ECAI 2023
K. Gal et al. (Eds.)

doi:10.3233/FAIA230315

PiercingEye: Identifying Both Faint and Distinct Clues for Explainable Fake News Detection with Progressive Dynamic Graph Mining

Yasan Ding[a], Bin Guo[a;*], Yan Liu[b], Hao Wang[a], Haocheng Shen[a] and Zhiwen Yu[a]

[a]Northwestern Polytechnical University
[b]Peking University

Abstract. Explainability is crucial for the successful use of AI for fake news detection (FND). Researchers aim to improve the explainability of FND by highlighting important descriptions in crowd-contributed comments as clues. From the perspective of law and sociology, there are distinct clues that are easy to discover and understand, and faint clues that require careful observation and analysis. For example, in fake news related to COVID-Omicron showing increased pathogenicity and transmissibility, distinct clues might involve virologists' opinions regarding the inverse correlation between pathogenicity and transmissibility. Meanwhile, faint clues might be reflected in an infected person's claim that the symptoms are milder than a cold (indirectly indicating reduced pathogenicity). Occasionally, the statements of some ordinary eyewitnesses can decisively reveal the truth of the news, leading to the judgment of fake news. Existing methods generally use static networks to model the entire news life-cycle, which makes it fail to capture the subtle dynamic interactions between individual clues and news. Thereby faint clues, whose relations to the truth of news are challenging to be characterized and extracted directly, are more likely to be overshadowed by distinct clues. To address this issue, we propose an explainable FND method, dubbed as PiercingEye, which leverages dynamic interaction information to progressively mine valuable clues. PiercingEye models the news propagation topology as a dynamic graph, with interactive comments serving as nodes, and employs the time-semantic encoding mechanism to refine the modeling of temporal interaction information between comments and news to preserve faint clues. Subsequently, it utilizes the self-attention mechanism to aggregate distinct and faint clues for FND. Experimental results demonstrate that PiercingEye outperforms state-of-the-art methods and is capable of identifying both faint and distinct clues for humans to debunk fake news.

1 Introduction

The proliferation of fake news on social networks causes confusion and undermines social stability, especially during epidemics and wars. To promote AI social governance, researchers have proposed various deep learning-based methods to automatically detect fake news [23]. Nevertheless, the fake news detection (FND) differs from typical classification tasks in that, in addition to verifying the authenticity of news, it also needs human-understandable decision-making support to persuade users and prevent further dissemination, i.e., explainable FND [9]. Early explorations rely on specific linguistic patterns [36] or propagation topology [11] to explain why a news record is fake, but this remains incomprehensible to humans.

The interactions that occur during the news dissemination facilitate the explainable FND. Identifying important comments in crowd-contributed content [9] not only guides the model's learning of news features, but also provides material for human to make judgements. We refer to these important comments, which directly or indirectly support decision-making in FND, as "clues". Clues encompass several semantic types, as shown in Table 1. For instance, during the spread of fake news, eyewitnesses tend to provide first-hand accounts of the scene in their comments, while authoritative experts may use their relevant knowledge to debunk inaccurate information.

Table 1. The Different Semantic Types of Clues for Explainable FND

Type	Description
Narratives from eyewitnesses	Objective description of news events
Expert opinion	Conclusions from experts, e.g., journalists
Statistical data	Survey results or reports
Universal principle	Widely accepted facts and principles
Knowledge of similar events	Relevant facts from previous similar events
Multimedia material	Videos/images directly related to the news

From the perspective of law and sociology, the aforementioned clues are categorized as distinct clues and faint clues [1, 3]. Distinct clues are easily discernible and directly prove news to be fake, whereas faint clues require more delicate analysis and may indirectly prove news to be fake. Concretely, in the fake news claiming that *COVID-Omicron showing increased pathogenicity and transmissibility*, as shown in Fig. 1, distinct clues include medical experts' statements on the inverse correlation between pathogenicity and transmissibility, as well as analysis of indicators such as infection coefficient and mortality rate. Faint clues, on the other hand, may involve the claims of an infected person that the symptoms are milder than a cold, which indirectly suggests a decrease in the pathogenic-

* Corresponding Author. Email: guob@nwpu.edu.cn

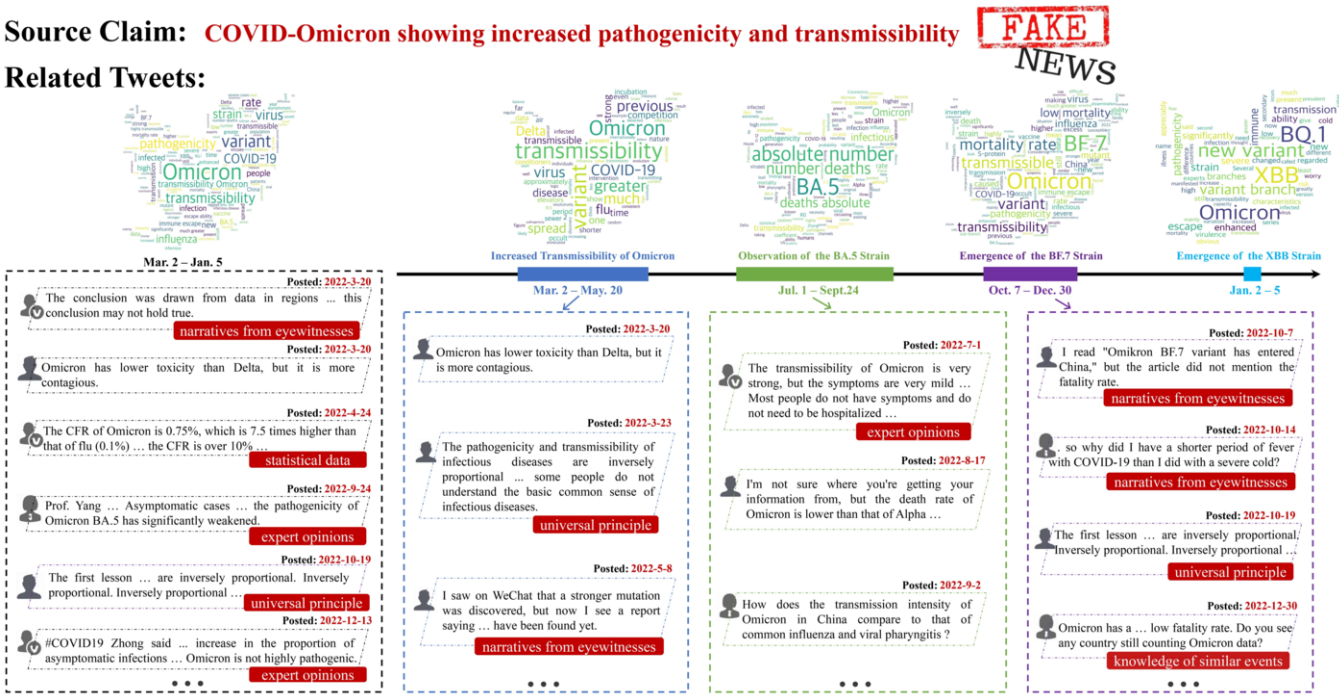

Figure 1. A preliminary analysis of a piece of fake news circulating on Twitter. The accompanying figure depicts the word clouds and representative clues (types) based on tweets related to the source tweet from March 2022 to January 2023 (all tweets translated from Chinese to English). The left panel is drawn based on the complete life-cycle data, while the right is displayed in a timeline format, detailing the evolution of news dissemination in a more intricate way.

ity of COVID-Omicron. During the early stages of news propagation, when the majority of users express support for the source tweet, this faint clue reflecting the actual situation based on their personal experience is especially valuable. As the source tweet spreads, this faint clue begins to attract rich interactions and resonate with multiple users, and results in an increasing number of critical voices appearing in comments, which is as indispensable as distinct clues to the entire FND process. Moreover, the content and semantic types of both faint and distinct clues undergo a continuous evolution through the dissemination of news, as Fig. 1 showing. By adaptively fusing such clues, the performance of FND can be improved in three ways: (1) *providing a comprehensive content* that merges the surface-level depiction of news events with the underlying logic, (2) *offering an objective analytical perspective* that can rectify misinformative content, and (3) *developing an efficient judgement* that can distinguish decisive clues from unimportant ones.

Existing methods tend to prioritize the identification of distinct clues whilst disregarding faint clues, which commonly utilize attention mechanisms [4, 8, 20] to select clues related to news content. For instance, Chen et al. [4] introduce self-attention into recurrent neural networks (RNN) for selective inference of distinct words or phrases with respect to inaccurate information. Similarly, Wu et al. [28] develop co-attention self-attention networks to achieve precise evidence selection. However, they primarily rely on identifying distinct clues based on the semantic similarity of keywords, e.g., *pathogenicity*, *transmissibility*, and *Omicron*, as depicted in Fig. 1. Conversely, faint clues have a wide variety of semantic associations, and hence, it is challenging to capture them based solely on relevance. Moreover, the increase in the number of candidate clues makes it difficult for them to address long-range semantic dependencies, overlooking clues that are stated earlier in sequential comments.

To further capture faint clues, some studies have proposed graph-based approaches [2, 32] to discover the intricate interactions between clues and news, which represent candidate clues as nodes and their connections as edges. For instance, Xu et al. [32] partition comments into news-related snippets and construct a semantic graph based on the co-occurrence relationships of these snippets. They then utilize the semantic relationships learned from the graph to mine faint clues fragments. However, most methods primarily model the propagation of news using static networks [24], which fail to capture the subtle dynamic interactions between faint clues and news. According to the temporal analysis results in Fig. 1, whenever a new virus variant appears, discussions about its pathogenicity and transmissibility reignite on Twitter. However, in the static global analysis results, the keywords associated with each virus variant are ignored due to their lower frequency, leading to an incomplete understanding of the dynamic evolution of the news. To prevent faint clues from being overshadowed by distinct clues, it is essential to effectively model the dynamic interaction features between clues and news for explainable FND [18].

In this paper, we aim to improve explainable FND by a fine-grained characterization and integration of faint and distinct clues throughout the news dissemination process. Drawing inspiration from dynamic graph neural networks (DGNNs) [29, 35], we intend to model the impact of clues on news dissemination based on continuous-time dynamic graphs. Despite the current ability of DGNN-based approaches to model the dynamic propagation of news, such as the Dynamic GCN [6] and the TGNF [24], they fail to consider the impact of semantic changes in comments on FND performance. To achieve our goals, we face two primary challenges: (1) How to characterize and distinguish between faint and distinct clues during news dissemination, given that clues have multiple semantic

types? (2) How to continuously aggregate decisive clues for the final explainable FND, given that different clues have varying impacts on the evolution of news propagation and their importance changes as new interactive comments emerge?

To address the aforementioned challenges, we propose an explainable FND method that utilizes continuous-time dynamic graphs to progressively extract decisive clues from user comments, dubbed as *PiercingEye*, mainly consisting of the *hybrid clues leaner*, the *dynamic clues tracker*, and the *fake news detector*. PiercingEye constructs a dynamic news propagation graph that includes source news and user comments as nodes and captures their interactions as edges that continue to expand as news propagates. By identifying key nodes, PiercingEye uncovers faint and distinct clues essential for FND. For challenge one, we leverage multi-granularity sentiment and semantic features to describe both faint and distinct clues, and incorporate temporal interactions through our time-semantics encoding mechanism. This ensures that PiercingEye effectively captures and learns two types of clues. For challenge two, we introduce a multi-head self-attention mechanism to emphasize the importance of various clues, and filter out irrelevant clues at each time interval to adapt to dynamic changes in news dissemination. Our contributions are as follows:

- We have pioneered the exploration of providing human-understandable decision-making clues from both distinct and faint clues for explainable FND. We propose a continuous-time dynamic graph-based detection method, dubbed as *PiercingEye*, which progressively extracts human-understandable clues from user comments to debunk fake news.
- We propose the *time-semantics encoding mechanism* to jointly learn the temporal interaction information of each clue and adaptatively aggregate clues using the *self-attention mechanism*, achieving a more accurate characterization of different clues during news dissemination.
- We demonstrate through experiments on two publicly available datasets that PiercingEye outperforms state-of-the-art methods in selecting both faint and distinct clues to debunk fake news.

2 Related Work

2.1 Explainable Fake News Detection

Explainable FND research concentrates primarily on two aspects: (1) creating detection techniques using interpretable machine learning models (e.g., decision trees, probabilistic graph models), and (2) clarifying the decision-making process or detection results. For example, Yang et al. [34] employ Bayesian networks to model the generation of news truth and user comments, incorporating the authority of news outlets and interacting users to infer the veracity of news. DTCA [28] builds an evidence extraction model for FND based on the decision tree, followed by a co-attention mechanism to learn the unverified news and evidence. Chien et al. [5] propose an explainable AI (XAI)-based FND method, namely XFlag, which enhances the transparency of the model by a situation awareness-based agent framework. In addition, the analysis of decision-making processes is accomplished by visualizing content or propagation characteristics that focused by models, or by presenting relevant evidence extracted from user comments. For instance, dEFEND [21] employs RNNs to model news content and user comments, and the co-attention mechanism to fuse information. Lu et al. [13] construct news propagation graphs using interacting users, and propose the Graph-aware Co-Attention Networks to detect fake news by analyzing user profiles. Bian et al. [2] utilize the bidirectional graph convolutional network to model the propagation structure and dispersal patterns of fake news. Xu et al. [32] propose a method that utilizes a semantic graph to represent snippets in source claims and evidence, capturing long-distance semantic dependencies through the attention mechanism. The XFake [33] comprehensively analyzes linguistic and semantic features in visual form, revealing the reasoning process through integrated analysis trees.

Current methods rely on retrospective data analysis to explain why a news record has been identified as fake, while this paper is screening for progressively identifiable clues for human checkers during FND process.

2.2 Dynamic Graph Neural Networks

Dynamic graph neural networks (DGNN) are categorized into discrete-time and continuous-time DGNNs. Discrete-time DGNNs input the graph snapshots under a certain timestamp into sequential models, e.g., WD-GCN [16] and EvolveGCN [19], to capture the temporal information in the graph. Continuous-time DGNNs, on the other hand, utilize RNN to update node embeddings in real-time, e.g., Streaming GNN [15], or use functions to encode continuous time, e.g., Temporal Point Process and Bochner's Theorem [25]. For example, TGAT [31] aggregates target node information with time points and neighborhood information using self-attention mechanisms to infer embeddings for new nodes as the graph evolves. Furthermore, the Dynamic GCN [6] represents the dissemination of news as a series of continuous snapshots of the graph and uses both sequential and temporal snapshots to model the evolution of news for FND. However, they model news dissemination in a discrete-time way, and interaction information can be easily ignored when the time interval between comments is too long. Song et al. [24] regard the propagation of news as a continuous-time dynamic graph and design a temporally evolving graph neural network for FND. Nevertheless, it still overlooks the semantic features contained in posts during news dissemination.

3 Method

3.1 Problem Formulation

Let p^{t_0} denote the news article posted on a social media platform at time t_0, from which we extract clues for the purpose of explainable FND. Each user comment is temporarily considered as a clue denoted as c_i. For the news article p^{t_0} at detecting time t, $\tilde{\mathcal{C}}(t) = \{c_1, c_2, \cdots, c_N\}$ represents the set of N candidate clues, ordered based on their posting time t_i. The news dissemination graph at time t, with the node set $V(t) = \left\{p_0^{t_0}, c_1^{t_1}, c_2^{t_2}, \cdots, c_N^{t_N}\right\}$ and the edge set $E(t) = \left\{e_{ij}^{t_x}\right\}_{i,j\in[1,N]}^{t_x \le t}$, is denoted by $G(t) = \langle V(t), E(t)\rangle$. Specifically, each edge e_{ij} in $E(t)$ indicates that c_i has responded to c_j at time t_x. The adjacency matrix of $G(t)$ is $\mathbf{A}(t) = a_{ij}(t)_{N\times N}$, where $a_{ij}(t)$ can be either 0 or 1 based on the existence of the edge e_{ij} in $E(t)$. We simplify the detection problem by classifying the textual content of news articles as either true or fake, where each news article has a label of 0 (fake) or 1 (true). The research problem is thereby formulated as: Given a dynamic news dissemination graph $G(t) = \langle V(t), E(t)\rangle$ at time t, explainable FND is to learn a classifier: $\mathcal{F} : G(t) \mapsto \hat{y}$ that predicts whether the authenticity of the news and outputs the final set of clues, $\mathcal{C}(t)$, that indicate the facts.

3.2 The PiercingEye Framework

The framework of *PiercingEye* is shown in Fig. 2, which includes the *hybrid clues learner*, *dynamic clues tracker*, and *fake news detector* three components (the following contents are introduced based on the source news p^{t_0} as the target node):

- Hybrid clues learner: Building the target news and its candidate clues into a dynamic graph, learning semantic and sentiment features of each clue, and computing a high-dimensional embedding vector for each clue during the news dissemination;
- Dynamic clues tracker: Utilizing the *time-semantics encoding* mechanism to learn the latent feature vector of each clue at any time point, aggregating the feature information of all interactive clues at the current time based on the *decisive clues aggregating* mechanism, and feeding the intermediate representations of the news to the subsequent component;
- Fake news detector: Using a multi-layer fully connected network with a softmax function to determine whether the target news is fake or not based on the dynamic graph representation.

3.2.1 Hybrid Clues Learner

After each candidate clue c_i preprocessed, we utilize the Word2Vec [17] to compute the d_0-dimension embedding for each word in c_i. Assuming that the maximum number of words in existing clues is k, the c_i is represented with the $k \times d_0$ dimensional original representation $\hat{\mathbf{x}}_i \in \mathbb{R}^{k \times d_0}$. To improve computational efficiency, the Text-CNN [10] is utilized to extract the initial textual semantic features from $\hat{\mathbf{x}}_i$, denoted as $\mathbf{x\_t}_i \in \mathbb{R}^{d_{sf}}$. Note that we utilize n_f filters of varying sizes in the convolution operations (where the width is fixed at 1 and the height varies as h), followed by the max-pooling to compress these feature maps.

In addition to semantic features, it is necessary to explicitly characterize clues using additional features to enable PiercingEye to focus on changes in clues as news spreads. According to Fig. 1, there are substantial differences in sentimental expression between faint and distinct clues. Distinct clues, e.g., expert opinions, are relatively objective and lacking in strong personal sentiments, tending towards non-negativity. On the other hand, faint clues often involve narratives from eyewitnesses or knowledge of similar events, resulting in diverse and occasionally exaggerated expressions with strong emotional tones. Moreover, the sentiment distribution of clues varies dynamically as news develops. Consequently, we utilize multi-granularity sentiment features to support mining clues, including coarse-grained *sentiment scores* and fine-grained *statistical features of sentiment words*.

The sentiment score of c_i, as a positive or negative value, is calculated by TextBlob [12], which considers the effect of *negation words*, *adverbs of degree*, *punctuation marks*, and *emoticons*. Users often convey their emotions through simple characters, such as "?" indicating doubt and "(T_T)" signifying sadness, which could be more effective than only using sentiment words [7]. The formula for computing sentiment score $Sent_i \in \mathbb{R}^{d_{se}}$ is:

$$\begin{cases} Sent_i = \frac{\sum_{i=0}^{M}(-1/2)^n \cdot S_{punc} \cdot S_{emo} \cdot S_{i\_adv}}{M} \\ S_{i\_adv} = \max\left(-1, \min\left(S_i \cdot S_{adv}, 1\right)\right) \end{cases} \tag{1}$$

where M and n respectively represent the number of sentiment and negation words that modify sentiment words in c_i; S_{punc}, S_{emo}, S_{adv}, and S_i represent the sentiment value of punctuation marks, emoticons, adverbs of degree, and the current sentiment word, respectively. The term S_{i_adv} represents the sentiment value of the current sentiment word weighted by the adverb of degree. As shown in Fig. 2, the PF, NF, DF, and RF represent the frequency of positive/negative sentiment words, adverbs of degree, and negation words in c_i, respectively. Suppose the dimension of the statistical features of sentiment words is d_{ss}, then $Stat_i \in \mathbb{R}^{d_{ss}}$. The sentiment embedding of c_i is represented as $\mathbf{x\_s}_i = Sent_i \bigoplus Stat_i \in \mathbb{R}^{d_{se}+d_{ss}}$, where $\bigoplus$ denotes concatenation. Finally, the hybrid clues learner concatenates the semantic features $\mathbf{x\_t}_i$ and the sentiment features $\mathbf{x\_s}_i$ of c_i, i.e., $\mathbf{x}_i = \mathbf{x\_t}_i \bigoplus \mathbf{x\_s}_i$, where $\mathbf{x}_i \in \mathbb{R}^{d_S}$ and $d_S = d_{sf} + d_{se} + d_{ss}$.

3.2.2 Dynamic Clues Tracker

The dynamic clues tracker is consisting of the *time-semantics encoding* mechanism and the *decisive clues aggregating* mechanism.

Time-Semantics Encoding is the mapping of time to a specific dimension vector to obtain temporal semantic information. There are a set of time points when both faint and distinct clues trigger interactions from onlookers, which can motivate the dynamic expansion of the news dissemination graph. This provides sufficient temporal information for PiercingEye to learn the entire news dissemination process. To harmoniously integrate the temporal and semantic information of clues, we need a continuous function $\mathbf{T}(\cdot)$ to map the time to a d_T-dimensional vector space (this function is expressed in vector form), instead of a simple time index. Inspired by related works [30, 31], we define $\mathbf{T}(\cdot)$ by utilizing a sinusoidal function as follows:

$$\begin{cases} \mathbf{T}(t) = \left[\sin\left(t/\omega_1\right), \cos\left(t/\omega_2\right), \cdots, \sin\left(t/\omega_{d_T}\right), \cos\left(t/\omega_{d_T}\right)\right] \\ \omega_i = 2^{(i/d)\cdot\log_2^{t_{max}}} \end{cases} \tag{2}$$

where ω_i is the frequency parameter of the i-th dimensional vector, and t_{max} denotes the length of the news life-cycle. The $\sin(\cdot)$ and $\cos(\cdot)$ functions help our model learn the periodic patterns in time series, thereby improving its generalizability. For each clue, we integrate its time embeddings $\mathbf{T}(t_i)$ and semantic embeddings $\mathbf{x}_i$ to form an intermediate representation $\mathbf{h}_i(t_i) \in \mathbb{R}^{d_S+d_T}$. Concretely, let $X\left(p_0^{t_0}; t\right) = \{\mathbf{x}_1(t_1), \mathbf{x}_2(t_2), \cdots, \mathbf{x}_N(t_N)\}$ represent the set of semantic embeddings for all current candidate clues, and their respective time embeddings compose the set $\mathcal{T}\left(p_0^{t_0}; t\right) = \{\mathbf{T}(t_1), \mathbf{T}(t_2), \cdots, \mathbf{T}(t_N)\}$. The time-semantics encoding mechanism takes inputs from $X\left(p_0^{t_0}; t\right)$ and $\mathcal{T}\left(p_0^{t_0}; t\right)$, and outputs the intermediate clues' representations set $H\left(p_0^{t_0}; t\right)$:

$$\begin{aligned} H\left(p_0^{t_0}; t\right) = \{\mathbf{h}_i(t_i)\}_{i=1}^{N} = \{\ \mathbf{x}_1(t_1) \bigoplus \mathbf{T}(t_1), \\ \mathbf{x}_2(t_2) \bigoplus \mathbf{T}(t_2), \cdots, \mathbf{x}_N(t_N) \bigoplus \mathbf{T}(t_N)\ \} \end{aligned} \tag{3}$$

It is more informative to consider the time intervals between posted times of clues and that of the news than absolute timestamps, which reflect the popularity of news and interaction patterns. Following Xu et al. [31], we apply the translation-invariant property of $\mathbf{T}(\cdot)$ to rewrite Eq. 3:

$$\begin{aligned} H\left(p_0^{t_0}; t\right) = \{\ \mathbf{x}_1(t_1) \bigoplus \mathbf{T}(t - t_1), \mathbf{x}_2(t_2) \bigoplus \\ \mathbf{T}(t - t_2), \cdots, \mathbf{x}_N(t_N) \bigoplus \mathbf{T}(t - t_N)\ \} \end{aligned} \tag{4}$$

For simplicity, the $\mathbf{h}_i(t_i)$ notation is still utilized in the subsequent text to refer to $\mathbf{x}_i(t_i) \bigoplus \mathbf{T}(t - t_i)$.

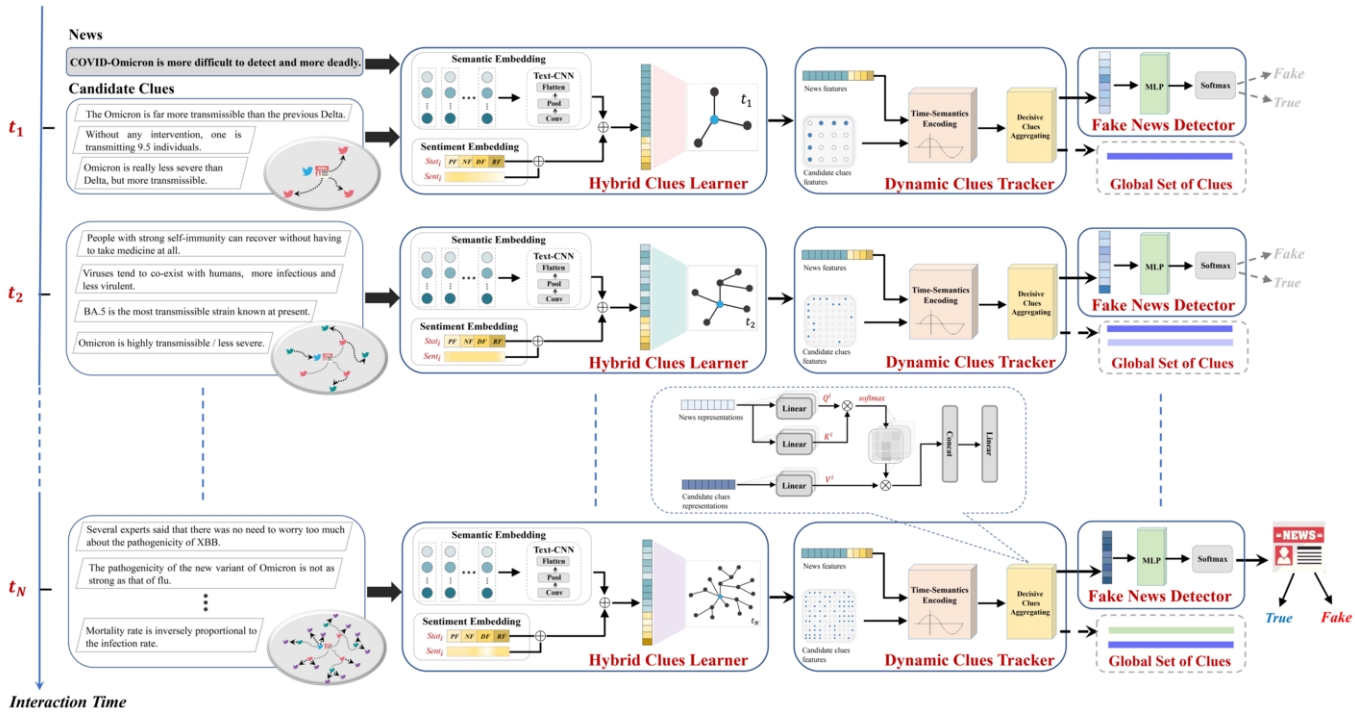

Figure 2. The architecture of PiercingEye.

Decisive Clues Aggregating is responsible for computing the time-aware feature representation of p_0 at time t. The node representation of p_0 without aggregating clues is denoted as $\mathbf{x}_0(t) \bigoplus \mathbf{T}(0)$. Concretely, the multi-head self-attention mechanism [26] is used to learn the importance of different clues and selectively aggregate their features in $G(t)$, and we define the three types of vectors required to compute the importance scores, i.e., *query* $\mathbf{q}(t)$, *key* $\mathbf{k}(t)$, and value *value* $\mathbf{v}(t)$. Specifically, $\mathbf{q}(t)$ represents the feature of the target news article that needs to aggregate clues, $\mathbf{k}(t)$ represents the features of all interacting clues with p_0, and $\mathbf{v}(t)$ is used to preserve the features of all clues. To perform decisive clues aggregating, we set n heads as follows:

$$\begin{cases} \mathbf{q}^j(t) = \mathbf{h}_0(t)\,\mathbf{W}_q^j \\ \mathbf{k}^j(t) = [\mathbf{h}_1(t_1), \mathbf{h}_2(t_2), \cdots, \mathbf{h}_N(t_N)]\,\mathbf{W}_k^j \\ \mathbf{v}^j(t) = [\mathbf{h}_1(t_1), \mathbf{h}_2(t_2), \cdots, \mathbf{h}_N(t_N)]\,\mathbf{W}_v^j \end{cases} \tag{5}$$

where $\mathbf{W}_q^j$, $\mathbf{W}_k^j$, $\mathbf{W}_k^j$ ($j \in [1, n]$) are all parameter matrices. After aggregating features from all the candidate clues, the news feature representation can be represented as:

$$\mathbf{h}^j(t) = softmax\left(\frac{\mathbf{q}^j(t)\left[\mathbf{k}^j(t)^\top\right]}{\sqrt{d_T + d_S}}\right)\mathbf{v}^j(t) \tag{6}$$

Finally, these n weighted feature representations are concatenated with the representation of the source news $\mathbf{h}_0(t)$ and mapped into the final form using the matrix $\mathbf{W}_o$ and bias $\mathbf{b}_o$:

$$\mathbf{h}(t) = \left[\mathbf{h}^1(t) \parallel \mathbf{h}^2(t) \parallel \cdots \parallel \mathbf{h}^n(t) \parallel \mathbf{h}_0(t)\right]\mathbf{W}_o + \mathbf{b}_o \tag{7}$$

3.2.3 *Fake News Detector*

After the hybrid clues learner and dynamic clues tracker stages, the fake news detector performs mean-pooling on $\mathbf{h}(t)$ to obtain the final representation of $G(t)$. Thereafter, this representation is fed into two fully-connected layers with a softmax function to predict the label $\hat{y}$ (i.e., true or fake) of the source news. PiercingEye is trained using the supervised binary cross-entropy loss, as shown in Eq. 8:

$$\mathcal{L} = \mathbb{E}\left[y log\hat{y} + (1-y)\,log\,(1-\hat{y})\right] \tag{8}$$

Concerning the final selection of faint clues and distinct clues set $\mathcal{C}(t)$ by PiercingEye, we rank them based on the weights of candidate clues calculated by the multi-head self-attention mechanism after model convergence. Afterward, a specific number of clues with higher weight values are grouped together to facilitate human experts in debunking fake news.

4 Experiments

4.1 *Datasets*

For fairly comparing the performance of PiercingEye and existing methods on explainable FND, we use two authoritative datasets to complete experiments, see details in Table 2.

- The *Rumdect-Weibo* [14] dataset comprises news content, comments, user profiles, and interaction timestamps. It consists of 2131 fake news records and 2207 non-fake news records.
- The Fakenewsnet [22] is one of the benchmarks for FND, and we select its PolitiFact [1] part as our dataset, denoted as *Fakenewsnet-*

[1] https://www.politifact.com/

PolitiFact. We exclude the news records with too short text length, and thus the polished dataset contains 432 fake news records and 317 true news records.

Table 2. Statistics of Datasets

Dataset	# of fake news	# of real news	# of users	Avg. time length	Avg. # of tweets
Rumdect -Weibo	2131	2207	1309645	1577 h	378
Fakenewsnet -PolitiFact	432	317	45109	1951 h	42

4.2 Implementation Details

4.2.1 Baselines

We have compared our proposed model with the following representative methods: (1) The *Text-CNN* [10] directly extracts fine-grained features of news content through multiple convolutional filters of different sizes to carry out FND. (2) The *GRU-2* employs a two-layer gated recurrent units to extract semantic features of the entire user comments to distinguish between true and fake news. (3) *dEFEND* [21] is an explainable FND pioneer consisting of a word encoder, sentence encoder, sentence-comment co-attention, and fake news detector. (4) *DTCA* [28] constructs an evidence extraction model based on the decision tree, selecting highly reliable user comments as evidence for explainable FND. (5) The *GAT* [27] introduces the self-attention mechanism on graph convolutional networks (GCN) to effectively aggregate local graph features to identify more accurate fake news characteristics. (6) The *BiGCN* [2] utilizes top-down and bottom-up GCNs to respectively extract propagation and dispersion features during news dissemination for training fake news classifiers. (7) The *Dynamic GCN* [6] represents news and user comments as a discrete-time dynamic graph, and utilizes the attention-based GCN to detect fake news. (8) The *TGNF* models the news propagation as a continuous-time dynamic graph, and aggregates temporal-topological features for FND.

4.2.2 Experimental settings

We construct news dissemination graphs using the source news records, user comments/replies as nodes, and their interactions as edges for each data item in both datasets. The dimension of the original word embeddings, d_0, is 300, and the number of convolutional filters, n_f, is set to 20, with the height, h, ranging from 1 to 4 as integers. We determine the maximum sentence length, k, based on 90% of the data records, which is 70 in Rumdect-Weibo and 100 in Fakenewsnet-PolitiFact. Also, the embedding dimension of the sentiment score, $Sent$, is 1, and that of the statistical features of sentiment words, $Stat$, is set to 4. Furthermore, we search for the optimal number of heads, n, in the multi-head attention mechanism from a candidate set $\{1, 2, 3, 4, 5\}$ based on the performance on both datasets, and finally set it to 4. The hidden sizes of the two fully connected layers in the fake news detector are 128 and 64, respectively. Moreover, we implement all the models based on PyTorch (version 1.6.0) and optimize them using the Adam optimizer with a learning rate of $1e-4$.

4.3 Fake News Detection Results

Table 3. Fake News Detection Performance on the Rumdect-Weibo

Method	Acc.	Fake News			True News		
		P	R	F1	P	R	F1
Text-CNN	0.788	0.786	0.789	0.787	0.789	0.786	0.788
GRU-2	0.811	0.808	0.796	0.791	0.828	0.820	0.809
DTCA	0.828	0.827	0.796	0.801	0.818	0.857	0.829
BiGCN	0.858	0.867	0.847	0.857	0.848	0.868	0.858
GAT	0.865	0.865	0.863	0.864	0.864	0.866	0.865
dEFEND	0.873	0.872	0.874	0.873	0.874	0.873	0.873
Dynamic GCN	0.885	0.886	0.883	0.884	0.884	0.887	0.885
TGNF	0.889	0.890	**0.888**	0.889	0.888	0.891	0.890
PiercingEye	**0.896**	**0.904**	0.883	**0.893**	**0.889**	**0.909**	**0.899**

Table 4. Fake News Detection Performance on the FakeNewsNet-PolitiFact

Method	Acc.	Fake News			True News		
		P	R	F1	P	R	F1
Text-CNN	0.707	0.715	0.698	0.706	0.700	0.717	0.709
GRU-2	0.732	0.740	0.724	0.731	0.725	0.741	0.733
DTCA	0.805	0.818	0.788	0.803	0.793	0.822	0.807
BiGCN	0.841	0.847	0.836	0.841	0.836	0.847	0.841
GAT	0.858	0.860	0.855	0.858	0.857	0.861	0.859
dEFEND	0.859	0.825	**0.891**	0.850	0.871	0.818	0.836
Dynamic GCN	0.864	0.867	0.862	0.865	0.861	0.866	0.863
TGNF	0.865	**0.879**	0.849	0.864	0.852	**0.882**	0.866
PiercingEye	**0.870**	0.869	0.866	**0.867**	**0.871**	0.874	**0.872**

4.3.1 Overall performance

In this section, we analyze the performance of models on datasets in terms of fake news detection accuracy (*Acc.*), precision (*P*), recall (*R*), and F1-score (*F1*). The results presented in Table 3 and 4 demonstrate that our proposed PiercingEye generally achieves optimal fake news detection performance on both datasets. Specifically, PiercingEye shows higher detection accuracy on Rumdect-Weibo dataset compared to BiGCN, DTCA, and GRU-2, respectively. In essence, the GRU-2 treats candidate clues as time-series data and extracts semantic information in small fragments, which makes it hard for deal with complex and long-distance semantic dependencies among clues. Although DTCA explicitly models the propagation topology of news, it fails to consider the interaction between different evidence branches under the source news. Users are affected by other parallel tweets while publishing comments, not only by the current tweet. The BiGCN learns features of the entire graph, and it often aggregates redundant or interfering information in candidate clues, which leads to the neglect of faint clues for FND during specific time periods. Furthermore, Table 3 shows that PiercingEye has a higher accuracy than Dynamic GCN, and it also improves precision in identifying fake and true news. This improvement can be credited to the Dynamic GCN model's way of learning the dynamics of news at fixed time intervals, i.e., discrete-time dynamic graphs. A more extended time interval between candidate clues might lead to the neglect of temporal interaction information between them. In

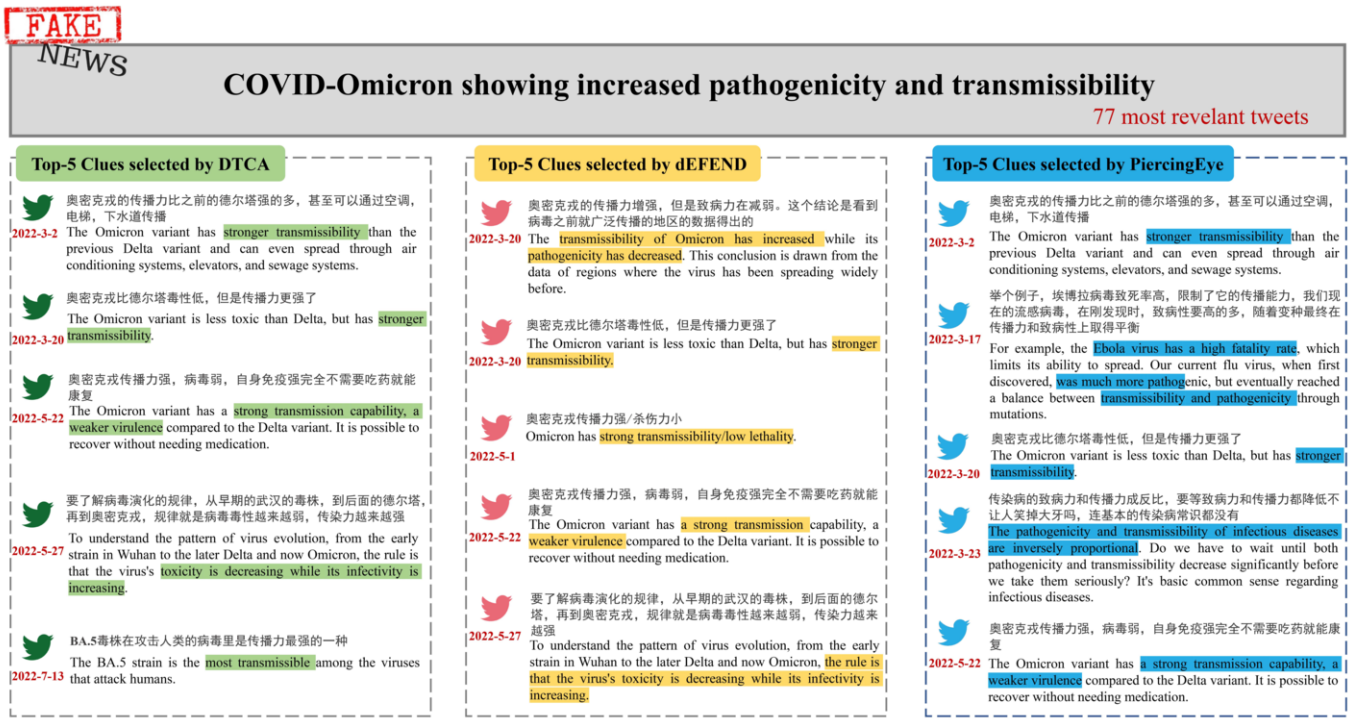

Figure 3. The comparison of clues extraction results (sorted by comment interaction time). The parts marked in different colors are visualizations of the key semantic segments of the selected clues, manually curated rather than automatically implemented by models.

both datasets, news has a relatively long lifespan and the distribution of candidate clues is not balanced across different time periods. Differently, our PiercingEye jointly encodes continuous temporal and semantic information to accurately capture decisive clues for FND. Additionally, PiercingEye outperforms TGNF in accuracy and shows an improvement in precision in identifying fake news, as presented in Table 4, which illustrates the effectiveness of our multi-granularity sentiment features in capturing important clues during FND.

4.3.2 *Case study for explainable FND*

We have conducted a comparative analysis of PiercingEye with dEFEND and DTCA in the context of explainable FND, wherein dEFEND provides explainability through the ranking of user comments and DTCA explains through the identification of evidence to expose inaccurate news fragments in user comments. On the other hand, PiercingEye considers important user comments as clues to uncover fake news. To facilitate ease of comparison and explanation, we refer to dEFEND's *explainable comments* and DTCA's *evidences* as the *clues*. In this context, Fig. 3 illustrates the clues selection outcomes of the three explainable FND methods using real Twitter data, which is the same dataset utilized in the preliminary analysis presented in Fig. 1. The results show that DTCA and dEFEND tend to extract clues that are semantically similar to the source news, while disregarding faint clues that our PiercingEye can capture. For example, the highly-focused clues of PiercingEye not only consist of distinct clues that contain semantic information highly similar to news content, but also the knowledge gained from the Ebola virus, which proves that the pathogenicity and transmissibility of viruses are inversely proportional. However, this decisive faint clue has been ignored by both the DTCA and dEFEND. Furthermore, the DTCA and dEFEND model the news dissemination data statically, which leads to a certain lag in the clues extracted in terms of interaction time compared to PiercingEye. As shown in Fig. 3, the clues extracted by PiercingEye are mainly concentrated in March, which is the early stage of the fake news dissemination. Hence, progressively mining decisive clues based on dynamic graph mining can also provide inspiration for early fake news detection.

5 Conclusion and Future Work

This article primarily focuses on the issue of explainable fake news detection (FND), and we aim at extracting human-understandable clues from user comments to debunk fake news. Since the semantic information of user comments constantly changes as news spreads, we propose a continuous-time dynamic graph neural network-based FND method, dubbed as *PiercingEye*, which consists of the hybrid clues learner, dynamic clues tracker, and fake news detector. Experimental results on two public datasets demonstrate that our PiercingEye outperforms state-of-the-art methods and has the ability to capture both the faint and distinct clues from user comments for explainable FND.

For the future work, we will attempt to introduce causal inference techniques to further enhance the explainability of FND. The current version of PiercingEye lacks a credibility evaluation of candidate clues, which may be susceptible to deceptive comments. In addition, it is necessary to more fundamentally characterize and distinguish faint clues and distinct clues in the future to fully utilize their role in explainable FND.

Acknowledgements

We would like to thank the reviewers for their comments, which helped improve this paper considerably. This work was partially supported by the National Science Fund for Distinguished Young Scholars (62025205), and the National Natural Science Foundation of China (No. 61960206008, 62032020).

References

[1] Terence Anderson, David Schum, and William Twining, *Analysis of evidence*, Cambridge University Press, 2005.

[2] Tian Bian, Xi Xiao, Tingyang Xu, Peilin Zhao, Wenbing Huang, Yu Rong, and Junzhou Huang, 'Rumor detection on social media with bi-directional graph convolutional networks', in *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 549–556, (2020).

[3] Peter Brooks, 'Clues, evidence, detection: Law stories', *Narrative*, **25**(1), 1–27, (2017).

[4] Tong Chen, Xue Li, Hongzhi Yin, and Jun Zhang, 'Call attention to rumors: Deep attention based recurrent neural networks for early rumor detection', in *Trends and Applications in Knowledge Discovery and Data Mining: PAKDD 2018 Workshops*, pp. 40–52. Springer, (2018).

[5] Shih-Yi Chien, Cheng-Jun Yang, and Fang Yu, 'Xflag: Explainable fake news detection model on social media', *International Journal of Human–Computer Interaction*, **38**(18-20), 1808–1827, (2022).

[6] Jiho Choi, Taewook Ko, Younhyuk Choi, Hyungho Byun, and Chongkwon Kim, 'Dynamic graph convolutional networks with attention mechanism for rumor detection on social media', *Plos one*, **16**(8), e0256039, (2021).

[7] Arjun Choudhry, Inder Khatri, and Minni Jain, 'An emotion-based multi-task approach to fake news detection (student abstract)', in *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 12929–12930, (2022).

[8] Manqing Dong, Lina Yao, Xianzhi Wang, Boualem Benatallah, Quan Z Sheng, and Hao Huang, 'Dual: A deep unified attention model with latent relation representations for fake news detection', in *Web Information Systems Engineering–WISE 2018: 19th International Conference, Dubai, United Arab Emirates, November 12-15, 2018, Proceedings, Part I 19*, pp. 199–209. Springer, (2018).

[9] Bin Guo, Yasan Ding, Lina Yao, Yunji Liang, and Zhiwen Yu, 'The future of false information detection on social media: New perspectives and trends', *ACM Computing Surveys (CSUR)*, **53**(4), 1–36, (2020).

[10] Yoon Kim, 'Convolutional neural networks for sentence classification', in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP*, pp. 1746–1751. ACL, (2014).

[11] David MJ Lazer, Matthew A Baum, Yochai Benkler, Adam J Berinsky, Kelly M Greenhill, Filippo Menczer, Miriam J Metzger, Brendan Nyhan, Gordon Pennycook, David Rothschild, et al., 'The science of fake news', *Science*, **359**(6380), 1094–1096, (2018).

[12] Steven Loria, P Keen, M Honnibal, R Yankovsky, D Karesh, E Dempsey, et al., 'Textblob: simplified text processing; 2018', *Online: https://textblob. readthedocs. io/en/dev/Accessed*, 08–02, (2019).

[13] Yi-Ju Lu and Cheng-Te Li, 'Gcan: Graph-aware co-attention networks for explainable fake news detection on social media', *arXiv preprint arXiv:2004.11648*, (2020).

[14] Jing Ma, Wei Gao, Prasenjit Mitra, Sejeong Kwon, Bernard J Jansen, Kam-Fai Wong, and Meeyoung Cha, 'Detecting rumors from microblogs with recurrent neural networks', (2016).

[15] Yao Ma, Ziyi Guo, Zhaocun Ren, Jiliang Tang, and Dawei Yin, 'Streaming graph neural networks', in *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*, pp. 719–728, (2020).

[16] Franco Manessi, Alessandro Rozza, and Mario Manzo, 'Dynamic graph convolutional networks', *Pattern Recognition*, **97**, 107000, (2020).

[17] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean, 'Distributed representations of words and phrases and their compositionality', *Advances in neural information processing systems*, **26**, (2013).

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

[19] Aldo Pareja, Giacomo Domeniconi, Jie Chen, Tengfei Ma, Toyotaro Suzumura, Hiroki Kanezashi, Tim Kaler, Tao Schardl, and Charles Leiserson, 'Evolvegcn: Evolving graph convolutional networks for dynamic graphs', in *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 5363–5370, (2020).

[20] Kashyap Popat, Subhabrata Mukherjee, Andrew Yates, and Gerhard Weikum, 'Declare: Debunking fake news and false claims using evidence-aware deep learning', in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 22–32, (2018).

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

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

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

[24] Chenguang Song, Kai Shu, and Bin Wu, 'Temporally evolving graph neural network for fake news detection', *Information Processing & Management*, **58**(6), 102712, (2021).

[25] Rakshit Trivedi, Hanjun Dai, Yichen Wang, and Le Song, 'Know-evolve: Deep temporal reasoning for dynamic knowledge graphs', in *international conference on machine learning*, pp. 3462–3471. PMLR, (2017).

[26] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin, 'Attention is all you need', *Advances in neural information processing systems*, **30**, (2017).

[27] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio, 'Graph attention networks', in *International Conference on Learning Representations*.

[28] Lianwei Wu, Yuan Rao, Yongqiang Zhao, Hao Liang, and Ambreen Nazir, 'Dtca: Decision tree-based co-attention networks for explainable claim verification', *arXiv preprint arXiv:2004.13455*, (2020).

[29] Louis-Pascal Xhonneux, Meng Qu, and Jian Tang, 'Continuous graph neural networks', in *International Conference on Machine Learning*, pp. 10432–10441. PMLR, (2020).

[30] Da Xu, Chuanwei Ruan, Evren Korpeoglu, Sushant Kumar, and Kannan Achan, 'Self-attention with functional time representation learning', *Advances in neural information processing systems*, **32**, (2019).

[31] Da Xu, Chuanwei Ruan, Evren Korpeoglu, Sushant Kumar, and Kannan Achan, 'Inductive representation learning on temporal graphs', *arXiv preprint arXiv:2002.07962*, (2020).

[32] Weizhi Xu, Junfei Wu, Qiang Liu, Shu Wu, and Liang Wang, 'Evidence-aware fake news detection with graph neural networks', in *Proceedings of the ACM Web Conference 2022*, pp. 2501–2510, (2022).

[33] Fan Yang, Shiva K Pentyala, Sina Mohseni, Mengnan Du, Hao Yuan, Rhema Linder, Eric D Ragan, Shuiwang Ji, and Xia Hu, 'Xfake: Explainable fake news detector with visualizations', in *The world wide web conference*, pp. 3600–3604, (2019).

[34] Shuo Yang, Kai Shu, Suhang Wang, Renjie Gu, Fan Wu, and Huan Liu, 'Unsupervised fake news detection on social media: A generative approach', in *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pp. 5644–5651, (2019).

[35] Zeyang Zhang, Xin Wang, Ziwei Zhang, Haoyang Li, Zhou Qin, and Wenwu Zhu, 'Dynamic graph neural networks under spatio-temporal distribution shift', in *Advances in Neural Information Processing Systems*, (2022).

[36] Zhe Zhao, Paul Resnick, and Qiaozhu Mei, 'Enquiring minds: Early detection of rumors in social media from enquiry posts', in *Proceedings of the 24th international conference on world wide web*, pp. 1395–1405, (2015).