Embedding Partial Propagation Network for Fake News Early Detection

Amila Silva[a], Yi Han[a], Ling Luo[a], Shanika Karunasekera[a] and Christopher Leckie[a]

[a]School of Computing and Information Systems, The University of Melbourne

Abstract

Detecting fake news as early as possible has attracted growing attention due to its fast-spreading nature and the significant harm it can cause. As demonstrated in recent studies, the propagation pattern of fake news on social media differs from that of real news, and a number of works have extracted different types of features from the propagation pattern for detection. However, a major limitation of this approach is that the propagation network is not fully available in the early stages, and may take a long time to complete. As a result, existing network-based fake news detection methods yield low accuracy during the early stages of propagation. To bridge the research gap, in this work we: (1) propose a novel network embedding algorithm, based on the investigation of a wide range of features obtained from the propagation network, which are not well studied in previous work; and (2) design an autoencoder-based neural architecture to predict the embedding of the complete propagation network using the partially available network in the early stages of propagation. Our experiments show that with the predicted embedding for the complete propagation network, our model can achieve state-of-the-art performance while only having access to the early stage propagation network.

Keywords

Fake News Detection, News Propagation Networks, Network Embedding

1. Introduction

While the growing popularity of social media has greatly facilitated the exchange of information, it also provides an ideal platform to spread fake news, especially intentional disinformation, which has already and will continue to cause significant damage.

Even though a number of independent fact-checking organisations have emerged globally over recent years, the sheer volume of fake news makes it infeasible to rely entirely on human investigation. In addition, what makes the task even more challenging is that fake news needs to be detected at an early stage before it becomes widespread, since it is difficult to correct people's perception towards an issue once it is formed, even if the previous impression is inaccurate [1]. Therefore, in our work we focus on **fake news early detection**: verifying the validity of a news item within a certain time limit from when it is published online. Here we use the definition in [2] that *fake news is intentionally and verifiably false news published by a news outlet*—similar definitions have also been used in previous studies on fake news detection [3, 4, 5, 6].

It has been demonstrated that the propagation pattern of news on social media, *e.g.*, tweets and retweets of news on Twitter, can facilitate the detection of fake news [7, 8, 9, 10, 11], since the propagation pattern of fake news exhibits distinctive characteristics. However, instead of relying on the entire propagation network, which may take days or even weeks to complete, we only use the initial network that, for instance, belongs to the first 100 tweets, or tweets posted within the first few hours, to verify a news item. Specifically, the main *contributions* of this work include (Figure 1 provides an overview):

- We investigate a range of local and global features of the propagation network, including temporal-based, text-based and user-based, and compare their contributions to the detection of fake news. Based on the observations, we propose a novel network representation learning algorithm to embed the propagation network;
- We train an autoencoder that takes as input the partial propagation network corresponding to the tweets posted within the detection deadline, and predicts the embedding of the complete propagation network;
- We perform extensive experiments to demonstrate that the predicted embedding of the complete propagation network can be used to achieve state-of-the-art performance in fake news early detection.

The remainder of this paper is organised as follows: Section 2 defines the problem of fake news early detection; Section 3 describes how to embed the propagation network; Section 4 introduces the network embedding-based detection algorithm; Section 5 provides the experimental verification of the designed algorithm; Section 6

Title of the Proceedings: "Proceedings of the CIKM 2020 Workshops October 19-20, Galway, Ireland"
Editors of the Proceedings: Stefan Conrad, Ilaria Tiddi
EMAIL: amila.silva@student.unimelb.edu.au (A. Silva); yi.han@unimelb.edu.au (Y. Han); ling.luo@unimelb.edu.au (L. Luo); karus@unimelb.edu.au (S. Karunasekera); caleckie@unimelb.edu.au (C. Leckie)
ORCID: 0000-0003-2042-9050 (A. Silva); 0000-0001-6530-4564 (Y. Han); 0000-0002-1363-8308 (L. Luo); 0000-0001-7080-5064 (S. Karunasekera); 0000-0002-4388-0517 (C. Leckie)

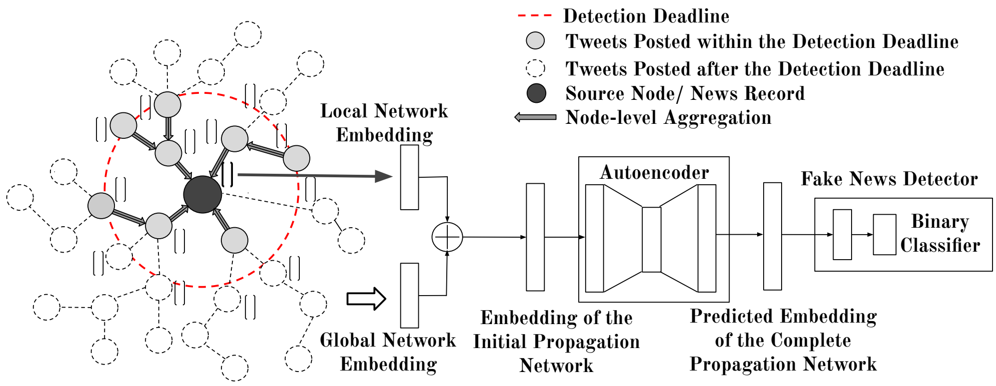

Figure 1: Overview of the Proposed Framework. Node-level and global attributes are extracted from the initial propagation network to generate the network embedding, which is used to train an autoencoder to predict the embedding for the complete propagation network.

reviews previous work on fake news detection; and finally Section 7 concludes the paper and offers directions for future work.

2. Problem Definition

We define the problem of fake news early detection as follows: let R^L be a set of labelled news records. Each record $r \in R^L$ is represented as a tuple $\langle t^r, W^r, G^r_t, H^r, y^r \rangle$, where (1) t^r is the timestamp when r is published online; (2) W^r is the text content of r; (3) G^r_t is the propagation network of r at timestamp $t^r + t$ (further explained below); (4) H^r is the set of timeline tweets posted by the users involved in G^r_t, *i.e.,* it provides background information of the news spreaders. Note that H^r does not necessarily always have to contain the latest timeline tweets; and (5) y^r is the label: y^r is 1 if r is false and 0 otherwise.

Each propagation network G^r_t is an attributed directed graph (V^r_t, E^r_t, X^r_t), where:

- V^r_t is the set of vertices/nodes, and each node is a tweet/retweet with the corresponding user. A special case is that an extra node representing the news is added to link the network together—it is called the source node hereafter.
- E^r_t is the set of edges. Here, edges represent how a news item spreads from one person to another as shown in Fig 1. However, Twitter APIs do not provide the immediate source of a retweet. To solve this problem, within each cascade we first sort the tweets by their timestamps, and then search for the potential source of a retweet from all the tweets that are published earlier. Specifically, there is an edge from node i to node j if (1) the user of tweet i mentions the user of tweet j; or (2) tweet i is public and tweet j is posted within a certain period of time after tweet i, *e.g.*, five hours. The follower and following relations are not included when constructing the edges of the propagation network, as they may not be available in real time due to the much stricter rate limit of the corresponding Twitter APIs, which prohibits the timely detection of fake news.
- X^r_t is the set of node-level and network-level features for G^r_t, which are explained in detail in Section 3.

The problem is to predict the label y^r for unlabelled news records $r \in R^U$ as false or real news records within a detection deadline Δt, where G^r_t for $r \in R^U$ is only available for $t \leq \Delta t$.

3. Representation Learning for Propagation Network

In this section, we propose a simple yet effective unsupervised network representation learning method to embed the propagation network. Formally, for a given propagation network G^r_t of news record r at timestamp $t^r + t$, representation learning aims to learn a mapping function $f : G^r_t \to h^r_t \in \mathbb{R}^d$ such that the obtained embedding h^r_t is useful for predicting the label y^r of the news record. Moreover, we analyse the informativeness of the learned embeddings for the initial propagation network at the detection deadline $G^r_{\Delta t}$ [1] and for the complete propagation network G^r_T $(T >> \Delta t)$.

Datasets. We conduct all our experiments on two publicly available datasets introduced in [12], which are collected from two fact-checking websites: (1) PolitiFact[2]; and (2) GossipCop[3]. Both datasets consist of labelled

[1] We denote the propagation network at the detection deadline $G^r_{\Delta t}$ as the initial propagation network.
[2] https://www.politifact.com/
[3] https://www.gossipcop.com/

Algorithm 1: Local Network Representation

Input: propagation network $G_t^r = (V_t^r, E_t^r, X_t^r)$
source node of r $v_s \in V_t^r$
gamma $\gamma \in [0, 1]$

Output: The local representation $f_{local}(G_t^r)$

1 $h_v^0 \leftarrow x_v \quad \forall v \in V_t^r$

2 **for** t *in* $1, 2, ..., k$ **do**

3   **for** v *in* V **do**

4         $h_v^t \leftarrow \gamma h_v^{t-1} + (1-\gamma)\frac{\sum_{\forall(v,u)\in E_t^r} h_u^{t-1}}{\sum_{\forall(v,u)\in E_t^r} 1}$

5   **end**

6 **end**

7 $f_{local}(G_t^r) \leftarrow h_{v_s}^k$

8 *return* $f_{local}(G_t^r)$

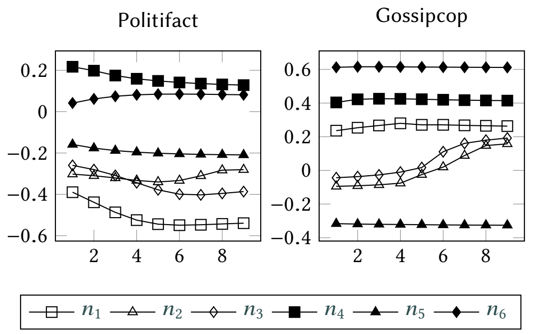

Figure 2: Correlation of the news labels with the source node representations using node-level user-based features ($n_1 - n_6$), at different iterations with the proposed label propagation scheme (X-axis: number of iterations, Y-axis: correlation values).

news records and all the tweets and retweets for each news item. Please refer to [4] for the descriptive statistics of the datasets.

3.1. Local Representation

In this subsection, we introduce how to embed node-level/local features.

3.1.1. Node-level Feature Aggregation

The nodes in a propagation network typically have complex multi-modal attributes e.g., temporal-based, text-based and user-based, which are useful to characterize the propagation network. Previous work [11, 13] mainly adopts simple averaging techniques to aggregate such node-level features, *e.g.*, the average time between tweets, or the average sentiment score of the tweets. The main limitation of these approaches is that they mostly ignore the structure of the network. To solve this problem, we propose an aggregation technique to summarise node-level attributes while preserving the structural properties of the network, which is elaborated in Algorithm 1.

The proposed approach iteratively updates the embedding of the nodes based on their one-hop neighbours. Specifically, the embedding of each node h_v^0 in the network is initialized using its features (Line 1 in Algorithm 1). Then for each iteration, the embeddings of one-hop neighbours (i.e., immediate successors in the directed graph) are propagated to the node following Line 4 in Algorithm 1. Here, γ controls the weight assigned to the propagated embeddings from the neighbours and the scale of the updated embedding. By running k iterations of the aforementioned label propagation scheme, each node can summarize its k-hop network based on the node-level features. Finally, the embedding of the source node v_s in the network G_t^r is returned as the local representation of the graph, *i.e.,* $f_{local}(G_t^r)$. In contrast to sophisticated neural architectures such as graph neural networks [14] that use data-driven trainable kernels to perform node-level aggregation, our approach is more straightforward and hence easier to interpret.

3.1.2. Node-level Features

We investigate three categories of node-level features: (1) user-based features; (2) text-based features; and (3) temporal features.

Node-level User-based Features. The following user-based features are studied in our experiments: whether the user is verified (n_1); the number of followers (n_2); the number of lists (n_3); the number of favourites (n_4); the number of tweets (n_5); and the number of friends mentioned per timeline tweet divided by the number of friends (n_6).

Such node-level user features can be useful to identify the differences in the way users engage with false news and real news. For example, less credible users are more likely to spread fake news as shown in [5]. n_1 and n_2 can be good indicators to identify less credible users. In addition, the finding in [13] shows that fake news spreaders tend to form larger clusters by their actions. n_6 can be useful to identify such user behaviours.

The correlations of features $n_1 - n_6$ with the news labels are shown in Figure 2. A positive (or negative) correlation in Figure 2 means that the corresponding feature values are higher (or lower) for fake news compared with real news. As can be seen, almost all the features show moderate correlation for at least one dataset. Specifically, n_1 and n_6 exhibit the highest correlation for PolitiFact and GossipCop, respectively, such that they are consistent with the aforementioned theory-driven explanations. However, feature n_1 shows opposite relations with the labels for PolitiFact (negative) and Gossipcop (positive),

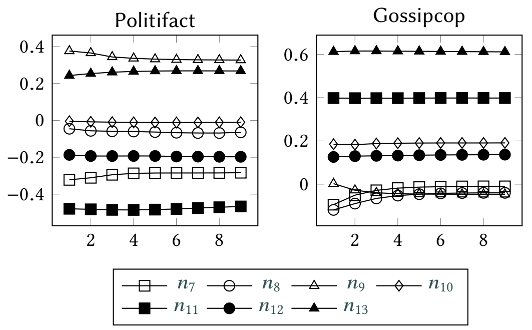

Figure 3: Correlation of the news labels with the source node representations using node-level text-based features ($n_7 - n_{14}$), at different iterations with the proposed label propagation scheme (X-axis: number of iterations, Y-axis: correlation values).

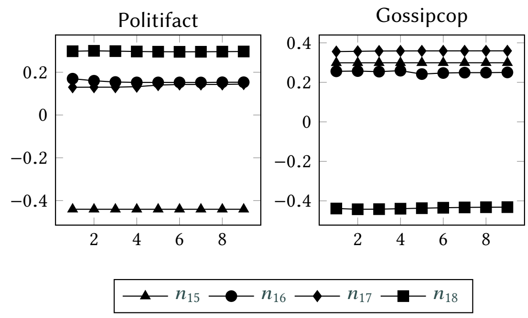

Figure 4: Correlation of the news labels with the source node representations using node-level temporal-based features ($n_{15} - n_{18}$), at different iterations with the proposed label propagation scheme (X-axis: number of iterations, Y-axis: correlation values).

which may be due to the domain difference of the two datasets.

Another interesting observation is that the correlation of each feature converges after a few iterations (≈ 8) using the proposed node-level aggregation approach. This observation indicates that the nodes that are close to the source node are more informative compared to the rest in the propagation networks.

Node-level Text-based Features. We further study text-based features as listed below: the sentiment scores computed using VADER[4] using text content in the tweets (n_7); the frequency of positive words (n_8); the frequency of negative words (n_9); the number of emojis (n_{10}); the number of mentions (n_{11}); the number of hashtags (n_{12}); and the percentage of tweets related to the target topic (n_{13})—we collect the timeline tweets for each user, and run tweet topic classification. For the dataset of PolitiFact (or GossipCop), we calculate the percentage of tweets whose topic is classified as "politics" (or "entertainment").

The node-level text features can be helpful to understand the linguistic differences of the text contents generated by the users engaging with fake news and real news. As shown in Figure 3, for both datasets a subset of the above features show relatively high correlation with the news labels, e.g., features n_9, n_{11} for PolitiFact, and features n_{11}, n_{13} for GossipCop. This aligns with well-defined theories—for example, it has been demonstrated that user-bias is a useful indicator to identify fake news spreaders [13]. A feature like n_{13} can help understand user bias to a particular domain, thus a user with a higher percentage of domain-specific posts (i.e., users with high n_{13}) is more likely to be a fake news spreader. In addition, correlation values in Figure 3 also converge after a few iterations of label propagation as seen in Figure 2.

[4] https://github.com/cjhutto/vaderSentiment

Node-level Temporal Features. Moreover, we analyse the following node-level temporal features to further capture the difference in the dissemination between fake and real news: the time difference with the source node (n_{15}); the time difference with the immediate predecessor (n_{16}); the average time difference with the immediate successors (n_{17}); user account timestamp (n_{18}).

According to the correlation analysis in Figure 4, the selected features show moderate correlations with the news labels for both datasets, and the results are also more consistent over different values of k, compared with the other node-level features.

In summary, the proposed label propagation scheme can capture up to k-hop neighbour information to generate the embedding for the source node based on the node-level features. Our empirical analysis shows that the nodes in close vicinity to the source node are mostly informative to generate useful local representations. Thus the proposed label propagation scheme with a limited k value is sufficient for performing node-level feature aggregation.

3.2. Global Representation

In addition to local features, the following network-level features are also extracted to represent the structural properties of each network G_t^r, which is denoted as the global representation $f_{global}(G_t^r)$ of the network.

- Wiener Index (g_1): The Wiener Index of a network is the sum of the lengths of the shortest paths between all pairs of vertices, which is a measure of the structural virality of a propagation network.
- Number of nodes (g_2): The number of nodes in a propagation network can be useful to understand the differences in the scale of user engagements for false and real news pieces.

Table 1
Correlation analysis of $g_1 - g_4$, where $g_i^{\Delta t}$ and g_i^{T} are the i^{th} global feature computed using the propagation networks at the detection deadline ($\Delta t = 5\ hours$) and using the complete propagation networks respectively. The statistically significant figures under correlation test are shown in bold.

Dataset	Politifact			Gossipcop		
Attribute	$corr(g_i^{\Delta t}, y)$	$corr(g_i^{T}, y)$	$corr(g_i^{\Delta t}, g_i^{T})$	$corr(g_i^{\Delta t}, y)$	$corr(g_i^{T}, y)$	$corr(g_i^{\Delta t}, g_i^{T})$
g_1	-0.0217	**-0.2825**	**0.2046**	0.0102	**0.2341**	**0.2341**
g_2	-0.0194	**-0.2589**	**0.2859**	**-0.0998**	**0.4099**	**0.2715**
g_3	0.0523	0.0107	**0.2959**	**-0.2573**	**0.2120**	**0.2808**
g_4	-0.0402	**-0.2151**	**0.3414**	**-0.0981**	**0.4249**	**0.2780**

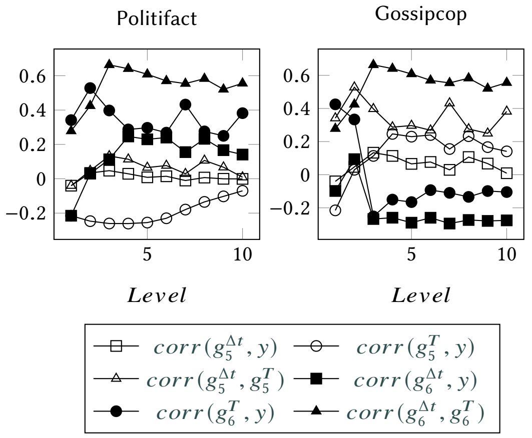

Figure 5: Correlation analysis of g_5 and g_6 at different network levels.

- Network depth (g_3): This measure captures how far the information is propagated via tweets and retweets.
- Maximum outdegree (g_4): This characterizes the most influential node in a propagation network.
- Number of nodes at different hops (g_5): This measure counts the number of k-hop neighbours with respect to the source node in a propagation network.
- Branching factor at different levels (g_6): For a given level l in a propagation network with respect to the source node, the branching factor at l is calculated as the ratio of the nodes at $l+1$ and the nodes at l.

Several observations can be made from the correlation analysis of $g_1 - g_6$ in Table 1 and Figure 5: (1) the global embeddings extracted from the initial propagation network do not show an obvious correlation with the news label; (2) the global embeddings from the complete propagation network, however, show much stronger correlation, which demonstrates the importance of having access to the complete network; and (3) there is a moderate correlation between the global embeddings generated from the initial network and from the corresponding complete network, which indicates the feasibility of using the initial network to predict the future embedding.

In addition, Figure 5 shows that the correlations of features $g_5 - g_6$ are stronger in close proximity to the source nodes, which is consistent with the observation in the node-level features. Hence, it further signifies the ability of the proposed label propagation scheme to preserve the network-level information.

After the local and global representations are obtained, we concatenate them to create the final embedding of the propagation network: $f(G_t^r) = f_{local}(G_t^r) \oplus f_{global}(G_t^r)$, where $\oplus$ is the concatenation operation.

4. Network-based Fake News Early Detection

As shown in Section 3, the embedding of the complete propagation network of news records have a relatively strong correlation with the labels. However, only the initial part of the propagation network is available at the early detection deadlines. Hence, we propose to train an autoencoder that takes the partial propagation network as input, and generates the embedding of the complete propagation network.

Formally, for a given news record r in the training set, denote the embedding of the initial network $f(G_{\Delta t}^r)$ and the complete network $f(G_T^r)$ as $f_{local}(G_{\Delta t}^r) \oplus f_{global}(G_{\Delta t}^r)$ and $f_{local}(G_T^r) \oplus f_{global}(G_T^r)$, respectively. The autoencoder is trained using the following reconstruction loss.

$$L_{recon} = ||Dec(Enc(f(G_{\Delta t}^r))) - f(G_T^r)||^2 \quad (1)$$

where Enc is the encoder: $f(G_{\Delta t}^r) \rightarrow l^r \in \mathbb{R}^{d'}$, Dec is the decoder: $l^r \rightarrow \overline{f(G_T^r)} \in \mathbb{R}^{d}$, d' is the latent dimension of the autoencoder, and $\overline{f(G_T^r)}$ is the predicted embedding for the complete propagation network. Both $Enc()$ and $Dec()$ mappings are modelled as 2-layer feedforward neural networks followed by a Sigmoid activation function (σ), which can be formally defined as follows:

$$Enc(f(G_T^r)) = \sigma(A_2(\sigma(A_1)f(G_T^r) + b_1) + b_2) \quad (2)$$

$$Dec(l^r) = \sigma(A_4(\sigma(A_3)l^r + b_3) + b_4) \quad (3)$$

where $\{A_1, A_4^T\} \in \mathbb{R}^{(2d',d)}$, $\{A_2, A_3^T\} \in \mathbb{R}^{(d',2d')}$, $\{b_1, b_4\} \in \mathbb{R}^{2d'}$, and $\{b_2, b_3\} \in \mathbb{R}^{d'}$ are trainable parameters. We leave the optimal neural architecture search for $Enc()$ and $Dec()$ in our model as future work.

Subsequently, the generated embedding of the complete network is used to classify the news record.

$$L_{predict} = BCE(\sigma(W * \overline{f(G_T^r)} + b), y^r) \quad (4)$$

where $BCE()$ is the standard binary cross entropy loss function and W, b are the trainable parameters of the fake news classifier.

The final loss function jointly optimises L_{recon} and $L_{predict}$:

$$L = L_{recon} + L_{predict} \quad (5)$$

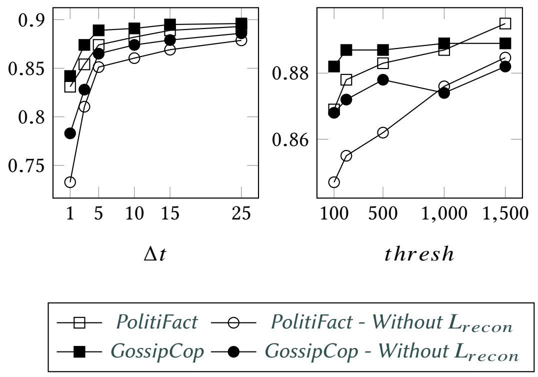

Figure 6: Accuracy of the proposed approach with different detection deadlines (Δt, in hour) and with different thresholds ($thresh$) for the maximum number of nodes in the initial network.

5. Experimental Verification

In this section, we present our experimental results to demonstrate the efficiency of the proposed algorithm.

5.1. Experimental Setup

Baselines. We compare our approach with four widely used text-based methods: (1) RST [15], (2) LIWC[5], (3) text-CNN [16], (4) HAN [17], one propagation network-based algorithm: HSA-BLSTM [18], and three mixed approaches: (1) TCNN-URG [19]; (2) CSI [6]; (3) dEFEND [4].

Parameter Settings. The proposed approach has three model-specific parameters: (1) the latent embedding dimension d'—the default value is set to 10, as we have empirically observed that the performance of the model plateaus for $d' \geq 10$; (2) detection deadline Δt—the default value is set to 5 $hours$, and we also analyse the model performance under other values of Δt; and (3) γ is set to 0.5. As for the baselines, please refer to [4] for the hyper-parameter settings. Note that all the propagation network-based baselines use the complete propagation network.

To evaluate the performance of the proposed approach, we adopt the commonly used metrics: (1) Accuracy (Acc); (2) Precision (Prec); (3) Recall (Rec); and (4) F1 Score (F1). Following the previous works [11, 4], we randomly choose 75% of news records for training and remaining 25% for testing, and the same process is performed for 5 different training and test splits and the average performance is reported.

5.2. Results for Fake News Detection

Table 2 shows the results for fake news detection. The proposed approach yields substantially better results for the GossipCop dataset, outperforming the best baseline by as much as 10% in accuracy.

For the PolitiFact dataset, the proposed approach outperforms all the baselines except dEFEND. However, the result for dEFEND is obtained using the complete propagation network for each news item, while our approach only requires the initial propagation network at the detection deadline. In other words, our method is more suitable for fake news early detection. In addition, dEFEND also extracts rich latent features from the news content, which may be manipulated by intelligent fake news generators to bypass detection—similar to the well-known adversarial attacks against machine learning models.

Ablation Study. Table 2 shows that without the reconstruction loss proposed in Eq. 1, *i.e.,* the model makes classification only based on the embedding of the initial propagation network, which is less informative as shown in Section 3, its accuracy drops by around 3% for both datasets. This clearly indicates the importance of predicting the embedding of the complete propagation network.

Furthermore, we analyse the contribution of different types of features. It can be seen that *Node-level User-based Features* are the most important among all *Node-level Features*, which is due to the high correlation of features like n_1 and n_6 with the actual news label. In addition, it is clear that *Global Features* are the least useful, as removing global features has minimum impact on the final result. The reason can be that most global features adopt simple sum operation and ignore the network structure. Overall, the removal of each type of feature in the ablation study decreases the final performance of the model, which verifies the positive contribution of the facets of the proposed model.

Parameter Sensitivity. In Figure 6, we have checked

[5] https://liwc.wpengine.com/

Table 2
Results for fake news detection of different methods, which are classified under three categories: (1) news record content-based approaches (N); (2) propagation network-based approaches (P); and (3) suitability for early fake news detection (i.e., ability to yield better performance with initial propagation networks) (E)

Method	Type			Politifact				Gossipcop			
	N	P	E	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1
RST	✓		✓	0.607	0.625	0.523	0.569	0.531	0.534	0.492	0.512
LIWC	✓		✓	0.769	0.843	0.794	0.818	0.736	0.756	0.461	0.572
text-CNN	✓		✓	0.653	0.678	0.863	0.760	0.739	0.707	0.477	0.569
HAN	✓		✓	0.837	0.824	0.896	0.860	0.742	0.655	0.689	0.672
HPA-BLSTM		✓		0.846	0.894	0.868	0.881	0.753	0.684	0.662	0.673
TCNN-URG	✓	✓		0.712	0.711	0.941	0.810	0.736	0.715	0.521	0.603
CSI	✓	✓		0.827	0.847	0.897	0.871	0.772	0.732	0.638	0.682
dEFEND	✓	✓		**0.904**	**0.902**	**0.956**	**0.928**	0.808	0.729	0.782	0.755
Our Approach		✓	✓	0.874	0.878	0.870	0.871	**0.889**	**0.859**	**0.805**	**0.828**
Ablation Study											
(-) *Reconstruction Loss*				0.851	0.854	0.855	0.854	0.865	0.846	0.791	0.810
(-) *Global Features*				0.871	0.837	0.867	0.852	0.876	0.851	0.769	0.802
(-) *Node-level Features*				0.722	0.686	0.840	0.751	0.779	0.679	0.669	0.673
(-) *Node-level Text Features*				0.840	0.834	0.852	0.841	0.863	0.808	0.772	0.781
(-) *Node-level Temporal Features*				0.862	0.854	0.879	0.864	0.878	0.844	0.784	0.804
(-) *Node-level User Features*				0.782	0.772	0.815	0.791	0.857	0.806	0.768	0.778

the performance of the proposed model with different configurations for the initial network. As can be seen, if the initial network is too small due to low $thresh$ or Δt, the performance drops drastically if the predictions are made using the embedding of the initial networks (*i.e.,* without L_{recon}). In contrast, the model performs reasonably well with the predicted embedding of the complete propagation network even with a small initial network size.

6. Related Work

Existing work on fake news detection mainly relies on two sources of information: news content and social context. Based on this criterion, we classify prior work into two categories: content-based and context-based.

6.1. Content-based Fake News Detection

This type of method uses news headlines and body content to detect fake news. The content here is not limited to text-based, but can also include visual information. For example, Wang *et al.* [20] extract both text and visual features from posts to train a fake news detector and an event discriminator simultaneously. Other work that applies multi-modal techniques includes [21, 22, 23]. In addition to content, styles can assist differentiating between fake and real news, since fake news aims to mislead the public, and often exhibits distinct writing styles [24]. Furthermore, the idea of knowledge-based detection is discussed in [2].

6.2. Context-based Fake News Detection

Social context here refers to the interactions between users. These engagements can be transferred into different types of graphs to facilitate fake news detection. For example, a range of models have been applied to study the propagation patterns, including Propagation Tree Kernel [7], LSTM cells incorporated with RNNs [8], and GNNs [3]. Other methods that fall into this category include [25, 11, 10]. Our method is also context-based, although it only relies on the partial propagation network for fake news early detection.

In addition to the above two categories, a number of methods use a mixed strategy and rely on both news content and associated user inter-actions over social media to detect fake news [6, 26, 27].

7. Conclusions and Future Work

In this work, we have designed a novel representation learning framework for fake news early detection, by embedding news propagation networks using both global-level and node-level attributes. Subsequently, we propose to train an autoencoder to predict the embedding of the complete propagation network using the partial network at an early stage. We demonstrate that the predicted embedding for the complete propagation network can achieve better results for fake news early detection.

For future work, we intend to work on the following directions: (1) Our empirical studies show that some network attributes carry domain-specific relations with the news labels. Therefore, a model trained on the dataset from one domain using these features may perform very poorly for data from other domains. In order to solve this problem, we will study how to extend the proposed approach in a domain-agnostic manner. (2) We have not considered other attributes of news records such as the textual and image content. Hence, the integration of features from these modalities with the proposed framework can be another direction to explore.

References

[1] J. D. keersmaecker, A. Roets, 'Fake news': Incorrect, but hard to correct. the role of cognitive ability on the impact of false information on social impressions, Intelligence (2017).

[2] X. Zhou, R. Zafarani, Fake news: A survey of research, detection methods, and opportunities, arXiv:1812.00315 (2018).

[3] F. Monti, F. Frasca, D. Eynard, D. Mannion, M. M. Bronstein, Fake news detection on social media using geometric deep learning, arXiv:1902.06673 (2019).

[4] K. Shu, L. Cui, S. Wang, D. Lee, H. Liu, DEFEND: Explainable fake news detection, in: Proc. of KDD, 2019.

[5] K. Shu, A. Sliva, S. Wang, J. Tang, H. Liu, Fake news detection on social media: A data mining perspective, SIGKDD Explor (2017).

[6] N. Ruchansky, S. Seo, Y. Liu, CSI: A hybrid deep model for fake news detection, in: Proc. of CIKM, 2017.

[7] J. Ma, W. Gao, K.-F. Wong, Detect rumors in microblog posts using propagation structure via kernel learning, in: Proc. of ACL, 2017.

[8] L. Wu, H. Liu, Tracing fake-news footprints: Characterizing social media messages by how they propagate, in: Proc. of WSDM, 2018.

[9] Y. Liu, Y.-f. B. Wu, Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks, in: Proc. of AAAI, 2018.

[10] X. Zhou, R. Zafarani, Network-based fake news detection: A pattern-driven approach, arXiv:1906.04210 (2019).

[11] K. Shu, D. Mahudeswaran, S. Wang, H. Liu, Hierarchical propagation networks for fake news detection: Investigation and exploitation, arXiv:1903.09196 (2019).

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

[13] K. Shu, G. Zheng, Y. Li, S. Mukherjee, A. Hassan Awadallah, S. Ruston, H. Liu, Leveraging multi-source weak social supervision for early detection of fake news, arXiv:2004.01732 (2020).

[14] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, P. S. Yu, A comprehensive survey on graph neural networks, arXiv:1901.00596 (2019).

[15] V. Rubin, N. Conroy, Y. Chen, Towards news verification: Deception detection methods for news discourse, in: Proc. of HICSS, 2015.

[16] Y. Kim, Convolutional neural networks for sentence classification, in: Proc. of EMNLP, 2014.

[17] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, E. Hovy, Hierarchical attention networks for document classification, in: Proc. of NAACL-HLT, 2016.

[18] H. Guo, J. Cao, Y. Zhang, J. Guo, J. Li, Rumor detection with hierarchical social attention network, in: Proc. of CIKM, 2018.

[19] F. Qian, C. Gong, K. Sharma, Y. Liu, Neural user response generator: Fake news detection with collective user intelligence, in: Proc. of IJCAI, 2018.

[20] Y. Wang, F. Ma, Z. Jin, Y. Yuan, G. Xun, K. Jha, L. Su, J. Gao, EANN: Event adversarial neural networks for multi-modal fake news detection, in: Proc. of KDD, 2018.

[21] Z. Jin, J. Cao, Y. Zhang, J. Zhou, Q. Tian, Novel visual and statistical image features for microblogs news verification, IEEE Trans. on Multimedia (2017).

[22] Y. Yang, L. Zheng, J. Zhang, Q. Cui, Z. Li, P. S. Yu, TI-CNN: Convolutional neural networks for fake news detection, arXiv:1806.00749 (2018).

[23] X. Zhou, J. Wu, R. Zafarani, SAFE: Similarity-aware multi-modal fake news detection, in: Proc. of PAKDD, 2020.

[24] S. Volkova, K. Shaffer, J. Y. Jang, N. Hodas, Separating facts from fiction: Linguistic models to classify suspicious and trusted news posts on twitter, in: Proc. of ACL, 2017.

[25] S. Yang, K. Shu, S. Wang, R. Gu, F. Wu, H. Liu, Unsupervised fake news detection on social media: A generative approach, AAAI (2019).

[26] J. Zhang, B. Dong, P. S. Yu, FAKEDETECTOR: Effective fake news detection with deep diffusive neural network, arXiv:1805.08751 (2018).

[27] K. Shu, S. Wang, H. Liu, Beyond news contents: The role of social context for fake news detection, in: Proc. of WSDM, 2019.