Revisiting Fake News Detection: Towards Temporality-aware Evaluation by Leveraging Engagement Earliness

Junghoon Kim*
KAIST
Daejeon, Republic of Korea
jhkim611@kaist.ac.kr

Junmo Lee*
KAIST
Daejeon, Republic of Korea
bubblego0217@kaist.ac.kr

Yeonjun In
KAIST
Daejeon, Republic of Korea
yeonjun.in@kaist.ac.kr

Kanghoon Yoon
KAIST
Daejeon, Republic of Korea
ykhoon08@kaist.ac.kr

Chanyoung Park†
KAIST
Daejeon, Republic of Korea
cy.park@kaist.ac.kr

Abstract

Social graph-based fake news detection aims to identify news articles containing false information by utilizing social contexts, e.g., user information, tweets and comments. However, conventional methods are evaluated under less realistic scenarios, where the model has access to future knowledge on article-related and context-related data during training. In this work, we newly formalize a more realistic evaluation scheme that mimics real-world scenarios, where the data is *temporality-aware* and the detection model can only be trained on data collected up to a certain point in time. We show that the discriminative capabilities of conventional methods decrease sharply under this new setting, and further propose DAWN, a method more applicable to such scenarios. Our empirical findings indicate that later engagements (e.g., consuming or reposting news) contribute more to noisy edges that link real news-fake news pairs in the social graph. Motivated by this, we utilize feature representations of engagement earliness to guide an edge weight estimator to suppress the weights of such noisy edges, thereby enhancing the detection performance of DAWN. Through extensive experiments, we demonstrate that DAWN outperforms existing fake news detection methods under real-world environments[1]. The source code is available **here**.

CCS Concepts

- **Information systems** → **Social networks**;
- **Computing methodologies** → **Artificial intelligence**.

Keywords

Fake News Detection, Social Network Analysis, Graph Neural Networks, Graph Structure Learning

ACM Reference Format:
Junghoon Kim, Junmo Lee, Yeonjun In, Kanghoon Yoon, and Chanyoung Park. 2025. Revisiting Fake News Detection: Towards Temporality-aware Evaluation by Leveraging Engagement Earliness. In *Proc. of the Eighteenth ACM International Conference on Web Search and Data Mining (WSDM '25), March 10–14, 2025, Hannover, Germany*. ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3701551.3703524

1 Introduction

The advent of social media platforms has made it possible for news to travel to a vast amount of people, allowing for greater accessibility and efficiency when obtaining information. However, the propagation of news articles containing false information has also become much easier. The spreading of such fake news has a detrimental effect on societal security and public health, ranging from potentially affecting presidential election results [1] to inciting distrust and panic during a pandemic [2, 19]. As such, the field of *fake news detection*, which aims to identify fabricated news articles, has gained increasing attention and importance in recent years [28, 36].

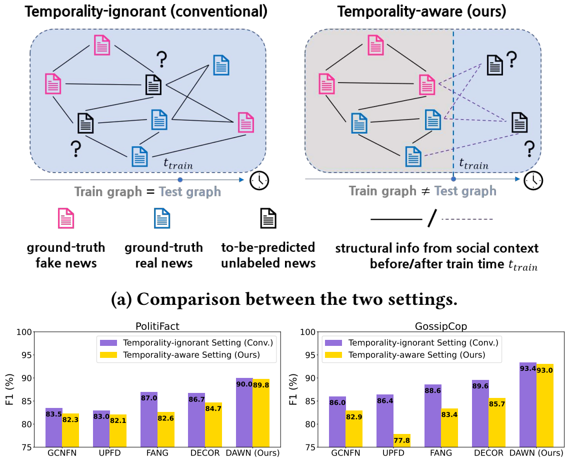
Recent fake news detection methods can be classified into two main categories. *Content-based* methods leverage patterns that can be obtained from the news article text itself, such as semantic representations or emotional features [6, 9, 10, 25]. On the other hand, *social graph-based* methods utilize social context knowledge, including user information, tweets and comments, in addition to textual contents [7, 22, 32, 33]. Specifically, social context knowledge can be beneficial for fake news detection since retweets on news pieces can reveal different propagation patterns for real and fake news, while user interactions and responses indicate how these types of news attract different user groups. Existing approaches model such contexts into graph structures, e.g., linking news article pairs based on the amount of tweets shared by readers. By utilizing Graph Neural Networks (GNNs) to model the relationship between news veracity and the structural patterns involved in rich social information, social graph-based methods often outperform their content-based counterparts and appear to be a promising area of research [26, 32].

However, we point out that conventional social graph-based methods are trained and evaluated under an unrealistic scenario [7, 32]. Specifically, they divide news articles into training and test sets via *random split* or use social contexts that occur *after* the training time for training the model. Such *temporality-ignorant* settings (See Fig. 1(a) left) would lead to information leakage as the model can access future knowledge not only on *article-related data* (e.g., textual

*Both authors contributed equally to this research.
†Corresponding author.
[1] An extended, more comprehensive version of this paper can be found **here**.

This work is licensed under a Creative Commons Attribution International 4.0 License.

WSDM '25, March 10–14, 2025, Hannover, Germany
© 2025 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-1329-3/25/03
https://doi.org/10.1145/3701551.3703524



(a) Comparison between the two settings.

(b) Performance of existing methods evaluated under the two settings.

Figure 1: (a) Conventional settings (left) for social graph-based fake news detection utilize future information and social context during training, as opposed to a more realistic temporality-aware setting (right). (b) Existing social graph-based fake news detection methods suffer performance degradation when applied on temporality-aware settings.

contents and veracity labels) but also *context-related data* (e.g., users, tweets and comments). We argue that this is inconsistent with real-world scenarios, where the detection model would be trained on offline data collected in advance, and then tested on online data delivered in real-time. Thus, it is more practical for the model to be trained by only utilizing data up to a certain point in time, while future data should only be available at test time, which we call a *temporality-aware* setting (See Fig. 1(a) right).

In Fig. 1(b), we examine the performance of existing social graph-based fake news detection methods (e.g., GCNFN [22], UPFD [7], FANG [23] and DECOR [32]) when they are trained and evaluated under the temporality-aware setting. Our empirical results on two prominent fake news datasets [27] reveal that when all related data are split into train/test considering the temporality (yellow bar), the performance of conventional methods decrease sharply, with a decline up to 8.6%p in F1 score. Such performance drops can be attributed to the inherent design flaws of these methods; that is, they construct a graph based on the *complete* information regarding social context given in the dataset regardless of the temporal information, leading to potential information leakage, which results in a single fixed structure for both training and testing (See Fig. 1(a) left). On the other hand, in the temporality-aware setting, the structure would change substantially after training (See Fig. 1(a) right).

In this work, we revisit the training and evaluation setting of social graph-based fake news detection methods, and propose a novel method that is applicable to real-world learning environments in which the temporal information should not be overlooked. The main idea is to utilize time-independent patterns in the graph, i.e., patterns representing general social user behaviors that occur similarly even at different points in time, so that the model is less affected by how the graph is constructed. Our method, called **D**etecting fake news via e**A**rliness-guided re**W**eighti**N**g (**DAWN**), takes into account the earliness-related patterns of users and tweets. These patterns are based upon fundamental and well-explored theory on social behaviors, i.e., the confirmation bias theory [24], and thus the features extracted from them differ less before and after training, making them more robust against temporality-aware settings.

Through an extensive data analysis in Section 4, we find that users on social media who have stronger opinions and thus more rapidly engage (e.g., consume or repost) with news articles are more inclined to be attracted to articles of the same veracity, i.e., either only real news or only fake news. From these observations, we obtain some valuable insights: **(1) earlier engagements tend to connect articles of the same veracity**, while **(2) the existence of later engagements leads to a higher probability of the labels being different**, which can manifest as noisy edges in social graphs (i.e., edges linking real news-fake news pairs). Based on our findings suggesting that appropriately utilizing knowledge on *engagement earliness* can help identify the edge noise, we suppress the weights of such noisy edges aiming at enhancing the model's discriminative capabilities. To this end, we employ a Graph Structure Learning (GSL) framework, where we train an edge weight estimator to assign new weights to existing edges. By leveraging feature representations of edge-specific engagement earliness, we can successfully guide the edge weight estimator to downweight noisy edges.

Overall, our contributions can be summarized as follows:

- **A realistic scenario for fake news detection.** We argue that the assumption made by existing social graph-based methods that all relevant information is accessible during training is unrealistic, leading to potential information leakage. As such, we present a temporality-aware setting for fake news detection (both article-wise and context-wise) in which existing methods underperform owing to their inherent design flaws.
- **Earliness-related empirical findings.** We analyze how the engagement earliness regarding users and tweets in a social network relate to the veracity label consistency of news article pairs.
- **Earliness-guided GSL.** We propose a novel GSL-based fake news detection framework, called DAWN, that leverages our earliness-related insights to downweight noisy edges in a social graph.
- **Effectiveness.** DAWN outperforms existing methods by a large margin on two real-world fake news datasets, highlighting the robustness of our simple yet effective earliness-based framework under the temporality-aware training and evaluation setting.

2 Related Works

2.1 Fake News Detection

Fake news detection aims to determine whether a given news article contains false information or not, and can generally be seen as a binary classification task that predicts its *veracity label* (0 if real, 1 if fake). Among the two main categories for fake news detection research, **content-based** methods utilize patterns from within the news article text itself. Such patterns include semantic representations [6, 20], emotional features [9, 10] and writing style [10, 25].

Meanwhile, **social graph-based** methods [7, 22, 23, 32, 33] additionally incorporate various social contexts including user information, tweets and comments, and have shown state-of-the-art performance, generally improving over content-based methods. For instance, GCNFN [22] constructs propagation trees for each news article by utilizing user responses and user following relations.

FANG [23] learns the representations of a heterogeneous social graph, consisting of users, news articles and sources. DECOR [32] links news article pairs based on co-user engagement knowledge. The social graphs constructed in these works utilize the *entire* information on social context, resulting in a single fixed structure for both training and testing. In other words, they are inherently designed to leverage future contextual data for model training leading to potential information leakage, which especially aggravates when the news articles are randomly split as well.

In this paper, we emphasize the importance of simulating real-world learning environments to enhance the applicability of fake news detection methods. Specifically, as information that appears during test time would not be available at training time under real-world scenarios, such *temporality-aware* settings should be properly deployed when evaluating model performance. To the best of our knowledge, we are the first to consider ***both* article-wise (textual contents and veracity labels) and context-wise (social knowledge on users and tweets) temporality-aware settings** for social graph-based fake news detection.

2.2 Graph Structure Learning

Various downstream tasks on graphs have been shown to be successfully tackled by Graph Neural Networks (GNNs) [11, 17, 29]. However, the performance of such GNNs are vulnerable to the existence of *noisy edges* that connect dissimilar nodes [8], either through structural adversarial attacks or inherent noise within the data [4, 5, 14, 16, 30, 34]. As such, many Graph Structure Learning (GSL) studies have aimed to downweight such noisy edges by optimizing the adjacency matrix. Guided by node feature similarity, prior studies have utilized various similarity metrics and link predictor training schemes to alleviate the effect of edge noise [4, 8, 13, 15, 31].

A recent study, called DECOR [32], has attempted to apply GSL to fake news detection, and has shown that constructing edge-specific features representing social patterns is much more effective than previous node feature-based methods when training the link predictor to suppress noisy edges. Despite its effectiveness, the degree-related patterns utilized by DECOR are obtained from a single fixed social graph, i.e., the patterns are subject to substantial changes under real-world environments where the graph structure greatly differs before and after training time, hindering detection performance. Motivated by this, our work aims to exploit new features whose underlying patterns are independent of time and thus are more robust under temporality-aware settings.

3 Preliminaries

3.1 Problem Statement

Definitions. Let $\mathcal{D} = (\mathcal{P}, \mathcal{U}, \mathcal{R})$ be a fake news detection dataset. $\mathcal{P}$ is a set of questionable news articles, where each news article $p_n \in \mathcal{P}$ contains the corresponding article text and the time of its publication t_n^p. $\mathcal{U}$ is a set of active users on social media, where each user has had at least m engagements (an *engagement* occurs when a user reposts a news article). $\mathcal{R}$ is a set of such engagements, where each engagement $r_l \in \mathcal{R}$ is defined as $\{(u, p, t_l^r) | u \in \mathcal{U}, p \in \mathcal{P}\}$ (i.e., user u has reposted news article p at time t_l^r).

To properly emulate real-world scenarios where accessible information changes by time, we adopt a *temporality-aware* setting. Specifically, we define timestamps t_{train}, t_{val} and t_{test} according to how much of the data we want to include in the training, validation and test set, respectively ($t_{train} < t_{val} < t_{test}$). Then, we construct subsets from $\mathcal{P}, \mathcal{U}$ and $\mathcal{R}$. More precisely, from $\mathcal{P}$, we obtain $\mathcal{P}_{train} = \{p_n | p_n \in \mathcal{P}, t_n^p \leq t_{train}\}, \mathcal{P}_{val} = \{p_n | p_n \in \mathcal{P}, t_{train} < t_n^p \leq t_{val}\}$ and $\mathcal{P}_{test} = \{p_n | p_n \in \mathcal{P}, t_{val} < t_n^p \leq t_{test}\}$. Similarly, $\mathcal{R}_{train} = \{r_l | r_l \in \mathcal{R}, t_l^r \leq t_{train}\}, \mathcal{R}_{val} = \{r_l | r_l \in \mathcal{R}, t_l^r \leq t_{val}\}$ and $\mathcal{R}_{test} = \{r_l | r_l \in \mathcal{R}, t_l^r \leq t_{test}\}$. Accordingly, $\mathcal{U}_{train}, \mathcal{U}_{val}$ and $\mathcal{U}_{test}$ are defined so that they contain users who have at least m engagements in $\mathcal{R}_{train}, \mathcal{R}_{val}$ and $\mathcal{R}_{test}$, respectively. Finally, $\mathcal{Y}_{train}$ and $\mathcal{Y}_{val}$ contain ground-truth veracity labels (1 if fake. 0 otherwise) associated with news articles in $\mathcal{P}_{train}$ and $\mathcal{P}_{val}$, respectively.

Temporality-aware Fake News Detection. Given a news dataset $\mathcal{D} = (\mathcal{P}, \mathcal{U}, \mathcal{R})$ and timestamps t_{train}, t_{val} and t_{test}, our goal is to learn a fake news detector, i.e., binary classifier. Specifically, we first train the classifier on $(\mathcal{P}_{train}, \mathcal{U}_{train}, \mathcal{R}_{train})$ and validate on $(\mathcal{P}_{val}, \mathcal{U}_{val}, \mathcal{R}_{val})$. Then, given $(\mathcal{P}_{test}, \mathcal{U}_{test}, \mathcal{R}_{test})$, the classifier would predict the veracity labels $\mathcal{Y}_{test}$ for news articles in $\mathcal{P}_{test}$.

3.2 Social Graph Construction

Following [32, 33], we construct a social graph that effectively captures the relationship between news articles through user engagements. Specifically, we first obtain an *engagement matrix* $E \in \mathbb{R}^{|\mathcal{U}| \times |\mathcal{P}|}$, where each element E_{ij} represents the number of times user u_i has interacted with news article p_j, the value of which being the number of the associated engagements in $\mathcal{R}$. Then, we construct a graph $\mathcal{G} = (\mathcal{P}, \mathcal{A})$, where a node denotes an article and an edge denotes the co-engagement patterns between a pair of articles. Specifically, the associated adjacency matrix $\mathcal{A} \in \mathbb{R}^{|\mathcal{P}| \times |\mathcal{P}|}$ is generated by retrieving co-engagement patterns from the engagement matrix, i.e., $\mathcal{A} = E^{\top} E$. Each element $\mathcal{A}_{ij}$ represents the edge weight between a pair of news articles p_i and p_j, where a weight of zero indicates no shared users between them (i.e., none of the users have reposted both articles). If there are multiple users who have reposted multiple responses on both articles, the weight would increase accordingly. In other words, the edge weights in $\mathcal{A}$ denote the intensity of the co-engagement between article pairs.

To incorporate the realistic temporality-aware setting, we expand on the above procedure and construct three separate graphs for training, validation and testing. In detail, we obtain a training engagement matrix $E_{train} \in \mathbb{R}^{|\mathcal{U}_{train}| \times |\mathcal{P}_{train}|}$ only utilizing $(\mathcal{P}_{train}, \mathcal{U}_{train}, \mathcal{R}_{train})$. From this we construct a training graph $\mathcal{G}_{train} = (\mathcal{P}_{train}, \mathcal{A}_{train})$, where $\mathcal{A}_{train} \in \mathbb{R}^{|\mathcal{P}_{train}| \times |\mathcal{P}_{train}|}$ is defined as $\mathcal{A}_{train} = E_{train}{}^{\top} E_{train}$. Similarly, we construct validation and test graphs $\mathcal{G}_{val} = (\mathcal{P}_{val}, \mathcal{A}_{val})$ and $\mathcal{G}_{test} = (\mathcal{P}_{test}, \mathcal{A}_{test})$ from $(\mathcal{P}_{val}, \mathcal{U}_{val}, \mathcal{R}_{val})$ and $(\mathcal{P}_{test}, \mathcal{U}_{test}, \mathcal{R}_{test})$, respectively.

4 Data Analysis: Engagement Earliness and Veracity Label Consistency

In this section, we explore the relationship between engagement earliness and the veracity label consistency of news article pairs. Our analysis is performed on two prominent fake news datasets PolitiFact and GossipCop from the FakeNewsNet [27] benchmark.

Following [33], we define a Fake News Affinity (FNA) score for each user $u \in \mathcal{U}$ (we set $m = 3$ for all analysis in this section) as:

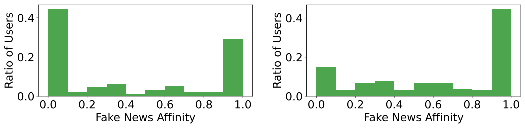
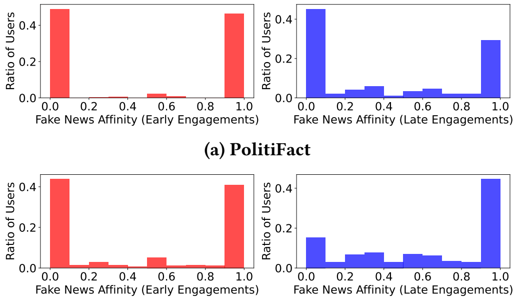


Figure 2: Users have a tendency to engage with either only fake news or real news (Left: PolitiFact, right: GossipCop).



(a) PolitiFact

(b) GossipCop

Figure 3: The skewed tendency intensifies with early engagements (red) over late engagements (blue).

$$FNA(u) = \frac{\text{\# of engagements with fake news by } u}{\text{\# of all engagements by } u}. \quad (1)$$

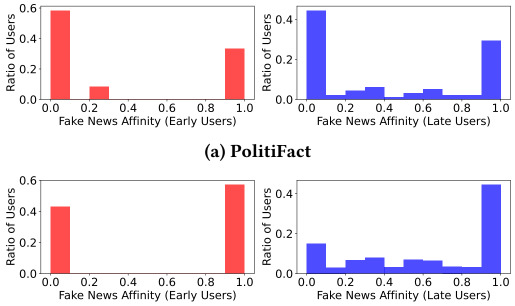
Fig. 2 shows the distribution of users' FNA scores via histograms. The results indicate that users generally have FNA scores close to 1 (i.e., only engage with fake news) or 0 (i.e., only engage with real news), implying that users tend to consume and spread news articles of the same veracity. In other words, users engaging with news articles on social media would be attracted to similar articles regarding veracity, according to their highly polarized opinions [18, 21]. This is in line with the well-known confirmation bias theory [24] stating that people have a tendency to be drawn to information that affirms and reinforces their prior beliefs and preferences.

Building upon this, we now investigate how the behaviors change in terms of *engagement earliness*. More precisely, as users with stronger opinions would generally act quicker when consuming and reposting news (i.e., engagements) that reflect their beliefs, we hypothesized that *confirmation bias would exacerbate in users displaying earlier engagement patterns*. In the following, we investigate earliness patterns in terms of each engagement in $\mathcal{R}$ and user in $\mathcal{U}$.

4.1 Engagement-wise Earliness Patterns

First, we defined a deadline t_d, where a certain engagement $r_l \in \mathcal{R}$ can be considered *early* if $t_r^l - t_n^p < t_d$, meaning the engagement occurred within a fixed timespan determined by the deadline after the corresponding news article has been posted. Engagements that occur after the deadline can be considered as *late*. We divided all engagements into two groups (early and late) according to the deadline (set to 30 minutes for PolitiFact and 5 minutes for GossipCop), then obtained the FNA scores separately within each group.

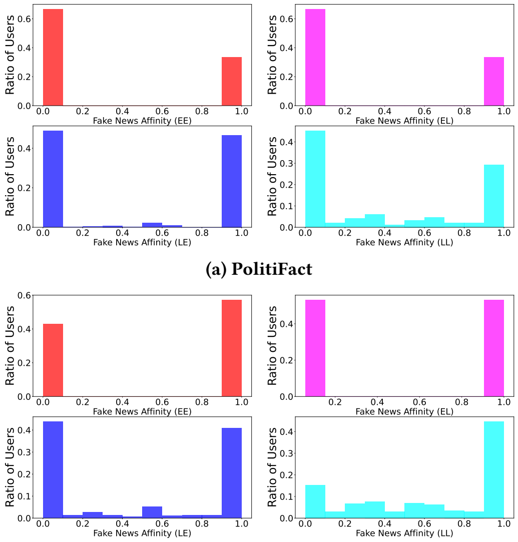
The difference in distributions are shown in Fig. 3, where the FNA scores associated with early engagements are much more skewed towards 0 or 1, compared to those associated with late engagements. In other words, we can observe that confirmation bias does indeed occur more intensely in users with "earlier engagements."



(a) PolitiFact

(b) GossipCop

Figure 4: The skewed tendency intensifies with engagements by early users (red) over late users (blue).



(a) PolitiFact

(b) GossipCop

Figure 5: The patterns can be further broken down when considering *both* engagement-wise and user-wise viewpoints.

4.2 User-wise Earliness Patterns

While our previous analysis assigned earliness categories in terms of each engagement, we also observed patterns when they are assigned in terms of each user. Specifically, we defined a User Earliness (UE) score for each user as:

$$UE(u) = \frac{\text{\# of early engagements by } u}{\text{\# of all engagements by } u}, \quad (2)$$

where early engagements are determined by the previously defined deadline. Users whose UE score exceeds a certain threshold $thres_u$ are classified as *early users*; those that don't are *late users*.

The distributions of the FNA scores of the two different user groups ($thres_u$ was set to 0.8 for both datasets) are shown in Fig. 4. We can once again observe that the skewness in the FNA scores increases among early users as opposed to that among late users. Such results indicate that engagements by "earlier users" are more likely to display patterns of confirmation bias.

4.3 Joint Earliness Patterns

Finally, we explore what happens when the previous viewpoints - both engagement-wise and user-wise - are taken into account simultaneously. Keeping t_d and $thres_u$ the same as before, we divided all engagements into four groups: early users' early engagements (EE), early users' late engagements (EL), late users' early engagements (LE) and late users' late engagements (LL).

The distributions of the FNA scores are shown in Fig. 5, where we observe some interesting results. **(1)** While in our previous observation the late engagement group's FNA scores were less skewed (See Fig. 3), dividing the group further through user-wise earliness we can see that group EL is more skewed compared to group LL. **(2)** Similarly, dividing the late user group further through engagement-wise earliness, it can be seen that group LE is more skewed compared to group LL.

Implications. Our extensive analyses on earliness-related patterns reveal that earlier user engagements have a stronger tendency to be linked with news articles of the same veracity, supporting our hypothesis. The major implication is that within the social graph constructed through the procedure detailed in Section 3.2, *edges containing later engagements have a higher likelihood of connecting "real news"-"fake news" pairs than those containing earlier engagements*. In other words, in the original $\mathcal{A}$ where all engagements are treated equally when determining edge weight, such "later" edges are more likely to be noisy edges that hinder the performance of existing social graph-based fake news detection models.

5 Proposed Method: DAWN

Based on our findings regarding engagement earliness, we propose **DAWN**, a novel method for **D**etecting fake news via e**A**rliness-guided re**W**eighti**N**g. Fig. 6 illustrates the overall framework of DAWN, consisting of three components. **(1)** We first construct edge-specific features for each existing edge in the social graph. Stemming from our insights in Section 4, we represent the joint earliness patterns within the edges as 4-dimensional vectors. **(2)** The features are then fed into an edge weight estimator f, which is a link predictor aiming to adjust the weights of existing edges. Further guided by a *ranking loss*, f suppresses the weights of noisy edges as opposed to that of clean edges (*noisy edges* connect real news-fake news pairs, while *clean edges* connect real news-real news or fake news-fake news pairs), resulting in a reweighted adjacency matrix $\mathcal{W}$. **(3)** Finally, a GNN classifier g utilizes the newly obtained $\mathcal{W}$ alongside node features extracted from the article texts to predict the veracity labels of the nodes.

5.1 Edge Feature Construction

Our empirical analysis indicates that earliness-related engagement patterns can help in determining the likelihood of a certain edge in the social graph being clean or noisy. Motivated by this, we construct earliness-related features specific to each edge, which can in turn successfully identify "later" edges. Our observations regarding the simultaneous consideration of both engagement-wise and user-wise earliness patterns suggest that focusing on either one on its own can lead to critical loss of information, e.g., ignoring the entire late user group will result in ignoring the LE group as well, despite them displaying more skewed patterns.

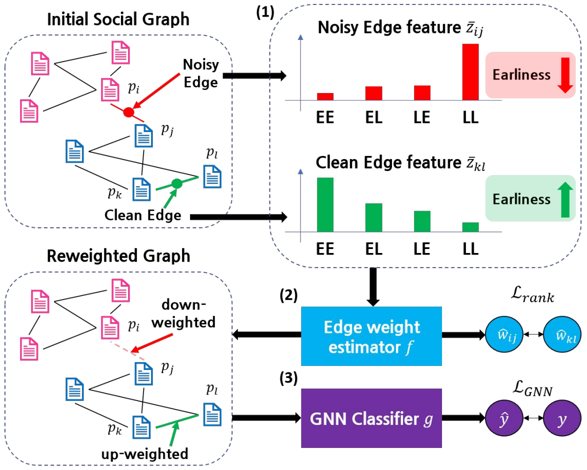


Figure 6: Overall framework of DAWN. After (1) extracting edge-specific features representing engagement earliness, (2) the features are fed into an edge weight estimator f, which distinguishes existing clean and noisy edges and adjusts their weights guided by a ranking loss. (3) Finally, a GNN classifier g is applied on the reweighted graph to predict node veracity.

As such, we design the edge features as joint representations that consider both viewpoints. Let $Z \in \mathbb{R}^{|\#\text{ edges}|\times 4}$ denote an edge feature matrix where each row corresponds to a 4-dimensional vector $z_{ij} \in \mathbb{R}^4$, the feature of the edge $\mathcal{A}_{ij}$ connecting news articles p_i and p_j. Each z_{ij} is constructed by following the strategy detailed in Section 4.3, where we divide the engagements corresponding to $\mathcal{A}_{ij}$ into four groups, i.e., EE, EL, LE and LL. Specifically, each element of z_{ij} represents the size of each group, respectively. After going through all existing edges, we perform a column-wise normalization on Z to obtain $\bar{Z}$, i.e., z_{ij} is normalized to $\bar{z}_{ij}$.

5.2 Noisy Edge Suppression Module

The normalized edge features are then fed into an MLP-based edge weight estimator f to obtain adjusted edge weights. This process is formulated as:

$$w_{ij} = f(\bar{z}_{ij}) = sigmoid(MLP(\bar{z}_{ij})), \tag{3}$$

where the MLP outputs a single value that is then activated through a sigmoid function, resulting in estimated edge weight $w_{ij} \in [0, 1]$. Our objective is to enable f to learn earliness-related patterns in $\bar{z}_{ij}$, allowing it to assign decreased weights to noisy edges and increased weights to clean edges within a fixed range. By replacing the edge weights of the original adjacency matrix $\mathcal{A}$ with the estimated weights, we can obtain a reweighted adjacency matrix $\mathcal{W}$.

To further guide f in distinguishing noisy edges displaying later patterns from clean edges displaying earlier patterns, we additionally introduce a regularization term for the estimated edge weights. Specifically, as we have access to the ground-truth veracity labels of news articles in the training set, we can divide all edges within the training graph into clean edge or noisy edge groups. A straightforward way of utilizing this information would be through a binary classification loss [4], penalizing noisy and clean edge weights when they are far from 0 and 1, respectively. However, as the rate of earliness is different for each edge, we discover that ignoring this and strictly sending all weights to 0 or 1 hinders performance (refer to Section 6.3). As such, we impose constraints that guide reweighting in a less strict manner, by utilizing a ranking loss. Specifically, we randomly sample K edges from both clean and noisy edge groups, and then emphasize the difference within each clean edge-noisy edge pair via the following loss function:

$$\mathcal{L}_{rank} = \frac{1}{K^2}\sum_{i=1}^{K}\sum_{j=1}^{K}\max(0, -(w_{clean}^{(i)} - w_{noisy}^{(j)}) + margin), \quad (4)$$

where $w_{clean}^{(i)}$ and $w_{noisy}^{(j)}$ denote the estimated edge weight of the i-th and j-th sample from the clean edge and noisy edge group, respectively. By penalizing cases where clean edges have smaller weights than noisy edges, further enhanced by a preset *margin* value, we can give regularization to relatively reduce the influence of noisy edges. During training, $\mathcal{L}_{rank}$ can guide the edge weight estimator f to obtain a reweighted training adjacency matrix $\mathcal{W}_{train}$. While comparing *all* possible pairs during training would be ideal, we empirically show that observing K^2 sampled pairs still achieves competitive performance in Section 6.4.

5.3 Fake News Detection Module

Following prior studies [7, 32], we extract the initial node feature x_n of a news article $p_n \in \mathcal{P}$ from the article text via a pre-trained BERT [6]. Utilizing the node features and reweighted adjacency matrix $\mathcal{W}$ obtained via edge weight estimator f, we can learn the representation of nodes through expressive GNN architectures [11, 17, 29]. Based on this learned representation, the veracity of article p_n is predicted in the form of $\hat{y}_n = softmax(h_n)$. $h_n \in \mathbb{R}^2$ is the output of the GNN classifier $g(\mathcal{X}, \mathcal{W})$ for article p_n, where $\mathcal{X}$ is the collection of all relevant node features in the form of a single matrix. During training, the inputs for g are the node features corresponding to news articles in the training set and $\mathcal{W}_{train}$, and the resulting GNN prediction loss is as follows:

$$\mathcal{L}_{GNN} = \sum_{p_n \in \mathcal{P}_{train}} l(\hat{y}_n, y_n), \quad (5)$$

$l(\hat{y}_n, y_n)$ denoting the cross entropy between $\hat{y}_n$ and $y_n \in \mathcal{Y}_{train}$.

5.4 Final Training Objective

The final loss function for training is as follows:

$$\mathcal{L}_{final} = \underset{\theta,\phi}{\arg\min}\mathcal{L}_{GNN} + \alpha\mathcal{L}_{rank}, \quad (6)$$

where θ and ϕ are the learnable parameters of GNN classifier g and edge weight estimator f, respectively. α is a hyperparameter for balancing the contribution of the ranking loss. DAWN follows an end-to-end approach in which f and g are learned simultaneously.

After training on $\mathcal{G}_{train}$, f adjusts the weights of $\mathcal{A}_{val}$, resulting in a reweighted $\mathcal{W}_{val}$. The best performing model on the validation set is then used for final prediction on the test set. Similarly, we adjust the test adjacency matrix through f to obtain $\mathcal{W}_{test}$, which is then fed into g alongside the node features. Through this procedure, DAWN effectively mitigates the effect of noisy edges and detects fake news using low-dimensional earliness-related edge features.

Table 1: Dataset statistics.

Dataset	PolitiFact	GossipCop
# News Articles	597	8,763
# Real News	282	6,764
# Fake News	315	1,999
# Users	162,262	129,820
# Tweets (Engagements)	255,227	516,172

6 Experiments

In this section, we conduct comprehensive experiments to answer the following research questions:

- **RQ1.** How well does our proposed DAWN perform in detecting fake news compared with baselines?
- **RQ2.** How effective are our constructed edge features and additional modules in enhancing DAWN's performance?
- **RQ3.** How does DAWN perform under various hyperparameters?
- **RQ4.** How efficient is DAWN on large-scale social networks?
- **RQ5.** Does DAWN successfully downweight noisy edges?

6.1 Experimental Setup

Datasets. We evaluate DAWN on two prominent fake news datasets, i.e., PolitiFact and GossipCop from the FakeNewsNet benchmark [27]. They contain news articles labeled as real or fake by leading fact-checking websites, along with related tweets by users on X (formerly known as Twitter). The statistics can be found in Table 1.

Under our temporality-aware evaluation scheme, t_{train}, t_{val} and t_{test} are set so that 70%, 10% and 20% of the news articles (in temporal order) are assigned to the training, validation and test sets, respectively. The social contexts regarding users and tweets are also split accordingly.

Baselines. We compare DAWN with the following baselines, which can be further categorized by model architecture: content-based methods (G1) (dEFEND\c [26], DualEmo\c [35], BERT [6] and GPT3.5), social graph-based methods (G2) (GCNFN [22], UPFD [7], FANG [23], GCN [17], GAT [29] and GraphSAGE [11]) and Graph Structure Learning methods (G3) (RS-GNN [4] and DECOR [32]).

Evaluation Metrics. Following previous works [7, 32], we adopt accuracy (**acc.**) and F1 score (**f1.**) to evaluate performance. For all experiments, we report the average of 5 independent runs.

6.2 Detection Performance (RQ1)

Table 2 compares the performance of DAWN and baseline models, where the bold (underlined) values indicate the best (second best) results. We can observe that **(1)** social graph-based methods (G2) generally outperform content-based methods (G1), including LLMs. This highlights the importance of utilizing social contexts for fake news detection. **(2)** Within group G3, RS-GNN is shown to be less effective than DECOR. This indicates that GSL guided by node features is less suited for the fake news detection task as opposed to leveraging edge-specific features. **(3)** DAWN outperforms baseline models by a substantial margin, enhancing accuracy and F1 score by up to 5.6%p and 7.3%p over the most competitive baseline on the GossipCop dataset, respectively. This demonstrates that under the temporality-aware setting where accessible information and social structures change greatly before and after training, DAWN displays more robust detection performance despite its simpler methodology.

Table 2: Performance comparisons under our proposed temporality-aware setting.

	Method	PolitiFact		GossipCop	
		acc.	f1.	acc.	f1.
G1	dEFEND\c [26]	81.2	79.6	75.4	68.3
	DualEmo\c [35]	84.0	81.5	77.9	71.9
	BERT [6]	84.2	81.7	76.4	69.1
	GPT3.5	75.7	77.9	62.8	56.6
G2	GCNFN [22]	84.8	82.3	85.3	82.9
	UPFD [7]	84.6	82.1	82.0	77.8
	FANG [23]	85.5	82.6	86.1	83.4
	GCN [17]	86.4	83.0	87.4	84.7
	GAT [29]	86.7	79.5	86.5	83.6
	GraphSAGE [11]	85.4	80.9	87.5	85.5
G3	RS-GNN [4]	80.8	62.8	85.9	83.6
	DECOR [32]	87.4	84.7	88.1	85.7
Ours	DAWN	**91.9**	**89.8**	**93.7**	**93.0**

Table 3: Ablation studies on DAWN (F1 score (%)).

		PolitiFact	GossipCop
Ours	DAWN	**89.8**	**93.0**
Feature Variants	DAWN+RAND	58.9	79.7
	DAWN-USER	85.6	92.5
	DAWN-ENG	83.2	92.1
	DAWN+RATIO	76.1	89.2
	DAWN+NF	72.0	80.7
Component Variants	DAWN-RANK	76.5	91.9
	DAWN+BC	62.4	90.2

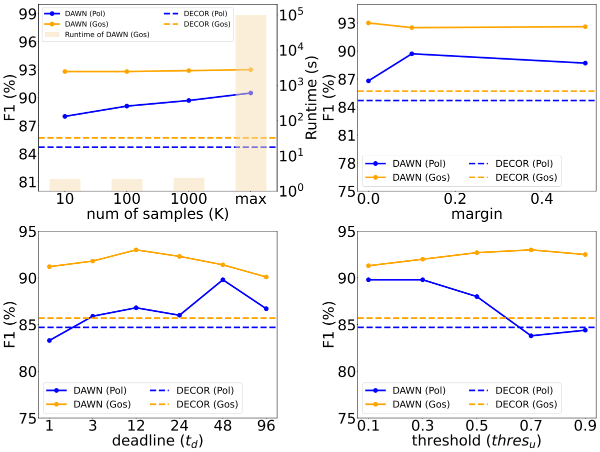


Figure 7: Hyperparameter sensitivity analysis.

As previously discussed, this is thanks to the proposed earliness-related patterns being based upon general social behaviors that occur similarly even at different points in time.

Additionally, we also investigate the performance of DAWN and existing social graph-based methods when evaluated under the *conventional temporality-ignorant setting*. In Fig. 1(b), we observe that DAWN significantly outperforms baselines and displays marginal performance difference between the two settings as opposed to previous methods that suffer substantial degradation. Such results further highlight the effectiveness of our time-independent features and demonstrate the versatility of our proposed method, as it displays superior performance under various scenarios.

6.3 Ablation Study (RQ2)

We conduct ablation studies to validate (1) the effectiveness of our earliness-related edge features proposed in Section 4.3, and (2) the model components of DAWN. Specifically, **DAWN+RAND** replaces our features with random values from the uniform distribution. **DAWN-USER** and **DAWN-ENG** remove user-wise and engagement-wise earliness patterns from the edge features, respectively. **DAWN+RATIO** constructs edge features in an alternate way, by retrieving the relative ratio of each group (EE, EL, LE and LL) size. **DAWN+NF** replaces our features with concatenated features of a node pair for each edge (i.e., concatenation of two 768-dimensional BERT embedding vectors). Meanwhile, **DAWN-RANK** removes $\mathcal{L}_{rank}$ from the final loss function by setting $\alpha = 0$, and **DAWN+BC** replaces $\mathcal{L}_{rank}$ with a binary classification loss as mentioned in Section 5.2.

The results are summarized in Table 3. DAWN is shown to outperform all variants. In detail, DAWN+RATIO, DAWN+RAND and DAWN+NF display worse results when compared to DAWN, highlighting the effectiveness of our earliness-related edge features and supporting our observation in Section 6.2. Additionally, DAWN outperforms DAWN-USER and DAWN-ENG as well, which is in line with our analysis in Section 4.3 that jointly considering both user-wise and engagement-wise earliness patterns results in richer patterns than focusing on either one on its own.

Further, we observe that DAWN outperforms DAWN-RANK, proving the effectiveness of $\mathcal{L}_{rank}$. In addition, DAWN+BC is inferior to DAWN-RANK, supporting our claim that strictly sending edge weights to 0 or 1 is unfit for our task and hinders performance.

6.4 Hyperparameter Sensitivity (RQ3)

In this section, we explore how the performance of DAWN is affected by varying the values of key hyperparameters, i.e., K, $margin$, t_d and $thres_u$. The results are shown in Fig. 7.

We observe that DAWN significantly outperforms DECOR under different variations of the sample size K and $margin$ used in Equation 4 (See Fig. 7 top). The detection performance shows an upward trend when increasing K, which is expected since more edge pairs are involved during training[2]. It is important to note that sampling only 10 edges per group (i.e., $K = 10$) significantly reduces computation time yet still shows performance comparable to the cases with larger Ks, even surpassing DECOR, indicating the effectiveness of DAWN[3].

In terms of deadline t_d and threshold $thres_u$ (See Fig. 7 bottom), we have some interesting observations. In detail, the performance on GossipCop peaks at a much shorter deadline t_d than on PolitiFact, i.e., engagements need to occur much quicker to be considered early in GossipCop. Similarly, the performance on GossipCop is subpar when the threshold $thres_u$ is set to lower values, while on PolitiFact a threshold value too large significantly hinders performance, meaning early users respond much more quickly to news in Gossipcop. Both results indicate that what is considered “early” varies depending on the dataset, where much stricter standards should be used for GossipCop over PolitiFact. This can be attributed to the fact that readers tend to be much more attracted to celebrity gossip (i.e., GossipCop) than political news (i.e., PolitiFact) [3, 12], resulting in stronger and quicker engagements. As such, applying

[2] *max* indicates comparing *all* clean edge-noisy edge pairs instead of K^2 pairs.
[3] Bars in Fig. 7 top denote log-scaled training time (s) of DAWN on GossipCop.

Table 4: Runtime and performance comparisons.

Method	GossipCop	
	Runtime (s)	f1.
GCN	0.286	84.7
DECOR	0.544	85.7
DAWN	2.41	93.0

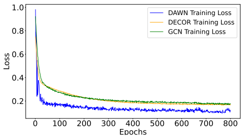


Figure 8: Model convergence comparisons.

Table 5: Homophily ratio of social graphs before and after reweighting the edges by DECOR and DAWN.

	Original graph	Adjusted by DECOR	Adjusted by DAWN
PolitiFact	0.724	0.807	0.821
GossipCop	0.932	0.943	0.954

stricter constraints when determining "early" user engagements can aid in distinguishing the earliness patterns of clean and noisy edges within GossipCop, while relatively lenient constraints are more appropriate for PolitiFact. Further investigating such trends for various news topics may provide valuable insights for future research.

6.5 Efficiency on Large Networks (RQ4)

To assess the applicability of DAWN on real-world tasks with massive social networks, we evaluate its computational cost on GossipCop as it is a much larger dataset compared with PolitiFact. Specifically, we train GCN, DECOR and DAWN under identical settings, and report the mean runtime per epoch and resulting F1 score in Table 4. We observe that although DAWN requires relatively higher computational cost, which is expected due to the introduction of $\mathcal{L}_{rank}$ in Equation 4, it achieves much quicker convergence, as shown in Fig. 8. In practice, DAWN's fast convergence is likely to offset the slightly higher computational cost per epoch, leading to an overall more efficient training process. Further, we point out that the performance of DAWN is significantly higher than that of GCN or DECOR. These results demonstrate DAWN's practical applicability in real-world fake news detection, as it substantially enhances performance while only modestly compromising efficiency.

6.6 Case Study (RQ5)

We conduct a case study to further illustrate how DAWN successfully distinguishes and reweights clean and noisy edges, especially compared to a similar reweighting framework DECOR [32].

In Fig. 9 we compare the neighborhood of a fake news article p after weight adjustment by DECOR (left) and DAWN (right). We observe that DAWN successfully upweights clean edges and significantly downweights the noisy edge, aiding in a correct prediction for p. On the other hand, DECOR wrongly downweights some clean edges and only marginally downweights the noisy edge, resulting in an incorrect prediction. Such differences are not limited to a single case, as we can see in Table 5 that DAWN exceeds DECOR in terms of the adjusted graph's homophily ratio (i.e., ($\sum$ clean edge weights) / ($\sum$ all edge weights)), proving DAWN is much more effective overall in suppressing noisy edges on the entire graph.

The reason for this is that DECOR's degree-based framework is unfit for temporality-aware settings. In detail, as test nodes and their related social contexts are unavailable during training, DECOR is vulnerable to unseen degree patterns that arise at test time. Further, even the degrees of training nodes are subject to substantial change at test time (See Fig. 9 left, where nodes that experience a greater change in degree are more negatively impacted when adjusting their corresponding edge weights) when new related social contexts occur (e.g., reposting a very old news article). Both cases hinder DECOR's reweighting process, leading to performance degradation.

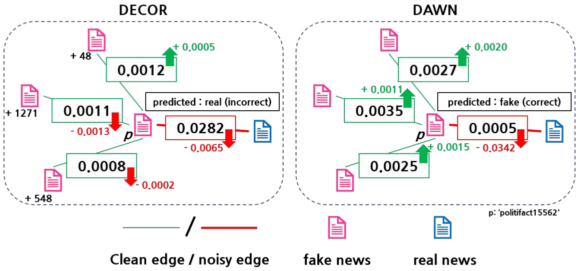


Figure 9: Case study comparison between DECOR (left) and DAWN (right). The boxed values denote the normalized edge weights adjusted by each method. The values next to the arrows show how much the weight has changed from the original adjacency matrix. For DECOR, the values next to the articles indicate the amount of degrees added in test time.

The earliness-related patterns utilized by DAWN, on the other hand, are based on *earlier user engagements having a higher tendency to be drawn to articles with the same veracity*. As previously discussed, this simple yet universal social trend is deeply rooted in the well-explored confirmation bias theory, and thus occurs similarly even at different points in time. Due to the time-independent nature of our earliness features, DAWN can easily adapt to newly observed or substantially changed edges and adjust their weights accordingly, resulting in successful noisy edge suppression and robust effectiveness under the temporality-aware setting.

7 Conclusion

In this paper, we revisit the training and evaluation setting of social graph-based fake news detection methods, and present a more realistic temporality-aware setting where future data (both article-wise and context-wise) is unavailable during training. We further propose DAWN, a method more applicable to such scenarios that utilizes time-independent patterns rooted in fundamental social behaviors. Based on our observation that later user engagements contribute more to noisy edges that link real news-fake news pairs in the social graph, DAWN leverages feature representations of engagement earliness to suppress the weights of such noisy edges through a GSL framework. Extensive analysis on two prominent datasets show the robust effectiveness of our proposed method in detecting fake news under the temporality-aware setting, demonstrating the practical applicability of DAWN to real-world tasks.

Acknowledgements. This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (RS-2024-00335098, RS-2024-00406985), and by the Ministry of Science and ICT (NRF-2022M3J6A1063021).

Ethics Statement

Regarding the adherence of ACM publication policy, to the best of our knowledge, there are no ethical issues with this paper. All datasets used for experiments are publicly available.

References

[1] Hunt Allcott and Matthew Gentzkow. 2017. Social media and fake news in the 2016 election. *Journal of economic perspectives* 31, 2 (2017), 211–236.

[2] Oberiri Destiny Apuke and Bahiyah Omar. 2021. Fake news and COVID-19: modelling the predictors of fake news sharing among social media users. *Telematics and Informatics* 56 (2021), 101475.

[3] Pablo J Boczkowski and Eugenia Mitchelstein. 2013. *The news gap: When the information preferences of the media and the public diverge.* MIT press.

[4] Enyan Dai, Wei Jin, Hui Liu, and Suhang Wang. 2022. Towards robust graph neural networks for noisy graphs with sparse labels. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*. 181–191.

[5] Hanjun Dai, Hui Li, Tian Tian, Xin Huang, Lin Wang, Jun Zhu, and Le Song. 2018. Adversarial attack on graph structured data. In *International conference on machine learning*. PMLR, 1115–1124.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[7] Yingtong Dou, Kai Shu, Congying Xia, Philip S Yu, and Lichao Sun. 2021. User preference-aware fake news detection. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*. 2051–2055.

[8] Negin Entezari, Saba A Al-Sayouri, Amirali Darvishzadeh, and Evangelos E Papalexakis. 2020. All you need is low (rank) defending against adversarial attacks on graphs. In *Proceedings of the 13th international conference on web search and data mining*. 169–177.

[9] Chuan Guo, Juan Cao, Xueyao Zhang, Kai Shu, and Miao Yu. 2019. Exploiting emotions for fake news detection on social media. *arXiv preprint arXiv:1903.01728* (2019).

[10] Hao Guo, Weixin Zeng, Jiuyang Tang, and Xiang Zhao. 2023. Interpretable Fake News Detection with Graph Evidence. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*. 659–668.

[11] Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems* 30 (2017).

[12] Tony Harcup and Deirdre O'neill. 2017. What is news? News values revisited (again). *Journalism studies* 18, 12 (2017), 1470–1488.

[13] Yeonjun In, Kanghoon Yoon, Kibum Kim, Kijung Shin, and Chanyoung Park. 2024. Self-Guided Robust Graph Structure Refinement. In *Proceedings of the ACM on Web Conference 2024*. 697–708.

[14] Yeonjun In, Kanghoon Yoon, and Chanyoung Park. 2023. Similarity preserving adversarial graph contrastive learning. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 867–878.

[15] Wei Jin, Yao Ma, Xiaorui Liu, Xianfeng Tang, Suhang Wang, and Jiliang Tang. 2020. Graph structure learning for robust graph neural networks. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*. 66–74.

[16] Junghoon Kim, Yeonjun In, Kanghoon Yoon, Junmo Lee, and Chanyoung Park. 2023. Class Label-aware Graph Anomaly Detection. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*. 4008–4012.

[17] Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).

[18] Emily Kubin and Christian Von Sikorski. 2021. The role of (social) media in political polarization: a systematic review. *Annals of the International Communication Association* 45, 3 (2021), 188–206.

[19] Vasileios Lampos, Maimuna S Majumder, Elad Yom-Tov, Michael Edelstein, Simon Moura, Yohhei Hamada, Molebogeng X Rangaka, Rachel A McKendry, and Ingemar J Cox. 2021. Tracking COVID-19 using online search. *NPJ digital medicine* 4, 1 (2021), 17.

[20] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692* (2019).

[21] Jasmine McNealy and Michaela Devyn Mullis. 2019. Tea and turbulence: Communication privacy management theory and online celebrity gossip forums. *Computers in Human Behavior* 92 (2019), 110–118.

[22] Federico Monti, Fabrizio Frasca, Davide Eynard, Damon Mannion, and Michael M Bronstein. 2019. Fake news detection on social media using geometric deep learning. *arXiv preprint arXiv:1902.06673* (2019).

[23] Van-Hoang Nguyen, Kazunari Sugiyama, Preslav Nakov, and Min-Yen Kan. 2020. Fang: Leveraging social context for fake news detection using graph representation. In *Proceedings of the 29th ACM international conference on information & knowledge management*. 1165–1174.

[24] Raymond S Nickerson. 1998. Confirmation bias: A ubiquitous phenomenon in many guises. *Review of general psychology* 2, 2 (1998), 175–220.

[25] 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).

[26] Kai Shu, Limeng Cui, Suhang Wang, Dongwon Lee, and Huan Liu. 2019. defend: Explainable fake news detection. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. 395–405.

[27] Kai Shu, Deepak Mahudeswaran, Suhang Wang, Dongwon Lee, and Huan Liu. 2020. Fakenewsnet: A data repository with news content, social context, and spatiotemporal information for studying fake news on social media. *Big data* 8, 3 (2020), 171–188.

[28] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake news detection on social media: A data mining perspective. *ACM SIGKDD explorations newsletter* 19, 1 (2017), 22–36.

[29] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903* (2017).

[30] Binghui Wang and Neil Zhenqiang Gong. 2019. Attacking graph-based classification via manipulating the graph structure. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. 2023–2040.

[31] Huijun Wu, Chen Wang, Yuriy Tyshetskiy, Andrew Docherty, Kai Lu, and Liming Zhu. 2019. Adversarial examples on graph data: Deep insights into attack and defense. *arXiv preprint arXiv:1903.01610* (2019).

[32] Jiaying Wu and Bryan Hooi. 2023. Decor: Degree-corrected social graph refinement for fake news detection. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2582–2593.

[33] Jiaying Wu, Shen Li, Ailin Deng, Miao Xiong, and Bryan Hooi. 2023. Prompt-and-align: prompt-based social alignment for few-shot fake news detection. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*. 2726–2736.

[34] Kanghoon Yoon, Yeonjun In, Namkyeong Lee, Kibum Kim, and Chanyoung Park. 2024. Debiased Graph Poisoning Attack via Contrastive Surrogate Objective. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*. 3012–3021.

[35] Xueyao Zhang, Juan Cao, Xirong Li, Qiang Sheng, Lei Zhong, and Kai Shu. 2021. Mining dual emotion for fake news detection. In *Proceedings of the web conference 2021*. 3465–3476.

[36] Xinyi Zhou and Reza Zafarani. 2020. A survey of fake news: Fundamental theories, detection methods, and opportunities. *ACM Computing Surveys (CSUR)* 53, 5 (2020), 1–40.