditions, with F1 scores dropping by -17% to -21% at 10% data versus -4% to -7% at full data. This demonstrates GNN-LP’s particular effectiveness in low-resource scenarios.

5.6 Time Cost Analysis

We present a detailed comparison of the computational cost of our proposed model against baselines in the Digg dataset, which has one of the largest networks for better comparison. Results are shown in Figure 5. While our model’s original training duration

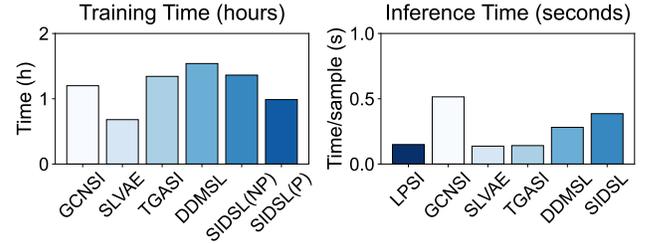


Figure 5: Training and inference time (per sample) comparison on Digg dataset. "NP" denotes the "non-pretrained" version. "P" denotes using the scheme of "pretraining+finetuning", and the overall training time is reported.

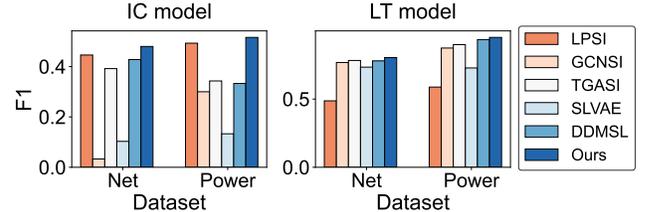


Figure 6: Performance under IC and LT propagation patterns. is the second highest, our model requires less time to pretrain on the synthetic data and finetune than initially training on the real data (-25%), which makes the training time the second lowest.

In inference time, our model is the second highest because of the iterative nature of the DDPM-based denoising process. We opt for DDPM as our foundation due to its classical design and proven effectiveness. It’s worth highlighting that SIDSL’s architecture is fully compatible with more computationally efficient diffusion variants, such as DDIM [28], which could substantially reduce the current computational overhead. This flexibility, combined with our model’s transfer capabilities, makes SIDSL particularly resource-efficient in practical deployments.

5.7 Results on Synthetic Propagation Patterns

It is a common practice in previous works [20, 31, 37] to evaluate methods’ performance on synthetic data with established propagation patterns. Following [20, 31], we conducted experiments on two commonly evaluated networks: *Net Science (Net)* and *Power Grid(Power)*. Using both IC and LT models, we simulated 100 steps until convergence to generate two sets of synthetic data. Strong baseline models were trained and tested on these datasets, with results shown in Figure 6. Our method outperforms the baselines,

achieving F1 score improvements of 7.6% and 4.7% under the IC model, and 2.7% and 1.7% under the LT model on the two datasets respectively. These results demonstrate our method’s ability to accurately identify sources with established propagation patterns.

6 Conclusion

In this paper, we presented SIDL, a structure-prior informed diffusion framework for robust source localization in graphs with limited real-world propagation data. By incorporating topology-aware priors through graph label propagation, a GNN-parameterized label propagation module, and a structure-prior biased denoising process, our framework effectively addresses three key challenges: unknown propagation patterns, complex topology-propagation relationships, and class imbalance issues. Extensive experiments on four real-world datasets demonstrate SIDL’s superior performance, achieving 7.5-13.3% improvements in F1 scores over state-of-the-art methods. More importantly, through effective pretraining with synthetic data, SIDL maintains robust performance with only 10% training data, showing remarkable advantages in both few-shot and zero-shot learning scenarios. These results validate our framework’s strong generalization capability and practical applicability in real-world source localization tasks where labeled data is scarce. Future directions include improving model efficiency through hierarchical decomposition approaches to handle larger-scale networks while maintaining localization accuracy.

References

- [1] Syed Shafat Ali, Tarique Anwar, and Syed Afzal Murtaza Rizvi. 2020. A revisit to the infection source identification problem under classical graph centrality measures. *Online Social Networks and Media* 17 (2020), 100061.
- [2] Massoud Amin and Phillip F Schewe. 2007. Preventing blackouts. *Scientific American* 296, 5 (2007), 60–67.
- [3] Le Cheng, Peican Zhu, Keke Tang, Chao Gao, and Zhen Wang. 2024. GIN-SD: source detection in graphs with incomplete nodes via positional encoding and attentive fusion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38, 55–63.
- [4] Jingtao Ding, Chang Liu, Yu Zheng, Yunke Zhang, Zihan Yu, Ruikun Li, Hongyi Chen, Jinghua Piao, Huandong Wang, Jiazhen Liu, and Yong Li. 2024. Artificial Intelligence for Complex Network: Potential, Methodology and Application. arXiv:2402.16887 [cs.SI] <https://arxiv.org/abs/2402.16887>
- [5] Ming Dong, Bolong Zheng, Nguyen Quoc Viet Hung, Han Su, and Guohui Li. 2019. Multiple rumor source detection with graph convolutional networks. In *Proceedings of the 28th ACM international conference on information and knowledge management*. 569–578.
- [6] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. 2017. Neural message passing for quantum chemistry. In *International conference on machine learning*. PMLR, 1263–1272.
- [7] Xizewen Han, Huangjie Zheng, and Mingyuan Zhou. 2022. Card: Classification and regression diffusion models. *Advances in Neural Information Processing Systems* 35 (2022), 18100–18115.
- [8] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems* 33 (2020), 6840–6851.
- [9] Dongpeng Hou, Yuchen Wang, Chao Gao, Xianghua Li, and Zhen Wang. 2024. New Localization Frameworks: User-centric Approaches to Source Localization in Real-world Propagation Scenarios. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*. 839–848.
- [10] Dongpeng Hou, Zhen Wang, Chao Gao, and Xuelong Li. 2023. Sequential attention source identification based on feature representation. *IJCAI* (2023).
- [11] Bosong Huang, Weihao Yu, Ruzhong Xie, Jing Xiao, and Jin Huang. 2023. Two-stage denoising diffusion model for source localization in graph inverse problems. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 325–340.
- [12] Lei Huang, Jie Qin, Yi Zhou, Fan Zhu, Li Liu, and Ling Shao. 2023. Normalization techniques in training dnms: Methodology, analysis and application. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2023).
- [13] Matt J Keeling and Ken TD Eames. 2005. Networks and epidemic models. *Journal of the royal society interface* 2, 4 (2005), 295–307.
- [14] David Kempe, Jon Kleinberg, and Éva Tardos. 2003. Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*. 137–146.
- [15] Jeffrey O Kephart and Steve R White. 1993. Measuring and modeling computer virus prevalence. In *Proceedings 1993 IEEE Computer Society Symposium on Research in Security and Privacy*. IEEE, 2–15.
- [16] William Ogilvy Kermack and Anderson G McKendrick. 1927. 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 (1927), 700–721.
- [17] Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).
- [18] Theodoros Lappas, Evimaria Terzi, Dimitrios Gunopulos, and Heikki Mannila. 2010. Finding effectors in social networks. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. 1059–1068.
- [19] Chen Ling, Tanmoy Chowdhury, Jie Ji, Sirui Li, Andreas Züfle, and Liang Zhao. 2024. Source Localization for Cross Network Information Diffusion. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 5419–5429.
- [20] Chen Ling, Junji Jiang, Junxiang Wang, and Zhao Liang. 2022. Source localization of graph diffusion via variational autoencoders for graph inverse problems. In *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*. 1010–1020.
- [21] Wuqiong Luo, Wee Peng Tay, and Mei Leng. 2013. Identifying infection sources and regions in large networks. *IEEE Transactions on Signal Processing* 61, 11 (2013), 2850–2865.
- [22] B Aditya Prakash, Jilles Vreeken, and Christos Faloutsos. 2012. Spotting culprits in epidemics: How many and which ones?. In *2012 IEEE 12th international conference on data mining*. IEEE, 11–20.
- [23] Ryan Rossi and Nesreen Ahmed. 2015. The network data repository with interactive graph analytics and visualization. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 29.
- [24] Xiaolei Ru, Jack Murdoch Moore, Xin-Ya Zhang, Yeting Zeng, and Gang Yan. 2023. Inferring patient zero on temporal networks via graph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37, 9632–9640.
- [25] Devavrat Shah and Tauhid Zaman. 2011. Rumors in a network: Who’s the culprit? *IEEE Transactions on information theory* 57, 8 (2011), 5163–5181.
- [26] Devavrat Shah and Tauhid Zaman. 2012. Rumor centrality: a universal source detector. In *Proceedings of the 12th ACM SIGMETRICS/PERFORMANCE joint international conference on Measurement and Modeling of Computer Systems*. 199–210.
- [27] Sushila Shelke and Vahida Attar. 2019. Source detection of rumor in social network—a review. *Online Social Networks and Media* 9 (2019), 30–42.
- [28] Jiaming Song, Chenlin Meng, and Stefano Ermon. 2020. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502* (2020).
- [29] Yinzhou Tang, Huandong Wang, and Yong Li. 2023. Enhancing Spatial Spread Prediction of Infectious Diseases through Integrating Multi-scale Human Mobility Dynamics. In *Proceedings of the 31st ACM International Conference on Advances in Geographic Information Systems* (<conf-loc>, <city>Hamburg</city>, <country>Germany</country>, </conf-loc>) (SIGSPATIAL ’23). Association for Computing Machinery, New York, NY, USA, Article 43, 12 pages. doi:10.1145/3589132.3625586
- [30] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).
- [31] Junxiang Wang, Junji Jiang, and Liang Zhao. 2022. An invertible graph diffusion neural network for source localization. In *Proceedings of the ACM Web Conference 2022*. 1058–1069.
- [32] Junxiang Wang and Liang Zhao. 2018. Multi-instance domain adaptation for vaccine adverse event detection. In *Proceedings of the 2018 World Wide Web Conference*. 97–106.
- [33] Zhen Wang, Dongpeng Hou, Shu Yin, Chao Gao, and Xianghua Li. 2024. Joint Source Localization in Different Platforms via Implicit Propagation Characteristics of Similar Topics. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, Kate Larson (Ed.). International Joint Conferences on Artificial Intelligence Organization, 2424–2432. doi:10.24963/ijcai.2024/268 Main Track.
- [34] Zheng Wang, Chaokun Wang, Jisheng Pei, and Xiaojun Ye. 2017. Multiple source detection without knowing the underlying propagation model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 31.
- [35] Duncan J Watts and Steven H Strogatz. 1998. Collective dynamics of ‘small-world’ networks. *nature* 393, 6684 (1998), 440–442.
- [36] Xovee Xu, Tangjiang Qian, Zhe Xiao, Ni Zhang, Jin Wu, and Fan Zhou. 2024. PGSL: A probabilistic graph diffusion model for source localization. *Expert Systems with Applications* 238 (2024), 122028.
- [37] Xin Yan, Hui Fang, and Qiang He. 2024. Diffusion model for graph inverse problems: Towards effective source localization on complex networks. *Advances in Neural Information Processing Systems* 36 (2024).

- [38] Cheng Yang, Hao Wang, Jian Tang, Chuan Shi, Maosong Sun, Ganqu Cui, and Zhiyuan Liu. 2021. Full-scale information diffusion prediction with reinforced recurrent networks. *IEEE Transactions on Neural Networks and Learning Systems* 34, 5 (2021), 2271–2283.
- [39] Shiqi Zhang, Jiachen Sun, Wenqing Lin, Xiaokui Xiao, Yiqian Huang, and Bo Tang. 2024. Information Diffusion Meets Invitation Mechanism. In *Companion Proceedings of the ACM on Web Conference 2024*. 383–392.
- [40] Liang Zhao. 2021. Event prediction in the big data era: A systematic survey. *ACM Computing Surveys (CSUR)* 54, 5 (2021), 1–37.
- [41] Kai Zhu, Zhen Chen, and Lei Ying. 2017. Catch'em all: Locating multiple diffusion sources in networks with partial observations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 31.
- [42] Kai Zhu and Lei Ying. 2014. Information source detection in the SIR model: A sample-path-based approach. *IEEE/ACM Transactions on Networking* 24, 1 (2014), 408–421.
- [43] Kai Zhu and Lei Ying. 2014. A robust information source estimator with sparse observations. *Computational Social Networks* 1, 1 (2014), 1–21.

A Comparison of multiple source localization methods

In Table 5, we compare the functionality, requirements, and application scenarios of mainstream source localization methods. “**Ind.**” refers to whether the method considers modeling the indeterminacy of source locations. “**Applicable patterns**” refers to the specific propagation pattern to which the method can be applied. “**Obs. input**” refers to the required input for the method to detect the sources. “**PI**” refers to whether the data-driven method is informed by prior knowledge.

From the demonstration, we can observe that our method is designed to be the most functional and capable of handling a broader range of real-world applications. Our method also requires less input data, which is more practical. The method proposed holds significant practical value and addresses the limitations of the existing methods. Additionally, as another method based on the diffusion model, DDMSL and TGASI require the propagation process data during training and the acquisition or calculation of parameters for the infectious model before source localization. This limitation restricts the model’s practical application value. Also, PGSL resembles SLVAE’s framework and merely utilizes a flow-based model to replace the VAE in SLVAE, while our diffusion model exhibits stronger distribution modeling capabilities. GINSD considers incomplete user data scenarios and utilizes a positional embedding module to distinguish incomplete nodes in the source inference process, and as we do not consider such circumstances, GINSD reduces to a simple GAT-based baseline similar to GCNSI.

It should also be noted that two recent works [19, 33] focus on source localization in a cross-platform setting, which is orthogonal to our research problem and thus not discussed.

B Analyzing source centrality in empirical data

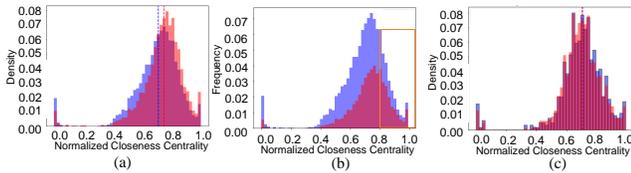


Figure 7: (a)(b) Normalized closeness centrality of infected users and sources for all cascades in *Digg*. The blue histogram shows the normalized closeness centrality distribution of infected nodes, while the red one shows that of source nodes. (a) is the density distribution of closeness centrality. The dashed line indicates the mean centrality for each node type. (b) is the frequency distribution of closeness centrality. The orange box highlights the part where centrality is above 0.8. (c) The closeness centrality(CC) probability density function of the predicted and ground truth source nodes on the *Digg* dataset. The blue histogram shows the normalized closeness centrality distribution of the ground truth sources, while the red one shows that of the predicted ones.

We believe that the source centrality assumption is not only common in most existing propagation patterns and real-world scenarios, as evidenced by the literature [1, 5], but also validated by the

competitive performance on real-world datasets of baselines like LPSI and GCNSI, which are devised based on similar assumptions. We show the analytical results demonstrating the effectiveness of the assumption in the following.

In our analysis of the real-world dataset *Digg*, we evaluate the normalized(max-min) closeness centrality density and frequency of the source nodes in the subgraph consisting of infected nodes to partially reflect the centrality characteristic of the sources. The closeness centrality (CC) specifically reflects the node topological distance to all other nodes in the subgraph, rather than its degree attribute. The result is shown in Figure 7(a). From Figure 7(a), the mean normalized closeness centrality of sources is higher than the average of all infected nodes, and source nodes cover over 63% of the nodes with the centrality score exceeding 0.8, as shown in Figure 7(b). The overall results demonstrate the crucial role of source nodes in the information diffusion process and their higher likelihood of being central to the network structure within the cascades.

Our proposed method, rather than strictly adhering to the centrality in outputting prediction results, exhibits stronger expressive capabilities. Intuitively, on homogeneous networks—where the probability of propagation along network edges is the same and fixed—the assumption can be strictly applied to locate the source of propagation. Such a propagation pattern that strictly obeys the centrality assumption is an indispensable subset that can be covered by the propagation patterns our model can characterize. As the proposed model leverages a simulated label propagation conditional module based on the centrality assumption but employs a graph neural network to learn the influence of the network’s heterogeneous topology from the data, other circumstances can also be modeled when learning from the data within our flexible data-driven framework. We have statistically analyzed the closeness centrality (CC) probability density function of the source nodes predicted by our trained SIDS model on the *Digg* dataset and compared it with the ground truth centrality of the source nodes in Figure 7(c). The mean and standard deviation for the CCs of the predicted sources are 0.7020 and 0.1444, and that for the CCs of the ground truth sources are 0.7044 and 0.1567, showing that there is no harmful bias in our method’s prediction. This statistical result indicates that our model captures the source distribution observed in empirical data, not just theoretical derivations. Our method not only uses knowledge to guide inference to accelerate learning but also learns distribution patterns beyond the knowledge, from the data.

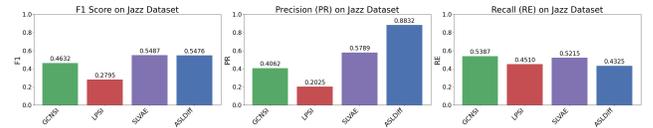


Figure 8: Additional experiments for simulated SIR scenarios on basic performance comparison.

Table 5: Comparison of different source localization methods. Ind.: Indeterminacy. Obs. input: Observation input. PI: Prior-knowledge informed

Category	Method	Ind.	Applicable patterns	Obs. input	PI
Rule-based	NetSleuth([22])	✗	SI	single snapshot	-
	OJC([41])	✗	SI, SIR	single snapshot	-
	LPSI([34])	✗	any patterns	single snapshot	-
Data-driven	GCNSI([5])	✗	any patterns	single snapshot	✓
	IVGD([31])	✗	IC	single snapshot	✗
	SLVAE([20])	✓	any patterns	single snapshot	✗
	SLDiff([11])	✗	any patterns	multiple snapshot	✗
	TGASI([10])	✗	any patterns	multiple snapshot	✗
	DDMSL([37])	✓	any patterns	multiple snapshot	✓
	PGSL([36])	✓	any patterns	single snapshot	✗
	GINSI([3])	✗	any patterns	single snapshot	✗
	NFSL([9])	✗	any patterns	single snapshot	✓
	SIDSL(Ours)	✓	any patterns	single snapshot	✓

C Additional Results of Performance under Other Propagation Patterns

We conduct additional experiments for simulated SIR scenarios on basic performance comparison. The results are shown in Figure 8. It can be seen that our model can still achieve competitive results compared to these baselines, proving our method’s applicability. The results also indicate that in terms of precision, ours achieved the highest score, more than 30% higher than the second-best SLVAE. Although we have a lower recall rate, a decrease of 0.09 only indicates around 1 node is not recalled from the ground truth, as only around 10 nodes are chosen as the ground truth sources in each infection. However, an increase of 0.3 in precision represents around 6 nodes correctly identified without false positives. Therefore, precision should be considered the more critical metric in source localization problems than recall when the F1 scores are similar, and SIDSL demonstrates the strongest competitiveness among the four methods.

D Dataset Description

The detailed description of the adopted datasets is presented as follows.

Table 6: Overview of networks in the datasets. "CC" denotes "clustering coefficient"

Dataset	Nodes	Edges	Mean Degree	CC
Digg	14511	194405	13.39	0.1353
Twitter	12619	309621	24.52	0.2962
Android	9958	42915	8.62	0.4121
Christianity	2897	30044	20.74	0.6027
Jazz	198	2742	13.84	0.6174
Net	1589	2742	1.72	0.6377
Power	4941	6594	1.33	0.0801

D.1 Real-world Propagation Dataset: Digg

The datasets selected, Digg and Twitter, represent real-world social networks where information propagation can be authentically traced, and are commonly used for evaluation in previous works [11, 20]. Both include propagation cascades demonstrating the time stamps and the information diffusion trace among users of each post or message. A connection network of all users is also provided in each dataset. Both datasets are pertinent to our study because they exemplify real-world dynamics of information spread.

Digg [23] is real-world social network data showcasing voting records of stories that made it to Digg’s front page in 2009, with each story’s spread counted as one diffusion cascade. We randomly choose 100 stories to form our dataset. The nodes (voters) involved in these stories form a subgraph of the original one, where the links represent the friendship of voters. The statistics of this friendship network are shown in Table 6. Drawing an analogy to the spread of a virus during a pandemic, it is often difficult to detect the virus at the very beginning, but after some time has passed—such as when the manifestation of symptoms—we can observe the infection status of the population. As a result, for each story cascade, we choose the top 10% of nodes and 30% of nodes as diffusion sources and observed influenced nodes based on their influenced time.

D.2 Real-world Propagation Dataset: Twitter

The Twitter [38] dataset is a collection of social network and public tweets written in English that were posted on the social media platform Twitter (a.k.a X) from March 24th to April 25th, 2012. The network statistics are shown in Table 6. Each tweet can be counted as one propagation cascade. Same as *Digg*, for each cascade, we choose the top 10% of nodes and 30% of nodes as diffusion sources and observed influenced nodes based on their influenced time.

D.3 Community Q&A Dataset: Android and Christianity

In Stack Exchange Q&A communities, knowledge dissemination occurs through a series of social interactions. When users post

questions, provide answers, leave comments, or cast votes, these activities form a complex social network structure. Each post is marked with specific tags (such as "google-pixel-2" in the Android community), and these posts with identical tags, arranged chronologically, create what we call "cascades." Our study focuses on two Stack Exchange communities with distinctly different themes: the Android community, which specializes in mobile device technical support, and the Christianity community, which centers on religious and theological discussions.

Same as *Digg*, for each cascade, we choose the top 10% of nodes and 30% of nodes as diffusion sources and observed influenced nodes based on their influenced time.

D.4 Synthetic Dataset

We synthesize propagation data under SIR, IC, and LT models on these three real-world networks: *jazz*, *network science* and *power grid*. These networks differ in scale, sparsity, and clustering characteristics, which enables us to investigate the model's performance on different types of networks. The statistic overview is presented in Table 6. For the propagation models, the propagation properties of the SIS infection model are determined by the inherent characteristics of the disease, applying homogeneity for all nodes/edges, i.e. the infection and recovery rates in SIR are constant; for the IC and the LT influence model, the heterogeneous propagation probability of each edge is considered, which is set to be inversely proportional to the degree of the target node. This aligns with real-world propagation patterns, where nodes with more connections tend to be less receptive to information from each neighbor.

- *Jazz* [23]. The provided dataset is a network of collaborations among Jazz musicians. Each node in the network represents a musician, and every edge connects two musicians who have performed together in a band. Rumors or infectious diseases are applicable to be propagated on such networks. We randomly choose 5% of nodes to be the spreading sources of each propagation and use SIS, IC, or LT models to simulate 100 steps or simulate until convergence.
- *Network Science (Net)* [23]. This is a coauthorship network of scientists working on network theory. Nodes represent scientists and edges represent collaborations. Influential information can be propagated on such networks. We randomly choose 0.5% of nodes to be the spreading sources of each propagation and use SIS, IC, or LT models to simulate 100 steps or simulate until convergence.
- *Power Grid (Power)* [35]. This is a topology network of the power grid network across the Western United States. In this network, each connection denotes a transmission line for electrical power. The nodes signify one of three components: a power generation unit, a transformer, or a distribution substation. Blackouts can be propagated on such a network. We randomly choose 0.5% of nodes to be the spreading sources of each propagation and use SIS, IC, or LT models to simulate 100 steps or simulate until convergence.

E Baselines

We compare the performance of SIDSLS against three state-of-the-art baselines of source localization methods using propagation

snapshot observations. To the best of our knowledge, these methods are the only ones that illustrate their superiority against other works on locating sources without knowing the underlying propagation pattern, which is the same as ours. The detailed information is presented as follows.

- **NetSleuth** [22] utilizes a minimum description length approach to filter nodes from multiple sources, yet it is exclusively designed to operate within the Susceptible-Infected (SI) model framework.
- **LPSI** [34] is a novel method for detecting multiple sources of information diffusion in networks without a predefined propagation model, leveraging the concept of source prominence and label propagation to identify probable sources based on local peaks in the propagation landscape. In our experiment, the parameter α in LPSI is determined by testing it among the values {0.1, 0.3, 0.5, 0.7, 0.9} for each evaluation dataset and then selecting the best one.
- **GCNSI** [5] introduces a deep learning approach for identifying multiple rumor sources in social networks without needing the underlying propagation model, using graph convolutional networks to enhance prediction precision through spectral domain convolution and multi-order neighbor information. The setting of this model follows the description in [5].
- **SLVAE** [20] is a probabilistic framework designed to tackle the challenge of source localization in graph diffusion problems using a variational autoencoder approach to quantify uncertainty and leverage prior knowledge. We follow the original implementation in the paper, tune the learning rate from 0.001 to 0.05, and select the best one.
- **TGASI** [10] is a sequence-to-sequence framework for multiple rumor source detection that considers heterogeneous user behavior in time-varying scenarios. It uses a GNN-based encoder to generate multiple features and a GRU-based decoder with temporal attention to infer sources. TGASI is designed with transferability and uses a unique loss function.
- **DDMSL** [37] proposes a novel probabilistic model for source localization and diffusion path reconstruction in complex networks. By formulating information propagation as a discrete diffusion process, DDMSL employs a reversible residual network to construct a denoising-diffusion model in discrete space. This approach allows for both accurate source identification and comprehensive reconstruction of information diffusion pathways.

F Experiments and Implementation Details

For each dataset, the ratio of training, validation, and testing portion is 6:1:1. For the diffusion framework of SIDSLS, we use $T = 500$ maximum diffusion timestep and linear schedule for noise scheduling. In the denoising network, we leverage a 2-layer graph convolutional network (GCN) to forward the LPSI estimation X_{est} . The GNN encoder and decoder comprise 3-layer GCNs with a hidden dimension of 128. The residual GNN of the conditioning module is a 2-layer GCN, with a hidden dimension of 8. The learning rate is searched from 0.01, 0.005, 0.001, and the maximum number of

training epochs is set to 500 for all datasets. In the few-shot learning experiments, the maximum pretrain/finetune epoch is set to 300. We train our model using Adam optimizer and a learning rate scheduler with a linear decay. Our model is trained on a single NVIDIA GeForce RTX 2080 Ti. The code implementation can be found at <https://anonymous.4open.science/r/ASLDiff-4FE0>.

G Limitations

Our proposed method also exhibits certain limitations. Our approach may, to some extent, depend on the accuracy of the advice provided by soft labels, despite our application of various sophisticated designs to enhance the model’s adaptability. As a result,

when confronted with more complex scenarios, our method might reveal limitations. On the other hand, the sampling speed of our multi-step diffusion model may be slower compared to some deep learning methods, which could become a bottleneck for applications requiring real-time localization. While computational constraints currently limit our model’s direct application to million-node networks, the core principles we developed can be integrated into hierarchical approaches. This hierarchical strategy would effectively reduce the network scale, allowing us to leverage our method’s proven strength in accurate source localization for moderately-sized networks. We will continue to conduct in-depth research in these areas.