Social Network Analysis and Mining (2022) 12:66
https://doi.org/10.1007/s13278-022-00890-z

ORIGINAL ARTICLE

Fake news spreader detection using trust-based strategies in social networks with bot filtration

Bhavtosh Rath[1] · Aadesh Salecha[1] · Jaideep Srivastava[1]

Received: 18 February 2021 / Revised: 9 May 2022 / Accepted: 16 May 2022 / Published online: 26 June 2022

Abstract

An important aspect of preventing fake news spreading in social networks is to proactively detect the users that are likely going to spread such news. Research in the domain of spreader detection is at a nascent stage compared to fake news detection. In this paper, we propose a graph neural network-based framework to identify nodes that are likely to become spreaders of false information. Using the community health assessment model and interpersonal trust (quantified using network topology and historical behavioral data), we propose an inductive representation learning framework to predict nodes of densely connected community structures that are most likely to spread fake news, thus making the entire community vulnerable to the infection. We also analyze the performance of our model in the presence and absence of bots detected using an existing state-of-the-art bot detection model. Using topology- and activity-based trust properties sampled and aggregated from neighborhood of nodes, we are able to predict false information spreaders better than refutation information spreaders.

1 Introduction

Social media platforms have become a ubiquitous part of daily lives. People use these platforms to connect with loved ones, for entertainment and increasingly rely on them as their primary source of news. Research has in fact shown than around 70% of people now get their news from online sources and 37% of this is made up of social media platforms entirely (Newman et al. 2020). But with this increase in reliance, there has also been a simultaneous rise in the massive diffusion of misinformation through these networks. This rise has brought a whole host of consequences with it, from swaying popular opinion during elections to generating mass panic during pandemics. Therefore, it is no surprise that researchers have been increasingly studying computational models for the detection and prevention of false information (popularly called *fake news*). Most of the literature have focused on identifying the veracity of information. But it is not only important to detect false information but also identify people who are most likely to believe and spread the false information. The development of these detection strategies can help contain and prevent the rapid spreading of fake news in social networks. While most existing work in fake news detection systems has focused on content- and propagation-based features, we propose a complementary approach that quantifies interpersonal trust using the social network topology and historical user activity. As the COVID-19 virus spread around the world, so did various rumors and false information regarding various aspects pertaining to it. The need for a spreader detection model and mitigation strategy for fake news has never been more evident. Thus, in this paper, we propose a novel spreader detection model that uses inductive representation learning allowing it to quickly identify spreaders before the false information penetrates deep into any densely connected community. The main contributions of the paper are as follows:

1. We identify a gap in existing literature related to a lack of an authoritative benchmark dataset and thus collect and publish the MinFN dataset (Rath 2021) consisting of real-world Twitter data from 10 unique news events along with their related fake and true tweets, users who have retweeted these tweets, their user-metadata and their *follower-followee* networks.

✉ Bhavtosh Rath
rathx082@umn.edu

Aadesh Salecha
salec006@umn.edu

Jaideep Srivastava
srivasta@umn.edu

[1] Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN, USA

2. We propose a fake news spreader detection framework using the Community Health Assessment model (Rath et al. 2019) and computational trust (Roy et al. 2016). To the best of our knowledge, this is the first fake news spreader detection model proposed that integrates features extracted from both underlying network structure (instead of the propagation structure) and historical behavioral data (instead of the content of the news).

3. We implement our framework using inductive representation learning (Hamilton et al. 2017) where we sample neighborhood of nodes in a weighted network and aggregate their trust-based features.

4. We evaluate our proposed interpersonal trust-based framework the MinFN and empirically show that trust-based modeling helps us identify false information spreaders with high accuracy, which makes the technique useful for fake news mitigation. We also investigate the effects of bots on our models and show that a bot-filtration step is essential to ensure optimal performance of our models.

5. We further observe that our model's accuracy when detecting false information spreaders is higher than that for true information spreaders. This indicates that people are usually able to reason about true information from analyzing the content, and thus, trust in their neighbors is not a very significant factor. However, determining the veracity of *false information that is plausibly true* from content itself is difficult and hence we have to rely on sources we trust to make this judgement. This makes nodes that are fake news spreaders and at the same time highly trusted by lots of people in the network, especially dangerous. We acknowledge that not all such *uber-spreaders* have ill intentions as some might be just ignorant. They all, nonetheless, have power to spread false information far and wide, with great speed.

This paper is an extended version of Rath et al. (2020). We build on the ideas and framework presented by first accounting for the effects of bots in our networks. We treat bots and humans separately, which is a major difference from Rath et al. (2020). We use stat-of-the-art bot detection techniques to accurately detect bots and then study the effects of their presence by running our models on networks void of bots which are more representative social networks comprised of actual people. We also present an extension of the dataset used in Rath et al. (2020) by providing a more comprehensive activity-feature-set for each user. We build new models that leverage this new set of features and compare their performance to our previous models. In addition, we also make the MinFN dataset public for other researchers to build on and evaluate their models furthering efforts to create a universal benchmark (Rath 2021).

The rest of the paper is organized as follows: We first discuss related work, then describe a motivating example for spreader detection from a network structure perspective, and summarize past ideas that the proposed research builds upon. We then explain the proposed framework and how we model interpersonal trust with it followed by experimental analysis. We do further analysis after bot filtration and increasing timeline data volume. Finally, we give our concluding remarks and proposed future work.

2 Related work

In this section, we highlight related work from four domains that our proposed framework build upon. They are (1) False Information in social networks, (2) Graph Neural Networks, (3) Computational Trust in social networks, (4) Detection of Bots in social networks.

2.1 False information in social networks

Research in the domain of fake detection and containment of false information is vast and varied. We discuss work along three main dimensions: Fake News Detection, Fake News Spreader Detection, and Fake News Datasets. Our work lies in intersection of these three fields.

2.1.1 Fake news detection

In order to study the credibility and gauge the validty of claims, researchers have employed techniques that generally fall into four buckets:

Content-based methods: These methods rely on lexical features, syntactic features and topic features. Pérez-Rosas et al. (2017) identify five major categories of differences between fake and true content—'Ngrams,' 'punctuation,' 'psycholinguistic features,' 'readability' and 'syntax.' Researchers have used these features to detect fake news (Potthast et al. 2017; Hu et al. 2014; Ito et al. 2015).

Social Context-based methods: These methods use the valuable data present in terms of human-content interaction data. From these data, researchers have extracted post-based features, which rely on users' individual opinions about the piece of information. Long (2017) concluded that the addition of these profile features led to a performance in existing fake news detection models. Guess et al. (2019) found correlations between party affiliation and how likely a user was to share fake posts on Facebook. Other researchers have extracted propagation-based features from this interaction data, which uses the overall information dissemination network. Wu et al. (2015) gleaned from propagation networks that fake messages are first posted by an ordinary user then forwarded by opinion leaders before finally reaching a large number of ordinary users, whereas in the case of true messages, it is first posted by opinion leaders and then reaches a large number of ordinary users. They developed a hybrid SVM model which utilized propagation structure to detect fake news. Jin et al. (2014) build on previous studies and built a propagation-based model that used microblogs, sub-events and events for information credibility validation.

Feature Fusion-based methods: Since content-based features and propagation-based features can be complementary, researchers have built fusion models that leverage both types of features (Shu et al. 2019; Della Vedova et al. 2018; Volkova and Jang 2018)

Deep Learning-based methods: These methods seek to use techniques to glean a abstracted view of fake news spread. The most widely used methods involve Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), Generative Adversarial Networks (GAN), and more recently Graph Convolutional Networks (GCN). Li et al. (2018) proposed a Bidirectional GRU model that utilized both directions of interaction information for fake news detection. Liu and Wu (2018) used CNNs and GRU to distinguish between true and false propagation paths. Chen et al. (2018) proposed a deep attention neural network that captured contextual variations of relevant posts over time. We discuss recent works that utilize Graph Neural Networks and GCNs in the subsequent sections as it closely relates to our work.

Table 1 Summary of dataset characteristics

Dataset	#Users	Time	Text	User	Network	Trust
TwitterDS	491,229	✓	✓	✓		
Twitter15		✓	✓	✓		
Twitter16		✓	✓	✓		
PolitiFact	1,540,190	✓	✓	✓	✓	
GossipCop	1,354,724	✓	✓	✓	✓	
PHEME-R	56,099	✓	✓	✓	✓	
PHEME		✓	✓	✓		
MinFN	Table 4	✓	✓	✓	✓	✓

#Users refer to the total number of unique users. Time refers to temporal time-stamp information associated with tweets, Text is the tweet text, User is the user metadata and Network is the follower-followee network for spreaders

2.1.2 Fake news spreader detection

Although most work in the domain of fake news has focused on detection of the content or the news itself. There is a smaller body of work for the detection of users that are most likely to spread fake news. Early work in this field used common user-metadata features like *number of followers*, *number of followees* (Almaatouq et al. 2016), *user-profile-name*, *email*, etc., (Arapakis et al. 2017) to detect suspicious profiles. Other work has used camera-based sensors (Castillo et al. 2011) and mobile phone tracking data (Carlini et al. 2016), to detect spammers and fake profiles. Pennycook and Rand (2020) conducted a survey-based study that studied the cognitive basis for identifying people who would believe fake news. Karami et al. (2021) used psychological profiling, and Shu et al. (2018) used other explicit and implicit profile features to identify trait differences between true news and fake news spreaders. One of the shared tasks at the PAN @ CLEF conference in 2020 was *Profiling Fake News Spreaders on Twitter* (Rangel et al. 2020). There were over 60 submissions that all tried to address this task (Cardaioli et al. 2020; Pizarro 2020; Vogel and Meghana 2020). The best accuracy that any of the submissions achieved was 75% as opposed to our best model which achieved an accuracy of 93% when detecting fake news spreaders. Giachanou et al. (2020) used CNNs and word embeddings to differentiate between users who spread fake news and who fact check it. This is the work that is closest to ours, but while Giachanou et al. (2020) proposes a model to differentiate between spreaders and checkers only, our framework is more general as it is able to distinguish between spreaders and any other type of user. We also achieve better performance with our model than Giachanou et al. (2020) (F1 scores of 0.59 vs 0.93).

2.1.3 Fake news datasets

Guo et al. surveyed all the publicly available datasets used in the domain of Fake News research and found that although there are a few open datasets for each of the major social networks, there is still a lack of standardized universal benchmark dataset (Guo et al. 2020). They attribute this to the time-consuming and labor-intensive nature of collecting fake news spread data. Some popular datasets that use data from *Twitter* are summarized as follows:

1. *TwitterDS*: Detecting Rumors from Microblogs with Recurrent Neural Networks (Ma et al. 2016)
2. *Twitter15*: Detect Rumors in Microblog Posts Using Propagation Structure via Kernel Learning (Ma et al. 2017)
3. *Twitter16*: Detect Rumors in Microblog Posts Using Propagation Structure via Kernel Learning (Ma et al. 2017)
4. *PolitiFact*: Fakenewsnet: A data repository with news content, social context and spatiotemporal information for studying fake news on social media (Shu et al. 2018)
5. *GossipCop*: Fakenewsnet: A data repository with news content, social context and spatiotemporal information for studying fake news on social media (Shu et al. 2018)
6. *PHEME-R*: Analyzing How People Orient to and Spread Rumors in Social Media by Looking at Conversational Threads (Zubiaga et al. 2016)
7. *PHEME*: All-in-one: Multi-task Learning for Rumor Verification (Kochkina et al. 2018)
8. *MinFN*: The *Min*nesota *F*ake *N*ews dataset that we publish as a part of this paper at (Rath 2021) (Table 1)

TwitterDS, Twitter15, Twitter16, PHEME are popular choices of datasets used in fake news research, but they all do not contain social-network information. Twitter15 and Twitter16 contain the tweet propagation-tree data as well while TwitterDS does not. Politifact, GossipCop, and PHEME-R datasets have almost all the information that our model intends to utilize, but they lack in the volume of data that they examine.

The critical difference in MinFN is that it provides network and twitter activity metadata for a false information and its refutation information. We summarize the MinFN dataset's characteristics in Table 4 which makes this difference more apparent.

2.2 Graph neural networks

Graph neural networks (Scarselli et al. 2008) are an emerging field of research that generalizes neural network models to graph structures. They have shown better performance over other node embeddings approaches which implement shallow learning methods. Some domains in which they have shown state-of-the-art improvement include computer vision (Defferrard et al. 2016; Monti et al. 2017), natural language processing (Yao et al. 2019; Zhang et al. 2018), molecular feature extraction (Duvenaud et al. 2015), extracting features from highly multi-relational data (Schlichtkrull et al. 2018), neuroimage analysis to perform disease progression modeling for people with Parkinson's disease (Zhang et al. 2018), traffic prediction (Yu et al. 2017) and for recommender systems (Monti et al. 2017; Ying et al. 2018) to name a few. Hu et al. (2019) is a recent work that proposed a graph neural network model for fake news detection using news content, but they model propagation paths while we model network topology. Usefulness of Graph neural networks can be explained by the fact that we want to analyze how the network structure and the trust-based node features can together be used to distinguish false information spreader from true information spreader. Other machine learning models do not capture this information.

2.3 Computational trust in social networks

Computational trust in social networks is a widely studied domain in which researchers have tried to assign trust scores to nodes of a network. Mui (2002) proposed a computational model for trust and reputation in social networks based on history of past interactions. Eigentrust by Kamvar et al. (2003) assigned global trust value to people sharing and distributing files in a P2P network which helped an ordinary user in the network to identify malicious peers and isolate them from the network. Mishra and Bhattacharya (2011) proposed a model to compute the bias and prestige of nodes in social networks which used an iterative matrix algorithm using edge weights. Roy et al. (2016) proposed the Trust in Social Media algorithm that assigned a pair of complementary trust scores called trustingness and trustworthiness to nodes. Our proposed research builds upon Roy's work.

2.4 Detection of bots in social networks

In order to mitigate the ill-effects of bots on social media platforms, researchers have proposed various bot detection methodologies. Ferrara et al. (2016) proposed a taxonomy for bot detection methods which included graph-based, feature-based, crowd-sourcing-based and hybrid approaches. Although there are a plethora of methods of differentiating between bots and humans, like that of Chu et al. (2013) that uses behavioral biometrics like mouse and keystroke patterns, our research leverages existing feature-based machine learning approaches to detect bots. BotOrNot Davis et al. (2016) is a well-known publicly available system that uses a wide array of features to decide if a certain account is a bot. Amleshwaram et al. (2013) proposed CATS which uses a list of URLs, entropy and community structures to detect spam bots on Twitter. Kudugunta and Ferrara proposed multiple models that use account-level features, tweet data or a combination of the two. One of their proposed deep learning architectures was able to detect bots using only a single tweet and user metadata (Kudugunta and Ferrara 2018). We build on models that they proposed for bot detection based on account-level features.

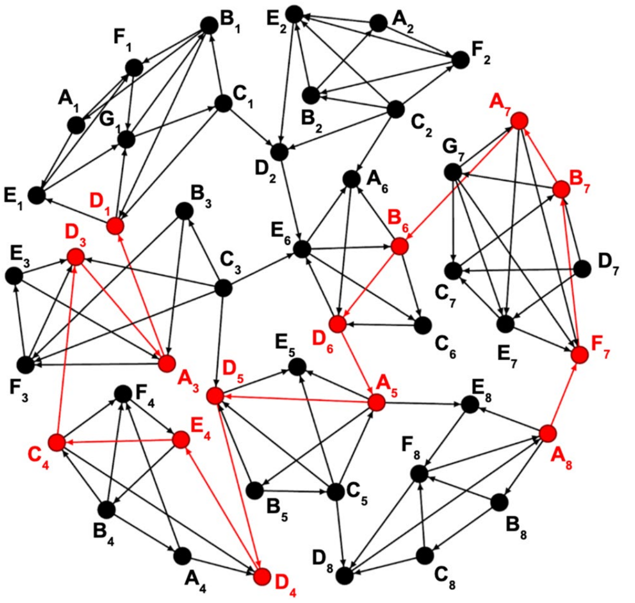


Fig. 1 Motivating example. Red nodes denote fake news spreaders

3 Motivation and preliminaries

To understand the role of network structure in fake news spreader detection, consider the scenario illustrated in Fig. 1. The network contains eight communities. Subscript of a node denotes the community it belongs to. In the context of Twitter, directed edge $B_1 \rightarrow A_1$ represents B_1 follows A_1. Thus, a tweet flows from A_1 to B_1. If B_1 decides to retweet A_1's tweet, we say that B_1 has endorsed A_1's tweet, and that B_1 trusts A_1. Communities in social networks are *modular* groups, where within-group members are tightly connected, and intra-community trust is higher, compared to trust between members in different communities, who are at best loosely connected. The more B trusts A, the higher the chance that B will retweet A's tweet, and thus propagate A's message, whether it is true or false. The figure illustrates the spread of fake news starting from D_1 as it spreads across the network through A_3 till A_8. We consider two scenarios for spreader detection:

1. Information reaches neighborhood of a community: Consider the scenario when a message is propagated by D_1, a neighborhood node for community 3. Node A_3 is exposed and is likely to spread the information, thus beginning spread of information into a densely connected community. Thus, it is important to predict nodes in the boundary of communities that are likely to become information spreaders.

2. Information penetrates the community: Consider the scenario where A_3 decides to propagate a message. Nodes B_3, D_3 and E_3, which are immediate followers of A_3, are now exposed to the information. Due to their close proximity, they are vulnerable to believing the endorser. The remaining nodes of the community (C_3, F_3) are two steps away from A_3. Similarly, for community 8 when the message has reached node A_8, nodes D_8 and F_8 are one step away and remaining community members (E_8, C_8, B_8) are two steps away. Intuitively, in a closely knit community structure, if one of the nodes decides to spread a piece of information, the likelihood of it spreading quickly within the entire community is very high. Thus, it is important to detect nodes within a community that are likely to become information spreaders to protect the health of the entire community.

Next, we discuss some concepts that our proposed model builds upon.

3.1 Community health assessment model

A social network has the characteristic property to exhibit community structures that are formed based on inter-node interactions. Communities tend to be modular groups where within-group members are highly connected, and across-group members are loosely connected. Thus, members within a community would tend to have a higher degree of trust among each other than between members across different communities. If such communities are exposed to fake news propagating in its vicinity, the likelihood of all community members getting infected would be high. Motivated by the idea of ease of spreading within a community, we use the Community Health Assessment model. The model identifies three types of nodes with respect to a community: neighbor, boundary and core nodes, which are explained below:

1. *Neighbor nodes*: These nodes are directly connected to at least one node of the community. The set of neighbor nodes is denoted by $\mathcal{N}_{com}$. They are not a part of the community.
2. *Boundary nodes*: These are community nodes that are directly connected to at least one neighbor node. The set of boundary nodes is denoted by $\mathcal{B}_{com}$. It is important to note that only community nodes that have an outgoing edge toward a neighbor nodes are in $\mathcal{B}_{com}$.
3. *Core nodes*: These are community nodes that are only connected to members within the community. The set of core nodes is denoted by $\mathcal{C}_{com}$.

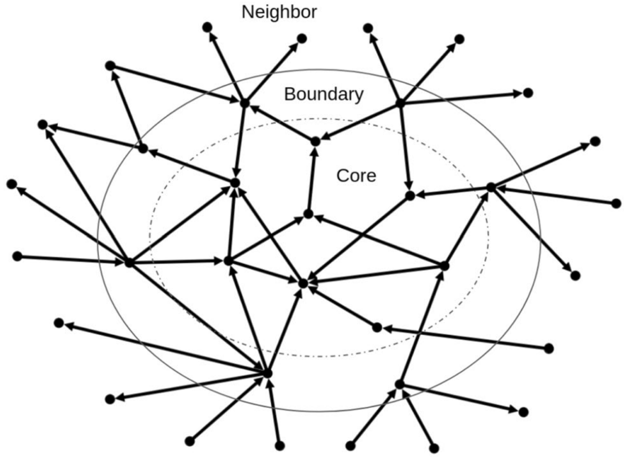


Fig. 2 Community health assessment model

Table 2 Neighbor, boundary and core nodes for communities in Fig. 1

com	$\mathcal{N}_{com}$	$\mathcal{B}_{com}$	$\mathcal{C}_{com}$
1	D_2	C_1	A_1,B_1,E_1,D_1,F_1,G_1
2	A_6,E_6	C_2,D_2	A_2,B_2,E_2,F_2
3	D_1,D_5,E_6	A_3,C_3	B_3,D_3,E_3,F_3
4	D_3	C_4	A_4,B_4,D_4,E_4,F_4
5	D_4,D_8,E_8	D_5,A_5,C_5	E_5,B_5
6	A_5	D_6	A_6,B_6,C_6,E_6
7	B_6	A_7	B_7,C_7,D_7,E_7,F_7, G_7
8	$F7$	A_8	B_8,C_8,D_8,E_8,F_8

The idea was proposed in Rath et al. (2019) to show how trust plays a more important role in spreading fake news compared to true news. The neighbor, boundary, and core nodes for communities in Fig. 1 are listed in Table 2 (Fig. 2).

3.2 Trustingness and trustworthiness

In the context of social media, researchers have used social networks to understand how trust manifests among users. The Trust in Social Media (TSM) algorithm is a technique that assigns a pair of complementary trust scores to each node in a network called *Trustingness* and *Trustworthiness*. *Trustingness (ti)* quantifies the propensity of a node to trust its neighbors and *Trustworthiness (tw)* quantifies the willingness of the neighbors to trust the node. The TSM algorithm takes a user network, i.e., a directed graph $\mathcal{G}\Leftarrow\mathcal{V}\Leftrightarrow\mathcal{E}\Rightarrow$, as input together with a specified convergence criteria or a maximum permitted number of iterations. In each iteration for every node in the network, trustingness and trustworthiness are computed using the following equations:

$$\left(ti(v) = \sum_{\forall x \in out(v)} \left(\frac{w(v,x)}{1 + (tw(x))^s} \right) \right) \tag{1}$$

$$tw(u) = \sum_{\forall x \in in(u)} \left(\frac{w(x,u)}{1 + (ti(x))^s} \right) \tag{2}$$

where $u, v, x \in \mathcal{V}$ are user nodes, $ti(v)$ and $tw(u)$ are trustingness and trustworthiness scores of v and u, respectively, $w(v, x)$ is the weight of edge from v to x, $out(v)$ is the set of out-edges of v, $in(u)$ is the set of in-edges of u, and s is the involvement score of the network. Involvement is basically the potential risk a node takes when creating a link in the network, which is set to a constant empirically. The details of the algorithm are excluded due to space constraints and can be found in Roy et al. (2016).

3.3 Believability

Believability is an edge score derived from the Trustingness and Trustworthiness scores (Rath et al. 2017). It helps us to quantify the potential or strength of directed edges to transmit information by capturing the intensity of the connection between the sender and receiver. Believability for a directed edge is computed as a function of the trustworthiness of the sender and the trustingness of the receiver.

More specifically, given users u and v in the context of microblogs such as Twitter, a directed edge from u to v exists if u follows v. The believability quantifies the strength that u trusts on v when u decides to follow v. Therefore, u is very likely to believe in v if:

1. v has a high trustworthiness score, i.e., v is highly likely to be trusted by other users in the network, or
2. u has a high trustingness score, i.e., u is highly likely to trust others.

So, the believability score is supposed to be proportional to the two values above, which can be jointly determined and computed as follow:

$$\text{Believability}(u \to v) = tw(v) * ti(u) \tag{3}$$

The idea has been previously applied in Rath et al. (2017) where a classification model was built to identify rumor spreaders in Twitter user network based on believability measure. Based on Zhao and Rosson (2009), *information posted by a person the reader has deliberately selected to follow on Twitter is perceived as useful and trustworthy*, which intuitively implies that follow relation can be considered as proxy for trust.

4 Proposed approach

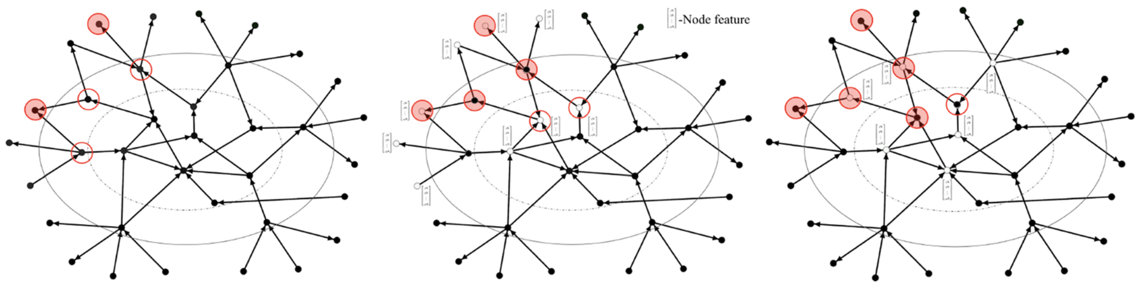
4.1 Problem formulation

Given a directed social network $\mathcal{G}\Leftarrow\mathcal{V}\Leftrightarrow\mathcal{E}\Rightarrow$ comprising disjoint modular communities (ϕ), with each community ($com \in \phi$) having well-defined neighbor nodes ($\mathcal{N}_{com}$), boundary nodes ($\mathcal{B}_{com}$) and core nodes ($\mathcal{C}_{com}$). Aggregating topology-based (top) and activity-based (act) trust properties from nodes sampled from depth K (where $Nbr_{K=1}(b) \subseteq \mathcal{N}_{com}$), we want to predict boundary nodes b that are most likely to become information spreaders (b_{sp}). Similarly, we aggregate nodes sampled from depth K (where $Nbr_{K=1}(c) \subseteq \mathcal{B}_{com}$) to predict core nodes c that are most likely to become information spreaders (c_{sp}).

4.2 Inductive representation learning model

Most researchers have studied fake news dissemination after the spreading of the information. But, any mitigation strategy that seeks to minimize the spread of fake news will have to work dynamically and adapt to the fast-changing nature of fake news spread. Therefore, keeping this in mind, we employ a machine learning technique that is scalable and can adapt efficiently to growing graph structures.

Most studies have analyzed the dissemination of fake news after the spreading of the news. But, any viable mitigation system will have to work in real time and adapt to the fast evolving nature of fake news network. Therefore, keeping this in mind, we employ a adaptive and scalable technique that is efficient for large evolving graph structures. It is important that the model is able to quickly learn meaningful representations for newly seen (i.e., exposed) nodes without relying on the complete network structure. Most graph representation learning techniques, however, employ a *transductive* approach to learning node representations which optimizes the embeddings for nodes based on the entire graph structure. We employ an *inductive* approach inspired from GraphSAGE (Hamilton et al. 2017) to generate embeddings for the nodes as the information spreading network gradually evolves. It learns an aggregator function that generalizes to unseen node structures which could become potential information spreaders. The idea is to simultaneously learn the topological structure and node features from the neighborhood (Nbr) nodes, by training a set of aggregator functions instead of individual node embeddings. Using an inductive representation



(a) Message reaches $\mathcal{N}_{com}$ (b) Message reaches $\mathcal{B}_{com}$ (c) Message reaches $\mathcal{C}_{com}$

Fig. 3 Community health assessment model perspective for fake news spreading

learning model, we learn features of the exposed population (i.e., followers of the spreaders) by aggregating trust-based features from their neighborhood nodes. Fig. 3 shows how we model the proposed approach with community health assessment perspective. Nodes outside the solid oval represent $\mathcal{N}_{com}$, between solid and dotted oval represents $\mathcal{B}_{com}$ and within the dotted oval represents $\mathcal{C}_{com}$. (a) shows that false information spread has reached the two neighbor nodes (highlighted in red). Three boundary nodes (circled in red) are exposed to the information. In (b), we learn representations for the exposed boundary nodes by aggregating features of their local neighborhood structure (denoted by white nodes). Two out of the three boundary nodes that become spreaders are highlighted and the exposed core nodes are circled. Similarly, in (c), we learn representations for the exposed core nodes by aggregating their local neighborhood features. One core node becomes a spreader and the community is now vulnerable to fake news spreading.

The proposed framework is explained as follows: First, we generate a weighted information spreading network based on interpersonal trust. We then sample neighborhood with a probability proportional to the trust-based edge weights. For the sampled neighborhood, we aggregate their feature representations. Finally, we explain the loss function used to learn parameters of the model.

4.3 Generating weighted graph

Graph of the information spreading network has edge weights that quantify the likelihood of trust formation between senders and receivers. Once we compute these edge scores using techniques mentioned in Table 3, we normalize weights for all out-edges connecting the boundary node.

$$\hat{w}_{bx} = \frac{\text{bel}_{bx}}{\sum_{\forall x \in out(b)} \text{bel}_{bx}} \tag{4}$$

Similarly, we normalize weights for all in-edges connecting the boundary node.

4.4 Sampling neighborhood

Instead of sampling neighborhood as a uniform distribution, we sample a subset of neighbors proportional to the weights of the edges connecting them. Sampling is done recursively till depth K. The idea is to learn features from neighbors proportional to the level of inter-personal trust. Algorithm 1 explains the sampling strategy.

```
Algorithm 1: Sample neighborhood ($SA$)
  Input: $\mathcal{G}(\mathcal{V}, \mathcal{E})$: Information spreading network,
  $K$: Sampling depth, $\mathcal{B}_{com}$: Boundary nodes of community.
  Output: $Nbr_K(b)$: Sampled neighborhood for $b$ till depth $K$.
  $\phi \leftarrow$ Disjoint modular communities in $\mathcal{G}$;
  for each $com \in \phi$ do
      for each $b \in \mathcal{B}_{com}$ do
          $Nbr_0(b) \leftarrow \{b\}$
          for $k = 1 \ldots K$ do
              $Nbr_k(b) \leftarrow Nbr_{k-1}(b) \cup SA_k(b)_{Eq\ 4}$
          end for
      end for
  end for
```

4.5 Aggregating features

After sampling neighborhood as an unordered set, we aggregate the embeddings of sampled nodes till depth K recursively for each boundary node. The intuition is that at each depth, the boundary nodes incrementally learn trust-based features from the sampled neighborhood. Three aggregation architectures namely mean, LSTM and pooling explained in Hamilton et al. (2017) can be used. For simplicity, we only apply the mean aggregator, which takes the mean of representations h_u^{k-1} where $u \in Nbr_{k-1}(b)$. The aggregator is represented as follows:

$$h_b^k \leftarrow \sigma(W_b^k.Mean(\{h_b^{k-1}\} \cup \{h_{u(\forall u \in Nbr(b))}^{k-1}\})) \tag{5}$$

Algorithm 2 explains the aggregation strategy.

```
Algorithm 2: Aggregate features (GE)
  Input: $\mathcal{G}(\mathcal{V}, \mathcal{E})$: Information spreading network,
  $K$: Sampling depth, $\mathcal{B}_{com}$: Boundary nodes of community, $x_{v(\forall v \in \mathcal{V})}$:
   Node features.
  Output: $z_b^k$: Embedding vector for $b$.
  $\phi \leftarrow$ Disjoint communities in $\mathcal{G}$;
  for each $com \in \phi$ do
      for each $b \in \mathcal{B}_{com}$ do
          $h_b^0 \leftarrow x_b$
          for $k = 1 \dots K$ do
              $h_{Nbr(b)}^k \leftarrow GE_k(h_{u(\forall u \in Nbr(b))}^{k-1})$
              $h_b^k \leftarrow \sigma(W_b^k.Concat(h_b^{k-1}, h_{Nbr(b)}^k))_{Eq.\ 5}$
          end for
          $h_b^k \leftarrow h_b^k / ||h_b^k||_2$
      end for
      $z_b^k \leftarrow h_b^k$
  end for
```

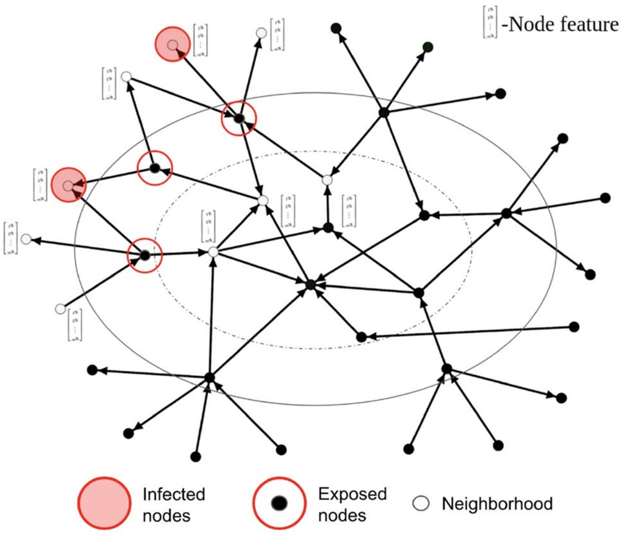


Fig. 4 Inductive representation learning model for detection of false information spreaders

4.6 Learning parameters

The weight matrices in Algorithm 2 are tuned using stochastic gradient descent on a loss function in order to learn the parameters. We train the model to minimize cross-entropy.

$$\text{Loss}(\hat{y}, y) = -\sum_{\forall b \in \mathcal{B}_{com}} \sum_{i \in \{b_{Sp}, b_{\bar{Sp}}\}} y_i \log \hat{y}_i \tag{6}$$

The loss function is modeled to predict whether the boundary node is an information spreader (b_{Sp}) or a non-spreader ($b_{\bar{Sp}}$). y represents the actual class (2-dimensional

Table 3 Trust based strategy for sampling and aggregating

Sample		Topology (top)	Activity (act)
	w_{xv}	bel$_{xv}$	RT$_{xv}$
Aggregate	*trusting others*	$ti(x)$	$\frac{\Sigma_{\forall i \in t} \begin{cases} 1 & \text{if } i = RT_x \\ 0 & \text{otherwise.} \end{cases}}{n(t)}$
	trusted by others	$tw(x)$	$\frac{\Sigma_{\forall i \in t}\, i_{n(RT_x)}}{n(t)}$

multinomial distribution of [1,0] for spreader and [0,1] for non-spreader) and $\hat{y}$ represents the predicted class.

We extend the model for $\mathcal{C}_{com}$ to identify the core node spreaders (c_{Sp}) and non-spreaders ($c_{\bar{Sp}}$). Considering boundary nodes have denser neighborhood compared to core nodes, we later analyze whether the proposed model is more sensitive to density of neighborhood structure or the aggregated features. Fig. 4 shows visual representation of our model. The implementation code is made publicly available.[1]

4.7 Modeling interpersonal trust

Interpersonal trust has been shown to be effective in the detection of rumor spreaders (Rath et al. 2017). We thus use trust measures as edge weights for our networks. We first apply a non-uniform neighborhood sampling strategy using weighted graph (where edge weights quantify the likelihood of trust formation). We then aggregate two trust features: (1) the likelihood of *trusting others* and (2) the likelihood of being *trusted by others*. We use two kinds of interpersonal-trust: topology-based (*top*) computed from the social network topology and activity-based (*act*) computed using timeline activity data collected for every node using Twitter API. We use trustingness ($ti(x)$) and trustworthiness ($tw(x)$) scores of node x obtained from the TSM algorithm as proxy for topology-based trust features and the fraction of timeline statuses of x that are retweets (RT_x) denoted by $\sum_{\forall i \in t}\{1 \text{ if } i = \mathrm{RT}_x \text{ else } 0\}/n(t)$ and average number of times x's tweets are retweeted ($n(\mathrm{RT}_x)$) denoted by $\sum_{\forall i \in t} i_{n(\mathrm{RT}_x)}/n(t)$ as activity-based trust features (t represents most recent tweets posted on x's timeline). For an edge from x to v, the topology-based edge weight is the believability score (bel_{xv}) and activity-based edge weight is the number of times x is retweeted by v (RT_{xv}). Trust-based sampling and aggregation strategy are summarized in Table 3.

5 Experiments and results

5.1 Construction of the MinFN dataset

In order to validate our model, we empirically test it out on real-world Twitter data belonging to 10 unique new events. For each news event, we collect tweets that spread some fake news about the event and also collect corresponding refuting true tweets. We rely on *altnews.in*, a popular fact checking website, to discern the validity of a tweet. From a source tweet, we extract the source tweeter and the retweeters of this tweet (proxy for spreaders). We then collect the follower-following network of the spreaders (proxy for network) and also the timeline data for all nodes in the network (to generate trust-based features) using the Twitter API.[2]. We evaluate our model on false information networks (F) and the refuting true information networks (T) separately, we also evaluated on the networks obtained by combining them ($F \cup T$). Metadata for the network dataset aggregated for all news events are summarized in Table 4.

5.2 Settings and protocols

We generated the topology-based trust measures by running the Trust Scores in Social Media (TSM) algorithm on every network to obtained *ti*, *tw* for all nodes and bel for all edges. We set the hyper-parameters using recommendations from Roy et al. (2016) (number of iterations = 100, involvement score = 0.391). We extract the disjoint modular communities of every network using Louvain community detection algorithm (Blondel et al. 2008) and identified the neighbor, boundary and core nodes for every community using the Community Health Assessment model. We then generated the activity-based trust measures from timeline data of the nodes. The embeddings are generated using the forward propagation method shown in Algorithm 2, assuming that the model parameters are learnt using Equation 6. Due to a class imbalance, we undersample the majority class to obtain balanced spreader and non-spreader class distribution. The size of hidden units is set to 128 and the learning rate is set to 0.001. We used rectified linear units as the non-linear activation function. The batch size was adjusted for optimal performance depending on the size of training dataset. Due to the heavy-tailed nature of degree distributions of edges in social networks, we downsample before modeling, which ensured that the neighborhood information is stored in dense adjacency lists. This drastically reduces our run time, which is ideal for early detection of spreaders. We also set sampling depth $K = 1$ because the network constitutes only immediate follower-following nodes of the spreaders. We compared results for the following three types of models:

5.2.1 Node feature only

Classification models that use only node features. Three baselines used are as follows:

(1) *Trusting others* Intuitively, users with high likelihood to trust others tend to be spreaders of false information. This model learns a threshold based on correlation between 'trusting others' features (both topology- and activity-based) and user ground truth.

[1] https://github.com/BhavtoshRath/Proactive_Spreader_Detection.

[2] https://developer.twitter.com/en/docs/twitter-api.

Table 4 A summary of the metadata of the MinFN dataset

	F	*T*
No. of nodes	1,709,246	1,161,607
No. of edges	3,770,532	2,086,672
No. of spreaders	2,337	671
No. of communities	63	40
No. of nodes in $\mathcal{N}$	205,975	94,707
No. of spreaders in $\mathcal{N}$	21,657	6,317
No. of nodes in $\mathcal{B}$	216,410	136,378
No. of spreaders in $\mathcal{B}$	2,159	618
No. of nodes in $\mathcal{C}$	1,492,836	1,025,229
No. of spreaders in $\mathcal{C}$	87	24

(2) *Trusted by others* Intuitively, users with high likelihood to be trusted by others tend to be spreaders of false information. Like the previous model, this model learns a threshold based on correlation between 'trusted by others' features (both topology- and activity-based) and user ground truth.

(3) *Interpolation* This model linearly combines 'trusting others' and 'trusted by others' features to find an optimal threshold.

5.2.2 Network structure only

Classification model that uses features extracted from graph structure. One baseline used is

(4) *LINE*: This model applies LINE (Tang et al. 2015) which serves as transductive learning baseline.

5.2.3 Node feature + network structure

Classification models that use features extracted using both network structure and node features. Following models, including the baseline (GCN) and proposed models are as follows:

(5) GCN_{top}: This model implements graph convolutional networks (Kipf and Welling 2017)-based transductive learning model that aggregates topology features from neighborhood.

(6) GCN_{act}: This is the graph convolutional networks-based model that aggregates activity features from neighborhood.

(7) $\text{SA}_{\text{rand}}\text{GE}_{\text{top}}$: This model applies the inductive learning by sampling neighborhood considered as uniform distribution and aggregating only topology-based features.

(8) $\text{SA}_{\text{rand}}\text{GE}_{\text{act}}$: This model applies the inductive learning by sampling neighborhood considered as uniform distribution and aggregating only activity-based features.

(9) $\text{SA}_{\text{top}}\text{GE}_{\text{top}}$: Instead of random sampling, we sample on the believability (bel) weighted network and aggregate their topology-based features.

(10) $\text{SA}_{\text{top}}\text{GE}_{\text{act}}$: Sampling approach is identical to (11) but we aggregate neighborhood's activity-based features.

(12) $\text{SA}_{\text{act}}\text{GE}_{\text{top}}$: We sample neighborhood non-uniformly on the retweet count (RT) weighted network and aggregate their topology-based features.

(13) $\text{SA}_{\text{act}}\text{GE}_{\text{act}}$: Sampling approach is identical to (14) but we aggregate neighborhood's activity-based features.

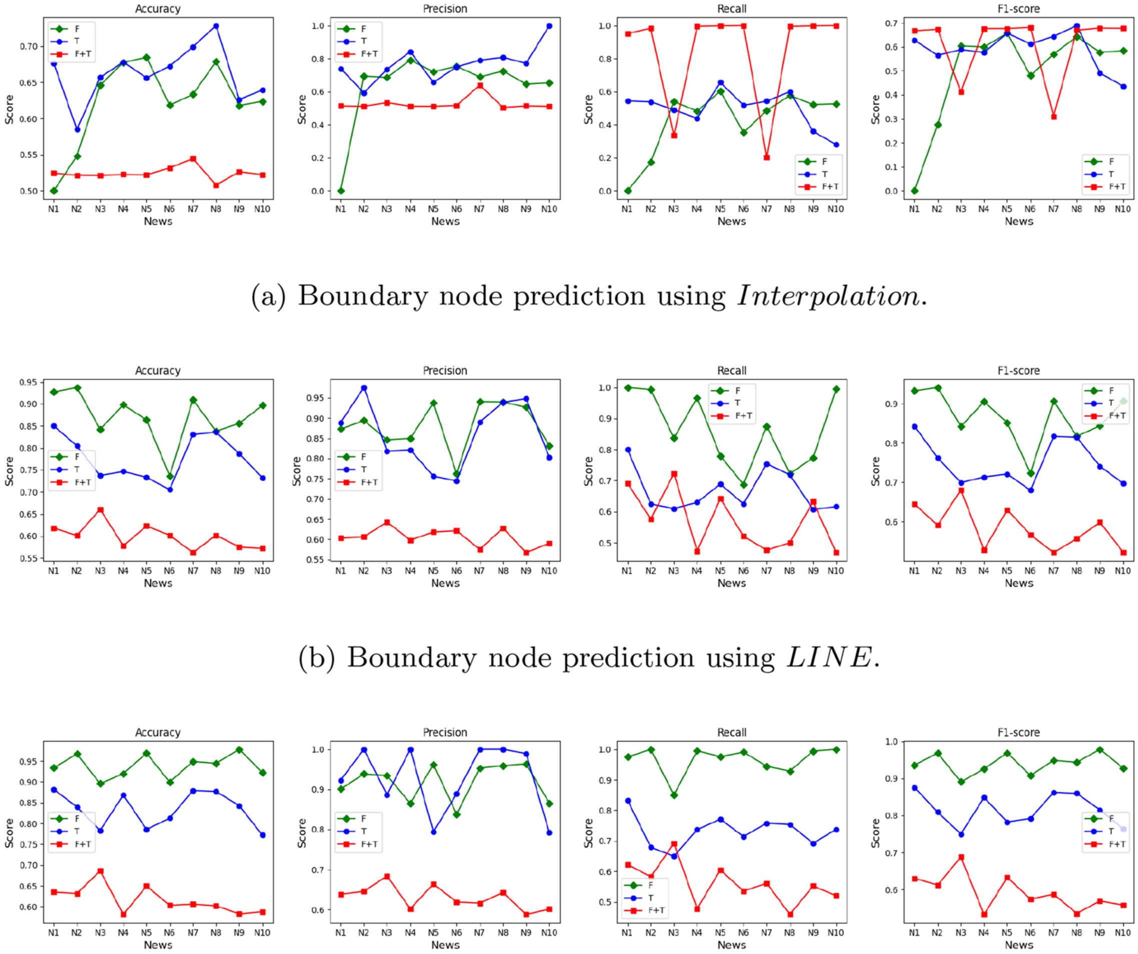
We compare our models against baseline models (1)–(3) inspired from (Rath et al. 2018) that considers features based on trust. Baseline model (4) considers features based on network structure only (Tang et al. 2015). Proposed models (5)–(13) integrate both neighborhood structure and node features. We analyze the best combination of sampling and aggregating strategy that predicts spreader node with highest accuracy. For evaluation, we did a 80-10-10 train-validation-test split of the dataset. We used fivefold cross-validation and four common metrics: Accuracy, Precision, Recall and F1 score. Accuracy is defined over the two classes as follows: Accu. $= \frac{\#\text{of correctly predicted users}}{\text{Total\#of users}}$, Precision is defined as Prec. $= \frac{\text{TP}}{\text{TP}+\text{FP}}$, the Recall is defined as Rec. $= \frac{\text{TP}}{\text{TP}+\text{FN}}$, and $F1$ is defined as $F1 = \frac{2*\text{Prec.}*\text{Rec.}}{\text{Prec.}+\text{Rec.}}$, where TP, FP and *FN* are true positive rate, false positive rate and false negative rate, respectively. We only show results for the spreader class.

5.3 Results and analysis

We evaluated our proposed model on 10 debunked news events. For each news event, we obtained three types of networks: network for the false information (F), for the true information (T) refuting it and the network obtained by combining them ($F \cup T$). Thus, we ran our models on 30 large-scale networks.

5.3.1 Boundary node analysis (less dense *Nbr*)

Table 5 summarizes results for the boundary node prediction aggregated for all news. The results show that F performs better than T on almost every metrics, while $F \cup T$ performs poorly. The poor performance of $F \cup T$ networks could be attributed to the fact that there is minimal overlap of nodes in F and T networks (12%) which causes the $F \cup T$ networks to have sparser communities. Also, false and true information spreaders are together considered as spreader class which could be affecting the model performance. While comparing the baseline models, *Trusted by others* model performs better than the *Trusting others* model with an improvement in accuracy of 4.8%, 5% and 1.5% for F, T and $F \cup T$ networks, respectively. *Interpolation* model shows a further improvement of 2.3%, 2.3% and 1.1% for F, T and $F \cup T$ networks, respectively, over *trustingness* model. *LINE* and *GCN* baselines show significant improvement on all metrics for F networks compared to T or $F \cup T$ networks. We see further substantial



(a) Boundary node prediction using *Interpolation*.

(b) Boundary node prediction using *LINE*.

(c) Boundary node prediction using $SA_{top}GE_{top}$.

Fig. 5 Metric performance for boundary node prediction for news events (N1–N10) for the best performing **a** Node feature only, **b** Network structure only and **c** Node feature + network structure models

increase in performance for each type of network using inductive learning models. Comparing the two random sampler models (i.e., $\mathrm{SA_{rand}GE_{top}}$, $\mathrm{SA_{rand}GE_{act}}$), we see that topology-based features of the neighborhood perform better than activity-based features. Similar trend is observed for topology-based sampler models (i.e., $\mathrm{SA_{top}GE_{top}}$, $\mathrm{SA_{top}GE_{act}}$) where model using topology-based aggregator performs better than activity-based aggregator. Same is the case for activity-based sampler models (i.e., $\mathrm{SA_{act}GE_{top}}$, $\mathrm{SA_{act}GE_{act}}$). Integrating *top* and *act* does not show any significant improvement over *top* only models. Thus, we can conclude that interpersonal trust-based modeling in the inductive learning framework is able to predict false information spreaders better than true information spreaders. We also observe that topology-based sampling and aggregating strategies perform better than activity-based strategies. The low performance of activity-based strategies could be attributed to the fact that many Twitter users are either inactive users or users with strict privacy settings whose timeline data could not be retrieved. Also, recent 10 activities on a user's timeline might be insufficient data to capture activity-based trust dynamics. For each type of network, we observe that $\mathrm{SA_{top}GE_{top}}$ model performs the best, with F having accuracy of 93.3%, which is higher than 12.3% and 52.1% over T and $F \cup T$ networks, respectively. Figure 5 shows the performance metrics of this model for the 10 news events (N1–N10) for the best performing a) Node feature only (*Interpolation*), b) Network structure only (*LINE*) and c) Node feature + network structure ($\mathrm{SA_{top}GE_{top}}$) models. On comparing F1 metric performance, we observe clearer distinction (with F networks performing better than *T*, which in turn is better than $F \cup T$) in performance for $\mathrm{SA_{top}GE_{top}}$ compared to *Interpolation* and *LINE*. *Interpolation* model has the least distinction in performance which can be attributed to the fact that it does not capture the features based on the network structure. Thus, we

Table 5 Results comparison of our different models against existing baselines for boundary node spreader prediction

	F				*T*				**F ∪ T**			
	Accu.	Prec.	Rec.	F1	Accu.	Prec.	Rec.	F1	Accu.	Prec.	Rec.	F1
Trusting others	0.58	0.612	0.329	0.396	0.615	0.697	0.450	0.519	0.510	0.522	0.888	0.603
Trusted by others	0.608	0.631	0.384	0.455	0.646	0.713	0.500	0.585	0.518	0.513	0.916	0.638
Interpolation	0.622	0.635	0.426	0.498	0.661	0.768	0.496	0.588	0.524	0.526	0.846	0.611
LINE	0.709	0.784	0.593	0.669	0.692	0.763	0.567	0.647	0.589	0.602	0.517	0.554
$\text{SA}_{\text{rand}}\text{GE}_{\text{top}}$	0.870	0.879	0.862	0.866	0.776	0.858	0.667	0.748	0.599	0.605	0.570	0.583
$\text{SA}_{\text{rand}}\text{GE}_{\text{act}}$	0.777	0.845	0.689	0.754	0.728	0.814	0.612	0.688	0.566	0.572	0.539	0.547
GCN_{top}	0.839	0.887	0.784	0.832	0.775	0.921	0.595	0.723	0.592	0.649	0.646	0.647
GCN_{act}	0.807	0.849	0.750	0.796	0.740	0.835	0.591	0.693	0.576	0.640	0.612	0.626
$\text{SA}_{\text{top}}\text{GE}_{\text{top}}$	**0.937**	**0.918**	**0.965**	**0.939**	**0.834**	**0.927**	**0.732**	**0.815**	**0.616**	**0.630**	**0.561**	**0.592**
$\text{SA}_{\text{top}}\text{GE}_{\text{act}}$	0.912	0.899	0.935	0.915	0.800	0.884	0.699	0.777	0.584	0.601	0.504	0.545
$\text{SA}_{\text{act}}\text{GE}_{\text{top}}$	0.838	0.854	0.816	0.833	0.763	0.817	0.686	0.743	0.582	0.589	0.542	0.559
$\text{SA}_{\text{act}}\text{GE}_{\text{act}}$	0.804	0.853	0.737	0.786	0.735	0.800	0.634	0.706	0.561	0.570	0.542	0.539

Trusting others, *Trusted by others* and *Interpolation* are baseline models from (Rath et al. 2018). LINE is a transductive baseline from (Tang et al. 2015), and GCN_{top}, GCN_{act} serve as vanilla baselines of GCN models

Bold values represents the highest value in the metric column

can conclude that underlying network structure around false information is very different compared to the network structure around true information. An interesting observation is the high precision values for *T*. This is because the percentage of predicted spreaders which are non-spreaders tends to be lower for *T* network than for *F* network.

5.3.2 Core node analysis (more dense *Nbr*)

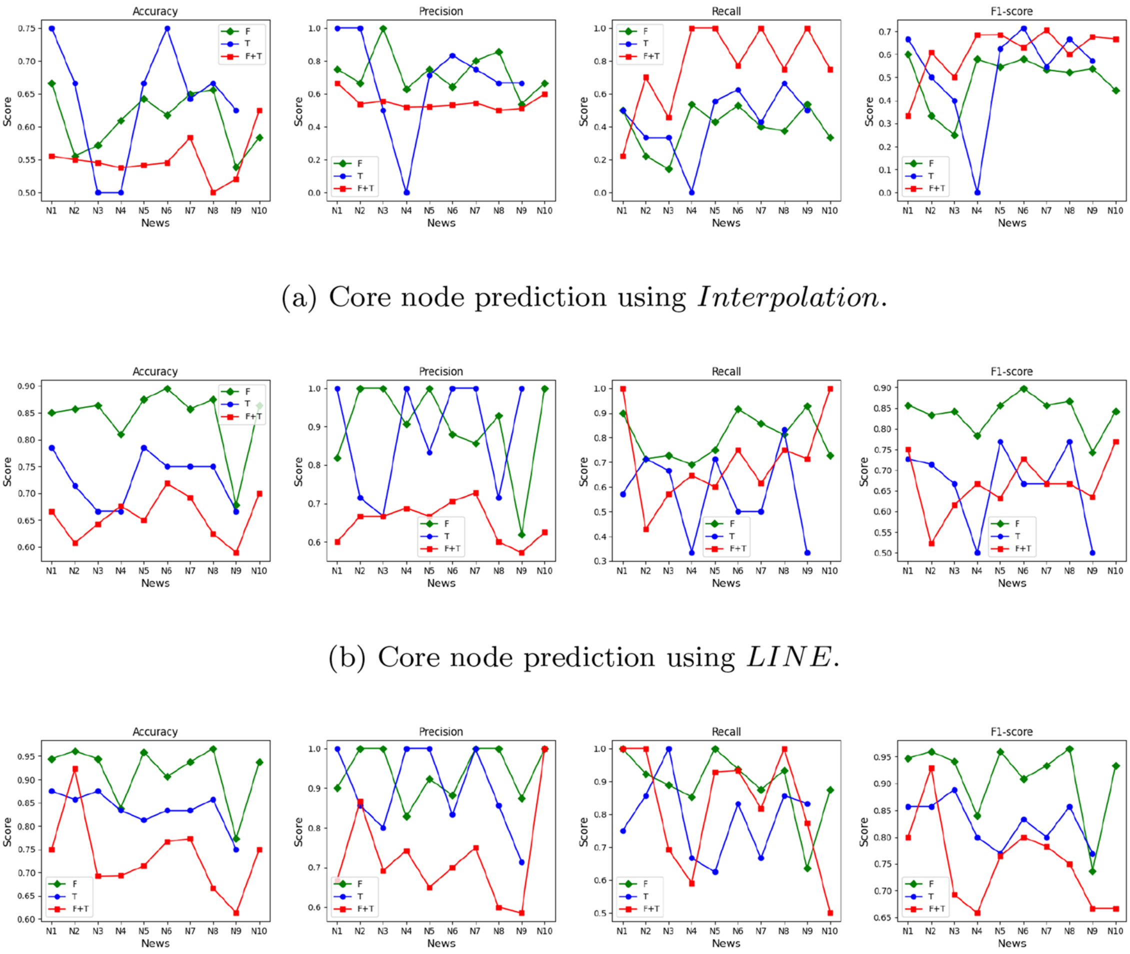
Table 6 summarizes results of the model for predicting core nodes aggregated for all news. The overall performance trend is identical to the results shown for boundary nodes in Table 5. Among the baseline models, *Interpolation* model performs better than *Trusted by others* and *Trusting others* models. *LINE*- and *GCN*-based models show significant improvement over trust feature baselines on all metrics. Among inductive learning models, topology-based trust modeling shows better performance than activity-based trust modeling. Also, F networks perform better than *T* networks, which in turn perform better than $F \cup T$ networks. Among random sampler models, $\text{SA}_{\text{rand}}\text{GE}_{\text{top}}$ has the highest accuracy of 84.2%, 72.6% and 65.6% for *F*, *T* and $F \cup T$ networks, respectively. Among topology-based sampler models, $\text{SA}_{\text{top}}\text{GE}_{\text{top}}$ performs better over $\text{SA}_{\text{top}}\text{GE}_{\text{act}}$ with an increase in accuracy of 2.8%, 4.5% and 7.1% for *F*, *T* and $F \cup T$ networks, respectively. Activity-based sampler models also show identical trend with $\text{SA}_{\text{act}}\text{GE}_{\text{top}}$ performing better than $\text{SA}_{\text{act}}\text{GE}_{\text{act}}$ with an increase in accuracy of 2.6%, 9% and 4.6% for *F*, *T* and $F \cup T$ networks, respectively. Among all models, $\text{SA}_{\text{top}}\text{GE}_{\text{top}}$ shows the best overall performance. Figure 6 shows the performance metrics of this model for the 10 news events (N1–N10) for the best performing a) Node feature only, b) Network structure only and c) Node feature + network structure models. As in Fig. 5, True information network for N10 is excluded from analysis as it did not have sufficient spreaders to train our model on. A clear observation is that the metric performance for the three types of networks is not as distinct as in Fig. 5. We notice that though the number of core nodes is much higher than boundary nodes, the number of core spreaders is much smaller than boundary node spreaders. Thus, the model fails to learn meaningful representations for core nodes due to smaller training dataset.

Summary: Sophisticated models that include both node features and network structure outperform simpler node feature only and network structure only models. Comparing the prediction performance of core and boundary spreaders, we can conclude that our model's performance is more sensitive to training dataset size compared to density of neighborhood.

6 Additional experimental analysis

6.1 Bot detection

Bot accounts interact with humans and influence interactions by spreading information rapidly. One of the most well-known instances of bots having a large effect on human behavior was that of the 2016 U.S elections (Bessi and Ferrara 2016). In order to obtain more representative networks, we filter out bots to include only humans to better quantify interpersonal and individual trust. We use a bot detection model proposed by Kudugunta and Ferrara (2018) where they used an AdaBoost classifier trained on an exhaustive set



(a) Core node prediction using $Interpolation$.

(b) Core node prediction using $LINE$.

(c) Core node prediction using $SA_{top}GE_{top}$.

Fig. 6 Metric performance for core node prediction for news events (N1–N10) for the best performing **a** Node feature only, **b** Network structure only and **c** Nade feature + network structure models

of user-based features to achieve an accuracy of 99.81%. The MinFN dataset (Rath 2021) consisted of the following metadata for each user: *Id, Screen Name, Name, Statuses Count, Favorites Count, Followers Count, Friends Count, Listed Count, Verified, Protected, Created At, Location* which was fewer than the features trained by Kudugunta et al. So we first tested the performance of the model with our limited set of features on a publicly available bot detection dataset (Cresci et al. 2017). The classifier achieved an accuracy of 98% thus proving that using our set of user features could identify bots with an almost identical accuracy.

6.2 Effects of bots on performance

Table 7 shows the analysis of bots in our dataset. We found that the bots are just as prevalent in fake news networks as the refuting true news network, further emphasizing the role of humans in fake news spread. We notice that around 5% of nodes in each network were classified as bots. Among spreaders (Table 8), we observe that higher number of false information spreaders tend to be bots compared to true information spreaders (except for N1 and N9). We then analyzed performance of our model after filtering out bots from the network as trust-based features are more representative of actual people's behavior. In Table 9, we can observe the increase in performance when we train and test our models on networks without bots as compared to networks with a mixture of bots and genuine users. We observe that when training on 100 historical tweets on networks without bots, the performance increases by 2.8% for $\mathrm{SA_{rand}GE_{act}}$, 1% for $\mathrm{SA_{top}GE_{act}}$, 4.6% for $\mathrm{SA_{act}GE_{top}}$ and 3.6% for $\mathrm{SA_{act}GE_{act}}$. The performance gains are even higher for T and $F \cup T$

Table 6 Results comparison of different models and baselines for core node spreader prediction

	F				T				$\mathbf{F \cup T}$			
	Accu.	Prec.	Rec.	F1	Accu.	Prec.	Rec.	F1	Accu.	Prec.	Rec.	F1
Trusting others	0.553	0.643	0.298	0.388	0.569	0.585	0.338	0.414	0.521	0.511	0.95	0.659
Trusted by others	0.569	0.628	0.411	0.481	0.614	0.694	0.503	0.508	0.540	0.523	0.952	0.673
Interpolation	0.609	0.730	0.400	0.492	0.640	0.681	0.438	0.521	0.550	0.548	0.764	0.608
LINE	0.721	0.821	0.625	0.681	0.672	0.870	0.467	0.579	0.577	0.572	0.676	0.602
$\mathrm{SA}_{\mathrm{rand}}\mathrm{GE}_{\mathrm{top}}$	0.842	0.900	0.802	0.838	0.726	0.880	0.574	0.664	0.656	0.651	0.707	0.665
$\mathrm{SA}_{\mathrm{rand}}\mathrm{GE}_{\mathrm{act}}$	0.798	0.893	0.700	0.764	0.658	0.742	0.448	0.523	0.597	0.631	0.512	0.548
GCN_{top}	0.755	0.972	0.524	0.681	0.739	0.698	0.839	0.762	0.683	0.731	0.537	0.619
GCN_{act}	0.731	0.741	0.705	0.722	0.701	0.735	0.641	0.684	0.657	0.691	0.561	0.619
$\mathrm{SA}_{\mathrm{top}}\mathrm{GE}_{\mathrm{top}}$	**0.916**	**0.940**	**0.892**	**0.912**	**0.836**	**0.895**	**0.787**	**0.825**	**0.734**	**0.725**	**0.823**	**0.750**
$\mathrm{SA}_{\mathrm{top}}\mathrm{GE}_{\mathrm{act}}$	0.891	0.929	0.849	0.884	0.800	0.931	0.684	0.769	0.685	0.703	0.677	0.682
$\mathrm{SA}_{\mathrm{act}}\mathrm{GE}_{\mathrm{top}}$	0.868	0.941	0.788	0.854	0.771	0.962	0.598	0.712	0.648	0.688	0.651	0.641
$\mathrm{SA}_{\mathrm{act}}\mathrm{GE}_{\mathrm{act}}$	0.846	0.847	0.858	0.846	0.707	0.827	0.581	0.661	0.619	0.694	0.522	0.567

Trusting others, *Trusted by others* and *Interpolation* are baseline models from Rath et al. (2018). LINE is a transductive baseline from Tang et al. (2015), and GCN_{top}, GCN_{act} serve as vanilla baselines of GCN model

Bold values represents the highest value in the metric column

network. In Table 10, we can observe similar trends with performance increase of 4.7% for $\mathrm{SA}_{\mathrm{rand}}\mathrm{GE}_{\mathrm{act}}$, 0.5% for $\mathrm{SA}_{\mathrm{top}}\mathrm{GE}_{\mathrm{act}}$, 1.5% for $\mathrm{SA}_{\mathrm{act}}\mathrm{GE}_{\mathrm{top}}$ and 0.4% for $\mathrm{SA}_{\mathrm{act}}\mathrm{GE}_{\mathrm{act}}$. Similar trend is observed for T and $F \cup T$ network. The performance gains from the bot filtration process are visualized in Fig. 7. This leads us to conclude that the filtration of bots is an essential step to better predict likely spreaders using trust-based features. Thus, including a pre-processing phase of bot filtration helps further increase the performance of the inductive learning model.

6.3 Effects of timeline data volume on performance

A major bottleneck for *act*-based models analyzed in Sect. 5.3 was having limited timeline data (recent 10 activities on a user's timeline). We further extended our dataset by collecting 100 recent timeline tweets in order to test whether increasing timeline data to capture more representative activity-based trust features increased performance of *act*-based models, and whether sampling ($\mathrm{SA}_{\mathrm{act}}$) or aggregating ($\mathrm{GE}_{\mathrm{act}}$) strategy showed better improvements. We report the F1 score performance of *act*-based inductive representation learning models ($\mathrm{SA}_{\mathrm{rand}}\mathrm{GE}_{\mathrm{act}}$, $\mathrm{SA}_{\mathrm{top}}\mathrm{GE}_{\mathrm{act}}$, $\mathrm{SA}_{\mathrm{act}}\mathrm{GE}_{\mathrm{top}}$ and $\mathrm{SA}_{\mathrm{act}}\mathrm{GE}_{\mathrm{act}}$) for 10- and 100-most recent timeline tweets.

Table 9 shows results for the boundary node prediction problem. We observe that for network with bots, the performance increases by 6.2% for $\mathrm{SA}_{\mathrm{rand}}\mathrm{GE}_{\mathrm{act}}$, 4.3% for $\mathrm{SA}_{\mathrm{top}}\mathrm{GE}_{\mathrm{act}}$, 6.9% for $\mathrm{SA}_{\mathrm{act}}\mathrm{GE}_{\mathrm{top}}$ and 5.2% for $\mathrm{SA}_{\mathrm{act}}\mathrm{GE}_{\mathrm{act}}$ when 100 timeline tweets are used to quantify trust features

Table 7 Analysis of information spreaders that are bots

	No. of Nodes ($F+T$)	No. of bots ($F+T$)	% ($F+T$)	No. of Nodes (F)	No. of bots (F)	% (F)	No. of Nodes (T)	No. of bots (T)	% (T)
N1	2,677,924	123,934	4.63	90,199	1,797,059	5.02	49,697	1,164,162	4.27
N2	1,230,559	49,654	4.04	35,458	885,598	4.00	19,666	453,537	4.34
N3	2,198,524	114,168	5.19	60,098	1,228,479	4.89	66,679	1,169,681	5.70
N4	2,900,925	152,346	5.25	135,771	2,607,629	5.21	27,343	433,616	6.31
N5	3,019,066	180,577	5.98	143,824	2,150,820	6.69	58,483	1,168,820	5.00
N6	2,420,000	99,824	4.12	96,315	2,387,610	4.03	32,570	1,297,371	2.51
N7	1,606,924	89,669	5.58	37,626	627,147	6.00	65,082	1,166,528	5.58
N8	2,663,392	114,324	4.29	78,471	2,036,162	3.85	52,539	1,058,482	4.96
N9	4,030,000	229,246	5.69	75,418	1,197,935	6.30	167,828	2,999,865	5.59
N10	2,729,312	111,462	4.08	81,237	2,174,023	3.74	39,979	704,006	5.68

Number of users in F + number of users in $T \neq$ number of users in $F \cup T$ because there are users that are common between F and T

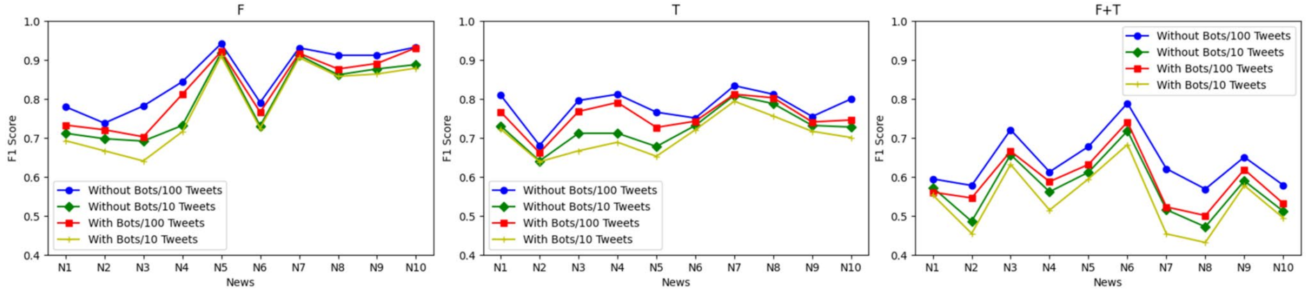


Fig. 7 F1 scores sensitivity analysis for $\text{SA}_{\text{act}}\text{GE}_{\text{act}}$ due to timeline data volume and bot filtration

Table 8 Analysis of bots among spreaders of fake news and true news

	F		*T*	
	Spreaders	Bots	Spreaders	Bots
N1	2721	1	454	2
N2	968	2	436	1
N3	1402	6	528	4
N4	4764	101	485	0
N5	3410	0	314	0
N6	3598	9	496	0
N7	716	2	867	0
N8	955	2	505	0
N9	2521	0	1977	2
N10	2329	3	747	1

instead of 10 timeline tweets. For network without bots, the performance increases by 5.5% for $\text{SA}_{\text{rand}}\text{GE}_{\text{act}}$, 1.9% for $\text{SA}_{\text{top}}\text{GE}_{\text{act}}$, 9% for $\text{SA}_{\text{act}}\text{GE}_{\text{top}}$ and 6.8% for $\text{SA}_{\text{act}}\text{GE}_{\text{act}}$. Similar trend is observed for T and $F \cup T$ networks as well. Table 10 shows results for the core node prediction problem. For network with bots, the performance increases by 3.8% for $\text{SA}_{\text{rand}}\text{GE}_{\text{act}}$, 3.8% for $\text{SA}_{\text{top}}\text{GE}_{\text{act}}$, 7.5% for $\text{SA}_{\text{act}}\text{GE}_{\text{top}}$ and 4.9% for $\text{SA}_{\text{act}}\text{GE}_{\text{act}}$ when using 100 timeline tweets instead of 10. For network without bots, the performance increases by 5.7% for $\text{SA}_{\text{rand}}\text{GE}_{\text{act}}$, 3% for $\text{SA}_{\text{top}}\text{GE}_{\text{act}}$, 6.2% for $\text{SA}_{\text{act}}\text{GE}_{\text{top}}$ and 4.8% for $\text{SA}_{\text{act}}\text{GE}_{\text{act}}$. We observe identical trend for T and $F \cup T$ networks.

An interesting observation is that the performance of $\text{SA}_{\text{top}}\text{GE}_{\text{act}}$ is better than $\text{SA}_{\text{act}}\text{GE}_{\text{act}}$, i.e., sampling strategy based on topology features is better than activity features

Table 9 F1 score comparison of inductive learning models with/without bots and for 10/100 timeline tweets (t) for *boundary* node prediction

	F				*T*				**F ∪ T**			
	With bots		Without bots		With bots		Without bots		With bots		Without bots	
	10 t	100 t	10 t	100 t	10 t	100 t	10 t	100 t	10 t	100 t	10 t	100 t
$\text{SA}_{\text{rand}}\text{GE}_{\text{act}}$	0.754	0.801	0.781	0.824	0.688	0.734	0.713	0.759	0.547	0.588	0.566	0.621
$\text{SA}_{\text{top}}\text{GE}_{\text{act}}$	**0.915**	**0.955**	**0.947**	**0.965**	**0.777**	**0.827**	**0.787**	**0.861**	**0.545**	**0.588**	**0.607**	**0.631**
$\text{SA}_{\text{act}}\text{GE}_{\text{top}}$	0.833	0.891	0.855	0.932	0.743	0.768	0.756	0.837	0.559	0.616	0.577	0.643
$\text{SA}_{\text{act}}\text{GE}_{\text{act}}$	0.786	0.827	0.802	0.857	0.706	0.756	0.726	0.782	0.539	0.591	0.569	0.639

Bold values represents the highest value in the metric column

Table 10 F1 score comparison of inductive learning models with/without bots and for 10/100 timeline tweets (t) for *core* node prediction

	F				*T*				**F ∪ T**			
	With bots		Without bots		With bots		Without bots		With bots		Without bots	
	10 t	100 t	10 t	100 t	10 t	100 t	10 t	100 t	10 t	100 t	10 t	100 t
$\text{SA}_{\text{rand}}\text{GE}_{\text{act}}$	0.764	0.793	0.786	0.831	0.523	0.678	0.567	0.691	0.548	0.59	0.566	0.673
$\text{SA}_{\text{top}}\text{GE}_{\text{act}}$	**0.884**	**0.918**	**0.896**	**0.923**	**0.769**	**0.838**	**0.788**	**0.874**	**0.682**	**0.761**	**0.692**	**0.799**
$\text{SA}_{\text{act}}\text{GE}_{\text{top}}$	0.854	0.918	0.877	0.932	0.712	0.766	0.738	0.809	0.641	0.689	0.677	0.703
$\text{SA}_{\text{act}}\text{GE}_{\text{act}}$	0.846	0.888	0.851	0.892	0.661	0.701	0.677	0.713	0.567	0.617	0.598	0.645

Bold values represents the highest value in the metric column which suggests that during sampling phase (i.e., choosing neighbors whose features are aggregated), network topology is more crucial than activity-based features being aggregated. This suggests the efficacy of believability-based weights assigned to edges is a better measure of interpersonal trust compared to simple retweet-based weights. Another interesting observation is that the best performing model in Table 9 and 10 (i.e., $\mathrm{SA}_{\mathrm{top}}\mathrm{GE}_{\mathrm{act}}$) outperforms the best performing model in Tables 5 and 6 (i.e., $\mathrm{SA}_{\mathrm{top}}\mathrm{GE}_{\mathrm{top}}$) on the same network (network with bots). This can be attributed to the fact that using aggregation strategy-based on activity features from 100 timeline tweets generates more representative features of trust compared to 10 timeline tweets, and also they outperform topology features. *We thus conclude that* $\mathrm{SA}_{\mathrm{top}}\mathrm{GE}_{\mathrm{act}}$ *outperforms* $\mathrm{SA}_{\mathrm{top}}\mathrm{GE}_{\mathrm{top}}$ *when sufficiently large number of timeline tweets are used to quantify act-based trust features.*

Figure 7 compares the sensitivity of our model on the presence of bots in the network and the volume of timeline data used to aggregate trust-based features for all news events for $\mathrm{SA}_{\mathrm{act}}\mathrm{GE}_{\mathrm{act}}$ model specifically. We conclude that inductive learning model performs better in the absence of bots and when we have larger volume of timeline data to extract features from.

7 Conclusions and future work

In this paper, we proposed a framework that uses inductive representation learning and community health assessment model to identify fake news spreaders. We also make public a massive dataset comprised of real-world Twitter data from 10 unique news events and use this dataset to empirically validate our framework.

Using interpersonal trust-based properties, we could identify spreaders with high accuracy and also showed that the proposed model identifies false information spreaders more accurately than true information spreaders. We analyzed our models on networks comprised of only humans and a mixture of humans and bots. We found that a bot-filtration step is quintessential to ensure a representative network and found significant performance increases in the absence of bots.

The key hypothesis we tested is that interpersonal trust plays a significantly more important role in identifying false information spreaders than true information spreaders. The intuition behind this being that true information is usually easy to accept, and blatantly false information is easy to reject; however, most false information is *false, yet plausibly true*, making it harder for people to accept/reject it on their own, and thus depending on the source they received it from. Identified false information spreaders can then be quarantined and true news spreaders can be promoted, thus serving as an effective mitigation strategy.

Using experimental analysis on real-world Twitter data, we showed that topology-based features and sampling strategies help in spreader detection more than activity-based features and sampling. And although topology-based features are more important, we did find that having more representative activity features (using larger volume of timeline data) increases the performance. The proposed framework can be used to identify people who are likely to become spreaders in real time on large networks due to our usage of inductive representation learning which can adapt to fast evolving spread networks. In the future, we want to include other proxies of trust such as how long a user has been active on the social media platform, whether the user has a history of refuting false information, etc. In this paper, we used only the immediate follower-following network of the information spreaders. In the future, we would want to extend this to greater sampling depths and study its effect on our model's performance.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s13278-022-00890-z.

Acknowledgements The authors would like to acknowledge the contribution made by Mohana Krishna Vutukuru (vutuk004@umn.edu) in testing the bot detection model on the dataset from Cresci et al. (2017).

References

Almaatouq A, Shmueli E, Nouh M, Alabdulkareem A, Singh VK, Alsaleh M, Alarifi A, Alfaris A, Pentland A (2016) If it looks like a spammer and behaves like a spammer, it must be a spammer: analysis and detection of microblogging spam accounts. Int J Inf Sec 15(5):475–491

Arapakis I, Barreda-Angeles M, Pereda-Baños A (2017) Interest as a proxy of engagement in news reading: spectral and entropy analyses of eeg activity patterns. IEEE Trans Affective Comput 10(1):100–114

Chu Z, Gianvecchio S, Koehl A, Wang H, Jajodia S (2013) Blog or block: detecting blog bots through behavioral biometrics. Comput Netw 57(3):634–646

Ferrara E, Varol O, Davis C, Menczer F, Flammini A (2016) The rise of social bots. Commun ACM 59(7):96–104

Guo B, Ding Y, Yao L, Liang Y, Yu Z (2020) The future of false information detection on social media: new perspectives and trends. ACM Comput Surv (CSUR) 53(4):1–36

Kudugunta S, Ferrara E (2018) Deep neural networks for bot detection. Inf Sci 467:312–322

Pennycook G, Rand DG (2020) Who falls for fake news? the roles of bullshit receptivity, overclaiming, familiarity, and analytic thinking. J Person 88(2):185–200

Scarselli F, Gori M, Tsoi AC, Hagenbuchner M, Monfardini G (2008) The graph neural network model. IEEE Trans Neural Netw 20(1):61–80

Zubiaga A, Liakata M, Procter R, Wong Sak Hoi G, Tolmie P (2016) Analysing how people orient to and spread rumours in social media by looking at conversational threads. PloS one 11(3):e0150989

Amleshwaram AA, Reddy AN, Yadav S, Gu G, Yang C (2013) Cats: characterizing automation of twitter spammers. In: COMSNETS. Citeseer, pp 1–10

Bessi A, Ferrara E (2016) Social bots distort the 2016 us presidential election online discussion. First Monday 21(11-7)

Blondel VD, Guillaume J-L, Lambiotte R, Lefebvre E (2008) Fast unfolding of communities in large networks. J Stat Mech Theory Exp

Cardaioli M, Cecconello S, Conti M, Pajola L, Turrin F (2020) Fake news spreaders profiling through behavioural analysis. In: CLEF (working notes)

Carlini N, Mishra P, Vaidya T, Zhang Y, Sherr M, Shields C, Wagner D, Zhou W (2016) Hidden voice commands. In: 25th {USENIX} security symposium ({USENIX} security 16, pp. 513–530

Castillo C, Mendoza M, Poblete B (2011) Information credibility on twitter. In: Proceedings of the 20th international conference on World wide web, pp 675–684

Chen T, Li X, Yin H, Zhang J (2018) Call attention to rumors: deep attention based recurrent neural networks for early rumor detection. In: PAKDD

Cresci S, Di&nbsp;Pietro R, Petrocchi M, Spognardi A, Tesconi M (2017) The paradigm-shift of social spambots: Evidence, theories, and tools for the arms race. In: Proceedings of the 26th international conference on world wide web companion, pp 963–972

Davis CA, Varol O, Ferrara E, Flammini A, Menczer F (2016) Botornot: a system to evaluate social bots. In: Proceedings of the 25th international conference companion on world wide web, pp 273–274

Defferrard M, Bresson X, Vandergheynst P (2016) Convolutional neural networks on graphs with fast localized spectral filtering. In: NeurIPS

Della&nbsp;Vedova ML, Tacchini E, Moret S, Ballarin G, DiPierro M, de&nbsp;Alfaro L (2018) Automatic online fake news detection combining content and social signals. In: 2018 22nd conference of open innovations association (FRUCT). IEEE, pp 272–279

Duvenaud DK, Maclaurin D, Iparraguirre J, Bombarell R, Hirzel T, Aspuru-Guzik A, Adams RP (2015) Convolutional networks on graphs for learning molecular fingerprints. In: Advances in neural information processing systems

Giachanou A, Ríssola EA, Ghanem B, Crestani F, Rosso P (2020) The role of personality and linguistic patterns in discriminating between fake news spreaders and fact checkers. In: International conference on applications of natural language to information systems. Springer, pp 181–192

Guess A, Nagler J, Tucker J (2019) Less than you think: prevalence and predictors of fake news dissemination on facebook. Sci Adv 5(1):eaau4586

Hamilton W, Ying Z, Leskovec J (2017) Inductive representation learning on large graphs. In: Advances in neural information processing systems

Hu G, Ding Y, Qi S, Wang X, Liao Q (2019) Multi-depth graph convolutional networks for fake news detection. In: CCF international conference on natural language processing and Chinese computing

Hu X, Tang J, Gao H, Liu H (2014) Social spammer detection with sentiment information. In: 2014 IEEE international conference on data mining. IEEE, pp 180–189

Ito J, Song J, Toda H, Koike Y, Oyama S (2015) Assessment of tweet credibility with lda features. In: Proceedings of the 24th international conference on world wide web, pp 953–958

Jin Z, Cao J, Jiang Y-G, Zhang Y (2014) News credibility evaluation on microblog with a hierarchical propagation model. In: 2014 IEEE international conference on data mining. IEEE, pp 230–239

Kamvar SD, Schlosser MT, Garcia-Molina H (2003) The eigentrust algorithm for reputation management in p2p networks. In: Proceedings of the 12th international conference on World Wide Web, 2003, pp 640–651

Karami M, Nazer TH, Liu H (2021) Profiling fake news spreaders on social media through psychological and motivational factors. In: Proceedings of the 32nd ACM conference on hypertext and social media, pp 225–230

Kipf TN, Welling M (2017) Semi-supervised classification with graph convolutional networks. In: 5th international conference on learning representations

Kochkina E, Liakata M, Zubiaga A (2018) All-in-one: Multi-task learning for rumour verification. arXiv preprint arXiv:1806.03713

Li L, Cai G, Chen N (2018) A rumor events detection method based on deep bidirectional gru neural network. In: 2018 IEEE 3rd international conference on image, vision and computing (ICIVC). IEEE, pp 755–759

Liu Y, Wu Y-FB (2018) Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks. In: Thirty-second AAAI conference on artificial intelligence

Long Y (2017) Fake news detection through multi-perspective speaker profiles. Association for Computational Linguistics

Ma J, Gao W, Mitra P, Kwon S, Jansen BJ, Wong K-F, Cha M (2016) Detecting rumors from microblogs with recurrent neural networks

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

Mishra A, Bhattacharya A (2011) Finding the bias and prestige of nodes in networks based on trust scores. In: Proceedings of the 20th international conference on World wide web, pp 567–576

Monti F, Boscaini D, Masci J, Rodola E, Svoboda J, Bronstein MM (2017) Geometric deep learning on graphs and manifolds using mixture model cnns. In: CVPR

Mui L (2002) Computational models of trust and reputation: agents, evolutionary games, and social networks. Ph.D. dissertation, Massachusetts Institute of Technology

Newman N, Fletcher R, Andi S, Nielsen RK (2020) The Reuters Institute digital news report 2020. Reuters Institute for the Study of Journalism

Pérez-Rosas V, Kleinberg B, Lefevre A, Mihalcea R (2017) Automatic detection of fake news. arXiv preprint arXiv:1708.07104

Pizarro J (2020) Using n-grams to detect fake news spreaders on twitter. In: CLEF (working notes)

Potthast M, Kiesel J, Reinartz K, Bevendorff J, Stein B (2017) A stylometric inquiry into hyperpartisan and fake news.'arXiv preprint arXiv:1702.05638

Rangel F, Giachanou A, Ghanem BHH, Rosso P (2020) Overview of the 8th author profiling task at pan 2020: profiling fake news spreaders on twitter. In: CEUR workshop proceedings, vol 2696. Sun SITE Central Europe, pp 1–18

Rath B (2021) False and refutation information network and historical behavioral data. https://doi.org/10.7910/DVN/GHAMOE

Rath B, Gao W, Ma J, Srivastava J (2017) From retweet to believability: Utilizing trust to identify rumor spreaders on twitter. In: Proceedings of the 2017 IEEE/ACM international conference on advances in social networks analysis and mining 2017, pp 179–186

Rath B, Gao W, Ma J, Srivastava J (2018) Utilizing computational trust to identify rumor spreaders on twitter. SNAM

Rath B, Gao W, Srivastava J (2019) Evaluating vulnerability to fake news in social networks: a community health assessment model. In: ASONAM

Rath B, Salecha A, Srivastava J (2020) Detecting fake news spreaders in social networks using inductive representation learning. In: 2020 IEEE/ACM international conference on advances in social networks analysis and mining (ASONAM). IEEE, pp 182–189

Roy A, Sarkar C, Srivastava J, Huh J (2016) Trustingness & trustworthiness: a pair of complementary trust measures in a social network. In: ASONAM

Schlichtkrull M, Kipf TN, Bloem P, Van Den&nbsp;Berg R, Titov I, Welling M (2018) Modeling relational data with graph convolutional networks. In: European semantic web conference

Shu K, Mahudeswaran D, Wang S, Lee D, Liu H (2018) Fakenewsnet: a data repository with news content, social context and spatialtemporal information for studying fake news on social media. arXiv preprint arXiv:1809.01286

Shu K, Wang S, Liu H (2018) Understanding user profiles on social media for fake news detection. In: 2018 IEEE conference on multimedia information processing and retrieval (MIPR). IEEE, pp 430–435

Shu K, Wang S, Liu H (2019) Beyond news contents: the role of social context for fake news detection. In: Proceedings of the twelfth ACM international conference on web search and data mining, pp 312–320

Tang J, Qu M, Wang M, Zhang M, Yan J, Mei Q (2015) Line: large-scale information network embedding. In: WWW

Vogel I, Meghana M (2020) Fake news spreader detection on twitter using character n-grams. In: CLEF (working notes)

Volkova S, Jang JY (2018) Misleading or falsification: inferring deceptive strategies and types in online news and social media. In: Companion proceedings of the the web conference 2018, pp 575–583

Wu K, Yang S, Zhu KQ (2015) False rumors detection on sina weibo by propagation structures. In: 2015 IEEE 31st international conference on data engineering. IEEE, pp 651–662

Yao L, Mao C, Luo Y (2019) Graph convolutional networks for text classification. In: AAAI

Ying R, He R, Chen K, Eksombatchai P, Hamilton WL, Leskovec J (2018) Graph convolutional neural networks for web-scale recommender systems. In: SIGKDD

Yu B, Yin H, Zhu Z (2017) Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. IJCAI

Zhang X, He L, Chen K, Luo Y, Zhou J, Wang F (2018) Multi-view graph convolutional network and its applications on neuroimage analysis for parkinson’s disease. In: AMIA. American Medical Informatics Association

Zhang Y, Qi P, Manning CD (2018) Graph convolution over pruned dependency trees improves relation extraction. In: Conference on empirical methods in natural language processing

Zhao D, Rosson MB (2009) “How and why people twitter: the role that micro-blogging plays in informal communication at work. In: Proceedings of the ACM 2009 international conference on Supporting group work, pp 243–252