Beyond News Contents:
The Role of Social Context for Fake News Detection

Kai Shu
Arizona State University
kai.shu@asu.edu

Suhang Wang
Penn State University
szw494@psu.edu

Huan Liu
Arizona State University
huan.liu@asu.edu

ABSTRACT

Social media is becoming popular for news consumption due to its fast dissemination, easy access, and low cost. However, it also enables the wide propagation of *fake news*, i.e., news with intentionally false information. Detecting fake news is an important task, which not only ensures users receive authentic information but also helps maintain a trustworthy news ecosystem. The majority of existing detection algorithms focus on finding clues from news contents, which are generally not effective because fake news is often intentionally written to mislead users by mimicking true news. Therefore, we need to explore auxiliary information to improve detection. The social context during news dissemination process on social media forms the inherent *tri-relationship*, the relationship among publishers, news pieces, and users, which has potential to improve fake news detection. For example, partisan-biased publishers are more likely to publish fake news, and low-credible users are more likely to share fake news. In this paper, we study the novel problem of exploiting social context for fake news detection. We propose a tri-relationship embedding framework TriFN, which models publisher-news relations and user-news interactions simultaneously for fake news classification. We conduct experiments on two real-world datasets, which demonstrate that the proposed approach significantly outperforms other baseline methods for fake news detection.

KEYWORDS

Fake news detection; joint learning; social media mining

1 INTRODUCTION

People nowadays tend to seek out and consume news from social media rather than traditional news organizations. For example, 62% of U.S. adults get news on social media in 2016, while in 2012, only

49 percent is reported seeing news on social media[1]. However, social media is a double-edged sword for news consumption. The quality of news on social media is much lower than that of traditional news organizations. Large volumes of *fake news*, i.e., news with intentionally false information, are produced online for a variety of purposes, such as financial and political gain [2, 13].

Fake news can have detrimental effects on individuals and the society. First, people may be misled by fake news and accept false beliefs [18, 21]. Second, fake news could change the way people respond to true news[2]. Third, the wide propagation of fake news could break the trustworthiness of entire news ecosystem. Thus, it is important to detect fake news on social media. Fake news is intentionally written to mislead consumers, which makes it nontrivial to detect simply based on news content. To build an effective and practical fake news detection system, it is natural and necessary to explore *auxiliary information* from different perspectives.

The news ecosystem on social media provides abundant social context information, which involves three basic entities, i.e., publishers, news pieces, and social media users. Figure 1 gives an illustration of such ecosystem. In Figure 1, p_1, p_2 and p_3 are news publishers who publish news $a_1, \ldots, a_4$ and $u_1, \ldots, u_6$ are users who have engaged in sharing these news pieces. In addition, users tend to form social links with like-minded people with similar interests. As we will show, the *tri-relationship*, the relationship among publishers, news pieces, and users, contains additional information to help detect fake news.

First, sociological studies on journalism have theorized the correlation between the partisan bias of publisher and the veracity degree of news content [8]. The partisan bias means the perceived bias of the publisher in the selection of how news is reported and covered [6]. For example, in Figure 1, p_1 is a publisher with extreme left partisan bias and p_2 is a publisher with extreme right partisan bias. To support their own partisan viewpoints, they have high degree to distort the facts and report fake news pieces, such as a_1 and a_3; while for a mainstream publisher p_3 that has least partisan bias, he/she has a lower chance to manipulate original news events, and is more likely to write a true news piece a_4. Thus, exploiting the partisan bias of publishers to bridge the publisher-news relationships can bring additional benefits to predict fake news.

Second, mining user engagements towards news pieces on social media also help fake news detection. Previous approaches try to aggregate users' attributes to infer the degree of news veracity by assuming that either (i) all the users contribute equally for learning feature representations of news pieces [10]; or (ii) user features are
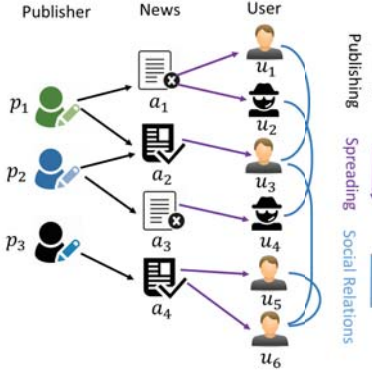
Figure 1: An illustration of tri-relationship among publishers, news pieces, and users, during the news dissemination process. For example, an edge $(p \rightarrow a)$ demonstrates that publisher p publishes news item a, an edge $(a \rightarrow u)$ represents news item a is spread by user u, and an edge $(u_1 \leftrightarrow u_2)$ indicates the social relation between user u_1 and u_2.

grouped locally for specific news and the global user-news interactions are ignored [4]. However, in practice, these assumptions may not hold. On social media, different users have different credibility levels. The credibility score, which means "the quality of being trustworthy" [1], has a strong indication of whether some user is more likely to share fake news or not. Those less credible users, such as malicious accounts or normal users who are vulnerable to fake news, are more likely to spread fake news. For example, u_2 and u_4 are users with low credibility scores, and they tend to spread fake news more than other highly credible users. In addition, users tend to form relationships with like-minded people [25]. For example, user u_5 and u_6 are friends on social media, so they tend to post those news that confirm their own views, such as a_4. Therefore, incorporating the user credibility levels to capture the user-news interactions has potentials to improve fake news prediction.

Moreover, the publisher-news relationships and user-news interactions both provide new and different perspectives of social context, and thus contain complementary information to advance fake news detection. In this paper, we investigate: (1) how to mathematically model the tri-relationship to extract feature representations of news pieces; and (2) how to take advantage of tri-relationship modeling for fake news detection. Our solutions to these two challenges results in a novel framework TriFN for fake news detection problem. Our main contributions are summarized as follows:

- We provide a principled way to model tri-relationship among publishers, news pieces, and users simultaneously;
- We propose a novel framework TriFN, which exploits both user-news interactions and publisher-news relations for learning news feature representations to predict fake news; and
- We conduct extensive experiments on two real-world datasets to assess the effectiveness of TriFN.

2 PROBLEM STATEMENT

Let $\mathcal{A} = \{a_1, a_2, ..., a_n\}$ be the set of n news pieces, and $\mathcal{U} = \{u_1, u_2, ..., u_m\}$ be the set of m users on social media posting these news pieces. We denote $\mathbf{X} \in \mathbb{R}^{n \times t}$ as the bag-of-word feature matrix
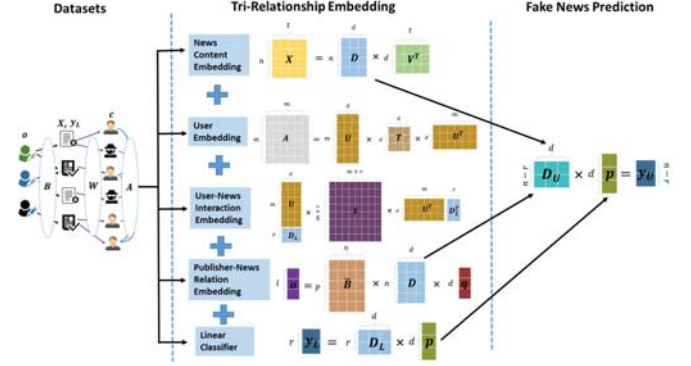

Figure 2: The tri-relationship embedding framework, which consists of five components: news contents embedding, user embedding, user-news interaction embedding, publisher-news relation embedding, and news classification.

of news pieces, where t is the dimension of vocabulary size. We use $\mathbf{A} \in \{0, 1\}^{m \times m}$ to denote the user-user adjacency matrix, where $\mathbf{A}_{ij} = 1$ indicates that user u_i and u_j are friends; otherwise $\mathbf{A}_{ij} = 0$. We denote the user-news interaction matrix as $\mathbf{W} \in \{0, 1\}^{m \times n}$, where $\mathbf{W}_{ij} = 1$ indicates that user u_i has shared the news piece a_j; otherwise $\mathbf{W}_{ij} = 0$. It's worth mentioning that we focus on those user-news interactions in which users agree with the news. For example, we only consider those users who share news pieces without comments, and these users share the same alignment of viewpoints with the news items [12]. We will introduce more details in Section 3.3. We also denote $\mathcal{P} = \{p_1, p_2, ..., p_l\}$ as the set of l news publishers. In addition, we denote $\mathbf{B} \in \mathbb{R}^{l \times n}$ as the publisher-news publishing matrix, and $\mathbf{B}_{kj} = 1$ means news publisher p_k publishes the news article a_j; otherwise $\mathbf{B}_{kj} = 0$. We assume that the partisan bias labels of some publishers are given and available (see more details of how to collect partisan bias labels in Sec 3.4). We define $\mathbf{o} \in \{-1, 0, 1\}^{l \times 1}$ as the partisan label vectors, where -1, 0, 1 represents left-, neutral-, and right-partisan bias.

Similar to previous research [10, 32], we treat fake news detection problem as a binary classification problem. In other words, each news piece can be true or fake, and we use $\mathbf{y} = \{y_1; y_2; ...; y_n\} \in \mathbb{R}^{n \times 1}$ to represent the labels, and $y_j = 1$ means news piece a_j is fake news; $y_j = -1$ means true news. With the notations given above, the problem is formally defined as,

Given news article feature matrix $\mathbf{X}$*, user adjacency matrix* $\mathbf{A}$*, user social engagement matrix* $\mathbf{W}$*, publisher-news publishing matrix* $\mathbf{B}$*, publisher partisan label vector* $\mathbf{o}$*, and partial labeled news vector* $\mathbf{y}_L$*, we aim to predict remaining unlabeled news label vector* $\mathbf{y}_U$*.*

3 A TRI-RELATIONSHIP EMBEDDING FRAMEWORK

In this section, we present the details of the proposed framework TriFN for modeling tri-relationship for fake news detection. It consists of five major components (Figure 2): a news contents embedding component, a user embedding component, a user-news interaction embedding component, a publisher-news relation embedding component, and a semi-supervised classification component.

In general, the news contents embedding component describes the mapping of news from bag-of-word features to latent feature space; the user embedding component illustrates the extraction of user latent features from user social relations; the user-news interaction embedding component learn the feature representations of news pieces guided by their partial labels and user credibilities; The publisher-news relation embedding component regularize the feature representations of news pieces through publisher partisan bias labels; The semi-supervised classification component learns a classification function to predict unlabeled news items.

3.1 News Contents Embedding

We can use news contents to find clues to differentiate fake news and true news. Recently, it has been shown that nonnegative matrix factorization (NMF) algorithms are very practical and popular to learn document representations [20, 28, 38]. It can project the news-word matrix $\mathbf{X}$ to a joint latent semantic factor space with low dimensionality, such that the news-word relations are modeled as the inner product in the space. Specifically, giving the news-word matrix $\mathbf{X} \in \mathbb{R}^{n \times t}$, NMF methods try to find two nonnegative matrices $\mathbf{D} \in \mathbb{R}_+^{n \times d}$ and $\mathbf{V} \in \mathbb{R}_+^{t \times d}$, where d is the dimension of the latent space, by solving the following optimization problem,

$$\min_{\mathbf{D}, \mathbf{V} \geq 0} \| \mathbf{X} - \mathbf{D}\mathbf{V}^T \|_F^2 + \lambda(\|\mathbf{D}\|_F^2 + \|\mathbf{V}\|_F^2) \qquad (1)$$

where $\mathbf{D}$ and $\mathbf{V}$ are the nonnegative matrices indicating low dimension representations of news pieces and words. Note that we denote $\mathbf{D} = [\mathbf{D}_L; \mathbf{D}_U]$, where $\mathbf{D}_L \in \mathbb{R}^{r \times d}$ is the news latent feature matrix for labeled news; while $\mathbf{D}_U \in \mathbb{R}^{(n-r) \times d}$ is the news latent feature matrix for unlabeled news. The term $\lambda(\|\mathbf{D}\|_F^2 + \|\mathbf{V}\|_F^2)$ is introduced to avoid over-fitting.

3.2 User Embedding

On social media, people tend to form relationships with like-minded people, rather than those users who have opposing preferences and interests. Thus, connected users are more likely to share similar latent interests in news pieces. To obtain a standardized representation, we use nonnegative matrix factorization to learn the users' latent representations. Specifically, giving user-user adjacency matrix $\mathbf{A} \in \{0, 1\}^{m \times m}$, we learn nonnegative matrix $\mathbf{U} \in \mathbb{R}_+^{m \times d}$ by solving the following optimization problem,

$$\min_{\mathbf{U}, \mathbf{T} \geq 0} \|\mathbf{Y} \odot (\mathbf{A} - \mathbf{U}\mathbf{T}\mathbf{U}^T)\|_F^2 + \lambda(\|\mathbf{U}\|_F^2 + \|\mathbf{T}\|_F^2) \qquad (2)$$

where $\mathbf{U}$ is the user latent matrix, $\mathbf{T} \in \mathbb{R}_+^{d \times d}$ is the user-user correlation matrix, $\mathbf{Y} \in \mathbb{R}^{m \times m}$ controls the contribution of $\mathbf{A}$, and $\odot$ denotes the Hadamard product operation. Since only positive links are observed in $\mathbf{A}$, following common strategies [19], we first set $\mathbf{Y}_{ij} = 1$ if $\mathbf{A}_{ij} = 1$, and then perform negative sampling and generate the same number of unobserved links and set weights as 0. The term $\lambda(\|\mathbf{U}\|_F^2 + \|\mathbf{T}\|_F^2)$ is to avoid over-fitting.

3.3 User-News Interaction Embedding

We model the user-news interactions by considering the relationships between user features and the labels of news items. We have shown (see Section 1) that users with low credibilities are more likely to spread fake news, while users with high credibilities are

less likely to spread fake news. To measure user credibility scores, we adopt the practical approach in [1]. The basic idea in [1] is that less credible users are more likely to coordinate with each other and form big clusters, while more credible users are likely to from small clusters. Specifically, the credibility scores are measured through the following major steps: 1) detect and cluster coordinate users based on user similarities; 2) weight each cluster based on the cluster size. Note that for our fake news detection task, we do not assume that credibility scores are directly provided, but inferred from widely available data, such as user-generated contents. By using the method in [1], we can assign each user u_i a credibility score $\mathbf{c}_i \in [0, 1]$. A larger $\mathbf{c}_i$ indicates that user u_i has a higher credibility, while a lower $\mathbf{c}_i$ indicates a lower credibility score. We use $\mathbf{c} = \{\mathbf{c}_1, \mathbf{c}_2, ..., \mathbf{c}_m\}$ to denote the credibility score vector for all users.

First, high-credibility users are more likely to share true news pieces, so we ensure that the distance between latent features of high-credibility users and that of true news is minimized,

$$\min_{\mathbf{U}, \mathbf{D_L} \geq 0} \sum_{i=1}^{m} \sum_{j=1}^{r} \mathbf{W}_{ij} \mathbf{c}_i (1 - \frac{1 + \mathbf{y}_{Lj}}{2}) \|\mathbf{U}_i - \mathbf{D}_{L_j}\|_2^2 \qquad (3)$$

and $(1 - \frac{1 + \mathbf{y}_{Lj}}{2})$ is to ensure we only include true news pieces (i.e., $\mathbf{y}_{Lj} = -1$), and $\mathbf{c}_i$ is to adjust the contribution of user u_i to the loss function. For example, if $\mathbf{c}_i$ is large (high-credibility) and $\mathbf{W}_{ij} = 1$, we put a bigger weight on forcing the distance of feature $\mathbf{U}_i$ and $\mathbf{D}_{Lj}$ to be small; if $\mathbf{c}_i$ is small (low-credibility) and $\mathbf{W}_{ij} = 1$, than we put a smaller weight on forcing the distance of feature $\mathbf{U}_i$ and $\mathbf{D}_{Lj}$ to be small.

Second, low-credibility users are more likely to share fake news pieces, and we aim to minimize the distance between latent features of low-credibility users and that of fake news,

$$\min_{\mathbf{U}, \mathbf{D_L} \geq 0} \sum_{i=1}^{m} \sum_{j=1}^{r} \mathbf{W}_{ij} (1 - \mathbf{c}_i) (\frac{1 + \mathbf{y}_{Lj}}{2}) \|\mathbf{U}_i - \mathbf{D}_{L_j}\|_2^2 \qquad (4)$$

and the term $(\frac{1 + \mathbf{y}_{Lj}}{2})$ is to ensure we only include fake news pieces (i.e., $\mathbf{y}_{L_j} = 1$), and $(1 - \mathbf{c}_i)$ is to adjust the contribution of user u_i to the loss function. For example, if $\mathbf{c}_i$ is large (high-credibility) and $\mathbf{W}_{ij} = 1$, we put a smaller weight on forcing the distance of feature $\mathbf{U}_i$ and $\mathbf{D}_{Lj}$ to be small; if $\mathbf{c}_i$ is small (low-credibility) and $\mathbf{W}_{ij} = 1$, then we put a bigger weight on forcing the distance of feature $\mathbf{U}_i$ and $\mathbf{D}_{Lj}$ to be small.

Finally, We combine Eqn 3 and Eqn 4 to consider the above two situations, and obtain the following objective function,

$$\min_{\mathbf{U}, \mathbf{D_L} \geq 0} \underbrace{\sum_{i=1}^{m} \sum_{j=1}^{r} \mathbf{W}_{ij} \mathbf{c}_i (1 - \frac{1 + \mathbf{y}_{Lj}}{2}) \|\mathbf{U}_i - \mathbf{D}_{L_j}\|_2^2}_{\text{True news}}$$

$$+ \underbrace{\sum_{i=1}^{m} \sum_{j=1}^{r} \mathbf{W}_{ij} (1 - \mathbf{c}_i) (\frac{1 + \mathbf{y}_{Lj}}{2}) \|\mathbf{U}_i - \mathbf{D}_{L_j}\|_2^2}_{\text{Fake news}} \qquad (5)$$

For simplicity, Eqn 5 can be rewritten as,

$$\min_{\mathbf{U}, \mathbf{D}_L \geq 0} \sum_{i=1}^{m} \sum_{j=1}^{r} \mathbf{G}_{ij} ||\mathbf{U}_i - \mathbf{D}_{L_j}||_2^2 \tag{6}$$

where $\mathbf{G}_{ij} = \mathbf{W}_{ij}(\mathbf{c}_i(1 - \frac{1+\mathbf{y}_{Lj}}{2}) + (1 - \mathbf{c}_i)(\frac{1+\mathbf{y}_{Lj}}{2}))$. If we denote a new matrix $\mathbf{H} = [\mathbf{U}; \mathbf{D}_L] \in \mathbb{R}^{(m+r) \times d}$, we can also rewrite Eqn. 6 as a matrix form as follows,

$$\min_{\mathbf{U}, \mathbf{D}_L \geq 0} \sum_{i=1}^{m} \sum_{j=1}^{r} \mathbf{G}_{ij} ||\mathbf{U}_i - \mathbf{D}_{L_j}||_2^2 \Leftrightarrow \min_{\mathbf{H} \geq 0} \sum_{i=1}^{m} \sum_{j=1+m}^{r+m} \mathbf{G}_{ij} ||\mathbf{H}_i - \mathbf{H}_j||_2^2$$

$$\Leftrightarrow \min_{\mathbf{H} \geq 0} \sum_{i,j=1}^{m+r} \mathbf{F}_{ij} ||\mathbf{H}_i - \mathbf{H}_j||_2^2 \Leftrightarrow \min_{\mathbf{H} \geq 0} \text{tr}(\mathbf{H}^T \mathbf{L} \mathbf{H}) \tag{7}$$

where $\mathbf{L} = \mathbf{S} - \mathbf{F}$ is the Laplacian matrix and $\mathbf{S}$ is a diagonal matrix with diagonal element $S_{ii} = \sum_{j=1}^{m+r} \mathbf{F}_{ij}$. $\mathbf{F} \in \mathbb{R}^{(m+r) \times (m+r)}$ is computed as follows,

$$\mathbf{F}_{ij} = \begin{cases} 0, & i, j \in [1, m] \text{ or } i, j \in [m+1, m+r] \\ \mathbf{G}_{i(j-m)}, & i \in [1, m], j \in [m+1, m+r] \\ \mathbf{G}_{(i-m)j}, & i \in [m+1, m+r], j \in [1, m] \end{cases} \tag{8}$$

3.4 Publisher-News Relation Embedding

Fake news is often written to convey opinions or claims that support the partisan bias of news publishers. Thus, a good news representation should be good at predicting the partisan bias of its publisher. We obtain the list of publishers' partisan scores from a well-known media bias fact-checking websites MBFC [3]. The partisan bias labels are checked with a principled methodology that ensures the reliability and objectivity of the partisan annotations. The labels are categorized into five categories: "left", "left-Center","least-biased","right-Center" and "right". To further ensure the accuracy of the labels, we only consider those news publishers with the annotations ["left","least-biased", "Right"], and rewrite the corresponding labels as [-1,0,1]. Thus, we can construct a partisan label vectors for news publishers as $\mathbf{o}$. Note that we may not obtain the partisan labels for all publishers, so we introduce $\mathbf{e} \in \{0, 1\}^{l \times 1}$ to control the weight of $\mathbf{o}$. If we have the partisan bias label of publisher p_k, then $\mathbf{e}_k = 1$; otherwise, $\mathbf{e}_k = 0$. The basic idea is to utilize publisher partisan labels vector $\mathbf{o} \in \mathbb{R}^{l \times 1}$ and publisher-news matrix $\mathbf{B} \in \mathbb{R}^{l \times n}$ to optimize the news feature representation learning. Specifically, we optimization following objective function,

$$\min_{\mathbf{D} \geq 0, \mathbf{q}} \ || \ \mathbf{e} \odot (\bar{\mathbf{B}} \mathbf{D} \mathbf{q} - \mathbf{o})||_2^2 + \lambda ||\mathbf{q}||_2^2 \tag{9}$$

where we assume that the latent feature of news publisher can be represented by the features of all the news it published, i.e., $\bar{\mathbf{B}} \mathbf{D}$. $\bar{\mathbf{B}}$ is the normalized user-news publishing relation matrix, i.e., $\bar{\mathbf{B}}_{kj} = \frac{\mathbf{B}_{kj}}{\sum_{j=1}^{n} \mathbf{B}_{kj}}$. $\mathbf{q} \in \mathbb{R}^{d \times 1}$ is the weighting matrix that maps news publishers' latent features to corresponding partisan label vector $\mathbf{o}$.

3.5 Proposed Framework - TriFN

We have introduced how we can learn news latent features by modeling different aspects of the tri-relationship. We further employ a semi-supervised linear classifier term as follows,

$$\min_{\mathbf{p}} \ || \ \mathbf{D}_L \mathbf{p} - \mathbf{y}_L||_2^2 + \lambda ||\mathbf{p}||_2^2 \tag{10}$$

where $\mathbf{p} \in \mathbb{R}^{d \times 1}$ is the weighting matrix that maps news latent features to fake news labels. With all previous components, TriFN solves the following optimization problem,

$$\min_{\mathbf{D}, \mathbf{U}, \mathbf{V}, \mathbf{T} \geq 0, \mathbf{p}, \mathbf{q}} ||\mathbf{X} - \mathbf{D} \mathbf{V}^T||_F^2 + \alpha ||\mathbf{Y} \odot (\mathbf{A} - \mathbf{U} \mathbf{T} \mathbf{U}^T)||_F^2$$

$$+ \beta \text{tr}(\mathbf{H}^T \mathbf{L} \mathbf{H}) + \gamma ||\mathbf{e} \odot (\bar{\mathbf{B}} \mathbf{D} \mathbf{q} - \mathbf{o})||_2^2 \tag{11}$$

$$+ \eta ||\mathbf{D}_L \mathbf{p} - \mathbf{y}_L||_2^2 + \lambda R$$

where $R = (||\mathbf{D}||_F^2 + ||\mathbf{V}||_F^2 + ||\mathbf{U}||_F^2 + ||\mathbf{T}||_F^2 + ||\mathbf{p}||_2^2 + ||\mathbf{q}||_2^2)$ is to avoid over-fitting. The first term models the news latent features from news contents; the second term extracts user latent features from their social relationships; and the third term incorporates the user-news interactions; and the fourth term models publisher-news relationships. The fifth term adds a semi-supervised fake news classifier. Therefore, this framework provides a principled way to model tri-relationship for fake news prediction.

4 AN OPTIMIZATION ALGORITHM

In this section, we present the detail optimization process for the proposed framework TriFN. If we update the variables jointly, the objective function in Eq. 11 is not convex. Thus, we propose to use alternating least squares to update the variables separately. For simplicity, we user $\mathcal{L}$ to denote the objective function in Eq. 11. Next, we introduce the updating rules for each variable in details.

Update D. Let Ψ_D be the Lagrange multiplier for constraint $\mathbf{D} \geq 0$, the Lagrange function related to $\mathbf{D}$ is,

$$\min_{\mathbf{D}} ||\mathbf{X} - \mathbf{D} \mathbf{V}^T||_F^2 + \beta \text{tr}(\mathbf{H}^T \mathbf{L} \mathbf{H}) + \gamma ||\mathbf{e} \odot (\bar{\mathbf{B}} \mathbf{D} \mathbf{q} - \mathbf{o})||_2^2$$

$$+ \eta ||\mathbf{D}_L \mathbf{p} - \mathbf{y}_L||_2^2 + \lambda ||\mathbf{D}||_F^2 - \text{tr}(\Psi_D \mathbf{D}^T) \tag{12}$$

and $\mathbf{D} = [\mathbf{D}_L; \mathbf{D}_U]$ and $\mathbf{H} = [\mathbf{U}; \mathbf{D}_L]$. We rewrite $\mathbf{L} = [\mathbf{L}_{11}, \mathbf{L}_{12}; \mathbf{L}_{21}, \mathbf{L}_{22}]$, where $\mathbf{L}_{11} \in \mathbb{R}^{m \times m}$, $\mathbf{L}_{12} \in \mathbb{R}^{m \times r}$, $\mathbf{L}_{21} \in \mathbb{R}^{r \times m}$, and $\mathbf{L}_{22} \in \mathbb{R}^{r \times r}$; and $\mathbf{X} = [\mathbf{X}_L, \mathbf{X}_U]$. The partial derivative of $\mathcal{L}$ w.r.t. $\mathbf{D}$ as follows,

$$\frac{1}{2} \frac{\partial \mathcal{L}}{\partial \mathbf{D}} = (\mathbf{D} \mathbf{V}^T - \mathbf{X}) \mathbf{V} + \lambda \mathbf{D} + \gamma \bar{\mathbf{B}}^T \mathbf{E}^T (\mathbf{E} \bar{\mathbf{B}} \mathbf{D} \mathbf{q} - \mathbf{E} \mathbf{o}) \mathbf{q}^T$$

$$+ [\beta \mathbf{L}_{21} \mathbf{U} + \beta \mathbf{L}_{22} \mathbf{D}_L + \eta (\mathbf{D}_L \mathbf{p} - \mathbf{y}_L) \mathbf{p}^T; \mathbf{0}] - \Psi_D \tag{13}$$

where $\mathbf{E} \in \mathbb{R}^{l \times l}$ is a diagonal matrix with $\{\mathbf{e}_k\}_{k=1}^{l}$ on the diagonal and zeros everywhere else. By setting the derivative to zero and using Karush-KuhnTucker complementary condition [3], i.e., $\Psi_D(i,j) \mathbf{D}_{ij} = 0$,we get,

$$\mathbf{D}_{ij} \leftarrow \mathbf{D}_{ij} \sqrt{\frac{\hat{\mathbf{D}}(i,j)}{\tilde{\mathbf{D}}(i,j)}} \tag{14}$$

$$\hat{D} = XV + \gamma(\bar{B}^T E^T Eoq^T)^+ + \gamma(\bar{B}^T E^T E\bar{B}Dqq^T)^-$$
$$+ \left[\eta(D_L pp^T)^- + \eta(y_L p^T)^+ + \beta(L_{21}U)^- + \beta(L_{22}D_L)^-; 0\right]$$
$$\tilde{D} = DV^T V + \lambda D + \gamma(B^T E^T E\bar{B}Dqq^T)^+ + \gamma(\bar{B}^T E^T Eoq^T)^-$$
$$+ \left[\beta(L_{21}U)^+ + \beta(L_{22}D_L)^+ + \eta(D_L pp^T)^+ + \eta(y_L p^T)^-; 0\right] \tag{15}$$

where for any matrix X, $(X)^+$ and $(X)^-$ denote the positive and negative parts of X, respectively. Specifically, we have $(X)^+ = \frac{ABS(X)+X}{2}$ and $(X)^- = \frac{ABS(X)-X}{2}$, $ABS(X)$ is the matrix with the absolute value of elements in X.

Update U, V and T. The partial derivative of the Lagrange objective function w.r.t. U and updating rule are as follows,

$$\frac{1}{2}\frac{\partial \mathcal{L}}{\partial U} = \alpha(Y \odot (UTU^T - A))UT^T + \alpha(Y \odot (UTU^T - A))^T UT$$
$$+ \lambda U - \Psi_U + \beta(L_{11}U + L_{12}D_L) \tag{16}$$

$$U_{ij} \leftarrow U_{ij}\sqrt{\frac{[\hat{U}](i,j)}{[\tilde{U}](i,j)}} \tag{17}$$

$$\hat{U} = \alpha(Y \odot A)UT^T + \alpha(Y \odot A)^T UT + \beta(L_{11}U)^- + \beta(L_{12}D_L)^-$$
$$\tilde{U} = \alpha(Y \odot UTU^T)UT^T + \alpha(Y \odot UTU^T)^T UT + \lambda U$$
$$+ \beta(L_{11}U)^+ + \beta(L_{12}D_L)^+ \tag{18}$$

The partial derivatives of the Lagrange objective w.r.t V and updating rule are,

$$\frac{1}{2}\frac{\partial \mathcal{L}}{\partial V} = (DV^T - X)^T D + \lambda V - \Psi_V \tag{19}$$

$$V_{ij} \leftarrow V_{ij}\sqrt{\frac{[X^T D](i,j)}{[VD^T D + \lambda V](i,j)}} \tag{20}$$

The partial derivative of the Lagrange objective w.r.t T and the updating rule are,

$$\frac{1}{2}\frac{\partial \mathcal{L}}{\partial T} = \alpha U^T(Y \odot (UTU^T - A))U + \lambda T - \Psi_T \tag{21}$$

$$T_{ij} \leftarrow T_{ij}\sqrt{\frac{[\alpha U^T(Y \odot A)U](i,j)}{[\alpha U^T(Y \odot UTU^T)U + \lambda T](i,j)}} \tag{22}$$

Update p and q. Optimization w.r.t p and q are essentially least square problems. By setting $\frac{\partial \mathcal{L}}{\partial p} = 0$ and $\frac{\partial \mathcal{L}}{\partial q} = 0$, the closed from solutions of p and q are as follows,

$$p = (\eta D_L^T D_L + \lambda I)^{-1}\eta D_L^T y_L$$
$$q = (\gamma D^T \bar{B}^T E\bar{B}D + \lambda I)^{-1}\gamma D^T \bar{B}^T Eo \tag{23}$$

Where I is an identity matrix, and $E \in \mathbb{R}^{l \times l}$ with $e_k, k = 1, \ldots, l$ on the diagonal and zeros everywhere else.

4.1 Optimization Algorithm of TriFN

We present the details to optimize TriFN in Algorithm 1. We first randomly initialize U, V, T, D, p, q in line 1, and construct the Laplacian matrix L in line 2. Then we repeatedly update related parameters through Line 4 to Line 8 until convergence. Finally, we predict the labels of unlabeled news y_U in line 10. The convergence of Algorithm 1 is guaranteed because the objective function is nonnegative

---

Algorithm 1 The optimization process of TriFN framework

Require: $X, A, B, W, Y, o, y_L, \alpha, \beta, \gamma, \lambda, \eta$
Ensure: y_U
1: Randomly initialize U, V, T, D, p, q
2: Precompute Laplacian matrix L
3: **repeat**
4:     Update D with Eqn 14
5:     Update U with Eqn 18
6:     Update V with Eqn 20
7:     Update T with Eqn 22
8:     Update p, q with Eqn 23
9: **until** convergence
10: Calculate $y_U = \text{Sign}(D_U p)$

---

and in each iteration it will monotonically decrease the objective value, and finally it will converge to an optimal point [15].

4.2 Complexity Analysis

The main computation cost comes from the fine-tuning variables for Algorithm 1. In each iteration, the time complexity for computing D is $O(nd + nld^2 + rd + rm + n^2)$. Similarly, the computation cost for V is approximately $O(tnd)$, for U is $O(m^4 d^3 + md)$, for T is about $O(m^4 d^3 + m^2 d^2)$. To update p and q, the costs are approximately $O(d^3 + d^2 + dr)$ and $O(d^2 ln + d^3 + dl)$. The overall time complexity is the sum of the costs of initialization and fine-tuning.

5 EXPERIMENTS

In this section, we present the experiments to evaluate the effectiveness of the proposed TriFN framework. Specifically, we aim to answer the following research questions:

- Is TriFN able to improve fake news classification performance by modeling publisher partisan and user engagements simultaneously?
- How effective are publisher partisan bias modeling and user engagement learning, respectively, in improving the fake news detection performance of TriFN?
- Can the proposed method handle early fake news detection when limited user engagements are provided?

5.1 Datasets

Table 1: The statistics of FakeNewsNet dataset

Platform	BuzzFeed	PolitiFact
# Users	15,257	23,865
# Engagements	25,240	37,259
# Social Links	634,750	574,744
# Candidate news	182	240
# True news	91	120
# Fake news	91	120
# Publisher	9	91

We utilize one of the comprehensive fake news detection benchmark dataset called FakeNewsNet [31, 32]. The dataset is collected

from two platforms with fact-checking: *BuzzFeed* and *PolitiFact*, both containing news content with labels and social context information. News content includes the meta attributes of the news (e.g., body text), and social context includes the related user social engagements of news items (e.g., user posting/sharing news in Twitter). The detailed statistics of the datasets are shown in Table 1.

5.2 Experimental Settings

To evaluate the performance of fake news detection algorithms, we use the following metrics, which are commonly used to evaluate classifiers in related areas: Accuracy, Precision, Recall, and F1. We randomly choose 80% of news pieces for training and remaining 20% for testing, and the process is performed for 10 times and the average performance is reported. We compare the proposed framework TriFN with several state-of-the-art fake news detection methods. Existing methods mainly focus on extracting*discriminative features* and feed them into a classification algorithm to differentiate fake news. Next, we introduce several representative features as follows,

- **RST** [26]: RST stands for Rhetorical Structure Theory, which builds a tree structure to represent rhetorical relations among the words in the text. RST can extract style-based features of news by mapping the frequencies of rhetorical relations to a vector space [4].
- **LIWC** [23]: LIWC stands for Linguistic Inquiry and Word Count, which is widely used to extract the lexicons falling into psycholinguistic categories. It's based on a large sets of words that represent psycholinguistic processes, summary categories, and part-of-speech categories. It learns a feature vector from a psychology and deception perspective [5].
- **Castillo** [4]: Castillo extract various kinds of features from those users who have shared a news item on social media. The features are extracted from user profiles and friendship network. We also include the credibility score of users inferred in Sec 3.3 as an additional social context feature.
- **RST+Castillo**: RST+Castillo represents the concatenated features of RST and Castillo, which include features extracted from both news content and social context.
- **LIWC+Castillo**: LIWC+Castillo represents the concatenated features of LIWC and Castillo, which consists of feature information from both news content and social context.

Note that for a fair and comprehensive comparison, we choose the above feature extraction methods from following aspects: 1) only extract features from **news contents**, such as RST, LIWC; 2) only construct features from **social context**, such as Castillo; and 3) consider both **news content and social context**, such as RST+Castillo, LIWC+Castillo

5.3 Performance Comparison

We evaluate the effectiveness of the proposed framework TriFN for fake news classification. We determine model parameters with cross-validation strategy, and we repeat the generating process of training/test set for three times and the average performance is reported. We first perform cross validation on parameters $\lambda \in$

$\{0.001, 0.01, 0.1, 1, 10\}$, and choose those parameters that achieves best performance, i.e., $\lambda = 0.1$. We also choose latent dimension $d = 10$ for easy parameter tuning, and focus on the parameters that contribute the tri-relationship modeling components. The parameters for TriFN are set as $\{\alpha = 1e - 4, \beta = 1e - 5, \gamma = 1, \eta = 1\}$ for BuzzFeed and $\{\alpha = 1e - 5, \beta = 1e - 4, \gamma = 10, \eta = 1\}$ for PolitiFact.

We test the baseline features on different learning algorithms, and choose the one that achieves the best performance (see Table 2). The algorithms include Logistic Regression (LogReg for short), Naïve Bayes (NBayes), Decision Tree (DTree), Random Forest (RForest), XGBoost, AdaBoost, and Gradient Boosting (GradBoost). We used the open-sourced *xgboost* [5] package and *scikit-learn* [22] machine learning framework in Python to implement all these algorithms. To ensure a fair comparison of features, we ran all the algorithms using default parameter settings. We also show the performances for each learning algorithm and report the average performance on both datasets. Due to the space limitation, we only show the results of F1 score (Table 3 and Table 4). We observe similar results for other metrics in terms of average performance. Based on Table 2, Table 3, and Table 4, we have following observations:

- For news content based methods RST and LIWC, we can see that $LIWC > RST$ for both best performance and average performance, indicating that LIWC can better capture the linguistic features in news contents. The good results of LIWC demonstrate that fake news pieces are very different from real news in terms of choosing the words that reveal psychometrics characteristics.
- In addition, social context based features are more effective than news content based features, i.e., $Castillo > RST$ and $Castillo > LIWC$. It shows that social context features have more discriminative power than those only on news content for predicting fake news.
- Moreover, methods using both news contents and social context perform better than those methods purely based on news contents, and those methods only based on social engagements, i.e., $LIWC + Castillo > LIWC$ or $Castillo$ and $RST + Castillo > RST$ or $Castillo$. This indicates that features extracted from news content and corresponding social context have complementary information, and thus boost the detection performance.
- Generally, for methods based on both news content and social context (i.e., RST+Castillo, LIWC+Castillo, and TriFN), we can see that TriFN consistently outperforms the other two baselines, i.e., $TriFN > LIWC + Castillo$ and $TriFN > RST + Castillo$, in terms of all evaluation metrics on both datasets. For example, TriFN achieves average relative improvement of 4.72%, 5.84% on BuzzFeed and 5.91%, 4.39% on PolitiFact, comparing with LIWC+Castillo in terms of *Accuracy* and *F1* score. It supports the importance to model tri-relationship of publisher-news and news-user to better predict fake news.

5.4 Assessing Impacts of Users and Publishers

In previous section, we observe that TriFN framework improves the classification results significantly. In addition to news contents, we also captures user-news interactions and publisher-news relations.

---

[4]The code is available at: https://github.com/jiyfeng/DPLP
[5]The readers can find more details about the software and feature description at: http://liwc.wpengine.com/

Table 2: Best performance comparison for fake news detection

Datasets	Metric	RST	LIWC	Castillo	RST+Castillo	LIWC+Castillo	TriFN
BuzzFeed	Accuracy	0.600 ± 0.063	0.719 ± 0.074	0.800 ± 0.037	0.816 ± 0.052	0.825 ± 0.052	**0.864 ± 0.026**
	Precision	0.662 ± 0.109	0.722 ± 0.077	0.822 ± 0.077	0.879 ± 0.049	0.821 ± 0.061	**0.849 ± 0.040**
	Recall	0.615 ± 0.018	0.732 ± 0.171	0.776 ± 0.027	0.748 ± 0.098	0.829 ± 0.055	**0.893 ± 0.013**
	F1	0.633 ± 0.056	0.709 ± 0.075	0.797 ± 0.044	0.805 ± 0.066	0.822 ± 0.035	**0.870 ± 0.019**
PolitiFact	Accuracy	0.604 ± 0.060	0.688 ± 0.063	0.796 ± 0.052	0.838 ± 0.036	0.829 ± 0.052	**0.878 ± 0.017**
	Precision	0.564 ± 0.064	0.725 ± 0.087	0.767 ± 0.056	0.851 ± 0.052	0.821 ± 0.116	**0.867 ± 0.034**
	Recall	0.705 ± 0.148	0.617 ± 0.100	0.889 ± 0.044	0.824 ± 0.063	0.879 ± 0.047	**0.893 ± 0.023**
	F1	0.615 ± 0.074	0.666 ± 0.092	0.822 ± 0.037	0.835 ± 0.043	0.843 ± 0.054	**0.880 ± 0.015**

Table 3: Average F1 of baselines for different learning algorithms on BuzzFeed. Best scores are highlighted.

Method	RST	LIWC	Castillo	RST +Castillo	LIWC +Castillo
LogReg	0.519	0.660	0.714	0.728	0.760
NBayes	0.511	0.370	0.600	0.716	0.680
DTree	0.566	0.581	0.736	0.681	0.772
RForest	0.538	0.709	0.767	0.805	0.733
XGBoost	0.480	0.672	0.797	0.795	0.782
AdaBoost	0.633	0.701	0.724	0.791	0.768
GradBoost	0.492	0.699	0.772	0.724	0.822

Table 4: Average F1 of baselines for different learning algorithms on PolitiFact. Best scores are highlighted.

Method	RST	LIWC	Castillo	RST +Castillo	LIWC +Castillo
LogReg	0.615	0.432	0.707	0.668	0.653
NBayes	0.537	0.486	0.442	0.746	0.687
DTree	0.514	0.661	0.771	0.792	0.772
RForest	0.463	0.586	0.767	0.835	0.836
XGBoost	0.552	0.648	0.822	0.783	0.823
AdaBoost	0.502	0.666	0.800	0.787	0.831
GradBoost	0.517	0.650	0.818	0.803	0.843

Now, we investigate the effects of these components by defining three variants of TriFN:

- TriFN\P - We eliminate the effect of publisher partisan modeling part $\gamma \|\mathbf{e} \odot (\bar{\mathbf{B}}\mathbf{D}\mathbf{q} - \mathbf{o})\|_2^2$ by setting $\gamma = 0$.
- TriFN\S - We eliminate the effects of user social engagements components $\alpha \|\mathbf{Y} \odot (\mathbf{A} - \mathbf{U}\mathbf{T}\mathbf{U}^T)\|_F^2 + \beta \mathrm{tr}(\mathbf{H}^T \mathbf{L}\mathbf{H})$ by setting $\alpha, \beta = 0$.
- TriFN\PS - We eliminate the effects of both publisher partisan and user social engagements, by setting $\alpha, \beta, \gamma = 0$. The model only consider news content embedding.

The parameters in all the variants are determined with cross-validation and the best performances are shown in Figure 3, we have following observations:

- When we eliminate the effect of user social engagements component , the performance of TriFN\S degrades in comparison with TriFN. For example, the performance reduces 5.2% and 6.1% in terms of F1 and Accuracy metrics on BuzzFeed, 7.6% and 10.6% on PolitiFact. The results suggest that social engagements in TriFN is important.

(a) BuzzFeed   (b) PolitiFact
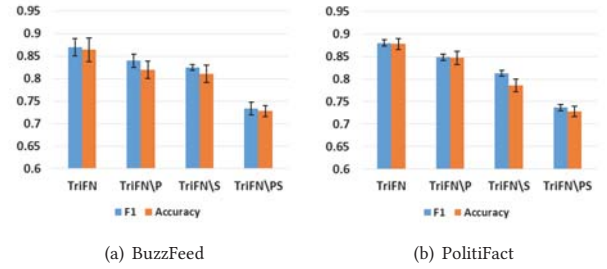
Figure 3: Impact analysis of users and publishers for fake news detection.

- We have similar observations for TriFN\P when eliminating the effect of publisher partisan component. The results suggest the importance to consider publisher-news relations through publisher partisan bias in TriFN.
- When we eliminate both components in TriFN\PS, the results are further reduced compared to TriFN\S and TriFN\P. It also suggests that components of user-news and publisher-news embedding are complementary to each other.

Through the component analysis of TriFN, we conclude that (i) both components can contribute to the performance improvement of TriFN; (ii) it's necessary to model both news contents and social engagements because they contain complementary information.

5.5 Early Fake News Detection

Early detection of fake news is very desirable to restrict the dissemination scope of fake news and prevent the future propagation on social media. Early fake news detection aims to give early alert of fake news, by only considering limited social context within a specific range of time delay of original news posted. Specifically, we change the delay time in [12, 24, 36, 48, 60, 72, 84, 96] hours. From Figure 4, we can see that: 1) generally, the detection performance is getting better when the delay time increase for those methods using social context information, which indicates that more social engagements of users on social media provide more additional information for fake news detection; 2) The proposed TriFN consistently achieves best performances on both datasets for accuracy and F1, which demonstrate the importance of embedding user-news interactions to capture effective feature representations; and 3) Even in the very early stage after fake news has been published, TriFN can already achieve good performance. For example, TriFN can achieve

F1 score more than 80% within 48 hours on both datasets, which shows promising potentials to combat fake new at the early stage.
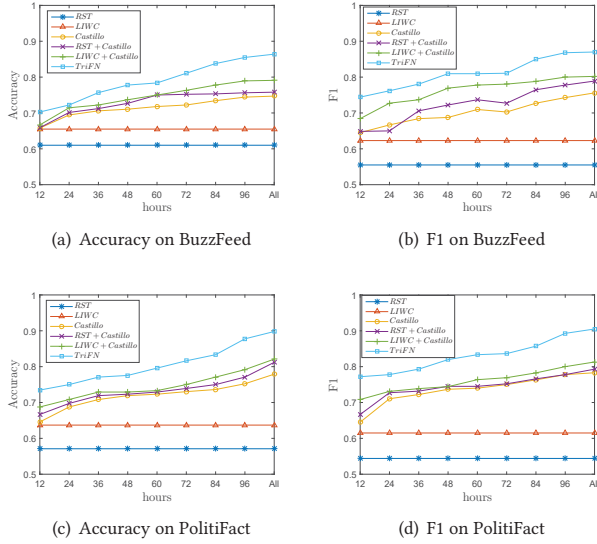

(a) Accuracy on BuzzFeed

(b) F1 on BuzzFeed

(c) Accuracy on PolitiFact

(d) F1 on PolitiFact

Figure 4: The performance of early fake news detection on BuzzFeed and PolitiFact in terms of Accuracy and F1.

5.6 Model Parameter Analysis

The proposed TriFN has four important parameters. The first two are α and β, which control the contributions from social relationship and user-news engagements. γ controls the contribution of publisher partisan and η controls the contribution of semi-supervised classifier. We first fix $\{\alpha = 1e-4, \beta = 1e-5\}$ and $\{\alpha = 1e-5, \beta = 1e-4\}$ for BuzzFeed and PolitiFact, respectively. Then we vary η as $\{1, 10, 20, 50, 100\}$ and γ in $\{1, 10, 20, 30, 100\}$. The performance variations are depicted in Figure 5. We can see i) when η increases from 0, eliminating the impact of semi-supervised classification term, to 1, the performance increase dramatically in both datasets. These results support the importance to combine semi-supervised classifier to feature learning; ii) generally, the increase of γ will increase the performance in a certain region, $\gamma \in [1, 50]$ and $\eta \in [1, 50]$ for both datasets, which easy the process for parameter setting. Next, we fix $\{\gamma = 1, \eta = 1\}$ and $\{\gamma = 10, \eta = 1\}$ for BuzzFeed and PolitiFact, respectively. Then we vary $\alpha, \beta \in [0, 1e-5, 1e-4, 1e-3, 0.001, 0.01]$. We can see that i) when α and β increase from 0, which eliminate the social engagements, to $1e-5$, the performance increases relatively, which again support the importance of social engagements; ii) The performance tends to increase first and then decrease, and it's relatively stable in $[1e-5, 1e-3]$.

6 RELATED WORK

We briefly introduce the related work about fake news detection on social media. Fake news detection methods generally focus on using *news contents* and *social contexts* [32, 40].

News contents contain the clues to differentiate fake and real news. For news content based approaches, features are extracted as linguistic-based and visual-based. Linguistic-based features capture
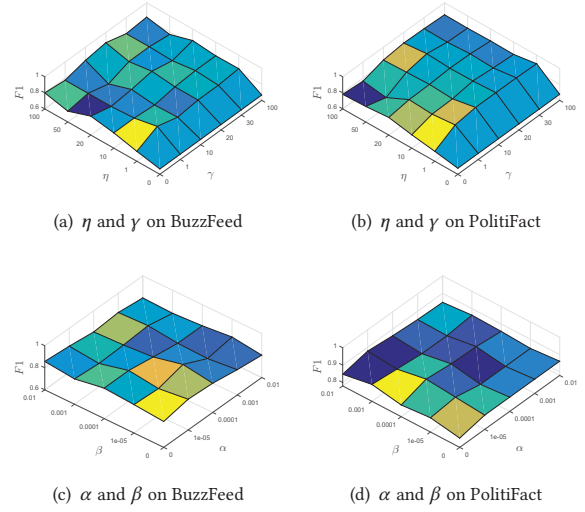

(a) η and γ on BuzzFeed

(b) η and γ on PolitiFact

(c) α and β on BuzzFeed

(d) α and β on PolitiFact

Figure 5: Model parameter analysis for TriFN on BuzzFeed and PolitiFact in terms of F1.

specific writing styles and sensational headlines that commonly occur in fake news content [24], such as lexical and syntactic features. Visual-based features try to identify fake images [9] that are intentionally created or capturing specific characteristics for images in fake news. News content based models include i) knowledge-based: using external sources to fact-checking claims in news content [17, 37], and 2) style-based: capturing the manipulators in writing style, such as deception [7, 27] and non-objectivity [24]. For example, Potthast *et al.* [24] extracted various style features from news contents and predict fake news and media bias.

In addition to news content, social context related to news pieces contains rich information to help detect fake news. For social context based approaches, the features mainly include user-based, post-based and network-based. User-based features are extracted from user profiles to measure their characteristics and credibilities [4, 14, 34, 39]. For example, Shu *et al.* [34] proposed to understand user profiles from various aspects to differentiate fake news. Yang *et al.* [39] proposed an unsupervised fake news detection algorithm by utilizing users' opinions on social media and estimating their credibilities. Post-based features represent users' social response in term of stance [10], topics [16], or credibility [4, 36]. Network-based features [29] are extracted by constructing specific networks, such as diffusion network [14] etc. Social context models basically include stance-based and propagation-based. Stance-based models utilize users' opinions towards the news to infer news veracity [10]. Propagation-based models assume that the credibility of news is highly related to the credibilities of relevant social media posts, which several propagation methods can be applied [10]. Recently, deep learning models are applied to learn the temporal and linguistic representation of news [11, 30, 35]. Shu *et al.* [33] proposed to generate synthetic data for augmenting training data to help improve the detection of clickbaits. It's worth mentioning that we can not directly compare the propagation-based approaches, because we assume we only have user actions, e.g., posting the

news or not. In this case, the propagation signals inferred from text are the same and thus become ineffective.

In this paper, we are to our best knowledge the first to classify fake news by learning the effective news features through the tri-relationship embedding among publishers, news contents, and social engagements.

7 CONCLUSION AND FUTURE WORK

Due to the inherent relationship among publisher, news and social engagements during news dissemination process on social media, we propose a novel framework TriFN to model tri-relationship for fake news detection. TriFN can extract effective features from news publisher and user engagements separately, as well as capture the interrelationship simultaneously. Experimental results on real world fake news datasets demonstrate the effectiveness of the proposed framework and importance of tri-relationship for fake news prediction. It's worth mentioning TriFN can achieve good detection performance in the early stage of news dissemination.

There are several interesting future directions. First, it's worth to explore effective features and models for early fake news detection, as fake news usually evolves very fast on social media; Second, how to extract features to model fake news intention from psychology's perspective needs further investigation. At last, how to identify low quality or even malicious users spreading fake news is important for fake news intervention and mitigation.

8 ACKOWLEDGMENTS

REFERENCES

[1] Mohammad Ali Abbasi and Huan Liu. 2013. Measuring User Credibility in Social Media.. In *SBP*. Springer, 441–448.
[2] Hunt Allcott and Matthew Gentzkow. 2017. *Social media and fake news in the 2016 election.* Technical Report. National Bureau of Economic Research.
[3] Stephen Boyd and Lieven Vandenberghe. 2004. *Convex optimization.* Cambridge university press.
[4] Carlos Castillo, Marcelo Mendoza, and Barbara Poblete. 2011. Information credibility on twitter. In *Proceedings of the 20th international conference on World wide web.* ACM, 675–684.
[5] Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining.* ACM, 785–794.
[6] Robert M Entman. 2007. Framing bias: Media in the distribution of power. *Journal of communication* 57, 1 (2007), 163–173.
[7] Song Feng, Ritwik Banerjee, and Yejin Choi. 2012. Syntactic stylometry for deception detection. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2.* Association for Computational Linguistics, 171–175.
[8] Matthew Gentzkow, Jesse M Shapiro, and Daniel F Stone. 2014. *Media bias in the marketplace: Theory.* Technical Report. National Bureau of Economic Research.
[9] Aditi Gupta, Hemank Lamba, Ponnurangam Kumaraguru, and Anupam Joshi. 2013. Faking sandy: characterizing and identifying fake images on twitter during hurricane sandy. In *Proceedings of the 22nd international conference on World Wide Web.* ACM, 729–736.
[10] Zhiwei Jin, Juan Cao, Yongdong Zhang, and Jiebo Luo. 2016. News Verification by Exploiting Conflicting Social Viewpoints in Microblogs.. In *AAAI.* 2972–2978.
[11] Hamid Karimi, Proteek Roy, Sari Saba-Sadiya, and Jiliang Tang. Multi-Source Multi-Class Fake News Detection. In *COLING'18.*
[12] Antino Kim and Alan R Dennis. 2017. Says Who?: How News Presentation Format Influences Perceived Believability and the Engagement Level of Social Media Users. (2017).
[13] David O Klein and Joshua R Wueller. 2017. Fake News: A Legal Perspective. (2017).
[14] Sejeong Kwon, Meeyoung Cha, Kyomin Jung, Wei Chen, and Yajun Wang. 2013. Prominent features of rumor propagation in online social media. In *Data Mining (ICDM), 2013 IEEE 13th International Conference on.* IEEE, 1103–1108.
[15] Daniel D Lee and H Sebastian Seung. 2001. Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems.* 556–562.
[16] Jing Ma, Wei Gao, Zhongyu Wei, Yueming Lu, and Kam-Fai Wong. 2015. Detect rumors using time series of social context information on microblogging websites. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management.* ACM, 1751–1754.
[17] Amr Magdy and Nayer Wanas. 2010. Web-based statistical fact checking of textual documents. In *Proceedings of the 2nd international workshop on Search and mining user-generated contents.* ACM, 103–110.
[18] Brendan Nyhan and Jason Reifler. 2010. When corrections fail: The persistence of political misperceptions. *Political Behavior* 32, 2 (2010), 303–330.
[19] Rong Pan and Martin Scholz. Mind the gaps: weighting the unknown in large-scale one-class collaborative filtering. In *KDD'09.*
[20] V Paul Pauca, Farial Shahnaz, Michael W Berry, and Robert J Plemmons. 2004. Text mining using non-negative matrix factorizations. In *Proceedings of the 2004 SIAM International Conference on Data Mining.* SIAM, 452–456.
[21] Christopher Paul and Miriam Matthews. 2016. The Russian "Firehose of Falsehood" Propaganda Model. *RAND Corporation* (2016).
[22] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in Python. *Journal of machine learning research* 12, Oct (2011), 2825–2830.
[23] James W Pennebaker, Ryan L Boyd, Kayla Jordan, and Kate Blackburn. 2015. *The development and psychometric properties of LIWC2015.* Technical Report.
[24] Martin Potthast, Johannes Kiesel, Kevin Reinartz, Janek Bevendorff, and Benno Stein. 2017. A Stylometric Inquiry into Hyperpartisan and Fake News. *arXiv preprint arXiv:1702.05638* (2017).
[25] Walter Quattrociocchi, Antonio Scala, and Cass R Sunstein. 2016. Echo chambers on facebook. (2016).
[26] Victoria L Rubin, N Conroy, and Yimin Chen. 2015. Towards news verification: Deception detection methods for news discourse. In *Hawaii International Conference on System Sciences.*
[27] Victoria L Rubin and Tatiana Lukoianova. 2015. Truth and deception at the rhetorical structure level. *Journal of the Association for Information Science and Technology* 66, 5 (2015), 905–917.
[28] Farial Shahnaz, Michael W Berry, V Paul Pauca, and Robert J Plemmons. 2006. Document clustering using nonnegative matrix factorization. *Information Processing & Management* 42, 2 (2006), 373–386.
[29] Kai Shu, H. Russell Bernard, and Huan Liu. 2018. Studying Fake News via Network Analysis: Detection and Mitigation. *CoRR* abs/1804.10233 (2018).
[30] Kai Shu, Deepak Mahudeswaran, and Huan Liu. 2018. FakeNewsTracker: a tool for fake news collection, detection, and visualization. *Computational and Mathematical Organization Theory* (2018), 1–12.
[31] Kai Shu, Deepak Mahudeswaran, Suhang Wang, Dongwon Lee, and Huan Liu. 2018. FakeNewsNet: A Data Repository with News Content, Social Context and Dynamic Information for Studying Fake News on Social Media. *arXiv preprint arXiv:1809.01286* (2018).
[32] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake News Detection on Social Media: A Data Mining Perspective. *KDD exploration newsletter* (2017).
[33] Kai Shu, Suhang Wang, Thai Le, Dongwon Lee, and Huan Liu. Deep Headline Generation for Clickbait Detection.. In *ICDM'18.*
[34] Kai Shu, Suhang Wang, and Huan Liu. 2018. Understanding User Profiles on Social Media for Fake News Detection. In *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR).* IEEE.
[35] Yaqing Wang, Fenglong Ma, Zhiwei Jin, Ye Yuan, Guangxu Xun, Kishlay Jha, Lu Su, and Jing Gao. EANN: Event Adversarial Neural Networks for Multi-Modal Fake News Detection. In *KDD'18.*
[36] Liang Wu and Huan Liu. Tracing fake-news footprints: Characterizing social media messages by how they propagate. In *WSDM'18.*
[37] You Wu, Pankaj K Agarwal, Chengkai Li, Jun Yang, and Cong Yu. 2014. Toward computational fact-checking. *Proceedings of the VLDB Endowment* 7, 7 (2014), 589–600.
[38] Wei Xu, Xin Liu, and Yihong Gong. 2003. Document clustering based on non-negative matrix factorization. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval.* ACM, 267–273.
[39] Shuo Yang, Kai Shu, Suhang Wang, Renjie Gu, Fan Wu, and Huan Liu. Unsupervised Fake News Detection on Social Media: A Generative Approach. In *AAAI'19.*
[40] Xinyi Zhou, Reza Zafarani, Kai Shu, and Huan Liu. Fake News: Fundamental Theories, Detection Strategies and Challenges. In *WSDM'19.*