Proceedings of the Nineteenth International AAAI Conference on Web and Social Media (ICWSM 2025)

LIN: Latent Influence Network for Discovering Hidden Directed Influence Links on Social Media

Chenhao Gu, Zainab Razia Zaidi, Ling Luo, Shanika Karunasekera

The University of Melbourne, Australia
{chenhao.gu, ling.luo, karus}@unimelb.edu.au, zainab.zaidi@monash.edu

Abstract

In the current social media landscape, the study of influence propagation and consensus formation has gained prominence. While user interactions like retweeting are apparent, the underlying pathways of influence often remain hidden and complex. This study proposes a novel network called Latent Influence Network (LIN), which advances the analysis of influence on social media. LIN's architecture and the process of parameter selection are meticulously discussed within the comprehensive Latent Influence Detection Framework (LIDET). Based on the user's behavior label, LIN identifies the optimal network configuration, revealing more accurate influence patterns. We applied the LIDET framework to four diverse datasets, each demonstrating substantial improvements in influence pattern recognition over traditional network models. Specifically, in a case study on a COVID-19 dataset, LIN achieved a classification accuracy of 99%, significantly outperforming conventional methods. These findings underscore the utility of LIN in capturing the dynamics of influence and enhancing our understanding of opinion formation on social media.

Introduction

In society, individuals are influenced by surrounding groups, while also exerting influence on them. Formally, *Social influence* is defined as "changes in an individual's thoughts, feelings, attitudes, or behaviors that result from interaction with another individual or a group" (Jendoubi et al. 2017). The intricacies of how ideas, behaviors, and opinions disseminated through networks, particularly on social media, often remain obscured and complex. For instance, even with the ability to track who shares a post, the scope of the full influence network, including indirect influence, is not completely revealed.

There has been a lot of research in understanding the influence and the process of reaching a consensus. With the advent of social networking platforms, the interactions and influences are more impactful and fast. After the US presidential election of 2016, the debate about social media's influence peaked, with numerous publications pointing to a clear link between Trump's election and his tweets (Brady, Kelly, and Stein 2017). People's participation in civic and political debates has risen dramatically due to social media. While this may benefit the political process, evidence of malevolent actors poisoning the information environment through disinformation and manipulative operations is emerging. While influence efforts, disinformation, and propaganda have always existed, social media has made it easier and extended its reach (Pacheco et al. 2021). There are efforts focused on measuring the influence of an individual or posts - using the number of likes, reposts, followers, etc. - or techniques to maximise influence over social networks, such as, for marketing purposes, etc.

At present, methods are available for tracking users who engage synchronously within specific time windows. The concept of the Latent Coordinated Network (LCN) (Weber and Neumann 2020) forms the cornerstone of our research. LCN is predicated on the notion that users exhibiting repetitive and similar behaviors within a time window are likely interconnected. The foundational premise of LCN is the existence of manipulated users in social networks, who may lack overt interactions but are highly correlated. Thus, the aim of LCN is to identify highly correlated user communities.

We propose that this concept can also be leveraged to identify the influence between users, revealing hidden associations among them. However, unlike LCN, LIN does not focus on identifying communities but on mapping out how influence travels through the network and impact users' behavior. Our approach is based on the intuition that accurate detection of influence links will facilitate accurate prediction of user behavior.

Our paper endeavours to address the following four research questions:

RQ1 *Can we infer a user's influence based on their social activity records?*

This question arises from the lack of full friend or follower network data in many datasets. While retweet networks are often used as proxies, they lack precision. Even with friend networks, influence is not always accurately reflected. We propose the Latent Influence Network (LIN) to address this gap, though it may still include spurious influence links, motivating RQ2.

RQ2 *How can we optimize and validate LIN's parameters and structure to better reflect realistic influence dynamics* *on platforms like Twitter?*

Due to the absence of explicit influence labels, we assume that better influence detection improves behavior prediction. Based on this principle, we introduce the Latent Influence Detection Framework (LIDET) to tune and assess LIN across datasets.

RQ3 *What insights emerge from LINs built using different user activities, and how do influence networks vary across datasets?*

This question explores how activity types (e.g., retweet, reply) shape LIN structure, and investigates the dataset-specific patterns and factors contributing to influence variation.

Our contributions can be summarised as follows:

1. We introduced LIN, a novel network tailored to capture user influence networks from users' historical activities.[1]
2. To assess LIN's performance across different parameters and compare it with conventional methods, we developed a comprehensive framework called LIDET. This framework facilitates the generation of user behaviour representations from diverse network structures and addresses a classification problem, allowing for a thorough evaluation of the strengths and weaknesses of various networks.
3. We evaluated LIN's performance across different datasets to provide a comprehensive understanding of its applicability and effectiveness.
4. Through a case study on the COVID-19 dataset, we conducted a detailed analysis of LIN's performance, particularly focusing on the impact of various parameter configurations within this specific context.

Related Works

Attempts to establish generic coordinated detection frameworks have been made since 2019. These studies create networks by finding regular indirect connections based on these "similar acting" accounts, which can involve retweeting the same tweets (Weber and Neumann 2020), using the same URLs (Magelinski, Ng, and Carley 2022), creating comparable text or images (Graham et al. 2020), and a mix of these interaction patterns(Ng and Carley 2022).

In general, researchers believe that two users are connected if they engage in identical activities within a short period of time, such as retweeting the same tweet. Weber and Falzon (2021) go on the formation of LCN in great depth. They have compared the LCNs produced for different sizes of time windows and discussed various scenarios that can occur when the time windows overlap.

In the meantime, many efforts attempt to predict users' future behaviors based on their social networks (Qiu et al. 2018). For instance, for a particular user, if their friends purchase certain products, will the user also purchase the same products in the future? These endeavors often assume that the friend network represents influence, implying that friends' behaviors will influence the target user. However, practical challenges exist. Obtaining friend network data can be difficult as many individuals choose to hide their social connections or restrict access to their followers' lists and other information. Consequently, researchers or analysts may not obtain complete and accurate friend network data in practice. Additionally, some platform APIs impose restrictions, further complicating the acquisition of user social network data. Furthermore, friends may not necessarily reflect true influence (Song et al. 2021). While friend networks are considered a significant factor influencing user behavior, not all friends truly reflect a user's influence. In social networks, individuals may be influenced by non-friends as well. Therefore, simply inferring user behavior based on friends' actions may overlook such differences, leading to biases in prediction results.

Motivated by the concept of LCN (Weber and Neumann 2020), we propose a new network, Latent Influence Networks (LIN) to focus more on the latent influences among users. This shift underscores our interest in the underlying influences driven by user interactions, rather than just the occurrence of similar activities. By interpreting these connections through the lens of influence (Friedkin and Johnsen 2011), we can not only detect coordinated behavior but also map the directional flows of influence within the network, offering a richer understanding of user interactions across various time windows and their influence pattern.

Framework Methodology

In this section, we present LIDET, a framework for analyzing the influence of users on social media platforms. Our methodology starts with the construction of Latent Influence Network (LIN) and then moves towards the parameter selection to identify the network configuration that best represents the real-world influence network. The input of this framework can be data from any social media platform, with timestamp information, corresponding posts, and user interactions as compulsory features. Figure 1 presents an overview of our framework for social media analysis using LIN.

The key components of the framework include:

1. Active User

   In practical applications, our analysis primarily focuses on the most active users within the network. To ensure manageability and relevance, we constrain the number of active users to a subset of approximately 3000 to 6000. This selection is based on their activity level, filtering out users who have minimal interactions (e.g., appearing only one or two times) within the dataset.
2. Building Candidate LINs

   LIN is a novel concept we propose to capture the hidden influences among users on social media. To select the most accurate representation of influence among users, we construct multiple LINs with different parameter settings before selecting the optimal LIN.
3. Node Embedding

   Once LINs are constructed, we employ node embedding techniques to extract feature vectors for each user from the network structure.

[1]Code available at github.com/GudanDeco/Latent-influence-network

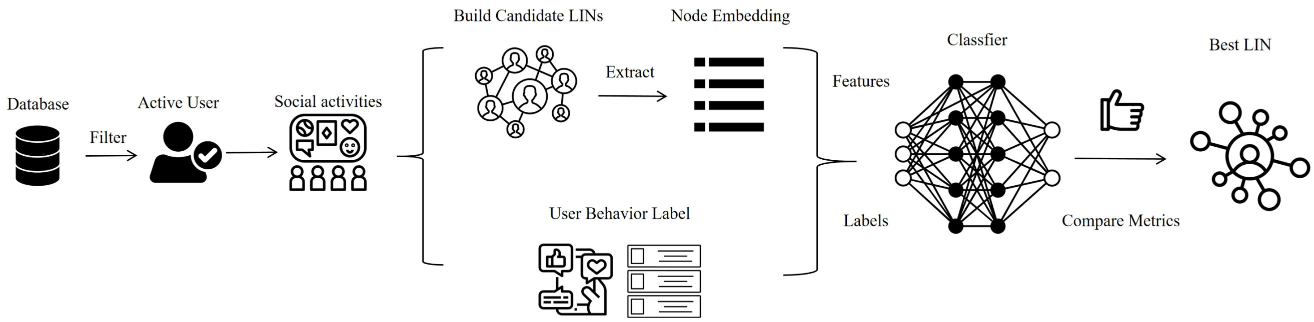


Figure 1: LIDET framework: This diagram represents the process of selecting the most effective LIN for social media analysis. It involves filtering active users, constructing candidate LINs using different parameters, applying node embedding techniques, and evaluating network effectiveness through machine learning models based on metrics such as accuracy and F1 score.

4. User Behavior Labelling

   For each dataset, user behaviors are labelled according to the specific categorization inherent to the dataset; for example, in the covid-19 dataset, users are classified by their stance.

5. User Classification

   Conduct classification tasks using the user features obtained from node embeddings and the labels derived from Behavior Labelling, to evaluate how well each influence network can help to classifier user behavior labels.

6. Determining the Most Effective Configuration of Influence Network

   Select the most effective network to capture the latent user influence based on the evaluation results of the classification and regression tasks.

This systematic framework LIDET, processes social media data to generate the most optimal influence networks, aims to provide a comprehensive tool for understanding and analyzing the dynamics of influence on social media platforms.

Latent Influence Network

The Latent Influence Network (LIN) is constructed specifically focusing on identifying the *shared activities* of user interactions. The activity plays a central role in LIN, representing various forms of user interaction such as retweeting the same tweet, mentioning the same hashtag or URL, and replying to the same user, among others.

Formally, let $LIN = (V, E)$ be a directed graph where V represents users and E denotes the influence edges. For each tweet t_i by user u_i, and subsequent tweet t_j by user u_j within a predefined time window, if $\tau(t_i) = \tau(t_j)$, indicating that both tweets share a common activity then an edge (u_i, u_j) is established in E.

Build Candidate LINs

Building candidate LINs involves a systematic approach to exploring various network configurations based on different parameter settings. This process is crucial in understanding how parameter variations can affect the structure and dynamics of the resulting networks. We detail the steps involved in this process below:

Algorithm 1: Build LIN

Input: A collection of tweets
Parameters: Time window W, Threshold θ, Activity τ
Output: LIN, a ***directed*** graph with ***weighted*** edges representing user influence

Initialize $LIN = (V, E)$ with $V = \emptyset$, $E = \emptyset$.
for each tweet t_i by user u_i **do**
    for each tweet t_j by user u_j within W of t_i **do**
        if $\tau(t_i) = \tau(t_j)$ **then**
            Add u_i and u_j to V if not present.
            if edge (u_i, u_j) exists in E **then**
                Increment weight of (u_i, u_j) by 1
            else
                Add edge (u_i, u_j) to E with weight of 1
            end if
        end if
    end for
end for
for each edge (u_i, u_j) in E **do**
    if the weight of (u_i, u_j) is less or equal to θ **then**
        Remove (u_i, u_j) from E.
    end if
end for
return LIN

Parameter Variations The parameters that primarily influence the construction of LINs are the time window W, threshold θ, and the activity τ. Each of these parameters plays a distinct role:

- **Time Window** W: Determines the temporal boundary for considering user interactions. A shorter window may capture more immediate and direct influences, while a longer window can reveal extended and possibly indirect influences.
- **Threshold** θ: Serves as a filter to determine the strength

of connections. A higher threshold may result in a network with stronger but fewer connections, whereas a lower threshold includes weaker and potentially more numerous links.

- **Activity** τ: Defines the specific user activities that form the basis of connections in the network.

For each set of parameters, we generate a variant of LIN to serve as a candidate for identifying the Best LIN.

Node Embedding

As we have seen different parameter choices for LIN result in various distinct networks. To select which network captures the actual influence interactions, we first obtain a vector representation of the node (user) based on the network structure, a process known as *node embedding*. These node embeddings will be used as user feature vectors, whereas the user stances will be used as labels in the classifier as shown in Fig. 1.

- DeepWalk (Perozzi, Al-Rfou, and Skiena 2014): Utilizes random walks to learn representations, preserving both local and global network structures.
- Node2Vec (Grover and Leskovec 2016): It offers a biased random walk procedure, which can be configured to capture the network's homophily or structural equivalence.
- SDNE (Structural Deep Network Embedding) (Wang, Cui, and Zhu 2016): Uses a deep autoencoder model to capture first and second-order proximity in networks.
- LINE (Large-scale Information Network Embedding) (Tang et al. 2015): Designed for large-scale information networks, it preserves both local and global network structures.

For simplicity, we employ one-hot embedding as the initial feature for each user node. This ensures that each node has a unique representation in the network without any prior bias. Then, we apply advanced embedding techniques like DeepWalk, Node2Vec, SDNE, and LINE for each user node, which serve as high-dimensional representations that capture the structural context and relations of each node within the network. The reason for using different node embedding algorithms is to reduce the bias introduced by the algorithms themselves.

User Behavior Labelling

This step focuses on extracting data from various datasets to illustrate user influence through behavioral indicators. In the COVID-19 dataset, we employed stance detection techniques to assign stances to each user. We postulate that a user's stance is significantly influenced by the stances of their neighbours, suggesting that with a known network of influence and the stances of a subset of trained users, it is feasible to predict the stances of other test users. This analytical approach is not limited to stance detection; it is applicable to any scenario that requires classifying user behavior based on dynamics of influence.

For instance, in the IRA dataset, users are categorized into coordinated users and normal users. In this context, different types of users exhibit varying structures of influence. By analyzing and identifying these patterns of influence, we can gain deeper insights into how users promote or inhibit the spread of information through their network positions and levels of influence.

To summarize, this section requires adjustments based on the specific conditions of different datasets to ensure the accuracy of the analysis. For example, if the relationship between user behavior labels and influence is not pronounced, the accuracy of the resulting network model may be affected, reflecting the predictive performance of the network.

User Classification

Different datasets have different labels for users, and with user embeddings and their corresponding labels, we can use them for classification tasks to evaluate the performance of different LINs.

The purpose of the classification task is to compare the effect of different network structures on users by predicting their labels. Using node embeddings as user features, we train a classifier to predict the dominant stance of a user's tweets and evaluate the performance using prediction accuracy and F1 score.

To mitigate biases introduced by different machine learning algorithms, we employed multiple methods, including Support Vector Machines, Random Forests, and Neural Networks. These were used as the foundational machine-learning models for both regression and classification tasks.

Determining the Most Effective Configuration of Influence Network

The final step of our framework is the selection of the optimal network settings for various activities. This process is crucial for identifying which LIN can reveal the hidden influence dynamics on real-world social media platforms most accurately. The selection outcome can be either a single network with the best metrics or a set of networks, each excelling in different aspects, depending on the application requirement.

Experiment Setup

Baseline Networks

Directed Retweet Network DRN is the most widely used retweet network. It is a directed graph where the retweeter is the target and the original tweeter is the source, which will serve as the baseline for our method. Retweet networks are among the most pivotal types of networks on Twitter (now called X).In the majority of literature, such networks are referred to as retweet networks (Cherepnalkoski and Mozetič 2016). In this paper, to differentiate from the latent influence retweet Network, we have adopted the term Directed Retweet Network.

Formally, DRN can be defined as a directed graph $DRN = (V, E)$, where V is the set of nodes representing users, and E is the set of directed edges denoting the retweet relationship. For each retweet t_x by user u_x of an original tweet t_0 by user u_0, an edge is formed from u_0 to

u_x in DRN, symbolizing the influence flow from the source u_0 to the target u_x.

Friends Network The Friends Network serves as an alternative baseline. This directed graph captures the following relationships among users, indicating who chooses to follow whom. This network helps to understand the patterns of influence and interaction that emerge from these directional connections.Unfortunately due to this information being more difficult to obtain, only two datasets have relevant data to construct friends network.

Latent Coordination Network LCN constructs undirected graphs to identify user communities based on their coordinated activities. To address the original LCN's lack of a parameter selection method, we integrated the LIDET framework to assist in selecting optimal parameters. This combined approach is referred to as LIDET_LCN.

Dataset Description

In this work, we utilize four different social network datasets. The case study is primarily related to the COVID-19 vaccine data. Because this dataset is fresh reflects current social network interaction patterns well, and has a large number of records of users' social activities, it is suitable for comparing different activities under the influence link.

1. **COVID-19 Vaccination Dataset**
   Our study used the COVID-19-related tweets datasets posted on IEEE Dataport by Rabindra Lamsal (Lamsal 2021). This dataset was collected using COVID-19 related keywords for global tweets in the English language. The details about the keywords and collection process can be accessed from (Lamsal 2021; Zaidi et al. 2023).

   We focused on the topic of vaccine hesitancy as there was a lot of relevant discussion on Twitter during the pandemic, and we hoped to get a large dataset of tweets in favour (pro-vax), against (anti-vax) and neutral stance with the topic of Vaccine hesitancy (Zaidi et al. 2023). Thus, in this dataset, our definition of influence is quite straightforward, i.e., whether users support vaccines. Different users should have different influence relationships, and knowing users' influence networks should allow us to infer the stances of other users. The accuracy brought about by influence networks is considered a criterion for evaluating the effectiveness of influence networks.
2. **Twitter15 and Twitter16**
   The second dataset is Twitter15 and Twitter16 (Ma, Gao, and Wong 2017), which are based on similar data processing methods and have been widely adopted as standard data in the field of rumor detection. The Twitter15 dataset contains 1490 tweet propagations, and Twitter16 contains 818 tweet propagations, each labelled with types such as non-rumor, false rumor, true rumor, and unverified rumor. In these datasets, we can categorize users by the type of tweets they send, e.g., non-rumor spreaders, false rumor spreaders, etc. In this network, influence is defined as whether a user is influenced by others to retweet within a certain time frame. Different spreaders have different influences; for instance, rumors often spread more easily than truths because they are often exaggerated. Therefore, if a user is a rumor spreader, their range of influence and linked nodes should differ significantly from those debunking rumors.
3. **IRA Dataset**
   The third dataset is the IRA dataset, consisting of highly coordinated Russian troll activities, along with added noise tweets (Machanayaka et al. 2024). In this dataset, we have normal users and highly coordinated users. In this dataset, we believe that the influence of coordinating users should differ from that of ordinary users. Thus, LIN should distinguish well between these two types of users.
4. **Digg**
   The fourth dataset is Digg (Hogg and Lerman 2012), a news aggregator that allows people to vote on web content. This dataset contains data on news promoted to the Digg homepage during one month in 2009. It also includes friendship relationships between users. Influence in this dataset means that if the target user likes a certain news item, he can influence who likes it as well.

Dataset Statistics Tabel 1 shows the number of nodes as well as edges left after different datasets have been filtered for active users. Since Digg has no notion of retweeting. Statistics shown for Digg is a friend network, and everything else is DRN.

	Covid-19	Twitter15	Twitter16	IRA	Digg
$\|V\|$	3,358	3,451	3,605	3,579	5,463
$\|E\|$	23,053	19,634	15,608	190,884	810,034

Table 1: Summary of datasets. $|V|$ and $|E|$ indicate the number of vertices and edges in the base graph for each dataset.

Data Preparation

Parameter Selection To facilitate the reproducibility of our results, here are the parameter choices we made:

For deepwalk, the main parameters are set as follows: the number of walking steps is 4, the number of walks is 80, and the vector size is 128. These parameters were selected after multiple trials on the COVID-19 dataset, yielding satisfactory results. Increasing the number of walking steps, walks, and vector size theoretically leads to more convergent results but significantly increases computational time. Moreover, it does not completely mitigate the inherent randomness in node embedding methods. It is noteworthy that our objective is not to obtain the optimal vector representation through node embedding methods. Different networks, structures, and training parameters can result in different outcomes. Additionally, due to the stochastic nature of node embedding methods, some random variations are inevitable. We aim to identify more universally applicable parameters to demonstrate the influence of network structure rather than changes induced by fine-tuning hyperparameters.

Similarly, for node2vec, we used 4 walking steps, 80 walks, with q and p parameters set to 1, and an output vector size of 128. The primary focus here is on the q and p parameters, which balance capturing the features of the target node against sampling predominantly from its vicinity. We opted for balanced parameters to incorporate both distant and nearby information.

Regarding LINE (Tang et al. 2015) and SDNE (Wang, Cui, and Zhu 2016), we employed a setting similar from their papers, while also standardizing the output vector size to 128. Further details can be found in the forthcoming code release.

After obtaining output vectors through node embedding, we fed them into various machine learning classifiers: logistic regression, SVM, random forest, xgboost, and neural networks. To mitigate the impact of classifiers on vector classification. In general, SVM performed the best. The neural network, due to its uniform parameterization across different dataset scenarios rather than fine-tuning, did not yield satisfactory results. These classifiers are parameterized using sklearn's (Pedregosa et al. 2011) default setting.

Additionally, we experimented with varying these parameters; while the specifics of the results differed, the overarching trend remained consistent.

Train and Test split The division of training and testing sets in each dataset was determined after identifying the number of active users. For example, in the COVID-19 dataset, active users were first quantified and then split in an 80-20 ratio for training and testing, respectively, using a random seed of 42. Since different networks might vary in the number of nodes—particularly when some nodes do not capture links in smaller time windows—the size of training and testing sets may differ across networks. This variability is necessary to avoid the inclusion of isolated nodes that have no observed interactions, which could bias the network analysis. To ensure consistency and fairness, however, the specific nodes used for training and testing are kept constant across different networks. This approach prevents discrepancies that might arise from using different sets of nodes in different network configurations.

Dataset	Network	AUC	Prec.	Rec.	F1
Covid-19	DRN	0.5768	0.6818	0.7724	0.6769
	Friends Network	0.8201	0.8171	0.7607	0.6777
	Retweet_LCN	0.9996	0.9920	0.9920	0.9920
	Retweet_LIN	**0.9998**	**0.9923**	**0.9923**	**0.9922**
	Hashtag_LCN	0.9901	0.9778	0.9778	0.9776
	Hashtag_LIN	0.9922	0.9803	0.9803	0.9801
	URL_LCN	0.9974	0.9838	0.9838	0.9838
	URL_LIN	0.9987	0.9867	0.9867	0.9867
	Mention_LCN	0.9975	0.9813	0.9813	0.9813
	Mention_LIN	0.9973	0.9850	0.9851	0.9850
	Reply_LCN	0.9223	0.9067	0.9013	0.9006
	Reply_LIN	0.9113	0.8723	0.8714	0.8707
Twitter15	DRN	0.7255	0.5032	0.4844	0.4400
	Retweet_LCN	0.8064	0.5885	0.5938	0.5878
	Retweet_LIN	**0.8403**	**0.6322**	**0.6354**	**0.6319**
Twitter16	DRN	0.7843	0.5562	0.5215	0.4877
	Retweet_LCN	0.8813	0.6951	0.6987	0.6937
	Retweet_LIN	**0.8962**	**0.6993**	**0.7012**	**0.6981**
IRA	DRN	0.9722	0.9421	**0.9853**	0.9632
	Hashtag_LCN	0.9966	0.9645	0.9807	0.9725
	Hashtag_LIN	**0.9973**	**0.9774**	0.9845	**0.9809**
	Mention_LCN	0.9923	0.9151	0.9442	0.9294
	Mention_LIN	0.9978	0.9298	0.9784	0.9535
	URL_LCN	0.9944	0.8930	0.7636	0.8223
	URL_LIN	0.9978	0.9298	0.9784	0.9535
Digg	Friends Network	**0.8600**	0.7181	0.5326	0.6115
	Like_LCN	0.8210	0.7640	**0.6854**	**0.7225**
	Like_LIN	0.8321	**0.7995**	0.6294	0.7043

Table 2: Performance metrics comparison across datasets. For each dataset, the best results are highlighted in bold.

Experiment Result

In Table 2, we compared the predictive performance of different networks across four datasets, presenting results for four metrics: Area under the ROC Curve (AUC), Precision (Prec.), Recall (Rec.), and F1-Measure (F1). For each dataset, we utilize the LIDET framework to select the best parameter settings for both LCN and LIN across different activities. The naming format LIDET_Activity_Network is used throughout, though the LIDET prefix is omitted from the table for brevity. Due to the randomness in the node embedding methods, we ran each combination 10 times and reported the average results.

For the COVID-19 dataset, data for both baselines, DRN and Friends Network, are available. It is evident that the Friends Network performs notably better than DRN, aligning with our hypothesis that the Friends Network is a key component in user influence. Consequently, it tends to outperform DRN. We then compared the performance of LCN and LIN under different activities, with Retweet_LIN emerging as the best. This is due to retweets being the most frequent activity, thus capturing more user interactions. The difference between LIDET_Retweet_LCN and LIDET_Retweet_LIN is minimal, but LIN outperforms LCN in all four metrics. This is partly because it is extremely difficult to improve beyond an accuracy close to 99%, and when there are many edges, node embedding struggles to capture the nuances of one or two edges. However, when comparing other activities, LIN consistently shows advantages over LCN, indicating that LIN indeed enhances influence at various levels.

The second dataset is Twitter15 and Twitter16. LIN shows a clear improvement over DRN in both datasets. In Twitter15, the improvement of LIN over LCN is more significant, whereas in Twitter16 the difference is less pronounced due to the relatively fewer tweets in the dataset, which means LIN captures less additional information compared to LCN.

The third dataset examined is the IRA, where the DRN model also exhibits strong performance, though this is due to specific characteristics inherent in the dataset. Originally, the IRA dataset was created by gathering tweets from coordinated users; tweets from normal users were later added as noise (Machanayaka et al. 2024). Further analysis indicated that while the hashtag distributions of normal users align with those of coordinated users, the latter group is distinguished by their retweet records, normal users only have original tweets with no retweet record. In the DRN model, only the retweets where coordinated users retweet normal users' tweets are recorded, which explains the DRN's effectiveness on this dataset. Therefore, Retweet LIN for this dataset would not include normal users, given their lack of retweet records, so we will not consider Retweet as one of the activities in this dataset. Nonetheless, the Hashtag LIN still delivers the best performance by effectively distinguishing between coordinated and normal users. Overall, across all activities, the LIN model demonstrates advantages over the LCN model.

The final dataset is Digg, where each network result has its strengths and weaknesses. Overall, the Friends Network performs poorly, showing an advantage only in the AUC metric. LIN and LCN each have their pros and cons. This is likely because activities on Digg involve liking news, where friends may have similar preferences, but liking is a less observable activity in comparison to retweeting or using hashtags and it seems like friends' liking preferences may not have a similar impact on users.

Overall, LIN shows advantages over LCN across multiple datasets and metrics, and even more substantial improvements over DRN and the Friends Network. This method holds significant potential for constructing user-influence networks.

Comparative Analysis of LIN and LCN

To better compare the differences between LCN and LIN, Table 3 presents T-tests and Cohen's d values calculated for various metrics of both models. They both go through the LIDET framework and get the best parameter setting. A p-value less than 0.05 in the T-tests indicates a statistically significant difference in their performance on the tasks. The effect size reported in the table is Cohen's D (Cohen 2013), a well-known statistical measurement. It quantifies the difference between two means expressed in terms of their pooled standard deviation, facilitating an understanding of the magnitude of the effect in a standardized way that is independent of the unit of measurement. The effect size indicates the practical significance of these differences: values below 0.2 suggest a negligible difference, not apparent in practical applications, while values above 0.8 indicate a substantial difference with significant practical implications. Negative values suggest an advantage for LCN. Our analysis reveals that in almost all datasets across all activities, there are significant differences between LIN and LCN, with LIN generally showing a greater advantage.

Given the overall advantages of LIN, it is pertinent to discuss the instances where LIN does not demonstrate superiority. Specifically, in the COVID-19 dataset, the Retweet activity shows no significant differences between LCN and LIN in three metrics; both approaches approached near-perfect performance and captured a large number of edges, making it difficult to discern performance differences in this activity. More notably, in the Reply activity, LCN outperforms LIN. This is likely due to the scarcity of Reply activities within the dataset, where LIN's refined definition of influence further granulates the limited data, thus creating a performance disparity. In subsequent datasets like Twitter15, Twitter16, and IRA, LIN consistently demonstrates significant advantages over LCN. In the Digg dataset, LCN overall performs better than LIN. This is primarily because activities like liking, which have a relatively weaker influence compared to other activities, tend to foster coordinated relationships, indicating a "like attracts like" phenomenon. Consequently, the coordination-based approach of LCN performs better than the influence-based approach of LIN in this context, aligning with our understanding of LCN and LIN. Despite this, LIN's overall performance remains comparable to LCN, with some metrics showing no significant differences.

With this comparison, we are able to see that LIN's assumptions about influence do have a positive impact on the task of distinguishing between these user behavioral labels, as well as being able to show performance beyond that of

Dataset	Activities	Stat	AUC	Prec.	Rec.	F1
Covid-19	Retweet	p	**0.0001**	0.2700	0.2631	0.2610
		d	**2.4576**	**0.4203**	**0.4117**	**0.3998**
	Hashtag	p	**0.0017**	**0.0030**	**0.0003**	**0.0004**
		d	**1.9570**	**3.0571**	**3.0592**	**3.0673**
	URL	p	**0.0002**	**0.0138**	**0.0020**	**0.0138**
		d	**3.5040**	**1.4775**	**1.4845**	**1.4797**
	Mention	p	0.6231	**0.0002**	**0.0002**	**0.0002**
		d	**0.4194**	**3.2219**	**3.2273**	**3.2333**
	Reply	p	**0.0006**	**0.0010**	**0.0021**	**0.0021**
		S	-2.2810	-2.2448	-1.9602	-1.9369
Twitter15	Retweet	p	**0.0002**	**0.0002**	**0.0002**	**0.0002**
		d	**10.026**	**5.7075**	**5.9564**	**5.9400**
Twitter16	Retweet	p	**0.0002**	0.2413	0.4723	0.1620
		d	**6.6068**	**0.5663**	**0.3481**	**0.6007**
IRA	Hashtag	p	**0.0211**	**0.0002**	**0.0006**	**0.0002**
		d	**1.1882**	**4.3497**	**1.7480**	**4.3453**
	Mention	p	**0.0002**	**0.0043**	**0.0001**	**0.0002**
		d	**7.3565**	**2.0068**	**5.6534**	**5.1135**
	URL	p	**0.0058**	**0.0002**	**0.0001**	**0.0002**
		d	**1.6270**	**3.9359**	**4.7522**	**4.4561**
Digg	Like	p	**0.0003**	0.0757	0.2228	0.6776
		d	-3.2003	**1.1354**	-0.4977	-0.0115

Table 3: Statistical Significance of Performance Differences between LCN and LIN Across Various Datasets and Activities. Bold entries indicate p-values less than 0.05 and positive Cohen's d values, suggesting significant differences where LIN demonstrates a comparative advantage.

LCN, which undoubtedly adds credibility to the superiority of the LIN model over LCN in tasks that involve distinguishing user behavior tags. The enhanced assumptions about influence within the LIN model not only provide a more refined analysis but also demonstrate its capability to capture nuances in user interactions more effectively than LCN. This distinction is crucial in scenarios where the subtleties of influence are pivotal to understanding complex social dynamics. This not only validates the theoretical underpinnings of LIN but also underscores its practical value in real-world applications where distinguishing between different types of user behaviors is essential.

Case Study: COVID-19 Vaccination Dataset

For our case study, we provide a comprehensive examination of the LIDET framework, illustrating its functionality and the insights it can produce. We utilized a COVID-19 dataset to demonstrate the LIDET framework and perform comparative analyses across different LINs within this framework. The dataset spans one month, specifically from February 21, 2021 to March 24, 2021, focusing on COVID-19 vaccine-related content. It was crucial to select a timeframe that wasn't excessively lengthy, yet offered substantial data. A month-long period was deemed an apt choice for this balance. Subsequently, we used the GPT model for stance detection, classifying users into three categories based on their stance towards vaccines: supportive, neutral, and opposed.

Then, we constructed LIN networks according to the methodology outlined in Algorithm 1. The hyperparameters used are as follows:

- Activities: Retweet, Hashtag, Mention, Reply, URL
- Time Windows: 1, 5, 10, 30, 60 minutes
- Thresholds: 0, 1, 2, 3, 4, 5

This approach resulted in the generation of 150 distinct LINs. Including the baseline DRN, the total number of networks analyzed amounts to 151.

Given that the LIN can encompass a wide range of links, to maintain the manageability of LIN connections, we opted to filter the network by selecting only the most active users. This approach involves displaying users who post tweets daily, significantly reducing the total number of nodes in the network. Such a method ensures the feasibility of visualization and highlights the interactions between the most active and influential participants in the network, thus more clearly representing the most dynamic elements within the LIN.

Illustrative Example: The Impact of Parameter Selection on Network Structure

The different parameters applied to LIN can yield a range of networks, each revealing distinct patterns of relationships and connections among user groups. This variation stems from the fact that certain parameters illuminate diverse aspects of user interactions. Notably, larger time windows and lower thresholds tend to include more spurious connections in the network. Therefore, the primary challenge for using LIN lies in fine-tuning these parameters to maximize real connections while effectively filtering out the spurious ones.

Figure 2 serves as an illustrative example in the Covid-19 dataset elucidating the necessity of parameter selection. The left side of the figure depicts an LIN with retweet as the target activity, a 30-minute time window, and a threshold of 0, while the network on the right side utilizes a 60-minute time window. In these networks, nodes represent different users, and colours indicate distinct communities identified by the Louvain algorithm.

In the 30-minute time window of the Retweet LIN, two isolated nodes are unconnected to the main network group. However, when the time window is extended to 60 minutes, these nodes establish connections with the main network. A review of the tweets from these four users revealed that they were focused on vaccine issues in a specific city in India, justifying their categorization within the same community.

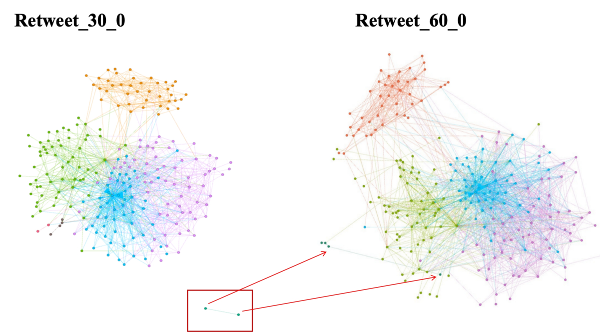


Figure 2: Visualization of two Retweet LINs with 30 and 60 minutes time windows

These observations underscore the significance of a 60-minute setting in revealing certain insights. However, it is also important to note that many of the connections within the 60-minute time window are coincidental and do not reflect real influence. It is impractical to manually inspect every influence link. Therefore, a more systematic and automated methodology is required. This necessity motivates us to develop the LIDET framework, focusing on parameter optimization within the context of LIN.

Classification Results

Table 4 displays the classification accuracy of various networks, where the network names follow the format Activity_Timewindow_Threshold. The models are named in two parts: the first part indicates the node embedding method used, and the second part denotes the machine learning model employed, namely Support Vector Machine (SVM), Logistic Regression (LG), Random Forest (RF), Gradient Boosting (GB), and Neural Network (NN). Due to limited space, for each activity, we have chosen to show only the results of the best network parameter settings.

From Table 4, it is evident that the traditional Directed Retweet Network (DRN) achieves accuracy from 77% to 79%. Given that users with a neutral stance are negligible in the network, this classification task can be treated as binary. Users with a pro-vax stance constitute approximately 65%

Models	DRN	HASHTAG_30_0	MENTION_60_1	REPLY_30_0	RETWEET_60_1	URL_60_1
deep_walk_SVM	0.7784	0.9651	**0.9630**	0.8911	0.9866	0.9821
deep_walk_LG	0.7735	0.9612	0.9546	0.8554	**0.9883**	0.9811
deep_walk_RF	0.7749	0.9667	0.9595	0.8796	0.9851	0.9824
deep_walk_GB	0.7635	0.9684	0.9571	0.8637	0.9833	**0.9849**
deep_walk_NN	0.7277	0.9662	0.9496	0.8197	0.9846	0.9841
node2vec_SVM	0.7805	**0.9741**	0.9610	**0.8943**	0.9853	0.9808
node2vec_LG	0.7696	0.9634	0.9548	0.8510	0.9874	0.9810
node2vec_RF	0.7749	0.9572	0.9573	0.8809	0.9878	0.9841
node2vec_GB	0.7607	0.9626	0.9581	0.8586	0.9863	0.9841
node2vec_NN	0.7283	0.9581	0.9561	0.8230	0.9840	0.9792
SDNE_SVM	0.7978	0.8850	0.8948	0.6680	0.9630	0.9276
SDNE_LG	0.7789	0.9498	0.9229	0.7160	0.9675	0.9725
SDNE_RF	**0.7983**	0.9469	0.9116	0.7100	0.9730	0.9715
SDNE_GB	0.7969	0.9522	0.9209	0.7280	0.9720	0.9767
SDNE_NN	0.7780	0.9493	0.9303	0.7100	0.9705	0.9772
LINE_SVM	0.7978	0.9605	0.9545	0.7220	0.9715	0.9839
LINE_LG	0.7915	0.9503	0.9526	0.7160	0.9770	0.9824
LINE_RF	**0.7983**	0.9157	0.9392	0.7020	0.9690	0.9815
LINE_GB	0.7965	0.9430	0.9442	0.7200	0.9720	0.9824
LINE_NN	0.7509	0.9518	0.9560	0.7120	0.9750	0.9848

Table 4: Classification accuracy of various models across different LINs. This table compares the performance of combinations of node embedding methods and machine learning models in classifying user stances within LINs. Each LIN is identified by its activity type, time window, and threshold setting. The highest accuracy in each LIN category is underlined to indicate the most effective model and parameter combination.

of the network. Consequently, using a naive ZeroR classifier, which predicts every user as pro-vax would achieve an accuracy of around 65%. The user features derived through DRN improve the accuracy by about 10%, which indicates DRN's network structure may have some effect on classifying users.

In contrast, the LINs exhibit a more distinct advantage, with the Retweet LIN approaching an impressive 99% accuracy. The URL LIN's performance is closely aligned with that of the Retweet LIN, and the visualisation of network structures in Figure 6 also show the similarity of these two types of LINs. Slightly lower, yet still notable, are the HASHTAG and MENTION categories, which achieve 95% accuracy. The weakest one is the Reply category, with fewer links, but it still manages to get a commendable performance close to 90%, which is a substantial improvement over the DRN.

This analysis underscores the effectiveness of LINs in social media data classification tasks, especially in comparison to traditional approaches like the DRN. The high accuracy rates in Retweet and URL LINs highlight the importance of using user interactions in predicting stances within a network. Despite its lower performance, the DRN still provides a basic understanding of user interactions and stances.

Detailed Analysis of the Retweet LIN Result We first focus on the Retweet LIN due to its superior performance compared to other LINs. By analyzing the most effective network initially, we can identify common traits and patterns that may apply to LINs based on other activities, providing a foundational understanding before moving to more detailed comparisons. In Fig. 3, the fluctuations of the accuracy (ACC) scores and F1 scores for the Retweet LIN are detailed across various thresholds and time windows. Both metrics vary with the expansion of the time window and threshold. The F1 score, a harmonic mean of precision and recall, and the accuracy rate, the proportion of correctly classified instances out of the total instances, are critical indicators of a classification model's performance. In a task approximating binary classification, where category distribution is imbalanced (such as 65 to 35), the F1 score is especially important as it offers a more equitable performance assessment for the

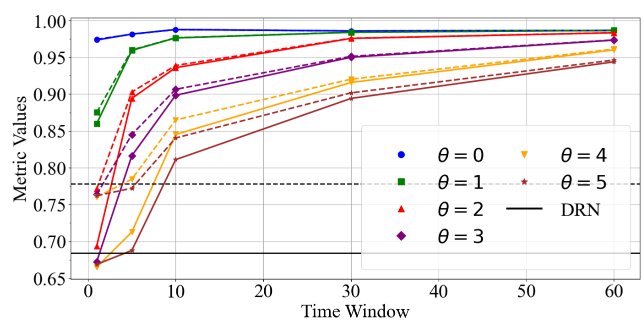


Figure 3: F1 scores (solid lines) and accuracy (dashed lines) of LIN models on the Retweet task. Each line corresponds to a different threshold θ, distinguished by color and marker. Black lines represent the DRN baseline.

minority class by balancing precision and recall.

In Fig. 3, we observe:

- Overall Trend: There is an ascending trajectory for both F1 scores and accuracy with the increase in time window, indicating an enhancement in model performance for classification tasks as the considered time frame broadens. This likely occurs because a larger time window allows the model to capture a richer array of user behaviours and influence patterns.
- Impact of Different Thresholds: Each line style, defined by a unique combination of color and marker, represents a different threshold θ used in the LIN framework. Lower thresholds (e.g., $\theta = 0$ or $\theta = 1$) tend to show sharper performance improvements in earlier time windows, indicating sensitivity to immediate influence. In contrast, higher thresholds (e.g., $\theta = 4$ or $\theta = 5$) result in more stable gains over longer time windows, suggesting that a broader temporal context can effectively suppress noise while still capturing influence dynamics. These trends highlight the trade-off between responsiveness to recent interactions and robustness to noisy fluctuations.
- DRN Baseline: The black lines represent the baseline for accuracy and F1 scores of the DRN. Compared to other curves, the DRN's performance is lower, emphasizing the substantial improvements in these two metrics by methods like Retweet LIN over traditional DRN approaches.
- Choice of Time Window: Selecting an appropriate time window is crucial for practical applications. Smaller time windows may fail to capture sufficient information, while larger ones might include excessive noise. The chart indicates that performance gains tend to plateau beyond a certain point, suggesting that extending the time window past a certain threshold does not significantly enhance model performance. For Retweet LIN, a low threshold setting of 5 to 10 minutes already achieves commendable performance, negating the need for excessively long time windows.

Comparative Analysis of LINs Across Different Activities For other LINs, we observe characteristics similar to those of the Retweet LIN. In the Hashtag LIN, a time window of approximately 5 to 10 minutes, along with a threshold of 0, yields commendable performance, with limited gains from subsequent increases in the time window. However, for higher thresholds, enhancing the time window can also significantly improve performance. Overall, a lower threshold and shorter time window are preferable.

In the URL LIN, a threshold of 0 also achieves good performance within a shorter time window. However, it is noteworthy that in the URL LIN, we observe a more rapid performance improvement with longer time windows and higher thresholds compared to the hashtags and Retweet LINs. These observations are not accompanied by figures due to space constraints. This may be due to URLs representing a larger quantity of information that is less likely to be repeated, leading to fewer incidental connections, thereby being more adaptable to longer time windows and higher thresholds.

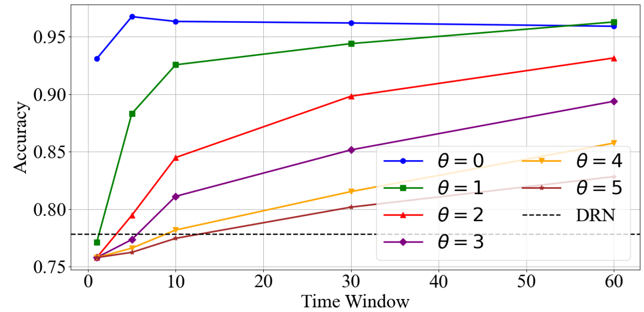


Figure 4: Accuracy performance in the Mention LINs

In the Mention LIN, the pattern is even more pronounced as shown in Fig 4, with the network at 60 minutes with a threshold of 1 surpassing the performance of the 60-minute network at threshold 0. Moreover, with the change in time window, there is a noticeable decline in performance at threshold 0. Thus, the Mention LIN is suited to a time window of 5 minutes or less and a low threshold; otherwise, a high threshold for a longer duration is recommended.

In the Reply LIN result as shown in Fig 5, since it carries less information, increasing the threshold can result in performance worse than the DRN. Overall, Reply performs best at a threshold of 0, with significant performance gains within 0 to 30 minutes, whereas performance tends to decline at 60 minutes.

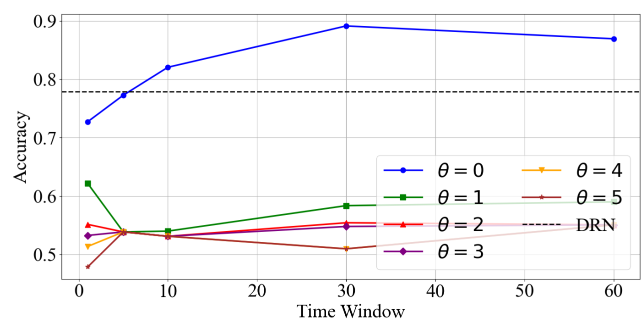
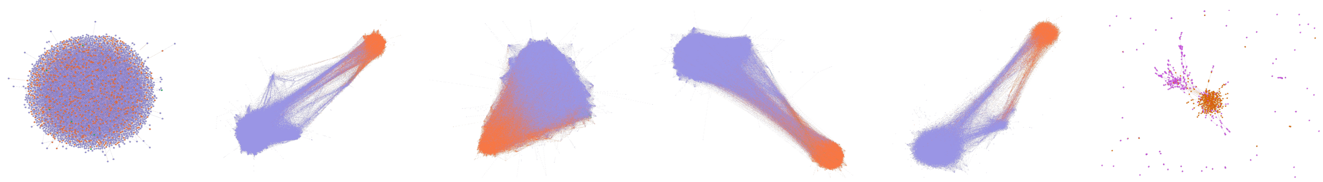


Figure 5: Accuracy performance in the Reply LINs

Visualization of Network Structures

We examine different network structures across various categories using visualization. Figure 6 displays LINs for each activity, wherein the parameters selected are those that yielded the best classification results within the category. The node colours represent the users' stance on vaccines: magenta for pro-vaccine, orange for anti-vaccine, and green for neutral. As these are highly active users, those users that maintain a completely neutral stance are exceedingly rare, making green nodes almost imperceptible in the visualizations. The majority are orange and magenta.

In the DRN, after applying the ForceAtlas2 layout algorithm, the network generally assumes a circular shape. Notably, pro-vaccine and anti-vaccine users are interspersed,



(a) DRN (b) Retweet LIN (c) Hashtag LIN (d) URL LIN (e) Mention LIN (f) Reply LIN

Figure 6: Visual representation of different network structures, including DRN and various LINs based on retweets, hashtags, mentions, URLs, and replies.

lacking a clear demarcation between the two groups. This observation transitions into a stark contrast when examining the Retweet LIN, where the influence and impact over users' opinions become significantly clearer. The Retweet LIN distinctly showcases the concentration of specific stances within user groups, a level of clarity that the DRN fails to achieve.

For the LIN visualizations, it is evident that distinct pro and anti-vaccine communities are visually discernible. The LIN networks formed by retweets and URLs have similar shapes, but the retweet LIN has more small communities linking between the pro and anti-vaccine clusters. The network formed by hashtags is more similar to the DRN, with closely connected nodes positioned nearer to each other. Hence, in the hashtag LIN, the distance between the pro and anti-vaccine clusters is not as pronounced as other LINs, possibly because both pro and anti-vaccine users might use the same hashtags, indicating lower distinctiveness. However, this is still a marked improvement over the DRN. The mention LIN features a prominent community linking pro and anti-vaccine clusters, likely due to both groups may mentioning the same users. For instance, conspiracy theories regarding microchips in vaccines mention Bill Gates and pro-vaccine groups also mention Bill Gates in relation to his vaccine support tweet. Finally, the reply LIN is noticeably sparser due to fewer interactions through replies, leading to many isolated communities outside the main network. But still, we could observe pro-vax and anti-vax communities in the reply LIN.

Discussion

Difference Between LCN and LIN

Since LIN was inspired by LCN, it is necessary to describe the differences that LIN brings. Comparing LIN with LCN illustrates how LIN enhances the understanding of user interactions and influence patterns, providing additional insights. In the LCN method, data is analyzed within fixed time intervals such as every 10 minutes or hour, without considering interactions across these periods, leading to potential cutoffs at the boundaries. Conversely, the LIN method adjusts time windows based on user activity; for instance, a window might start with user A's post and extend for ten minutes afterward. This adaptive timing allows for overlapping windows that can capture dynamic interactions more effectively than the rigid intervals of LCN.

Moreover, LCN primarily functions to identify highly correlated user communities. In LCN, edges represent coordinated activities between users and have no direction. However, in reality, coordination may also have direction, for example, if A and B both retweet C's post, it is possible that A is the dominant player in the process, and the next time A retweets D's post, B will retweet it as well, meanwhile, if B retweets D's post, A may not necessarily retweet it. Therefore, we propose LIN whose edges represent influence channels with directionality, indicating the flow of influence among users. Unlike LCN, LIN does not focus on identifying communities but on mapping out how influence travels through the network. As a result, LIN addresses the issue of semantic ambiguity in network connections present in LCN, providing clarity on the significance and importance of these connections.

Future Work

In exploring the potential of LIN, future research could focus on refining the timing of user influence windows to better reflect the real patterns of user behavior. For instance, users might follow certain patterns, such as browsing tweets during specific time periods, making them more susceptible to influence from tweets published in these intervals. Additionally, the influence should logically wane over time, necessitating the introduction of a time-based weighting system. Moreover, another pivotal question for future exploration is whether there exists a method to summarize all the information from these various LINs to construct a network that more accurately reflects real-world influence patterns. For instance, shorter time windows in LINs could be more adept at detecting influence between retweet bots, while longer windows might capture extended influence trends more effectively. Investigating the potential for combining these diverse insights could lead to a more comprehensive and realistic representation of influence dynamics on social networks, enhancing the utility of LIN in both analytical and practical applications.

Conclusion

This paper proposes a novel approach to understanding influence propagation on social media platforms. We introduced the concept of a Latent Influence Network (LIN), which significantly advances the traditional methodology of retweet networks by incorporating the direction of influence. This innovation allows for a more nuanced and granular representation of user interactions, providing deeper insights into how opinions and information are disseminated through social media.

Our approach diverges from traditional practices by focusing on the temporal order of activities among users. Through LIDET framework, we successfully identify the optimal parameters for building meaningful network structures that overcome the limitations of traditional retweet networks. A comparative analysis of Twitter activity in these influence networks reveals that LINs, especially those constructed from retweeting and URL sharing activity, have a clear advantage in distinguishing different types of users.

We tested LIN's usability on four different social network datasets, and LIN shows a large improvement for DRNs as well as friend networks, and a clear advantage for LCN. This highlights the robustness and practicality of our approach. This improvement is not only quantitatively significant but also qualitatively important for understanding the dynamics of social media influence.

References

Brady, J. T.; Kelly, M. E.; and Stein, S. L. 2017. The Trump Effect: With No Peer Review, How Do We Know What to Really Believe on Social Media? *Clinics in Colon and Rectal Surgery*, 30(4): 270–276. Epub 2017 Sep 12.

Cherepnalkoski, D.; and Mozetič, I. 2016. Retweet networks of the European Parliament: Evaluation of the community structure. *Applied network science*, 1: 1–20.

Cohen, J. 2013. *Statistical power analysis for the behavioral sciences*. Routledge.

Friedkin, N. E.; and Johnsen, E. C. 2011. *Social influence network theory: A sociological examination of small group dynamics*, volume 33. Cambridge University Press.

Graham, T.; Bruns, A.; Zhu, G.; and Campbell, R. 2020. Like a virus: The coordinated spread of Coronavirus disinformation. https://apo.org.au/node/305864. Accessed on 21-07-2022.

Grover, A.; and Leskovec, J. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 855–864.

Hogg, T.; and Lerman, K. 2012. Social dynamics of digg. *EPJ Data Science*, 1(1): 1–26.

Jendoubi, S.; Martin, A.; Liétard, L.; Hadji, H. B.; and Yaghlane, B. B. 2017. Two evidential data based models for influence maximization in twitter. *Knowledge-Based Systems*, 121: 58–70.

Lamsal, R. 2021. Design and analysis of a large-scale COVID-19 tweets dataset. *Applied Intelligence*, 51(5): 2790–2804.

Ma, J.; Gao, W.; and Wong, K.-F. 2017. Detect rumors in microblog posts using propagation structure via kernel learning. Association for Computational Linguistics.

Machanayaka, I.; Zaidi, Z.; Karunasekera, S.; and Leckie, C. 2024. Identifying Coordinated Activities on Online Social Networks Using Contrast Pattern Mining. In *in IEEE WCCI (IJCNN)*.

Magelinski, T.; Ng, L. H. X.; and Carley, K. M. 2022. A synchronized action framework for responsible detection of coordination on social media. *Journal of Online Trust and Safety*, 1.

Ng, L. H. X.; and Carley, K. M. 2022. Online coordination: methods and comparative case studies of coordinated groups across four events in the united states. In *Proceedings of the 14th ACM Web Science Conference 2022*, 12–21.

Pacheco, D.; Hui, P. M.; Torres-Lugo, C.; Truong, B. T.; Flammini, A.; and Menczer, F. 2021. Uncovering coordinated networks on social media: methods and case studies. In *In Proceedings of the AAAI international conference on web and social media (ICWSM)*.

Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Vanderplas, J.; Passos, A.; Cournapeau, D.; Brucher, M.; Perrot, M.; and Duchesnay, E. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12: 2825–2830.

Perozzi, B.; Al-Rfou, R.; and Skiena, S. 2014. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 701–710.

Qiu, J.; Tang, J.; Ma, H.; Dong, Y.; Wang, K.; and Tang, J. 2018. Deepinf: Social influence prediction with deep learning. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, 2110–2119.

Song, C.; Wang, B.; Jiang, Q.; Zhang, Y.; He, R.; and Hou, Y. 2021. Social recommendation with implicit social influence. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, 1788–1792.

Tang, J.; Qu, M.; Wang, M.; Zhang, M.; Yan, J.; and Mei, Q. 2015. Line: Large-scale information network embedding. In *Proceedings of the 24th international conference on world wide web*, 1067–1077.

Wang, D.; Cui, P.; and Zhu, W. 2016. Structural deep network embedding. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 1225–1234.

Weber, D.; and Falzon, L. 2021. Temporal Nuances of Coordination Networks.

Weber, D.; and Neumann, F. 2020. Who's in the gang? Revealing coordinating communities in social media. In *2020 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, 89–93. IEEE.

Zaidi, Z.; Ye, M.; Samon, F.; Jama, A.; Gopalakrishnan, B.; Gu, C.; Karunasekera, S.; Evans, J.; and Kashima, Y. 2023. Topics in antivax and provax discourse: yearlong synoptic study of COVID-19 vaccine tweets. *Journal of Medical Internet Research*, 25: e45069.

Paper Checklist

1. For most authors...
   (a) Would answering this research question advance science without violating social contracts, such as violating privacy norms, perpetuating unfair profiling, exacerbating the socio-economic divide, or implying disrespect to societies or cultures? Yes
   (b) Do your main claims in the abstract and introduction accurately reflect the paper's contributions and scope? Yes
   (c) Do you clarify how the proposed methodological approach is appropriate for the claims made? Yes
   (d) Do you clarify what are possible artifacts in the data used, given population-specific distributions? NA
   (e) Did you describe the limitations of your work? Yes
   (f) Did you discuss any potential negative societal impacts of your work? No, because we do not see any negative societal impact of our work.
   (g) Did you discuss any potential misuse of your work? No, because the risk that our work on influence can be exploited by malicious actor for creating misinformation campaign is very low.
   (h) Did you describe steps taken to prevent or mitigate potential negative outcomes of the research, such as data and model documentation, data anonymization, responsible release, access control, and the reproducibility of findings? Yes, our research is approved by the institution's ethics committee and conforms to the guidelines of responsible use of social media data.
   (i) Have you read the ethics review guidelines and ensured that your paper conforms to them? Yes
2. Additionally, if your study involves hypotheses testing...
   (a) Did you clearly state the assumptions underlying all theoretical results? NA
   (b) Have you provided justifications for all theoretical results? NA
   (c) Did you discuss competing hypotheses or theories that might challenge or complement your theoretical results? NA
   (d) Have you considered alternative mechanisms or explanations that might account for the same outcomes observed in your study? NA
   (e) Did you address potential biases or limitations in your theoretical framework? NA
   (f) Have you related your theoretical results to the existing literature in social science? NA
   (g) Did you discuss the implications of your theoretical results for policy, practice, or further research in the social science domain? NA
3. Additionally, if you are including theoretical proofs...
   (a) Did you state the full set of assumptions of all theoretical results? NA
   (b) Did you include complete proofs of all theoretical results? NA
4. Additionally, if you ran machine learning experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? Yes, we include the instructions needed to reproduce the main experimental results. Unfortunately due to the changes in Twitter (now X)'s policies, accessing the tweets from the shareable tweet IDs may not be possible for some researchers following our work, but this is beyond our control.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? Yes, we describe some details in the experiment section. Also, we are planning to publish our code on GitHub on publication of this paper.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? Yes, we report the T-test and effect size.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? No, because the focus of this article is not on training large machine learning models and the computation is not large.
   (e) Do you justify how the proposed evaluation is sufficient and appropriate to the claims made? Yes, we discuss that in detail in our Experiments and Conclusions sections.
   (f) Do you discuss what is "the cost" of misclassification and fault (in)tolerance? Yes, we discussed model biases and classification outcomes in detail. The explicit cost of misclassification is not central to our study.
5. Additionally, if you are using existing assets...
   (a) If your work uses existing assets, did you cite the creators? Yes, in the dataset section.
   (b) Did you mention the license of the assets? NA
   (c) Did you include any new assets in the supplemental material or as a URL? NA
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? No, our use of Twitter (now X)'s data conforms to their policies.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? No, because no user is identifiable in this paper.
   (f) If you are curating or releasing new datasets, did you discuss how you intend to make your datasets FAIR? NA
   (g) If you are curating or releasing new datasets, did you create a Datasheet for the Dataset? NA
6. Additionally, if you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots? NA
   (b) Did you describe any potential participant risks, with mentions of IRB approvals? NA

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? NA

(d) Did you discuss how data is stored, shared, and deidentified? NA