New Localization Frameworks: User-centric Approaches to Source Localization in Real-world Propagation Scenarios

Dongpeng Hou
hdp0918@mail.nwpu.edu.cn
Northwestern Polytechnical University
Xi'an, Shaanxi, China

Yuchen Wang
wany810@mail.nwpu.edu.cn
Northwestern Polytechnical University
Xi'an, Shaanxi, China

Chao Gao
cgao@nwpu.edu.cn
Northwestern Polytechnical University
Xi'an, Shaanxi, China

Xianghua Li*
li_xianghua@nwpu.edu.cn
Northwestern Polytechnical University
Xi'an, Shaanxi, China

Zhen Wang
w-zhen@nwpu.edu.cn
Northwestern Polytechnical University
Xi'an, Shaanxi, China

Abstract

Source localization in social platforms is critical for managing and controlling the misinformation spreading. Despite all the recent advancements, existing methods do not consider the dynamic and heterogeneous propagation behaviors of users and are developed based on simulated data with strong model assumptions, limiting the application in real-world scenarios. This research addresses this limitation by presenting a novel framework for source localization, grounded in real-world propagation cascades from platforms like Weibo and Twitter. What's more, recognizing the user-driven nature of users in information spread, we systematically crawl and integrate user-specific profiles, offering a realistic understanding of user-driven propagation dynamics. In summary, by developing datasets derived from real-world propagation cascades, we set a precedent in enhancing the authenticity and practice of source identification for social media. Our comprehensive experiments not only validate the feasibility and rationale of our novel user-centric localization approaches but also emphasize the significance of considering user profiles in real-world propagation scenarios. The code is available at https://github.com/cgao-comp/NFSL.

CCS Concepts

• **Human-centered computing** → **Social network analysis**; **Social networks**.

Keywords

Source Localization, Real-world Cascades, User Profiles

ACM Reference Format:
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 (CIKM '24), October 21–25, 2024, Boise, ID, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3627673.3679796

CIKM '24, October 21–25, 2024, Boise, ID, USA.

ACM ISBN 979-8-4007-0436-9/24/10
https://doi.org/10.1145/3627673.3679796

1 Introduction

The rapid development of internet technologies has significantly boosted the popularity of social platforms and has changed the way of information propagation. While people enjoy considerable convenience in exchanging information, this also provides opportunities for the spread of malicious information such as fake news and rumors, posing significant economic and societal challenges [10, 11, 36]. It is necessary to locate harmful sources rapidly [7, 12, 32].

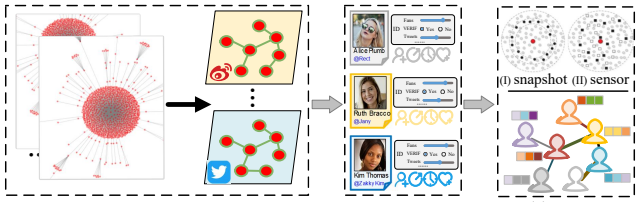

Figure 1: Considering that current localization methods largely rely on simulated cascade data from theoretical propagation dynamics, we model the localization framework directly from the real-world propagation cascades in social media. Recognizing users as pivotal in the propagation, the user profiles are crawled for user-centric source localization. The available observation information is consistent with the current snapshot and sensor based localization paradigms.

The source identification methods mainly include infection-status-based and sensor-based methods. The former relies on snapshots captured at some timestamps to locate the source, while the latter obtains propagation information from pre-deployed sensors within the network [22]. However, all of the source localization methods primarily rely on employing simulated data for localization performance verification based on theoretical propagation dynamics models, such as the Susceptible-Infected (SI) model and the Independent-Cascade (IC) model [3]. Though effective in some

specific environments, these methods do not consider the impact of heterogeneous user behaviors on propagation. In real-world scenarios, this oversight can lead to reduced scalability and practicality of these methods.

Against this backdrop, one necessary work is to bridge the source localization gap from theory to practice. As depicted in Fig. 1, we provide a novel view that refines the localization of information spread in social media by integrating real-world propagation patterns with user-centric data. Specifically, Fig. 1(a) illustrates the modeling of authentic propagation data from social media. In this model, we transform the raw propagation data into a graph based representation where users are depicted as nodes. And the edges represent direct propagation interactions, such as comments or reposts, forming links from successor users to a given user, thereby establishing a directed graph that captures the flow of information propagation cascades. Further, as shown in Fig. 1(b), considering that the propagation is driven by users, we crucially align and inject the crawled profiles[1] of nearly 1 million users from social platforms based on the unique user identification (UID) provided from the raw data. Lastly, Figure 1(c) demonstrates the source identification phase where we follow the paradigms in the source localization domain, including snapshot and sensor based classification. And only the available observation information obtained from the corresponding paradigm is used to infer the source.

Further, by applying the novel real-world data to the existing classification paradigm of the localization field, specifically sensor based and snapshot based localization, we consequently derive a framework for source localization in social network scenarios rather than relying on simulated data from propagation dynamics models. As illustrated in Fig. 2, leveraging collected historical propagation cascades from platforms like Weibo and Twitter, we construct a historical relationship network using the commonly used idea of "union shared users (i.e., the same UID) in different cascades" [23]. Then, based on the historical relationship network, source localization can be performed when new propagation cascades appear. Considering that the propagation is driven by users, we crucially align and inject the crawled profiles of nearly 1 million users from social platforms by indexing the UID in these cascades. Next, we explain the insights of transferring to real-world social media by following two localization classification paradigms.

From Sensors to Social Bots: When adapted to the social media environment, the concept of sensors in network science closely aligns with social bots. By strategically selecting some users or accounts as deployed bots in an area of attention, these bots function as monitors in capturing the information dynamic, such as the timing and direction of information spreading. Although deploying social bots offers continued monitoring, determining deployment strategies and the bot placements incur costs.

From Snapshots to Time-interval Capture: The snapshot based approach in network science can be seen as regularly capturing every user's states of the partial network in social media. More specifically, the crawler system periodically captures the propagation information based on a fixed time interval. To highlight the feasibility and rationality of information capture in large-scale social networks, we cannot expect to get comprehensive cascade data, like complete cascade graphs including directions and timestamps. Instead, only the participants in the area of attention are recorded when the time interval is achieved.

Based on the real-world datasets, we further consider the impact of user profiles on propagation and design localization methods following information capture paradigms of these two classifications. And the experiments demonstrate that user profiles have a significant impact on localization methods in real-world propagation scenarios. The major contributions are summarized as follows.

- Our study bridges the gap of source localization from theory to practice by introducing real-world datasets for source localization in social media, uniquely built upon crawled profiles of approximately one million users.
- Recognizing the often-overlooked user-centric nature of propagation dynamics, our research considers the effectiveness of user profiles during the source inference process. The utility of localization models can be enhanced.
- Through the ablation experiment of each feature in user profiles, the necessity of user profiles in real-world scenarios is validated. This underscores the necessary transition from relying solely on simulated propagation data to real-world cascade data including user profiles.

2 Related Work

2.1 Diffusion Models

Many diffusion models have been proposed to simulate the propagation data which is applied for evaluating the performance of localization methods, such as the Susceptible-Infected (SI) model [19, 35, 37] and the Susceptible-Infected-Recovered (SIR) model [26, 38]. However, a key limitation of these traditional models is their homogeneous nature. In reality, every individual in social networks has unique features, leading to diverse responses to the same information. Many researchers have further considered the heterogeneous diffusion models that characterize the difference of individuals [4, 13]. For instance, models like the heterogeneous SI (HSI) and heterogeneous SIR (HSIR) consider varied infection and recovery rates respectively. In addition to these diffusion models, influence models such as the Independent-Cascade (IC) and Linear Threshold (LT) models [5, 6] have been adopted. These models underscore the mutual influence dynamics between individuals. However, these simplified models are proposed based on strong assumptions. Therefore, it is necessary to utilize real-world propagation cascade data implicitly containing user interaction features to improve the rationality and practicality of downstream localization methods [9].

2.2 Source Localization Methods

There are two main classifications in the field of source localization, namely snapshot based and sensor based methods. The snapshot based methods infer sources based on the infection snapshots captured from various diffusion models. Studies on the snapshot branch are due to the convenience and low cost of snapshot acquiring. The innovative LPSI method introduces a label propagation process based on source prominence to locate the source. Dong et al. introduce a Graph Convolutional Network (GCN) based source

[1]The profiles of each user include verification status (indicating authoritativeness), number of tweets (representing tweet activity), registration date (indicative of user seniority), number of fans (denoting popularity), number of followings (indicating information seeking behavior), ratio of fans to followings (reflecting credibility).

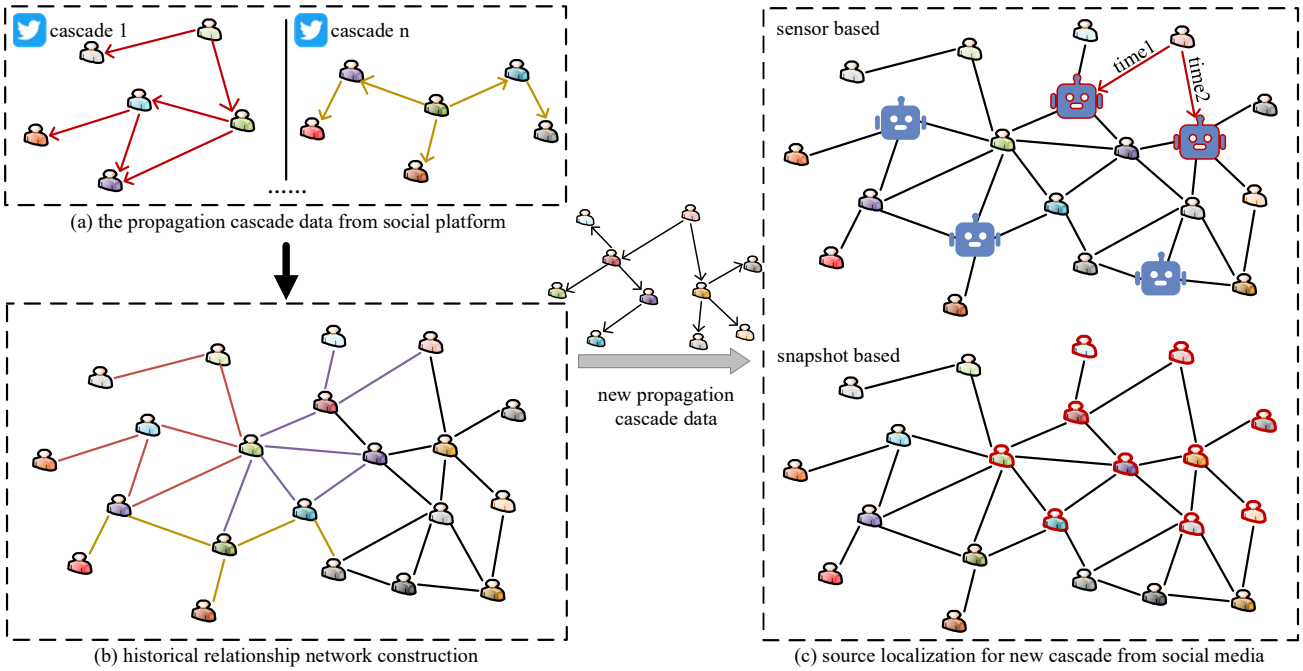

Figure 2: The source localization framework in social media based on two existing paradigms. The social bots based classification offers the advantage of prior deployment, serving as a monitor for information propagation. However, its limitations are twofold: firstly, it captures information only from deployed bots, resulting in the partial receptive field. Secondly, the deployment strategy significantly impacts the channels of information acquisition and requires certain costs. The time-interval capture based classification obtains the snapshots through periodic captures. Generally, only user's participation status (i.e., infection state in simulation models) is included in the snapshot [3, 33], and the cost of obtaining other information is still high.

identification model, GCNSI, to address the multiple rumor source detection problem [3]. Additionally, some methods consider the dynamic features of propagation before the source inference process, such as IVGD [29], MCGNN [25] and SL_VAE [14]. Another branch that involves higher overhead but captures more propagation information (e.g., direction) is sensor based localization, including statistical techniques of Monte Carlo simulations [1], multivariate Gaussian estimation [22], and Pearson correlation [28]. These methods, grounded in simulated data for localization, do not consider the special impact of various user profiles in the propagation process. In reality, individuals are the primary drivers of information spreading. Therefore, the effectiveness of these localization methods in real-world scenarios is limited.

3 Methods

3.1 Social Bots based Localization

3.1.1 Problem Definition. Based on K available experienced propagation cascades $C_k=(\mathcal{V}_k, \mathcal{E}_k, \mathcal{F}_k)$ $(1 \le k \le K)$ from a social platform, where $\mathcal{V}_k$ is the user set with UID in a social platform, $\mathcal{E}_k$ is the set of user's directed propagation interaction, and $\mathcal{F}_k$ is the feature set associated with the users. We then construct the historical relationship network $\mathcal{G}=(\mathcal{V}, \mathcal{E}, \mathcal{F})$ to represent a group of people characterized by topic relevance, friendship ties, or consistent attributes. Further, given a new cascade $C=(V, E, F)$ with only partial information observed from this focused community or circle $\mathcal{G}$, social bots based localization includes the bots deployment phase and source inference phase. It's worth noting that adhering to the localization paradigm and real-world experience means that we cannot be aware of the other user's interactions or arrival times outside of social bots to ensure the reasonableness of the information acquisition cost. Therefore, in the first phase, a certain strategy (e.g., greedy selection) is picked to deploy a certain ratio (denoted as ξ) of accounts as social bots in the focused group $\mathcal{G}$ in advance, among them, $O \triangleq \{o_i\}_{i=1}^{|V|*\xi}$ is regarded as a bot set. And all bots are divided into two types, denoted as $O = \{O_Y, O_N\}$, where a bot $o \in O_Y$ or $o \in O_N$ indicates that o records or does not observe the diffusion information, respectively. And the propagation information observed by bots in O_Y includes the time and direction from the new cascade. Then in the second phase, the source inference strategy is executed to locate the source s (the first user who initializes to spread the information) from the candidate set (denoted as $\mathcal{V} \setminus O$) based on $\mathcal{G}$ and the limited information recorded by fractional bots in O_Y. After the user is predicted ($\hat{s}$), rigorous indicators are used to evaluate the localization performance by comparing $\hat{s}$ with s.

3.1.2 Framework. There are two necessary phases in the bots based localization, namely the bot deployment phase and the source inference phase. The first bot deployment phase mainly focuses on the information greedy capture of the deployed bots. However, the greedy strategy implies higher time or space consumption for the

facility. Therefore, balancing the greedy extent and facility consumption is one of the biggest challenges in this phase. The general definition of a constraint pattern $\Phi(O) \leq O(|V|^2)$, which limits the space and time complexity, to overcome the unsuitable problem for large-scale scenarios caused by the vast overhead of the global greedy strategy.

$$Deployment(\mathcal{G}, O) \approx \min \sum_{v_i \in (\mathcal{V} \setminus O)} d(v_i, O), \quad s.t.\ \Phi(O) \leq O(|\mathcal{V}| log |\mathcal{V}|). \tag{1}$$

Algorithm 1: The optimized random greedy based deployment strategy for social bots

Input: Network topology of $\mathcal{G}(\mathcal{V}, \mathcal{E})$, deployment ratio ξ of bots.
Output: The deployed bot set O.

```
Initialize the bot set O;
Construct a degree sequence of V as K={v_i}_{i=1}^{|V|}, where degree(i)≥degree(j) for i<j;
while |O| ≤ |V| * ξ do
    Select a new bot o={K[i] |argmin_i K[i] ∉ O};
    Add o to O;
    Record the eccentricity of o as E_o;
    for n=(1, E_o) do
        Select a new bot o' = {K[j] |d(o,K[j])=n, argmax_j K[j] ∉ O};
        Add o' to O;
RETURN O;
```

Specifically, an acknowledged greedy deployment method with an $O(N^3)$ complexity possesses the capability to effectively capture global information ranging from hub nodes (central and influential users) to leaf ones (fringe users). With the implementation of this strategy, every user in the network can have at least one bot within each hop of the neighborhood. Inspired by this, it seems rational to focus on high-degree central users, then perform a Depth-First Search (DFS) to randomly select a user from each hop of neighbors for bot deployment. To ensure the selected bot chain is complete coverage for the network instead of a half chain, so if the number of deployments is less than the graph's diameter, the DFS strategy can be executed multiple times for the same central user. Although each iteration of DFS has a high degree of greediness, a significant drawback is the close hops between users chosen in different discrete DFS runs.

To boost the innate greediness in information capture, while staying within the prescribed constraint patterns, a more cost-efficient yet even more greedy strategy is put forward. Specifically, leaf nodes in the network usually possess a high eccentricity, coming close to the graph's diameter. Thus, by shifting focus from the entire network diameter to the eccentricities of these leaf nodes, one can sidestep the burdensome $O(N^3)$ computation of diameter, which becomes significantly taxing in larger-scale networks. Based on this, starting from a leaf node, one can maintain a permutation based on maximum degrees as heuristic information representing influence. At each step of the new bot selection, the most influential neighbor, as derived from this permutation, is picked as a new hub. This process continues until the eccentricity of the started leaf node is reached. Such a strategy not only emphasizes fringe information but also ensures that at each step, the most influential neighbor is considered which circumvents the close hops issue from suboptimal bot selections from multiple independent DFS runs. Furthermore, while existing methodologies often presume knowledge of the network diameter, the computational overhead associated with such a determination shouldn't be dismissed. The strategy now presented ingeniously bypasses the need to calculate this diameter, further reducing both temporal and spatial overheads. The specific implementation of this strategy is illustrated in Alg. 1.

Next, in the second inference phase, when a new unknown cascade $C(V, E, F)$ from our concerned social circle $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{F})$ is observed by bots, a punishment-oriented source likelihood estimator considering user profiles importance is designed as follows.

$$\hat{s} = \underset{v \in \mathcal{V}_\theta}{\operatorname{argmin}} \left(\sum_{o \in O_Y} d_{o,v} \right)^p * \left(\sum_{o \in O_Y} \otimes(t_o, d_{o,v}) \right)^q * rp(v) * \Psi(v). \tag{2}$$

Among them, due to the greedy deployment strategy, the sum of the distances from the source to all bots in O_Y should be less than a threshold θ. Therefore, the aim of the candidate screening process is to lock the source within an early small participation area. We define the remaining candidates as reliable candidates, denoted as $v \in \mathcal{V}_\theta$. Next, in the punishment score estimator, t_o represents the normalized time at which o records a diffusion event. There are four items for evaluating the punishment score of $v \in \mathcal{V}_\theta$. First, item $d_{o,v}$ uses the real diffusion path in the source inference phase. Second, item $\otimes(t_o, d_{o,v})$ considers that the real diffusion distance and normalized observed time have a high positive correlation. Third, item $rp(v)$ further decreases or increases the neighbor's punishment score of bots in O_Y or O_N, respectively. Fourth, $\Psi(v)$ evaluates the user's negative influence on propagation based on real-world features.

$$t_o = \begin{cases} t_o^- - \min\left(t_o^-\right) + \omega^+, & if\ 6 \leq \min\left(t_o^-\right),\ \forall\ o \in O_Y \\ t_o^- - \min\left(t_o^-\right) + \omega, & else\ if\ 4 \leq \min\left(t_o^-\right),\ \forall\ o \in O_Y \\ t_o^- - \min\left(t_o^-\right) + \omega^-, & else\ if\ 3 \leq \min\left(t_o^-\right),\ \forall\ o \in O_Y \end{cases}, \tag{3}$$

$$\otimes(t_o, d_{o,v}) \triangleq \max\left(\frac{t_o}{d_{o,v}}, \frac{d_{o,v}}{t_o}\right), \tag{4}$$

$$d_{o,v} = d(v_{\psi(o)}, v) + 1, \tag{5}$$

$$rp(v) = \begin{cases} \varrho_1^{|\pi(v) \cap O_Y|}, & if\ \exists\ o \in O_Y,\ v \in \pi(o) \\ \varrho_2^{|\pi(v) \cap O_N|}, & if\ \exists\ o \in O_N,\ v \in \pi(o) \end{cases}, \tag{6}$$

where t_o^- is the original recorded time of a diffusion from $o \in O_Y$, $\min(t_o^-)$ is the earliest recorded time, and t_o is the standardized time of bot o, ω^+, ω and ω^- represent the penalty item of the observed time regularization, and the punishment is $\omega^+ > \omega > \omega^-$. $\varrho_1 \in (0, 1)$ is a reward coefficient and $\varrho_2 > 1$ is a punishment coefficient. $\pi(\cdot)$ is the neighbor set. Here, each cascade standardizes all related

timestamps in the following manner: First, by subtracting the minimum observed time, denoted as min (t_o^-). Second, depending on the range of min (t_o^-), each cascade's different user reception times will follow only one specific branch. This means that for any given cascade, only one of w^+, w, or w^- will be activated. What's more, the purpose of time regularization is to address the problem that the observed time with a bigger timestamp applied in the penalty term of $\otimes(t_o, d_{o,v})$ has a weaker source detection effect than the standardized time. Specifically, the recorded time observed by deployed bots is relatively later when in the condition of the low influence, it takes at least 30 timestamps to successfully infect a susceptible node with 95% probability when the user's influence (i.e., infection rates) is smaller than 0.1. Based on this assumption, it can be easy to find that $\frac{d_{o,v}}{t_o}$ in $\otimes(t_o, d_{o,v})$ can hardly act as a constraint because $d_{o,v}$ is much smaller than t_o. The single valid term $\frac{t_o}{d_{o,v}}$ indicates that a candidate $v \in V_\theta$ far away from the earlier observed bots has a smaller penalty than the candidate close to these bots. On the other hand, if the earliest (i.e., first) recorded time observed by a bot is directly regarded as 1, the fine information about the timestamps will be ignored because a diffusion scenario with the earliest recorded timestamp of 1 has a completely different meaning than the earliest recorded timestamp of 100. The normalization of the real observed time in Eq. (3) of the bot can effectively solve these two issues.

As for the punishment items: Sometimes, the items $\sum_{o \in O_Y} d_{o,v}$ and $\otimes(t_o, d_{o,v})$ of Eq. (2) do not well distinguish a small number of candidates because these candidates have a similar score. Therefore, a punishment-reward mechanism is proposed to correct candidates' source likelihood probability. That is, the candidates who are the neighbors of observed bots (i.e., $o \in O_Y$) have a higher probability of becoming the diffusion source while the neighbors of unobserved bots (i.e., $o \in O_N$) have a lower probability of becoming the diffusion source. Here, due to the nature of the punishment based scoring rule, where a higher penalty equates to a higher score and, consequently, a lower likelihood of being the diffusion source, we apply different coefficients to candidates based on their proximity to early bots. For users close to early bots, we assign a positive coefficient less than 1, while for those farther away, we apply a penalty coefficient greater than 1.

What's more, we also consider user profiles with six heterogeneous features Ψ to evaluate their impact on propagation, emphasizing the fact that propagation online is primarily human-driven. In our study, a set of six user profiles $\mathcal{F}$ is incorporated, including verification status (indicating authoritativeness), number of tweets (representing tweet activity), registration date (indicative of user seniority), number of fans (denoting popularity), number of followings (indicating information seeking behavior), ratio of fans to followings (reflecting credibility). And we can effortlessly evaluate the propagation influence of every individual using any chosen model. Then based on the four items in Eq. (2), the user with the smallest punishment score is the predicted source.

Here, in light of the fact that we cannot fully obtain V and E in the new cascade C, we extract a subgraph based on $\mathcal{V}_\theta$ in the historical relationship network $\mathcal{G}$ to reconstruct the adjacency matrix A. Significantly, to emphasize the direct one-hop users who spread the information to the bots in O_Y, the corresponding edges in the adjacency matrix are additionally weighted with a coefficient γ. This procedure aims to strengthen those propagation paths that directly influence the bots to underscore the very few confirmed routes that are captured in the new cascade. Next, we evaluate the ordering of feature importance using the chi-squared test. We first normalize $\mathcal{F}'$ using Min-Max normalization, which transforms each feature to lie within the [0,1] interval [21], and using chi-squared test based statistical analysis to obtain the weight of six features. Then based on the normalized feature representations for each user, we assign different ranking weights to each feature to compute the user's influence. Subsequently, we utilize the normalized adjacency matrix $\hat{A}$ form A to aggregate the influence from neighbors, refining the final influence of each user. Lastly, the negative influence can be achieved by negating or taking the inverse. It should be noted that we intuitively consider the influence of user profiles in propagation from a statistical perspective. Given that we develop the adjacency matrix information and the user's feature matrix, using larger facility consumption and employing some deep learning approaches may uncover more implicit propagation information, such as GCN [34] or GAT [27].

Algorithm 2: User's negative influence evaluation

Input: Crawled profile matrix $\mathcal{F}$ with six features for each user, reliable candidate $\mathcal{V}_\theta$ of cascade C, historical relationship network $\mathcal{G}$

Output: Negative Influence scores for each user

$\mathcal{F}' \leftarrow MinMaxScaler(\mathcal{F})$;

Evaluate feature importance using *chi-squared test* to get scores $\chi^2(f')$;

Normalize scores to get weights $w(f') = \frac{\chi^2(f')}{\sum_{f'} \chi^2(f')}$;

for *each user* u **do**

  Influence$(u) \leftarrow \sum_{f'} w(f') \times \mathcal{F}'(u, f')$;

Extract a subgraph based on $\mathcal{V}_\theta$ in $\mathcal{G}$ to construct the adjacency matrix A of new cascade C;

for *each bot* i *in* O_Y **do**

  Index the *one-hop neighbor* j that spreads the info to i;

  $A_{ij} \leftarrow \gamma + A_{ij}$, γ>0;

Influence$(u) = \sum_v (D^{-\frac{1}{2}}(A+I)D^{-\frac{1}{2}})_{uv} \times$ Influence(v);

for *each user* u **do**

  NegativeInfluence$(u) \leftarrow 1/($Influence$(u) + \epsilon)$;

RETURN Negative Influence scores for each user;

3.2 Time-interval Capture based Localization

3.2.1 Problem Definition. Having similarly constructed the historical relationship network $\mathcal{G}$=$(\mathcal{V}, \mathcal{E}, \mathcal{F})$, given a time-interval size $\mathcal{T}$, we perform snapshot sampling every interval of $\mathcal{T}$ and can rapidly capture a series of observed cascade snapshots $\{C_{s_j} = (V_{s_j}, E_{s_j}, F_{s_j}) \mid j = 1, 2, 3, ...\}$ with only participation state information available from this concerned community $\mathcal{G}$. The salient advantages of this classification are the periodic capturing for global viewing of a network, and low overhead instead of bots selection and deployment in advance. Consequently, certain intrinsic constraints

emerge when we aim to capture information in a globally periodic manner while maintaining low costs. In this way, the depth and breadth of diffusion information captured can be limited when $\mathcal{T}$ is small. Specifically, we focus solely on a primary type of information, i.e., the participation state of each user, while neglecting more diverse types, such as the time and direction of information propagation each user receives. Each user in the cascade with timestamp s_j has an associated state defined by $Y_{s_j} = \{Y_{s_j}(v_1), Y_{s_j}(v_2), ...\}$. $Y_{s_j}(v_i) = 1$ if user v_i is observed to participate, and $Y_{s_j}(v_i) = 0$ otherwise.

3.2.2 Framework. For a series of propagation snapshots $(C_{s_1}, C_{s_2}, ...)$ pertaining to a new cascade $C = (V, E, F)$, $Y_{s_j}(v_i) = 1$ indicates that a user v_i participates a cascade at snapshot time s_j. Given participants in each timestamp, we can extract the snapshot based subgraph G_{s_j} or adjacency A_{s_j} from the historical relationship network $\mathcal{G}$, on which the feature matrix F aligns with each user according to UID. It becomes a natural choice to consider using both the RNN and the GNN. Because RNNs are aptly suited for handling time-sequenced data, reflecting the temporal evolution of the cascade, GNNs effectively aggregate and propagate feature information across the graph, leveraging the underlying structural properties with user features. Therefore, the encoder-decoder framework with deep graph module and recurrent module can be developed in this scenario. To ensure the structural consistency of each A_{s_j} across different timestamps which is a preprocessing idea for the deep learning module, and to perceive the connections to surrounding non-participant neighbors, for a single cascade C, we further extend the snapshot based subgraph with more interaction users by considering the one-hop non-participant neighbors from the participant users in the latest captured timestamp $max(s_j)$. Therefore, we extract a larger subgraph $\tilde{G}(\tilde{V}, \tilde{E})$ or $\hat{A}$ from the focused area $\mathcal{G}$, whose scale is larger than the latest captured snapshot $G_{max(s_j)}$. Then, each G_{s_j} is mapped onto $\tilde{G}$ and is denoted as $\hat{G}_{s_j}$. Simultaneously, the state and characteristics of users in each timestamp are retained and the topological consistency of different timestamps is also guaranteed.

$$\tilde{V} = \{u \mid v \in V_{max(s_j)}, u \in N^{\mathcal{G}}(v) \setminus V_{max(s_j)}\} \cup V_{max(s_j)}, \quad (7)$$

$$\tilde{E} = \{(v_i, v_j) \mid v_i, v_j \in \tilde{V} \text{ and } (v_i, v_j) \in \mathcal{E}\}, \quad (8)$$

where $N^{\mathcal{G}}(v)$ is the neighbor set of user v in the historical relationship network $\mathcal{G}$=$(\mathcal{V}, \mathcal{E}, \mathcal{F})$.

After obtaining a series of snapshot based subgraphs with topological consistency $\hat{G}_{s_1}, \hat{G}_{s_2}, ...$, a sequence-to-sequence model is designed to locate the source user of propagation. First in the encoder phase, to better solve the user-level based source localization task, some unique propagation dynamic features are designed for each user combined with user profiles to differentiate each unique user. There are six explicit dynamic indicators in timestamp s_j (denoted as $H_1^{s_j}$-$H_6^{s_j}$) are selected to characterize the time-varying state based participation features of an individual. Among them, the ratio of participant neighbors and the ratio of non-participant neighbors of v_i in timestamp s_j are shown in Eq. (9) and Eq. (10), respectively.

$$H_1^{s_j}(v_i) = \frac{\sum_{v_k \in N^{\hat{G}_{s_j}}(v_i)} Y_{s_j}(v_k)}{|N^{\hat{G}_{s_j}}(v_i)|}, \quad (9)$$

$$H_2^{s_j}(v_i) = \frac{|N^{\hat{G}_{s_j}}(v_i)| - \sum_{v_k \in N^{\hat{G}_{s_j}}(v_i)} Y_{s_j}(v_k)}{|N^{\hat{G}_{s_j}}(v_i)|}. \quad (10)$$

What's more, we also consider the normalized number of participant neighbors and the normalized number of non-participant neighbors of v_i in timestamp s_j are shown in Eq. (11) and Eq. (12), respectively.

$$H_3^{s_j}(v_i) = \frac{\sum_{v_k \in N^{\hat{G}_{s_j}}(v_i)} Y_{s_j}(v_k)}{\max_{u \in V}\left(N^{\hat{G}_{s_j}}(u)\right)}, \quad (11)$$

$$H_4^{s_j}(v_i) = \frac{|N^{\hat{G}_{s_j}}(v_i)| - \sum_{v_k \in N^{\hat{G}_{s_j}}(v_i)} Y_{s_j}(v_k)}{\max_{u \in V}\left(N^{\hat{G}_{s_j}}(u)\right)}. \quad (12)$$

Here, as for these six dynamic indicators, the inspiration for these features is drawn from an existing work [8], which highlighted the challenges of modeling with limited information from propagation snapshots that only provide binary participation statuses (1 or 0) without any prior knowledge. The authors emphasized the necessity of explicitly constructing propagation features to reduce the learning difficulty for models. They considered original state features and the ratio of the neighbor's states at different propagation timestamps. Building upon this, we introduced additional considerations: the normalized total number of participant neighbors (H_3) and the normalized total number of non-participant neighbors (H_4). These features shift the focus from the ratio to the exact number of a node's neighbor states. For instance, consider a node with only one neighbor, which is a participant. In this case, H_1 would have a relatively large participant ratio indicator, but such a node cannot be conclusively identified as a source. Therefore, while H_1 and H_2 account for the participant and non-participant ratios of neighbors, H_3 and H_4 evaluate the actual numbers of participant and non-participant neighbors. This comprehensive characterization offers a more nuanced understanding of a node's neighboring participation states. After these six propagation dynamic features are obtained, the complete feature embedding H^{s_j} in the timestamp s_j can be obtained by concatenating another six normalized user profile features in F. Then the feature embedding $H^{s_j}(v_i)$ of user v_i in the timestamp s_j is constructed.

Next, a sequential decoder needs to be designed to predict the source based on the discrete temporal embedding. Propagation information from earlier stages is the prior knowledge for later source localization, and the inverse process of forward diffusion is the localization task. So we use a bidirectional lightweight time series module, Bi-LSTM, for source prediction. However, while Bi-LSTM typically processes sequence data, it is also expected to learn the network topology information inductively. Hence, rules are required when incorporating graph data as input. Intuitively and feasibly, the theoretical principle is to design a graph based recurrent layer in the update process, which is similar to the message aggregation process of GNN. The vectorization version of the graph

based recurrent layer for all nodes is shown in Eq. (13).

$$h_t = \sigma\Big(\hat{A}H^{s_j}W_{ih} + b_{ih} + \hat{A}h_{t-1}W_{hh} + b_{hh}\Big), \widehat{A} = D^{-\frac{1}{2}}(\tilde{A} + I_n)D^{-\frac{1}{2}}, \tag{13}$$

where σ is the activation function, h_t is the hidden layer output for all nodes in the timestamp s_t, h_{t-1} is the hidden state of the layer in the timestamp s_{t-1} or the initial empty hidden state, W_{ih} and W_{hh} are the learnable weight matrices, b_{ih} and b_{hh} are bias. Further, after using the softmax activation, the source probability for each timestamp can be derived. Every node will possess a probability at each timestamp, and the node with the highest probability is identified as the propagation source.

In summary, the GNN efficiently embeds user influence information into the message-passing process, leveraging its robust embedding capabilities for node attributes. The LSTM model has a outstanding ability to process time series data. In our method, The embedding of the snapshot sequence $H^{s_1}, H^{s_2}, H^{s_3}, ...H^{s_\zeta}$ is inputted into a graph based Bi-LSTM model, and the forward LSTM infer the source based on the snapshots from the timestamp s_1 to s_ζ, and the forward output predicted binary classification for all nodes of the hidden layer in the timestamp s_j is denoted as $\overrightarrow{h_j} \in \mathbb{R}^{|V|\times 2}$. Similarly, the backward LSTM processes the source inferring process from the timestamp s_ζ to s_1, and the backward output predicted of the hidden layer is represented as $\overleftarrow{h_j} \in \mathbb{R}^{|V|\times 2}$. Finally, the two hidden layer states of the timestamp s_j are concatenated, i.e., $R_j = [\overrightarrow{h_j}, \overleftarrow{h_j}] \in \mathbb{R}^{|V|\times 4}$. Then the output from Bi-LSTM for each node $\hat{R}_j(v)$ has 4 elements. Therefore, the Bi-LSTM model processes the embedded snapshot sequence, with forward and backward LSTMs inferring the source based on timestamps and predicting binary classifications for nodes. The dual hidden layer states at each timestamp are concatenated to form a comprehensive representation. This combination of GNN and Bi-LSTM allows for effective modeling of both spatial and temporal dimensions in the data, enhancing the accuracy of our source localization process. Finally, a loss function is required to realize the parameters optimization of the encoder-decoder. The weighted binary classification based cross entropy is used to pay more attention to the minority group of source users.

4 EXPERIMENT

4.1 Datasets and Baseline

We used three datasets collected from two real-world social platforms, Weibo and Twitter, for source localization, namely Weibo [18], Twitter15, and Twitter16 [15, 17]. Furthermore, we have crawled user profile information from six dimensions for each user based on the UID in the propagation cascade, achieving user profile alignment on the social platform. The relevant information of the three datasets is shown in Tab. 1.

To demonstrate the validity and novelty of the two proposed localization methods, some SOTA methods are selected from both snapshot and sensor based localization for comparative analysis. For sensor based classification, we compare our method with GRSL [31], DISGE [35], M-PTV [19], GMLA [20], and SNF [28]. For snapshot based classification, we consider TGASI [8], IVGD [29], SL_VAE [14], GCSSI [2], and MCGNN [25] for comparison.

Table 1: Statistics of the datasets. $\mathcal{G}$ is the largest component of the joint historical relationship network based on the unique UIDs.

Statistic	Twitter15	Twitter16	Weibo
#users	480,987	289,675	2,856,741
#profile dimensions	6	6	6
#users in $\mathcal{G}$	480,405	289,504	2,856,519
#relations in $\mathcal{G}$	565,948	334,603	3,508,596
#cascades	1490	818	4664
#rumors	372	207	2244
#non-rumors	744	410	2082

4.2 Setting and Metrics

We validate the source localization experiment from two perspectives. Firstly, we verify it from the viewpoint of a new localization framework using real propagation data in social media. Notably, in practice, several factors can hinder accurate identification of the root source: (1) information diffusion data obtained through rapid crawling might be incomplete; (2) source users might delete the original tweets; and (3) instead of simply retweeting or sharing the initial post, subsequent users might appropriate and modify existing tweets as their own original content, perhaps as a tactic to boost their own visibility or popularity. These complexities can make locating the source user challenging. Consequently, we adopt a cascading random propagation failure strategy, making some activated relationships unknown. We then proceed with the localization based on the rules from the social bot (Sec. 3.1.1) and time-interval capture (Sec. 3.2.1) for the respective localization categories. As for the two platforms, we employ a rigorous ten-fold cross-validation to minimize the randomness and variability in our outcomes. For example, the Weibo dataset has 2244 independent rumor cascades. To divide these cascades into 10 parts, we first randomized the list of cascades to eliminate any potential order bias. After randomization, we evenly distributed these cascades into 10 separate groups. Each group, therefore, contains approximately 224 rumor cascades. We designate one of these parts, selected at random, as the set of unknown real propagation events for future rumor source localization tasks, denoted as a test dataset. The remaining nine parts are used as prior propagation data, denoted as a training dataset.

Secondly, since the two novel classifications strictly adhere to the current localization paradigms, we also validate the effectiveness of the two proposed localization methods using traditional heterogeneous propagation models in comparison with SOTA methods, such as adhering to the commonly used heterogeneous SI model [8, 28, 31]. Therefore, consistent with the SOTA methods, we uniformly employ the heterogeneous SI and SIR propagation models with infection rates ranging from 0.1 to 0.3 and recovery rates from 0 to 0.01 throughout the simulation process. Here the selected network is *Page-Large* with 22,470 nodes and 171,002 edges [24]. It is worth mentioning that we apply the optimal experimental settings of all comparison methods unless otherwise specified.

To demonstrate the source prediction performance of all methods rigorously, the widely used $F1-score = \frac{2*Precision*Recall}{Precision+Recall}$ in localization studies is chosen as the evaluation metric [3, 8, 30, 33].

4.3 Results of Social Bot

We initially present the localization results based on the paradigm of sensor based classification, as illustrated in Tab. 2. The group based on the propagation simulation in the Page-Large network is solely to demonstrate the robustness and feasibility of the proposed method from a traditional widely recognized scenario. In this group, we cannot apply user features because there is no concept of user profiles in simulated propagation environments. Here, we calculate the average results of 500 simulations and present the setting with a 1% deployment rate (i.e., ξ) due to the limited space[2]. It is worth mentioning that the parameter ξ includes 1%, 5%, 10%, 15%, 20%, 25%, and 30% in our experiments. For reference, the settings of reference [28] include 5%, 10%, 15%, and 20%. So we provide a more comprehensive experimental environment. Due to the limited space, we only present the results related to 1% deployment ratio in our manuscript. And the reason for showcasing only the lowest bound 1% is twofold: (1) It emphasizes the vast scale of social networks, making the deployment of more extensive measures impractical. A representation of 1% (or even less) illustrates the feasibility of the algorithm's potential application in real scenarios. (2) An increase in the deployment ratio does indeed enhance the accuracy of source localization, and our method consistently shows the best performance than SOTA methods. Due to page limitations, we conservatively opted to present the lowest bound of detection performance with a 1% deployment ratio in the manuscript.

As evidenced by the results, our method consistently outperforms the SOTA on both real-world and simulated datasets. Specifically, when benchmarked against the optimal baseline GRSL using the F1-score metric, our approach exhibits a 47% improvement on real-world datasets and a 38% enhancement on simulated cascades from the Page-Large dataset based on the heterogeneous propagation models. There are two primary reasons for this observed superiority. (1) Our bot deployment strategy judiciously considers fringe nodes. Built upon this, in every step, we greedily select new influential nodes as bots in a deep search way. This deployment idea ensures not only a globally expansive coverage but also highlights the influential prominence with high information acquisition capability of each single individual node. (2) The more significant advantage in real-world cascades than Page-Large based simulation can be attributed to the user impact evaluation module which is grounded on user profiles in the proposed source estimator. The user profiles module thoroughly accounts for user characteristics affecting propagation, which is a lack of absence in SOTA methods.

4.4 Results of Time-interval Capture

Then we present the localization results based on the paradigm of snapshot based classification, as illustrated in Tab. 3. Similar to the results from the Social Bot classification experiments, our method surpasses the SOTA on both real-world and simulated datasets.

[2]Deploying monitors in real-world scenarios is a complex endeavor. Some studies have highlighted that the feasible deployment resources in real situations are drastically below the current theoretical benchmarks of 10% [16]. For the first time, we attempt to deploy at an even more constrained level of just 1%.

Table 2: Performance compared to sensor based methods. The bold values represent the best results.

Datasets	Real-world Cascades			Page-Large	
Group	*T15*	*T16*	*Wb*	*HSI*	*HSIR*
GRSL [31]	0.411	0.371	0.329	0.207	0.154
DISGE [35]	0.223	0.254	0.231	0.076	0.067
M-PTV [19]	0.241	0.281	0.236	0.078	0.059
GMLA [20]	0.185	0.194	0.228	0.046	0.025
SNF [28]	0.154	0.176	0.122	0.020	0.013
Bot (Ours)	**0.592**	**0.537**	**0.506**	**0.282**	**0.216**

Moreover, this advantage is even more pronounced in real-world cascades. There are three main reasons supporting the notable superiority of our method. (1) The user profiles are embedded in the framework encoder to imply the user's unique characteristics at each timestamp. (2) The propagation dynamic features and static topology features are designed to fit the source localization task. (3) Users' relational information is quantified and integrated into the matrix with topological consistency. Subsequently, a graph based recurrent layer is designed in the decoder of the source inferring process, ensuring a comprehensive consideration of users' contact topological information.

Table 3: Performance compared to snapshot based methods.

Datasets	Real-world Cascades			Page-Large	
Group	*T15*	*T16*	*Wb*	*HSI*	*HSIR*
TGASI [8]	0.559	0.511	0.445	0.447	0.411
IVGD [29]	0.417	0.366	0.321	0.036	0.027
SL_VAE [14]	0.344	0.352	0.340	0.028	0.044
GCSSI [2]	0.208	0.225	0.265	0.014	0.005
MCGNN [25]	0.226	0.271	0.188	0	0
TI Capture (Ours)	**0.629**	**0.644**	**0.576**	**0.492**	**0.436**

4.5 Ablation Study: Necessity of User Profiles

Compared with SOTA methods, the performance improvement of proposed methods in real-world datasets is more distinct. An intuitive rationale is that the optimal baseline inherently does not consider user characteristics. Thus, removing user profiles to evaluate the inherent localization capacity of our proposed methods is imperative. Furthermore, by comparing the inclusion and exclusion of user features, we aim to elucidate the fundamental necessity of user characteristics in real-world propagation scenarios. Specifically, within the social bots framework, we set the influence evaluation for all users to a constant value of 1 (denoted as Bot_Norm). In the time-interval capture, we assign a value of 0 to every item of six user profile features (denoted as TW_Norm). From Tab. 4, it's obvious that even without user features, our methods outperform the optimal baseline in real-world cascade scenarios. Furthermore, the performance degradation when omitting user features underscores the importance of considering user characteristics in real-world scenarios.

Table 4: The performance without user profiles.

Paradigm	*Datasets*	*T15*	*T16*	*Wb*
Sensor	Opt Basline	0.411	0.371	0.329
	Bot_Norm (Ours)	0.531	0.495	0.466
	Bot (Ours)	**0.592**	**0.537**	**0.506**
Snapshot	Opt Baseline	0.559	0.511	0.445
	TW_Norm (Ours)	0.566	0.552	0.478
	TI Capture (Ours)	**0.629**	**0.644**	**0.576**

4.6 Computational Cost Comparisons

4.6.1 As for the computational cost of Social Bots based Localization: In the social bots deployment phase, the computational complexity of selecting social bots is approximately $O(|V|(log|V|)^2)$. And in the source inference phase, the computational complexity of selecting social bots is approximately $O(|E||F|)$. In the phase of selecting social bots, the computational complexity of selecting social bots is approximately $O(|V|(log|V|)^2)$. More specifically, first, it is necessary to compute the degrees of all users and to call the sorting function to sort users. The computational complexity of these corresponding steps is $O(|V|log|V|)$. Next, the eccentricity computation of a hub user or leaf user node is an incomplete Dijkstra process, which requires a time complexity of $O(|V|log|V|)$. The above steps are generally repeated $log|V|$ times to select all sensors, leading to a $O(|V|(log|V|)^2)$ complexity. In the phase of source inference, the computational complexity of source inference is approximately $O(|V|log|V|)+O(|E|)$. More specifically, firstly, each user in the step of candidate screening needs to calculate the distance from itself to all social bots $o \in O_Y$. So the computational complexity of computing the distance information is $O(|O_Y||V|log|V|)$. Next, in the step of source estimation, each reliable candidate in V_θ is processed $|O_Y|$ times, so the total complexity is $O(|O_Y||V_\theta|)$. In the punishment-reward mechanism process, each reliable candidate verifies if its neighbors are members of O_Y or O_N. So the complexity of the mechanism is approximate to $O(|V_\theta|+2|V_\theta|\langle k \rangle)$. Further, in the process of user's negative influence evaluation, matrix-vector operations are employed for all nodes in one go, leading to a computational complexity of $O(|E||F|)$. Finally, using the sorting function to identify the source need $O(|V_\theta|log|V_\theta|)$. Here, $|V_\theta| << |V|$, $|O_Y| << |O|$, and $|F|$ is the dimensions of profiles. In summary, the computational cost is $O(|V|(log|V|)^2 + |E||F|)$. What's more, we conduct the experiments to compare the runtime between our proposed algorithm and SOTA methods. The proposed algorithm is the same complexity as GRSL, and 10-100 times faster than multivariate Gaussian distribution based methods (DISGE, M-PTV, GMLA).

4.6.2 As for the computational cost of Time-window Capture based Localization: Since the method adopted is a deep learning method based on improved message-passing techniques, the computational complexity is consistent with that of GCN based methods. We measure complexity by comparing the training times of all methods. GCSSI, which only considers time series information, is faster than our proposed GCN based methods by a fifth. However, it has the lowest detection performance among SOTA methods, and our method achieves an accuracy that is three times higher than that of GCSSI. Compared to TGASI and MCGNN, our method is 60% faster. It also doubles the speed of the IVGD framework and is three times faster than SL_VAE.

4.7 Parameter Analysis and Ablation Study of Designed Modules

We also conduct comprehensive analyses to ensure the optimal values of parameters or hyperparameters and the necessity of the designed modules. For example, we test the reward coefficient ranging from 0.8 to 1 and the penalty coefficient from 1 to 1.2, incrementing by 0.01. The best localization performance is achieved under the reward-penalty pair of (0.9, 1.02).

Then we conduct ablation experiments to assess the contribution of each set of features (participant ratio of neighbors H_1 and H_2, normalized participant number of neighbors H_3 and H_4, and original participant states H_5 and H_6). More specifically, we mask them by setting their values to zero, ensuring they do not influence the model's source localization process. The results clearly demonstrated that each group of features significantly contributes to the accuracy of source localization. Removing any group of features led to a noticeable decline in localization accuracy.

Table 5: The performance evaluation of variant models from the time-interval capture based algorithm. TI_R- is the proposed TI Capture based algorithm without considering the the ratio of participant neighbors and the ratio of non-participant neighbors (H_1 and H_2) of a user in a timestamp. Similarly, TI_N- does not consider the normalized number of participant neighbors and non-participant neighbors (H_3 and H_4). And TI_S- does not consider the participant or non-participant state of a user (H_5 and H_6).

Paradigm	Variants	Twitter15	Twitter16	Weibo
Snapshot	**TI Capture**	**0.629**	**0.644**	**0.576**
	TI _R-	0.548	0.537	0.506
	TI _N-	0.579	0.581	0.543
	TI _S-	0.612	0.621	0.562

5 Conclusion

Our study introduces a novel perspective for source localization by bridging the gap from theoretical methodologies to practical user-centric approaches, which is a change from simulated propagation data to real-world propagation cascades. By considering and integrating user-specific profiles in the source inference process, a more realistic application for the proposed approaches is guaranteed. And the comprehensive experiments highlight the efficacy and feasibility of the presented framework, and validate the importance of considering user profiles.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Nos. 62271411, U22A2098, 62261136549, U22B2036), Fok Ying-Tong Education Foundation, China (No. 171105), the Fundamental Research Funds for the Central Universities (No. G2024WD0151) and the Tencent Foundation and XPLORER PRIZE.

References

[1] Ameya Agaskar and Yue M. Lu. 2013. A fast Monte Carlo algorithm for source localization on graphs. In *SPIE Optical Engineering and Applications*. 88581N.

[2] Ming Dong, Bolong Zheng, Guohui Li, Chenliang Li, Kai Zheng, and Xiaofang Zhou. 2022. Wavefront-Based Multiple Rumor Sources Identification by Multi-Task Learning. *IEEE Transactions on Emerging Topics in Computational Intelligence* (2022).

[3] 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.

[4] Glenn Ellison. 2020. *Implications of heterogeneous SIR models for analyses of COVID-19*. Technical Report. National Bureau of Economic Research.

[5] Jacob Goldenberg, Barak Libai, and Eitan Muller. 2001. Talk of the network: A complex systems look at the underlying process of word-of-mouth. *Marketing letters* 12, 3 (2001), 211–223.

[6] Mark Granovetter. 1978. Threshold models of collective behavior. *Amer. J. Sociology* 83, 6 (1978), 1420–1443.

[7] Dongpeng Hou, Chao Gao, Zhen Wang, Xiaoyu Li, and Xuelong Li. 2024. Random Full-Order-Coverage Based Rapid Source Localization With Limited Observations for Large-Scale Networks. *IEEE Transactions on Network Science and Engineering* (2024).

[8] Dongpeng Hou, Zhen Wang, Chao Gao, and Xuelong Li. 2023. Sequential Attention Source Identification Based on Feature Representation. (2023), 4794–4802.

[9] Dongpeng Hou, Shu Yin, Chao Gao, Xianghua Li, and Zhen Wang. 2024. Propagation Dynamics of Rumor vs. Non-rumor across Multiple Social Media Platforms Driven by User Characteristics. *arXiv preprint arXiv:2401.17840* (2024).

[10] Roland Imhoff and Pia Lamberty. 2020. A bioweapon or a hoax? The link between distinct conspiracy beliefs about the Coronavirus disease (COVID-19) outbreak and pandemic behavior. *Social Psychological and Personality Science* 11, 8 (2020), 1110–1118.

[11] Md Saiful Islam, Tonmoy Sarkar, and Sazzad Hossain Khan. 2020. COVID-19–related infodemic and its impact on public health: A global social media analysis. *The American Journal of Tropical Medicine and Hygiene* 103, 4 (2020), 1621.

[12] Jiaojiao Jiang, Sheng Wen, Shui Yu, Yang Xiang, and Wanlei Zhou. 2016. Identifying propagation sources in networks: State-of-the-art and comparative studies. *IEEE Communications Surveys & Tutorials* 19, 1 (2016), 465–481.

[13] Brian Karrer and Mark EJ Newman. 2010. Message passing approach for general epidemic models. *Physical Review E* 82, 1 (2010), 016101.

[14] 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.

[15] Xiaomo Liu, Armineh Nourbakhsh, Quanzhi Li, Rui Fang, and Sameena Shah. 2015. Real-time Rumor Debunking on Twitter. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*. 1867–1870.

[16] Linyuan Lü, Tao Zhou, Qian-Ming Zhang, and H Eugene Stanley. 2016. The H-index of a network node and its relation to degree and coreness. *Nature communications* 7, 1 (2016), 10168.

[17] Jing Ma, Wei Gao, Prasenjit Mitra, Sejeong Kwon, Bernard J. Jansen, Kam-Fai Wong, and Cha Meeyoung. 2016. Detecting Rumors from Microblogs with Recurrent Neural Networks. In *The 25th International Joint Conference on Artificial Intelligence*. AAAI, 3818–3824.

[18] Jing Ma, Wei Gao, and Kam-Fai Wong. 2017. Detect Rumors in Microblog Posts Using Propagation Structure via Kernel Learning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vol. 1. 708–717.

[19] Robert Paluch, Łukasz Gajewski, Krzysztof Suchecki, and Bolesław Szymański. 2021. Enhancing maximum likelihood estimation of infection source localization. In *Simplicity of Complexity in Economic and Social Systems*. 21–41.

[20] Robert Paluch, Xiaoyan Lu, Krzysztof Suchecki, Bolesław K Szymański, and Janusz A Hołyst. 2018. Fast and accurate detection of spread source in large complex networks. *Scientific Reports* 8, 1 (2018), 1–10.

[21] SGOPAL Patro and Kishore Kumar Sahu. 2015. Normalization: A preprocessing stage. *arXiv preprint arXiv:1503.06462* (2015).

[22] Pedro C Pinto, Patrick Thiran, and Martin Vetterli. 2012. Locating the source of diffusion in large-scale networks. *Physical Review Letters* 109, 6 (2012), 068702.

[23] Maryam Ramezani, Aryan Ahadinia, Amirmohammad Ziaei Bideh, and Hamid R Rabiee. 2023. Joint Inference of Diffusion and Structure in Partially Observed Social Networks Using Coupled Matrix Factorization. *ACM Transactions on Knowledge Discovery from Data* 17, 9 (2023), 1–28.

[24] Benedek Rozemberczki, Carl Allen, and Rik Sarkar. 2021. Multi-scale attributed node embedding. *Journal of Complex Networks* 9, 2 (2021), 1–22.

[25] Xincheng Shu, Bin Yu, Zhongyuan Ruan, Qingpeng Zhang, and Qi Xuan. 2021. Information Source Estimation with Multi-Channel Graph Neural Network. *Graph Data Mining: Algorithm, Security and Application* (2021), 1–27.

[26] Wenchang Tang, Feng Ji, and Wee Peng Tay. 2018. Estimating infection sources in networks using partial timestamps. *IEEE Transactions on Information Forensics and Security* 13, 12 (2018), 3035–3049.

[27] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2018. Graph attention networks. *International Conference on Learning Representations* (2018).

[28] Hongjue Wang and Kaijia Sun. 2020. Locating source of heterogeneous propagation model by universal algorithm. *Europhysics Letters* 131, 4 (2020), 48001.

[29] Junxiang Wang, Junji Jiang, and Liang Zhao. 2022. An Invertible Graph Diffusion Neural Network for Source Localization. In *Proceedings of the ACM Web Conference*. 1058–1069.

[30] Zhen Wang, Dongpeng Hou, Chao Gao, Jiajin Huang, and Qi Xuan. 2022. A Rapid Source Localization Method in the Early Stage of Large-scale Network Propagation. In *Proceedings of the ACM Web Conference*. 1372–1380.

[31] Zhen Wang, Dongpeng Hou, Chao Gao, Xiaoyu Li, and Xuelong Li. 2023. Lightweight source localization for large-scale social networks. In *Proceedings of the ACM Web Conference 2023*. 286–294.

[32] 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*. 217–223.

[33] 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*. PALO ALTO, CA 94303 USA, San Francisco, CA, 217–223.

[34] Max Welling and Thomas N Kipf. 2016. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR 2017)*.

[35] Fan Yang, Shuhong Yang, Yong Peng, Yabing Yao, Zhiwen Wang, Houjun Li, Jingxian Liu, Ruisheng Zhang, and Chungui Li. 2020. Locating the propagation source in complex networks with a direction-induced search based Gaussian estimator. *Knowledge-Based Systems* 195 (2020), 105674.

[36] Shu Yin, Peican Zhu, Lianwei Wu, Chao Gao, and Zhen Wang. 2024. GAMC: An Unsupervised Method for Fake News Detection Using Graph Autoencoder with Masking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 347–355.

[37] Wenyu Zang, Chuan Zhou, Li Guo, and Peng Zhang. 2015. Topic-aware source locating in social networks. In *Proceedings of the 24th International Conference on World Wide Web*. 141–142.

[38] 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.