Meta-Path-based Fake News Detection Leveraging Multi-level Social Context Information

Jian Cui
S2W Inc.
Seongnam, South Korea
cj19960819@gmail.com

Kwanwoo Kim
Korea Advanced Institute of Science and Technology
Daejeon, South Korea
kw2128@kaist.ac.kr

Seung Ho Na
Korea Advanced Institute of Science and Technology
Daejeon, South Korea
harry.na@kaist.ac.kr

Seungwon Shin*
Korea Advanced Institute of Science and Technology
Daejeon, South Korea
claude@kaist.ac.kr

ABSTRACT

Fake news, false or misleading information presented as news, has a significant impact on many aspects of society, such as in politics or healthcare domains. Due to the deceiving nature of fake news, applying Natural Language Processing (NLP) techniques to the news content alone is insufficient. Therefore, more information is required to improve fake news detection, such as the multi-level social context (news publishers and engaged users in social media) information and the temporal information of user engagement. The proper usage of this information, however, introduces three chronic difficulties: 1) multi-level social context information is hard to be used without information loss, 2) temporal information of user engagement is hard to be used along with multi-level social context information, and 3) news representation with multi-level social context and temporal information is hard to be learned in an end-to-end manner. To overcome all three difficulties, we propose a novel fake news detection framework, *Hetero-SCAN*. We use Meta-Path, a composite relation connecting two node types, to extract meaningful multi-level social context information without loss. We then propose Meta-Path instance encoding and aggregation methods to capture the temporal information of user engagement and learn news representation end-to-end. According to our experiment, *Hetero-SCAN* yields significant performance improvement over state-of-the-art fake news detection methods.

CCS CONCEPTS

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

KEYWORDS

Fake News Detection; Graph Representation Learning

*Corresponding author

CIKM '22, October 17–21, 2022, Atlanta, GA, USA

ACM ISBN 978-1-4503-9236-5/22/10…$15.00
https://doi.org/10.1145/3511808.3557394

ACM Reference Format:
Jian Cui, Kwanwoo Kim, Seung Ho Na, and Seungwon Shin. 2022. Meta-Path-based Fake News Detection Leveraging Multi-level Social Context Information. In *Proceedings of the 31st ACM International Conference on Information and Knowledge Management (CIKM '22), October 17–21, 2022, Atlanta, GA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3511808.3557394

1 INTRODUCTION

The wide dissemination of fake news has become a major social problem in the world. The most recent and infamous distribution of fake news was in the 2020 United States presidential election fraud [9] and COVID-19 rumors [1]. Both industry and government are making efforts to prevent the spread of fake news [10]. Nevertheless, fake news verification still relies on human experts and their manual efforts in analyzing the news contents with additional evidence. Therefore, there should be an automatic and efficient way to identify the veracity of the news.

The most typical way to detect fake news is applying Natural Language Processing (NLP) techniques on the news content [15, 18]. Considering that even people struggle in identifying the news authenticity by the news content alone, these NLP solutions are hard to be effective. Thus, more information is required to improve fake news detection.

The first important information is the users in social media. Social media is one of the most influential mediums to propagate information, and it has become a common practice for people to share their thoughts in social media. Even though regular users use social media as a communication tool, some users, known as instigators, intentionally spread fake news. Instigators usually have a highly partisan-biased personal description and a lot of followers and followings, which is significantly different from the profiles of regular users (See in Figure 1). Therefore, analyzing the users engaged in the news can provide additional evidence for identifying news authenticity. The publisher information can also play an important role because certain partisan-biased publishers are more likely to publish fake news [3, 5, 6]. As such, information of users and publishers can be viewed as multi-level social context information, and they provide additional clues for fake news detection.

In addition to multi-level social context information, temporal information of user engagement (temporal information for short) is another instrumental information in fake news detection. Fake and

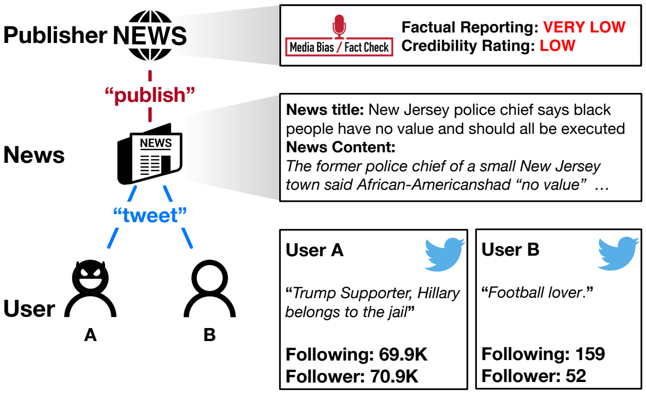


Figure 1: Example of fake news distribution and dissemination. Publishers publish the news, and users tweet the news. Some publishers are regarded as low credibility sources according to the famous fact-checking website, MBFC. User A is an example of an instigator in Twitter, and User B is an example of a regular user.

real news show different propagation properties in social media: Fake news is periodically mentioned by people and usually lasts longer, but real news receives attention only at the beginning of the news publication [27]. In this context, the temporal information should be included in the news representation along with multi-level social context information.

Using multi-level social context and temporal information, however, leads to three chronic difficulties. Firstly, due to the heterogeneity of multi-level social context information, it is hard to use this information without loss. Secondly, temporal information is hard to be used along with multi-level social context information. The graph is a typical way to present social context and its connectivity to the news, but the graph itself has complications in presenting temporal information. The last difficulty is to learn the news representation end-to-end. The information we attempt to use: multi-level social context and temporal information, involves different kinds of information, which will increase the difficulty of adopting end-to-end learning. Although different kinds of information can be dealt with by making their respective sub-tasks, as FANG [31] did, it is inevitable to suffer from error propagation problem: the errors from sub-tasks can propagate to the final fake news detection. Therefore, it is necessary to adopt end-to-end learning to improve the fake news detection performance.

To the best of our knowledge, existing approaches fail to address all three difficulties, so we propose a novel fake news detection framework, *Hetero-SCAN*, to tackle above-listed difficulties. In *Hetero-SCAN*, to preserve multi-level social context information, we use the *Meta-Path*. Meta-Path is a composite relation connecting two node types, aiming to capture the semantics in the heterogeneous graph. We define two Meta-Paths containing different aspects of news (users and publishers) to extract multi-level social context information without information loss. Moreover, Meta-Path instance encoding and aggregation methods are proposed to capture the temporal information of user engagement and learn the news representation end-to-end.

To show that our proposed method outperforms existing solutions, we test *Hetero-SCAN* with two real-world datasets [16, 31],

Table 1: Comparison of *Hetero-SCAN* with exiting graph-based fake news detection methods.

	Multi-level Social Context	Information Preserving	Temporal Information	End-to -end
CSI [38]	✗	✓	✓	✓
SAFER [13]	✗	✗	✗	✓
FANG [31]	✓	✓	✓	✗
AA-HGNN [36]	✗	✓	✗	✓
Hetero-SCAN	✓	✓	✓	✓

and the results show that *Hetero-SCAN* achieves significant improvement over previous approaches. Our code with data is released on the Zenodo [1] for reproducibility. Our major contributions are:

(1) We pose three chronic difficulties in social context aware fake news detection and address them by proposing a novel fake news detection framework, *Hetero-SCAN*.
(2) We conduct diverse experiments on the two real-world fake news datasets, covering the broad definition of fake news (Section 3), and demonstrate that *Hetero-SCAN* shows better performance than existing solutions.
(3) We analyze the temporal behavior differences of engaged users between intentional and unintentional fake news.

2 RELATED WORK

2.1 Fake News Detection

Fake news detection methods can be categorized into two types: content-based and graph-based approaches.

The content-based approach models the content of the news, such as headline or body text, to detect news authenticity. Some research on content-based approaches utilizes linguistic features such as stylometry, psycholinguistic properties, and rhetorical relations [12, 33, 34, 37]. Researchers also use Multi-modal approaches, the combination of visual and linguistic features to verify the news authenticity [20, 24, 35, 45, 47].

The graph-based approach, also known as the social context aware approach, adds auxiliary information of the user or publisher to model the news. CSI [38] is a framework that aims to capture the information of users and their temporal engagements. CSI, however, did not consider publishers, and the connections between users and news were also ignored. Bi-GCN [11] and SAFER [13] use Graph Convolution Network (GCN) [25] to obtain the news representation with user information. However, they suffer from a information loss since they present news and user information in a homogeneous graph. In other words, they fail to taking the node and relation types into account. Most recently, FANG [31] is proposed to preserve information by dividing the fake news detection task into several sub-tasks, such as textual encoding and stance detection. Nonetheless, dividing into sub-tasks causes the error propagation problem: If the sub-tasks have errors, the errors can propagate up to the final news representation and thereby deteriorate the detection performance. AA-HGNN [36] uses adversarial active learning and

[1]https://doi.org/10.5281/zenodo.6565547

extends Graph Attention Network (GAT) [44] into the heterogeneous graph to learn the news representation with limited data. Information of users and their temporal engagement, however, are not considered in AA-HGNN. Table 1 compares *Hetero-SCAN* and existing fake news detection methods.

2.2 Graph Neural Network

Graph Neural Network, the extension of the deep learning method into graphs, shows its effectiveness in graph-represented data. The first method proposed is Graph Convolutional Network (GCN) [25] which aggregates the features from the adjacent nodes in the graph. To further improve it, some methods adopt the attention mechanism and random work with restart sampling strategy, namely Graph Attention Network (GAT) [44] and GraphSAGE [22].

As these methods are designed for homogeneous graphs, they are not general enough to apply to the heterogeneous graph, so new approaches tailor to heterogeneous graphs are then proposed. To model the multi-relations in the graph, the Relation aware GCN (R-GCN) [39] is proposed first. HetGNN [50] uses a sampling strategy based on random walk with restart and Bi-LSTM to aggregate the node features in the heterogeneous graph. Later, the methods based on Meta-Path and attention mechanism, such as HAN [23] and MAGNN [19], are proposed.

3 PRELIMINARIES

*Definition 3.1 (**Broad Definition of Fake News**).* Fake news is false news.

*Definition 3.2 (**Narrow Definition of Fake News**).* Fake news is intentionally false news published by a news outlet.

The term fake news has recently been defined by the work of Zhou, Xinyi and Reza Zafarani [51]. They define fake news in two scopes, broad and narrow. The broad definition emphasizes the authenticity of the information, and the narrow one emphasizes the intention of the author. In line with most research on fake news detection, we also employed a broad definition of fake news for the evaluation.

*Definition 3.3 (**Heterogeneous Graph**).* A heterogeneous graph is defined as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ associated with a node type mapping function $\phi : \mathcal{V} \to \mathcal{A}$ and an edge type mapping function $\psi : \mathcal{E} \to \mathcal{R}$. $\mathcal{A}$ and $\mathcal{R}$ denotes the predefined sets of node types and edge types, respectively, with $|\mathcal{A}| + |\mathcal{R}| > 2$.

*Definition 3.4 (**Meta-Path**).* A Meta-Path P is defined as a path in the form of $A_1 \xrightarrow{R_1} A_2 \xrightarrow{R_2} \ldots \xrightarrow{R_l} A_l$ (abbreviated as $A_1A_2 \ldots A_l$), which describes a composite relation $R = R_1 \circ R_2 \circ \ldots \circ R_n$ between node types A_l and A_{l+1}, where $\circ$ denotes the composition operator on relations.

*Definition 3.5 (**Meta-Path Instance**).* Given a Meta-Path P of a heterogeneous graph, a Meta-Path instance p of P is defined as a node sequence in the graph following the schema defined by P.

4 METHODOLOGY

4.1 Graph Construction & Feature Engineering

To integrate multi-level social context information, we build a heterogeneous graph of news (Figure 2). The graph consists of three types of nodes (publisher, news, and users) and four types of edges (citation, publication, tweet, and following). Formally, the heterogeneous graph of news is noted as $\mathcal{G}(\mathcal{V}, \mathcal{E})$, and the set of three node types are symbolized as $\mathcal{A} = \{A_p, A_n, A_u\}$.

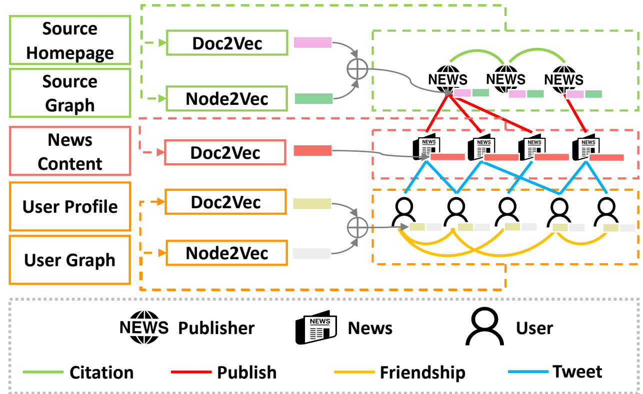


Figure 2: Heterogeneous Graph of News and Node Feature Engineering.

Before utilizing this heterogeneous graph, it is necessary to construct initial node features for three types of nodes in the graph. For news nodes, Doc2Vec [28] is applied to the news article to construct their initial features. The user and publisher nodes, however, need additional information to construct their respective initial features. Users' profiles are used for user nodes feature construction since the importance of the user profiles for detecting news authenticity has been proved by Shu, Kai et al. [42]. The distinct feature of each publisher is acquired from about-us pages on their official websites; If there is no about-us page on the publisher's official website, we use Wikipedia's description instead. Doc2Vec is applied again to leverage these text contents. To also include the structural role they play in their respective networks, we apply Node2Vec [21] to capture user connections and citations among publishers as features. By concatenating the two vectors obtained from Doc2Vec and Node2Vec, we construct the initial features of user and publisher nodes. Figure 2 shows the overall node feature construction process.

4.2 Meta-Path Instance Extraction

After constructing initial node features, we then need to learn the news representation containing multi-level social context and temporal information. Multi-level social context information should be used without loss, which is the first difficulty to overcome. To address this difficulty, we use the concept, Meta-Path (defined in Section 3). Meta-Paths can be used to extract meaningful social context with respect to publishers and users. We define two Meta-Paths that can reflect the method used in actual news factualness verification. When people verify the news authenticity, they need to cross-check both publisher and the news published by this publisher. The same goes for users: User information, as well as the news tweet by the user, needs to be reviewed. From these two intuitions, a set of Meta-Path $\mathcal{P}$ that we defined is:

$$\mathcal{P} \in \{\mathcal{P}_U, \mathcal{P}_S\} \tag{1}$$

where $\mathcal{P}_U$: *News* $\to$ *User* $\to$ *News* and $\mathcal{P}_S$: *News* $\to$ *Publisher* $\to$ *News*.

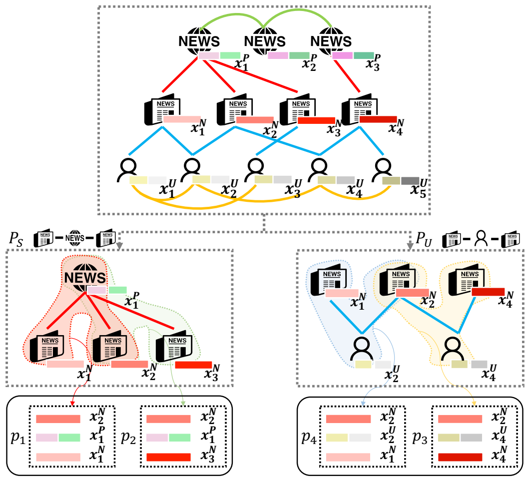


Figure 3: Extracting Meta-Path instances of the target news node x_2^N.

After defining a set of Meta-Path, we extract Meta-Path instances p following each Meta-Path, $\mathcal{P}_S$ or $\mathcal{P}_U$, for each target news node. To efficiently extract Meta-Path instances, we first divide the whole graph into two sub-graphs, which only contain the nodes types specified in the Meta-Path, $\mathcal{P}_S$ or $\mathcal{P}_U$. Then, in each sub-graph, the Meta-Path instances following each Meta-Path are extracted. The corresponding collection of features are fed into *Hetero-SCAN* to get the final representation of the target news node. The sets of instances following two Meta-Path $\mathcal{P}_S$ and $\mathcal{P}_U$ are denoted as $\mathbf{P}_S$ and $\mathbf{P}_U$ respectively. For instance, if we want to extract the Meta-Path instances of the target news node x_2^N in Figure 3, we first divide the whole graph into two sub-graphs. One is composed of news and publisher nodes, and the other is made of news and user nodes. Then, the Meta-Path instances follow Meta-Path $\mathcal{P}_S$ or $\mathcal{P}_U$ are selected from each sub-graph, and the corresponding features of nodes along these Meta-Path instances will be prepared for our model. The Meta-Path instance p_1 is made of features of nodes following Meta-Path $\mathcal{P}_S$: *News* $\rightarrow$ *Publisher* $\rightarrow$ *News*, which is x_1^N, x_1^P and x_2^N in the graph. In the same manner, $p2$, $p3$ and $p4$ are extracted. For the target node v, we use $\mathbf{P}_S$ and $\mathbf{P}_U$ to denote the set of Meta-Path instances follow each Meta-Path. In this case, $\mathbf{P}_S = \{p1, p2\}$ and $\mathbf{P}_U = \{p3, p4\}$ are set of Meta-Path instances of target node x_2^N.

There are usually a large number of users engaged per news in the real world. To cope with this situation, we extract Meta-Path instances from our heterogeneous graph of news with random sampling. Specifically, a certain number of Meta-Path instances are randomly sampled for each news node according to a pre-defined Meta-Path. At last, the Meta-Path instances from the Meta-Path $\mathcal{P}_U$ are sorted chronologically before being fed into the proposed model to make *Hetero-SCAN* aware of chronological information of Meta-Path instances. In the following sections, we assume that the Meta-Path instances from $\mathcal{P}_U$ are sorted in chronological order.

4.3 Model Architecture

Hetero-SCAN takes vectors from the previous step as input and processes them through four steps as shown in Figure 4 to tackle the yet addressed chronic difficulties.

Table 2: Formulation of Encoding Method.

Method	Original	In Our Paper
TransE	$\mathbf{e}_s + \mathbf{e}_p$	$MEAN[(\mathbf{h}_u + r + r^{-1}), (\mathbf{h}_w + r^{-1})]$
ConvE	$[\mathbf{e}_s \parallel \mathbf{e}_p] * \mathbf{W}$	$[\tilde{\mathbf{h}}_u \parallel \tilde{r} \parallel \tilde{\mathbf{h}}_w \parallel \tilde{r}^{-1}] * \mathbf{W}$
RotatE	$\mathbf{e}_s \odot \mathbf{e}_p$	$MEAN[(\mathbf{h}_u \odot r \odot r^{-1}), (\mathbf{h}_w \odot r^{-1})]$

4.3.1 ***Node Feature Transformation.*** The initial node features have different dimensions since different sources and techniques are used in the feature engineering process (Section 4.1). To make them lie in the same latent space, we apply the type-specific linear transform on the features of each type of node. Type-specific transformation refers to the linear projection of a vector into another dimension for each type of node in the graph. The transformed feature for a node $v \in \mathcal{V}_A$ of type $A \in \mathcal{A}$ is:

$$\mathbf{h}_v^A = \mathbf{W}_A \cdot \mathbf{x}_v^A \tag{2}$$

where $\mathbf{x}_v \in \mathbb{R}^{d_A}$ is the initial feature of node v, and $\mathbf{W}_A \in \mathbb{R}^{d' \times d_A}$ is the learnable type-specific weight matrix for node type A.

4.3.2 ***Meta-Path Instance Encoding.*** The first step transformed all the features of the node into the same dimension. We then need to efficiently summarize the Meta-Path instances for the remaining aggregation steps, which is important in capturing temporal information and learning the representation end-to-end. To efficiently encode node features, we adopted the method that shows excellent performance in knowledge graph triple embedding [17, 43, 48].

The major advantage of using knowledge graph triple embedding is the structural similarity between knowledge graph triples and our Meta-Paths. In the knowledge graph, the knowledge graph triple usually refers to the subject, predicate, and object (s, p, o). The Meta-Path we defined is similar to the knowledge graph triple in a sense that Meta-Path is the same format along with one more entity and relation. Formally,

$$\begin{gathered}\text{Knowledge graph triple:}\mathbf{e}_s \xrightarrow{\mathbf{e}_p} \mathbf{e}_o \\ \text{Meta-Path:}\mathbf{h}_u \xrightarrow{r} \mathbf{h}_w \xrightarrow{r^{-1}} \mathbf{h}_v\end{gathered} \tag{3}$$

where v is target node, u and w refer to the nodes along the Meta-Path. Considering the Meta-Path we defined in the Section 4.2, $v \in A_n$, $u \in A_n$, and $w \in \{A_p, A_u\}$. The r and r^{-1} is the relation between u, w and w, v respectively. $\mathbf{h}$ is the transformed embedding of the node as we stated in Section 4.3.1, and $\mathbf{e}$ is the embedding of the knowledge graph triple.

Several research on knowledge graph domain tackle the triple embedding problem [17, 43, 48]. We use TransE [48] as our main encoding method for the proposed model. TransE [48] represents relations as translations, so the object vector $\mathbf{e}_o$ in the triple is considered as a translation of subject vector $\mathbf{e}_s$ on predicate vector $\mathbf{e}_p$. Other than TransE, RotatE [43] and ConvE [17] knowledge graph embedding methods are also examined in our work. Ablation study on different knowledge graph triple embedding methods and their descriptions are provided in the Section 5.5.

In knowledge graph, there are usually explicit features for predicated ($\mathbf{e}_p$ in Equation 3), but in our case, there is no explicit features for the relations (r in Equation 3), so we use learnable embedding vectors to present relations. Inverse relationships, such

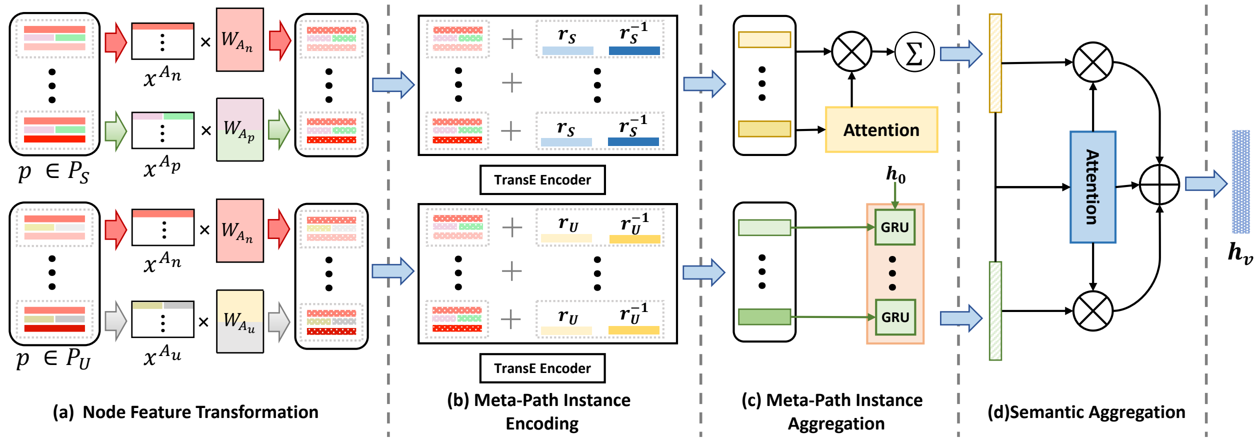


Figure 4: Architecture of *Hetero-SCAN*.

as *Publisher* − *News* and *News* − *Publisher*, are represented by taking the sign inverses. For instance, if we define r as the embedding of *Publisher* − *News* relationship, the inverse relationship, *News*−*Publisher* is $r^{-1} = -r$. Our encoding function f_{enc} is defined as:

$$\mathbf{h}_p = f_{enc}(p) = f_{enc}(\mathbf{h}_u, r, \mathbf{h}_w, r^{-1}) \tag{4}$$

The existing knowledge graph triple embedding methods explained above are designed for two nodes and the relation between them. In our Meta-Path, we have a total of three nodes and two relations in a Meta-Path instance. We deal with this by slightly tuning the formulation to fulfill our needs. The original formulation of knowledge graph triple embedding methods and ours are summarized in Table 2. In this table, the $\tilde{\mathbf{h}}$ means the reshape of vector $\mathbf{h}$ in a 2D form, and the $\odot$ and $\|$ represent the element-wise product and concatenation of vector, respectively.

4.3.3 ***Meta-Path Instance Aggregation***. The encoded vectors from two different Meta-Paths are aggregated by using different methods.

The encoded vectors from Meta-Path $\mathcal{P}_S$: *News* → *Publisher* → *News* contain information of other news from the same publisher. Among the news published by the publisher, not all news will contain valuable information for detection. Thus, the model should 'focus' on some of the news published by this publisher and include this information in the aggregated representation. For each Meta-Path instance $p \in \mathbf{P}_S$:

$$\begin{aligned} e_p &= LeakyReLU(\mathbf{a}^T \cdot \mathbf{h}_p) \\ \alpha_p &= softmax(e_p) = \frac{exp(e_p)}{\sum_{p' \in \mathbf{P}_S} exp(e_{p'})} \end{aligned} \tag{5}$$

where e_p is the attention value calculated by multiplying encoded Meta-Path instance $\mathbf{h}_p$ with attention vector $\mathbf{a} \in \mathbb{R}^{2d'}$, and it is normalized by a softmax function over all Meta-Path instances of the target node v, the result is denoted as α_p above.

To alleviate the effect of the high variance of the data in a heterogeneous graph, we adopt multi-head attention mechanism. K independent attention mechanisms execute the transformation as shown in Equation 6, and their features are concatenated after they pass the activation function σ. The output feature representation can be formulated as:

$$\mathbf{h}_v^{\mathcal{P}_S} = \overset{K}{\underset{k=1}{\|}} \sigma(\sum_{p \in \mathbf{P}_S} [\alpha_p]_k \cdot \mathbf{h}_p) \tag{6}$$

where $[\alpha_p]_k$ is the normalized attention value of Meta-Path instance p of target node v at the k-th attention head.

Temporal information of user engagement is another critical feature to determine the veracity of the given news, and incorporating this information is the second difficulty to resolve. To capture the temporal information, we aggregate the Meta-Path instances follow $\mathcal{P}_U$: *News* → *User* → *News* through Recurrent Neural Network (RNN). Since Meta-Path instances are already encoded in the previous step, we can directly feed them into the RNN. There are usually a large number of users engaged per news, so we choose GRU [14] as our RNN unit to avoid the vanishing or exploding gradients problem.

$$\mathbf{h}_v^{\mathcal{P}_U} = \mathbf{GRU}(\mathbf{h}_{p_1}, \mathbf{h}_{p_2}, ..., \mathbf{h}_{p_n}), p_i \in \mathbf{P_U} \tag{7}$$

The last hidden state of the GRU is used for the downstream task as it is the high-level representation that summarizes the temporal information of the user engagement.

4.3.4 ***Semantic Aggregation***. Two vectors, $\mathbf{h}_v^{\mathcal{P}_S}$ and $\mathbf{h}_v^{\mathcal{P}_U}$, from previous step represents two different aspects of the news. The final news representation is produced by fusing these two vectors, which enables us to learn the news representation end-to-end (the third difficulty). As two Meta-Paths show two different aspects of a given news, the model should be able to weigh the importance of the two aspects with different news. To this end, we adopt another attention mechanism. Before applying the attention mechanism, non-linear transformations are applied to summarize $\mathbf{h}_v^{\mathcal{P}_S}$ and $\mathbf{h}_v^{\mathcal{P}_U}$. Thus for $P \in \{\mathcal{P}_S, \mathcal{P}_U\}$:

$$s_P = \frac{1}{|\mathcal{V}|} \sum_{v \in \mathcal{V}} tanh(\mathbf{M}_A \cdot \mathbf{h}_v^P + \mathbf{b}_A) \tag{8}$$

Here, $\mathbf{M}_A \in \mathbb{R}^{d_m \times d'}$ and $\mathbf{b} \in \mathbb{R}^{d_m}$ is a learnable weight matrix and bias vector. $\mathcal{V}$ is the set of news nodes.

Then we apply the attention mechanism to aggregate two vectors to obtain our final news representation $\mathbf{h}_v$.

$$
\begin{aligned}
e_P &= tanh(q^T \cdot s_P) \\
\beta_P &= \frac{exp(e_P)}{\sum_{P' \in \mathcal{P}} exp(e_{P'})} \\
\mathbf{h}_v &= \sum_{P \in \mathcal{P}} \beta_P \cdot \mathbf{h}_v^P
\end{aligned}
\tag{9}
$$

where $q \in \mathbb{R}^{d_m}$ is the attention vector and β_P is the normalized importance of Meta-Path P.

4.4 Training

The final representation of the target news vector is passed to the classification layer to get the classification result. During training, our predictions and labels are used to calculate the loss, and we update the learnable parameters of the model by using the back-propagation algorithm. The loss function used in *Hetero-SCAN* is cross-entropy loss, which is:

$$\mathcal{L} = -\sum y \log \mathbf{P}_{fake} + (1-y) \log \mathbf{P}_{real} \tag{10}$$

5 EXPERIMENTAL RESULT AND ANALYSIS

5.1 Dataset and Settings

To test the effectiveness of our method, we conducted our experiments with two real-world datasets: FANG [31] and FakeHealth [16]. The dataset FANG is composed of standard benchmark datasets that are widely used in rumor and news classification study [26, 29, 40]. The original news content was obtained through the provided news url, and for the 100 news urls that did not have the news content available, resorted to manually searching the news title for the content. From provided tweet ids, users and their profiles on Twitter could be found through the Twitter API [8]. The labels of the news in FANG are obtained from two well-known fact-checking websites: Snopes [7] and PolitiFact [4]. FakeHealth is another publicly available benchmark dataset for fake news detection, mainly focused on healthcare. The dataset consists of two subsets, HealthStory and HealthRelease; HealthStory was used in our study due to the number of news articles in HealthRelease being too small. HealthStory is collected from the healthcare information review website HealthNewsReviews [2]. On this website, the professional reviewers gave scores of 1 to 5 for each news. Similar to the original study that published the FakeHealth dataset, an article is considered as fake if the score is less than three and real otherwise. The detailed statistics of the dataset used in our experiment are listed in Table 3.

Table 3: Dataset Statistics.

	FANG	HealthStory
# News	1,054	1,638
# Fake News	448	460
# Real News	606	1,178
# Users	52,357	63,723 (sampled)
# of Users per News	71.9	227.26
# Publishers	442	31

In each dataset, we used 70% of news articles as our training set, and the remaining 30% of news articles are further divided into equal sizes of validation and test set. For the hyper-parameters, the transformed hidden dimension and the learning rate are set to 512 and 0.0001, respectively. The early-stopping training strategy with patience 20 is adopted to avoid overfitting.

In addition, in the testing phase of the experiment, *Hetero-SCAN* could encounter new publishers or users, and we cannot make Node2Vec features for them because the Node2Vec model is incapable of generating embeddings for new nodes unless we rerun the Node2Vec for the graph including those new nodes. Therefore, in our experiment, only for seen users and publishers, we input the Node2Vec features calculated in the training step; for new users and publishers in the testing phase, we set all Node2Vec features to 0-vectors to guarantee a realistic test setting.

5.2 Evaluation of ML Algorithms on News Embedding

We trained *Hetero-SCAN* by connecting the output representation to a fully connected layer to classify the news. After training, we evaluated our news representation with five classical machine learning baselines, such as Naive Bayes, Logistic Regression, etc. The metrics used for comparison are precision, recall, accuracy, F1 score, and AUC score, and the evaluation results are summarized in Table 4.

As shown in Table 4, the trained classification layer gives relatively better results than the other machine learning algorithms in terms of F1 score and accuracy because the classification layer is optimized by the classification objective (cross-entropy loss). In terms of AUC score, SVM gives the best result in the FANG dataset, but Logist Regression gives the best result in the HealthStory dataset. Based on this, SVM and Logistic Regression are chosen as the classification algorithms for their respective dataset for subsequent evaluations. Note that **regardless of downstream classification methods, *Hetero-SCAN* surpass any existing fake news detection methods** (details in Section 5.3). In the dataset - HealthStory, *Hetero-SCAN* does not give an ideal result. The explanation for the result on the HealthStory dataset is further discussed in Section 6.2.

5.3 Comparison with Existing Methods

To show the efficacy of the proposed *Hetero-SCAN*, we compared *Hetero-SCAN* with existing fake news detection methods. The benchmarked detection methods can be categorized into text-based and graph-based approaches. For text-based approach, we use three different document embedding methods, TF-IDF, LIWC [32], and Doc2Vec [28], combined with SVM as baselines; and several representative graph-based fake news detection frameworks [13, 31, 36, 38] are also compared in this experiment. The explanations of aforementioned fake news detection methods are listed below.

- **TF-IDF + SVM**: TF-IDF is short for term frequency-inverse document frequency. It is intended to represent the importance of a word in a document. Feature vectors were extracted based on news article contents with TF-IDF, and SVM is applied to it.
- **LIWC** [32] **+ SVM**: LIWC stands for Linguistic Inquiry and Word Count. It is widely used to extract words falling into psychologically meaningful categories, and these words can be used to compose a feature vector.

Table 4: Detection result of *Hetero-SCAN* on two real-word dataset: FANG and FakeHealth. Bold numbers denote the best value in average, and underscored numbers denote the smallest variation (± stands for 95% confidence interval). The classification method with highest AUC score, was pointed out by ⋆ and was selected for the subsequent evaluation.

Dataset	Classification Method	Precision	Recall	F1 Score	Accuracy	AUC Score
FANG	Classification Layer	**0.845**±0.052	**0.843**±0.054	**0.843**±0.053	**0.843**±0.054	0.839±0.048
	Naive Bayes	0.839±0.053	0.837±0.058	0.835±0.057	0.837±0.058	0.840±<u>0.041</u>
	Logistic Regression	0.835±0.054	0.835±0.054	0.835±0.054	0.835±0.054	0.907±0.058
	⋆ **SVM**	0.832±0.036	0.839±0.053	0.840±0.053	0.839±0.053	**0.910**±0.047
	Random Forest	0.832±<u>0.036</u>	0.831±<u>0.037</u>	0.831±<u>0.037</u>	0.831±<u>0.037</u>	0.900±0.057
	AdaBoost	0.811±0.070	0.807±0.076	0.808±0.075	0.807±0.076	0.881±0.056
HealthStory	Classification Layer	0.529±<u>0.093</u>	**0.717**±<u>0.003</u>	0.599±<u>0.008</u>	**0.717**±<u>0.003</u>	0.500±<u>0.003</u>
	Naive Bayes	0.662±0.139	0.600±0.244	0.573±0.289	0.633±0.131	0.508±0.177
	⋆ **Logistic Regression**	0.660±0.065	0.595±0.206	0.594±0.185	0.584±0.180	**0.557**±0.076
	SVM	0.649±0.094	0.620±0.137	**0.612**±0.089	0.623±0.137	0.536±0.108
	Random Forest	**0.674**±0.117	0.550±0.272	0.526±0.327	0.520±0.269	0.513±0.134
	AdaBoost	0.656±0.129	0.539±0.302	0.492±0.303	0.540±0.301	0.554±0.076

Table 5: Comparison of AUC scores with existing methods. The AUC scores of CSI and FANG are from Nguyen, Van-Hoang, et al. [31]. FANG experiment on HealthStory dataset cannot be conducted since it needs additional labels.

Category	Method	FANG	HealthStory
Text-based	TF.IDF + SVM	0.735	0.526
	LIWC + SVM	0.511	0.534
	Doc2Vec + SVM	0.554	0.582
Graph-based	CSI	0.741	-
	SAFER	0.669	0.615
	FANG	0.750	-
	AA-HGNN	0.654	0.559
GNN-baselines	GCN	0.633	0.528
	GAT	0.630	0.541
	GraphSAGE	0.773	0.589
	R-GCN	0.753	0.500
	HAN	0.658	0.600
Hetero-SCAN	*w/ temporal*	**0.910**	0.557
	w/o temporal	0.823	**0.636**

- **Doc2Vec** [28] **+ SVM**: Doc2Vec is an unsupervised paragraph embedding technique based on Word2Vec [30]. It uses skip-gram to learn the representation vector.
- **SAFER** [13]: SAFER uses GCN and pre-trained RoBERTa model to embed news nodes in the heterogeneous graph. They concatenate two vectors and apply Logistic Regression.
- **CSI** [38]: CSI uses deep learning-based method to model the response, text, and user engagement of the news. The representation of response and text is concatenated with the user vector.
- **FANG** [31]: FANG divides the detection task into several sub-tasks, and the final detection object is optimized by aggregated loss function of each sub-task.
- **AA-HGNN** [36]: AA-HGNN uses active learning to tackle the limited training data problem and extends GAT [44] to learn the news representation in the graph.

Hetero-SCAN is also compared with some Graph Neural Network (GNN) methods to show that *Hetero-SCAN* performs better than just simply applying the GNN on the graph. The basic GNN methods [22, 25, 44], as well as the methods tailored to the heterogeneous graph, are compared [23, 39]. The brief descriptions of GNN baselines we compared with are listed below.

- **GCN** [25]: GCN is a deep learning based method on a graph-structured data. Each node is learned by aggregating the feature information from its neighbors and the feature of itself.
- **GAT** [44]: GAT is similar to GCN, but it introduces the attention mechanism to replace the statically normalized convolution operation in GCN.
- **GraphSAGE** [22]: GraphSAGE is a general inductive framework that learns a node representation by sampling its neighbors and aggregating features of sampled nodes.
- **R-GCN** [39]: R-GCN is an application of the GCN framework for modeling relational data. In R-GCN, edges can represent different relations.
- **HAN** [46]: HAN is an extension of GAT on the heterogeneous graph. Meta-Path extraction strategy and attention mechanism are adopted to learn the representation of a node.

The results of Table 5 indicates that *Hetero-SCAN* outperforms existing text-based or graph-based fake news detection methods. Because these existing approaches cannot produce representation with rich social context and temporal information as *Hetero-SCAN* do, i.e., they fail to tackle all three difficulties. CSI and SAFER, for example, did not use multi-level social context, and they also incurred some information loss as they ignored the node and relation types. AA-HGNN, including SAFER, miss temporal information in the news representation. AA-HGNN also did not use users as social context. FANG performs better than these methods since it tries to preserve multi-level social context and temporal information.

Table 6: Comparison of AUC score against other fake news detection methods by varying the size of the training data. (-t) and (t) refer to *Hetero-SCAN* without and with temporal information, respectively.

	10%	30%	50%	70%	90%
CSI	0.636	0.671	0.670	0.689	0.691
SAFER	0.546	0.689	0.666	0.692	0.669
FANG	0.669	0.704	0.717	0.723	0.752
AA-HGNN	0.573	0.598	0.656	0.657	0.642
Hetero-SCAN$_{(-t)}$	0.594	0.707	0.776	0.749	0.751
Hetero-SCAN$_{(t)}$	**0.764**	**0.835**	**0.878**	**0.889**	**0.900**

Nevertheless, to preserve information, FANG divides the fake news detection task into several sub-tasks, and each sub-task deals with certain information. Dividing into several sub-tasks is ineffective because errors in sub-task will be propagated up to the final news representation and thus harm the detection performance. As such, the result emphasizes the importance of resolving the proposed three difficulties in fake news detection.

For GNN baselines, the graph embedding methods made for homogeneous graphs, such as GCN, GAT, and GraphSAGE, did not give ideal results since node types and relations are ignored in these cases. R-GCN and HAN, which are designed for heterogeneous graph, also has no significant improvement, which implies that *Hetero-SCAN* is better than a simple application of these graph embedding methods on the heterogeneous graph of news. The failure of GNN baselines target on the heterogeneous graph can attribute to the missing temporal information of user engagement, which is the second difficulty that needs to be resolved in the social context-aware fake news detection.

5.4 Limited training data

Normally, the fake news dataset has limited training data due to the large-scale requirement of human labor, so the model should work well in the circumstance of limited training samples. To show that *Hetero-SCAN* outperforms existing methods given the circumstance of scarce training data, we gradually enlarge the training data, from 10% to 90%, and compare the fake news detection result with existing methods. Table 6 shows the comparison result.

The AUC score of *Hetero-SCAN* achieves over 0.8 with only 30% of training data and even outperforms the rest of the methods with 90% of the training data. AA-HGNN is designed to overcome the scarcity of training data issues in the fake news detection task, but *Hetero-SCAN* is still better than AA-HGNN even when the size of training data is small.

5.5 Ablation study on Meta-Path Instance Encoding Methods

In Section 4.3.2, we propose to use knowledge triple embedding methods to encode Meta-Path instances, and we adopt TransE in *Hetero-SCAN*. We wanted to examine the performance differences by changing the Meta-Path encoding method to other knowledge triple embedding methods, RotatE and ConvE. Descriptions of the three encoding methods are introduced below.

- **TransE** [48]: The TransE model represents relations as translation from the subject entity to the object entity.
- **RotatE** [43]: The RotatE model maps the entities and relations to the complex vector space and defines each relation as a rotation from the subject entity to the object entity.
- **ConvE** [17]: The ConvE model uses 2D convolution over embedding and multiple layers of nonlinear features to mode knowledge graphs. It reshapes the embedding of subject and predicates in a 2D form and apply the convolution on it.

Table 7: Performance of detection result when apply different Meta-Path encoding method. Bold texts indicate the highest value.

	F1 Score	Accuracy	AUC
TransE	**0.840**±0.053	**0.839**±0.053	**0.910**±0.047
RotatE	0.799±0.035	0.799±0.036	0.862±0.035
ConvE	0.532±0.174	0.526±0.079	0.665±0.021

To show the performance differences when different knowledge triple embedding methods are applied, F1 score, Accuracy, and AUC score were measured.

Table 7 indicates that TransE gives better results than the others. The reason can be drawn from the fact that TransE requires fewer parameters and operations than RotatE and ConvE. With limited training data, complex models are easy to suffer from over-fitting, which will cause performance degradation.

6 DISCUSSION

6.1 Inductiveness of *Hetero-SCAN*

A deep learning based approach dealing with graph-structured data should have generality to produce practical predictions for unseen data. A method is an inductive approach if it can generate embeddings for the nodes that were not seen during training. In contrast, it is called a transductive approach if the method cannot generate embeddings for the nodes appearing in the testing phase for the first time. For example, GCN is inductive, whereas Node2Vec is transductive.

In graph-based fake news detection, unseen nodes can appear in the testing phase. It might be newly published news, new publishers, or new users. Some approaches using matrix decomposition [38, 41] are not able to generate embedding for newly published news with social context information.

In *Hetero-SCAN*, however, the learnable parameters in our model are used after Meta-Path extraction with random sampling, and they are shared by all nodes. Therefore, *Hetero-SCAN* can generate news embeddings that are not seen during the training. Although we use Node2Vec in our feature engineering process, *Hetero-SCAN* can still be considered as an inductive framework. If we consider the actual adoption of our framework in the wild, it is not difficult for administrators to maintain a graph of users or publishers so that administrators can generate Node2Vec embeddings for them. Just like the evaluation settings mentioned in Section 5.1, the administrator can use the Node2Vec features from the existing graph; and for the new users and publishers, the administrator can set all the Node2Vec features to 0-vectors. Since our model performs well in this way as can be seen in the evaluation results (Section 5.3), we insist that our model is still inductive.

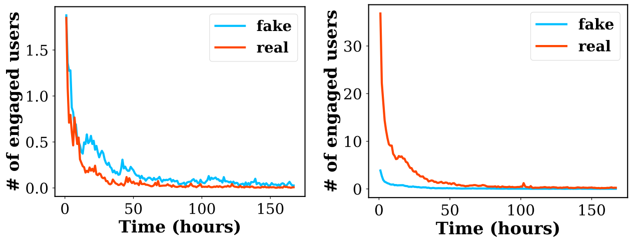


Figure 5: Comparison of temporal behaviors on two datasets. Both figures show the # of engagements (tweets) per news vs. time (hours) for FANG (left) and HealthStory (right).

6.2 Misinformation vs Disinformation

As mentioned in Section 3, we use the broad definition of fake news, that is, fake news in this paper including both false news with and without intention. According to the definition by Wardle et al. [49], the fake news can be further divided into *disinformation* and *misinformation*, which are false news with the intention of causing harm and false news without intention, respectively.

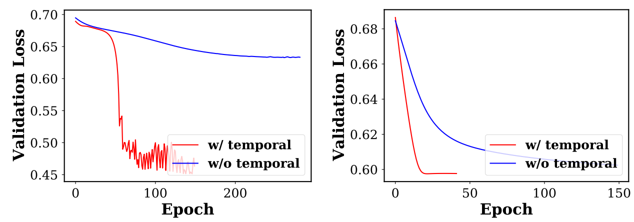
FANG dataset used in our experiment is mainly composed of news checked from PolitiFact and Snopes, which are political-related fact-checking websites. Thus, most of the fake news in this dataset is either partisan-biased news or some false information to demean certain politicians, which could be considered as information intended to harm the specific person or the organizations. Hence, it is likely that the fake news in this dataset is *disinformation*. On the contrary, the news in HealthStory is collected and fact-checked from Health News Review, which evaluates and rates the completeness and accuracy of news regarding to medical treatments, health care journalism, etc. By our design of considering lower score reviews as fake news in HealthStory, the nature of these news tends to be insufficient and innocently false, so we can regard them as *misinformation*. This characteristic difference between the two datasets, which is potentially related to the author's intention, can affect the model's detection performance. Thus, we conduct extended experiments to reveal the difference and its impact on our proposed model.

Figure 5 compares the number of engaged users along with the time to see how people react to fake and real news. The fake news in FANG has many periodic spikes, which means the users periodically talk about fake news. On the contrary, the fake news in HealthStory does not have any periodic spikes and converges to zero not long after the news is published, which is similar to the real news. As such, the temporal information of user engagement shows some differences between FANG and HealthStory dataset.

Also, to see the impact of temporal information in *Hetero-SCAN*, we replace the RNN in *Hetero-SCAN* with attention mechanism. In other words, we checked the detection performance difference between the *Hetero-SCAN* with and without temporal information. The evaluation result on the datasets can be found in Table 8. The results show that the RNN based approach performs better than the other one in FANG dataset, but for the HealthStory dataset, the performance is better when the attention is applied. Furthermore, in FANG dataset, the validation loss of *Hetero-SCAN* with RNN converges much faster than the one with the attention mechanism; by contrast, the convergence speed of the two approaches is similar in the HealthStory dataset. (See Figure 6 in Appendix)

Table 8: Performance of the *Hetero-SCAN* with and without temporal information.

Dataset	*Hetero-SCAN*	F1	Accuracy	AUC
FANG	*w/ temporal*	**0.840**	**0.839**	**0.910**
	w/o temporal	0.759	0.760	0.823
HealthStory	*w/ temporal*	0.594	0.584	0.557
	w/o temporal	**0.614**	**0.595**	**0.636**



(a) FANG **(b) HealthStory**

Figure 6: Validation loss during training. (Red and blue lines indicate the validation loss of *Hetero-SCAN* with and without temporal information, respectively.)

To sum up, the fake news in two dataset show some temporal behavior differences. In the dataset that has temporal behavior differences (FANG), *Hetero-SCAN* with RNN not only improves the detection performance but also accelerates the learning speed.

6.3 Limitation and Future Work

As expected, a single news article is engaged with by a large number of users. Using every single user's information as a feature is therefore impractical, and we eventually used simple random sampling to select a certain number of users. Hence, an improved method of screening important users is necessary for fake news detection to overcome the limitation. In addition, to apply the proposed method, we must first identify the relevant tweets for particular news. Since this paper focuses primarily on identifying the news in the context in which news and related tweets are given, finding relevant tweets for particular news is left as future work.

7 CONCLUSIONS

Fake news is a critical social problem threatening many aspects of the general public's lives. We pose three difficulties in social context aware fake news detection and address them by proposing an inductive fake news detection framework *Hetero-SCAN*. Our model overcomes the shortcomings of the previous graph-based approaches and exhibits state-of-the-art performance. We also study the propagation properties of different types of fake news (*misinformation* and *disinformation*) and their impact on our *Hetero-SCAN*. We believe *Hetero-SCAN* can be of aid in future studies not only residing to fake news detection but also various events concerning fake news.

ACKNOWLEDGMENTS

This research was supported by the Engineering Research Center Program through the National Research Foundation of Korea (NRF) funded by the Korean Government MSIT (NRF-2018R1A5A1059921)

REFERENCES

[1] 2020. Coronavirus: The viral rumours that were completely wrong. https://www.bbc.com/news/blogs-trending-53640964.

[2] 2020. Health News Review. https://www.healthnewsreview.org/.

[3] 2020. Left bias Publishers checked by MBFC. https://mediabiasfactcheck.com/left/.

[4] 2020. PolitiFact. https://www.politifact.com/.

[5] 2020. Questionable Publishers checked by MBFC. https://mediabiasfactcheck.com/fake-news/.

[6] 2020. Right bias Publishers checked by MBFC. https://mediabiasfactcheck.com/right/.

[7] 2020. Snopes. https://www.snopes.com/.

[8] 2020. Twitter API. https://developer.twitter.com/en/docs/twitter-api.

[9] 2020. US election 2020: Fact-checking Trump team's main fraud claims. https://www.bbc.com/news/election-us-2020-55016029.

[10] 2021. Facebook Media: Working to Stop Misinformation and False News. https://www.facebook.com/formedia/blog/working-to-stop-misinformation-and-false-news.

[11] Tian Bian, Xi Xiao, Tingyang Xu, Peilin Zhao, Wenbing Huang, Yu Rong, and Junzhou Huang. 2020. Rumor detection on social media with bi-directional graph convolutional networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 549–556.

[12] Carlos Castillo, Marcelo Mendoza, and Barbara Poblete. 2011. Information credibility on twitter. In *Proceedings of the 20th international conference on World wide web*. 675–684.

[13] Shantanu Chandra, Pushkar Mishra, Helen Yannakoudakis, and Ekaterina Shutova. 2020. Graph-based Modeling of Online Communities for Fake News Detection. *arXiv preprint arXiv:2008.06274* (2020).

[14] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555* (2014).

[15] Niall J Conroy, Victoria L Rubin, and Yimin Chen. 2015. Automatic deception detection: methods for finding fake news. In *Proceedings of the 78th ASIS&T Annual Meeting: Information Science with Impact: Research in and for the Community*. 1–4.

[16] Enyan Dai, Yiwei Sun, and Suhang Wang. 2020. Ginger cannot cure cancer: Battling fake health news with a comprehensive data repository. In *Proceedings of the International AAAI Conference on Web and Social Media*, Vol. 14. 853–862.

[17] Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. 2018. Convolutional 2d knowledge graph embeddings. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.

[18] Song Feng, Ritwik Banerjee, and Yejin Choi. 2012. Syntactic stylometry for deception detection. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. 171–175.

[19] Xinyu Fu, Jiani Zhang, Ziqiao Meng, and Irwin King. 2020. MAGNN: metapath aggregated graph neural network for heterogeneous graph embedding. In *Proceedings of The Web Conference 2020*. 2331–2341.

[20] Anastasia Giachanou, Guobiao Zhang, and Paolo Rosso. 2020. Multimodal Fake News Detection with Textual, Visual and Semantic Information. In *International Conference on Text, Speech, and Dialogue*. Springer, 30–38.

[21] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*. 855–864.

[22] William L. Hamilton, Rex Ying, and Jure Leskovec. 2017. Inductive Representation Learning on Large Graphs. In *NIPS*.

[23] Yi Han, Shanika Karunasekera, and Christopher Leckie. 2020. Graph neural networks with continual learning for fake news detection from social media. *arXiv preprint arXiv:2007.03316* (2020).

[24] Dhruv Khattar, Jaipal Singh Goud, Manish Gupta, and Vasudeva Varma. 2019. Mvae: Multimodal variational autoencoder for fake news detection. In *The World Wide Web Conference*. 2915–2921.

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

[26] Elena Kochkina, Maria Liakata, and Arkaitz Zubiaga. 2018. PHEME dataset for Rumour Detection and Veracity Classification. https://doi.org/10.6084/m9.figshare.6392078.v1

[27] Sejeong Kwon, Meeyoung Cha, Kyomin Jung, Wei Chen, and Yajun Wang. 2013. Prominent features of rumor propagation in online social media. In *2013 IEEE 13th international conference on data mining*. IEEE, 1103–1108.

[28] Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *International conference on machine learning*. PMLR, 1188–1196.

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

[30] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

[31] 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.

[32] James W Pennebaker, Ryan L Boyd, Kayla Jordan, and Kate Blackburn. 2015. *The development and psychometric properties of LIWC2015*. Technical Report.

[33] Verónica Pérez-Rosas, Bennett Kleinberg, Alexandra Lefevre, and Rada Mihalcea. 2018. Automatic Detection of Fake News. In *Proceedings of the 27th International Conference on Computational Linguistics*. 3391–3401.

[34] Martin Potthast, Johannes Kiesel, Kevin Reinartz, Janek Bevendorff, and Benno Stein. 2018. A Stylometric Inquiry into Hyperpartisan and Fake News. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 231–240.

[35] Shengsheng Qian, Jinguang Wang, Jun Hu, Quan Fang, and Changsheng Xu. 2021. Hierarchical multi-modal contextual attention network for fake news detection. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 153–162.

[36] Yuxiang Ren, Bo Wang, Jiawei Zhang, and Yi Chang. 2020. Adversarial active learning based heterogeneous graph neural network for fake news detection. In *2020 IEEE International Conference on Data Mining (ICDM)*. IEEE, 452–461.

[37] Victoria L Rubin and Tatiana Lukoianova. 2015. Truth and deception at the rhetorical structure level. *Journal of the Association for Information Science and Technology* 66, 5 (2015), 905–917.

[38] Natali Ruchansky, Sungyong Seo, and Yan Liu. 2017. Csi: A hybrid deep model for fake news detection. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. 797–806.

[39] Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *European semantic web conference*. Springer, 593–607.

[40] Kai Shu, Deepak Mahudeswaran, Suhang Wang, Dongwon Lee, and Huan Liu. 2018. FakeNewsNet: A Data Repository with News Content, Social Context and Dynamic Information for Studying Fake News on Social Media. *arXiv preprint arXiv:1809.01286* (2018).

[41] Kai Shu, Suhang Wang, and Huan Liu. 2019. Beyond news contents: The role of social context for fake news detection. In *Proceedings of the twelfth ACM international conference on web search and data mining*. 312–320.

[42] Kai Shu, Xinyi Zhou, Suhang Wang, Reza Zafarani, and Huan Liu. 2019. The role of user profiles for fake news detection. In *Proceedings of the 2019 IEEE/ACM international conference on advances in social networks analysis and mining*. 436–439.

[43] Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. 2019. Rotate: Knowledge graph embedding by relational rotation in complex space. *arXiv preprint arXiv:1902.10197* (2019).

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

[45] Nguyen Vo and Kyumin Lee. 2021. Hierarchical Multi-head Attentive Network for Evidence-aware Fake News Detection. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. 965–975.

[46] Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Yanfang Ye, Peng Cui, and Philip S Yu. 2019. Heterogeneous graph attention network. In *The World Wide Web Conference*. 2022–2032.

[47] Yaqing Wang, Fenglong Ma, Zhiwei Jin, Ye Yuan, Guangxu Xun, Kishlay Jha, Lu Su, and Jing Gao. 2018. Eann: Event adversarial neural networks for multi-modal fake news detection. In *Proceedings of the 24th acm sigkdd international conference on knowledge discovery & data mining*. 849–857.

[48] Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 28.

[49] Claire Wardle and Hossein Derakhshan. 2017. Information disorder: Toward an interdisciplinary framework for research and policy making. *Council of Europe report* 27 (2017), 1–107.

[50] Chuxu Zhang, Dongjin Song, Chao Huang, Ananthram Swami, and Nitesh V Chawla. 2019. Heterogeneous graph neural network. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 793–803.

[51] 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.