

Received June 6, 2021, accepted July 19, 2021, date of publication July 26, 2021, date of current version August 5, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3100245

# MVAN: Multi-View Attention Networks for Fake News Detection on Social Media

SHIWEN NI<sup>1</sup>, (Student Member, IEEE), JIAWEN LI<sup>1</sup>, AND HUNG-YU KAO, (Member, IEEE)

Department of Computer Science and Information Engineering, National Cheng Kung University, Tainan 701401, Taiwan

Corresponding author: Hung-Yu Kao (hykao@mail.ncku.edu.tw)

This work was supported in part by Qualcomm through the Taiwan University Research Collaboration Project, and in part by the Ministry of Science and Technology, Taiwan, under Grant MOST 109-2221-E-006-173 and Grant NCKU B109-K027D.

**ABSTRACT** Fake news on social media is a widespread and serious problem in today's society. Existing fake news detection methods focus on finding clues from Long text content, such as original news articles and user comments. This paper solves the problem of fake news detection in more realistic scenarios. Only source tweet and its retweet users are provided without user comments. We develop a novel neural network based model, Multi-View Attention Networks (MVAN) to detect fake news and provide explanations on social media. The MVAN model includes text semantic attention and propagation structure attention, which ensures that our model can capture information and clues both of source tweet content and propagation structure. In addition, the two attention mechanisms in the model can find key clue words in fake news texts and suspicious users in the propagation structure. We conduct experiments on two real-world datasets, and the results demonstrate that MVAN can significantly outperform state-of-the-art methods by 2.5% in accuracy on average, and produce a reasonable explanation.

**INDEX TERMS** Fake news detection, graph attention networks, attention, deep learning, social media.

## I. INTRODUCTION

With the rapid development of social media platforms, such as Twitter, fake news can spread rapidly on the internet and affect people's lives and judgment. On April 27, 2020, the president of the U.S.A Donald Trump said, "Fake news, the enemy of the people!" Those words indicate that fake news has been a serious social problem. Fake news refers to false statements and rumors on social media, including completely false information or gross misrepresentation of a real event. However, due to the limitations of expertise, time or space, it is difficult for ordinary people to separate fake news from the vast amount of information available online. Therefore, it is necessary to develop automated and auxiliary methods to detect fake news at an early stage. With the development of artificial intelligence (AI), many researchers have attempted to apply AI technology to automatically detect fake news [2].

Early research on automatic detection of fake news mainly focused on designing effective features from various information sources, including text content [2]–[4], publisher's personal information [2], [5] and communication

mode [6]–[8]. However, these feature-based methods are very time-consuming and labour-intensive. In addition, the performance of the model is very dependent on the quality of the artificial features, thus, the performance of this method is not ideal in most cases.

Driven by the success of deep neural networks, several recent studies [9], [10] have applied various neural network models to fake news detection. For example, a recurrent neural network (RNN) [11] is used to learn the representation of the tweet text on the posting timeline. Liu [9] modelled the propagation path as a multivariate time series and applied a combination of RNN and convolution neural network (CNN) to capture the changes of user characteristics along the propagation path. The main limitation of these methods is that they can only process sequence data but cannot process structured data, leading to the inability to properly simulate the real propagation structure.

We know that the dissemination structure of the news on social media can constitute a social network graph. Generally, tweets can be reposted by any other user. The structure of tweet propagation composed of retweet users is shown in Fig. 1. With the help of social media, a piece of Twitter news can be spread all over the world in a very short time. To capture the information hidden in the sequence text and

The associate editor coordinating the review of this manuscript and approving it for publication was Alba Amato<sup>1</sup>.

the structured propagation graph at the same time, RNN and graph neural network (GNN) were used to process these two kinds of data. Meanwhile, to make the model have better learning ability and certain interpretability, two different attention mechanisms, text semantic attention and propagation structure attention, were added to RNN and GNN.

The main contributions of this paper are summarized as follows:

(1) To the best of our knowledge, we are the first to adopt graph attention networks (GATs) to encode and represent the propagation structure of news.

(1) Experimental results on two real-world datasets show that the multi-view attention networks (MVAN) model achieves the highest accuracy and outperforms state-of-the-art models.

(3) Our model is more robust in early fake news detection and the model has some interpretability in both perspectives of text and propagation structure.

## II. RELATED WORK

The goal of fake news detection is to distinguish the authenticity of news published on social media platforms based on their relevant information (such as text content, comments, propagation structure, etc.). Related works can be divided into different categories as follows.

### A. FEATURE-BASED METHODS

Some early studies focused on fake news detection based on handcrafted features. These features are mainly extracted from text content and users' profile information. Castillo *et al.* (2011) [2] proposed a decision tree-based model, utilizing a large number of features for fake news detection on Twitter. Yang *et al.* (2012) [5] created two new features to enrich the feature set of previous researchers: client-based features and location-based features. They are used to automatically detect fake news on Sina Weibo. Wu *et al.* (2015) [12] used a propagation structure composed of 23 features in the hybrid support vector machines (SVM). These features are divided into three categories (message-based features, user-based features and retransmission-based features). Wu *et al.* (2017) [13] proposed a machine learning model based on time series fitting of tweet volume time characteristics. Ma *et al.* (2015) [11] proposed an SVM model that engineers each of the social context features. Rath *et al.* (2017) [14] extracted user information and combined them with an RNN model.

### B. CONTENT-BASED METHODS

Content-based methods rely on the text content to detect the truthfulness of the news article. Ma *et al.* (2015) [11] and Yu *et al.* (2016) [15] combined text and RNN or CNN for fake news detection. Chen *et al.* (2018) [16] combined the attention mechanism with the text to detect fake news at an early age. Liu *et al.* (2018) [9] used both RNN and CNN to encode the propagation structure for early fake news detection. Ajao *et al.* (2018) [17] proposed a hybrid CNN-long-

**TABLE 1. Comparison of recent related studies. Column notations: source news texts (ST), response comments (RC), user features (UF), structural information (SI), model explainability (ME), attention mechanism(AT) and multiple attention mechanisms (MA).**

Works	ST	RC	UF	SI	ME	AT	MT
BiLSTM [43]	✓	✓					
RCNN [45]	✓		✓	✓			
CSI [44]	✓	✓	✓				
dEFEND [20]	✓	✓			✓	✓	
GCAN [26]	✓		✓	✓	✓	✓	
G-SEGA [27]	✓	✓		✓			
Our work	✓		✓	✓	✓	✓	✓

short term memory (LSTM) model for fake news detection on Twitter. Yu *et al.* (2019) [18] proposed an attention-based convolutional approach for text authenticity detection. Shu *et al.* (2019) [19] proposed a sentence-comment co-attention sub-network to use both news contents and user comments for fake news detection and used an attention mechanism to provide explainability.

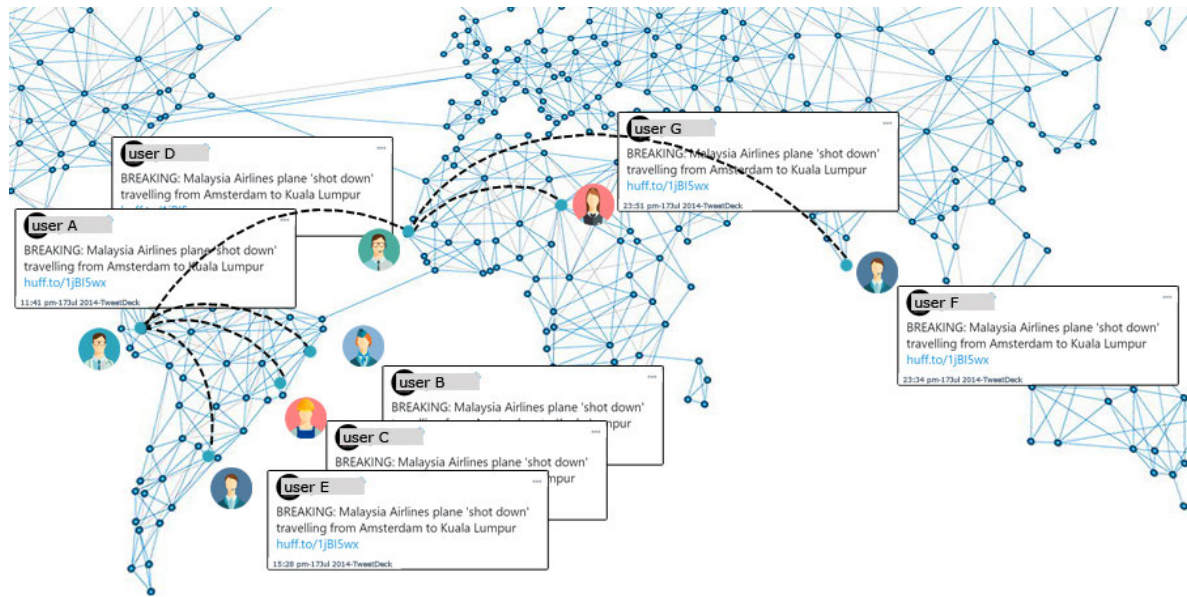
### C. STRUCTURE-BASED METHODS

Unstructured methods cannot handle structured data. In recent years, researchers have proposed some new approaches to use structural information. Ma *et al.* (2018) [21] constructed a recursive neural network to handle conversational structure. This model generates a tree structure by bottom-up or top-down propagation. Monti *et al.* (2019) [22] proposed propagation-based fake news detection using graph convolutional networks (GCN). Nguyen (2019) [23] detected fake news using a multimodal social graph. Li *et al.* (2020) [20] combined objective information and subjective factors for rumor detection. Bian *et al.* (2020) [24] proposed a novel bi-directional graph neural networks model, bi-directional graph convolutional networks (Bi-GCN), to explore both characteristics by operating on both top-down and bottom-up propagation of fake tweets. Lu and Li (2020) [25] developed graph-aware co-attention networks (GCAN) to detect fake news, which generated an explanation by highlighting the evidence on suspicious re-tweeters and the words of concern. Li *et al.* (2020) [26] built a conversation structure from source tweet and user comments, and used GNN to encode it. Li *et al.* (2020) [27] crawled user-follower information and built a friendly network based on the follow-followers relationship.

We compare our work and the most relevant studies in Table 1. The uniqueness of our work lies in: targeting at source news text, requiring no user response comments, analysing model explainability and Integrating multiple attention mechanisms.

## III. THE BACKGROUND OF THE RELATED DEEP LEARNING TECHNIQUES

Deep learning is a significant branch of machine learning. The mainstream of deep learning is based on neural network methods, and there are also some methods based on tree



**FIGURE 1.** Propagation structure of posts/tweets on Twitter. A tweet can be reposted by multiple people, and a person can also repost multiple tweets.

models [28], [29]. The deep learning in this paper refers specifically to deep neural networks. It has achieved unprecedented success in multiple natural language tasks, such as machine translation, emotion analysis, question answering system, etc. Deep learning is essentially a way of expressing learning that is different from traditional methods, which extracts features based on the manual method. Deep learning models can automatically generate appropriate vectors to represent words, phrases and sentences. This chapter focuses on relevant deep learning models used in this article. In this section, we introduce several deep learning techniques used in our model.

### A. RECURRENT NEURAL NETWORKS

RNN is one of the most commonly used deep learning networks in natural language processing (NLP) tasks of AI. RNN is a type of feed-forward neural network that can be used to model variable-length sequential information such as sentences or time series. Therefore, it has some advantages in learning the nonlinear characteristics of sequences. For each time step, RNN updates its hidden state  $h_t$  by extracting information from its last time step in a hidden state  $h_{t-1}$  and the input in this time step  $x_t$ . This process continues until all time steps have been evaluated. The algorithm iterates over the following equations:

$$a_t = f(h_{t-1}, x_t) \quad (1)$$

$$g(x) = \tanh(x) \quad (2)$$

$$a_t = g(\mathbf{W}_{hh} \cdot h_{t-1} + \mathbf{W}_{xh} \cdot x_t) \quad (3)$$

$$a_t = \tanh(\mathbf{W}_{hh} \cdot h_{t-1} + \mathbf{W}_{xh} \cdot x_t) \quad (4)$$

$$h_t = \mathbf{W}_{hy} \cdot a_t \quad (5)$$

where  $f$  and  $g$  are activation functions,  $\mathbf{W}_{hh}$  and  $\mathbf{W}_{xh}$  are parameter matrices and  $h$  means the hidden state an RNN.

### B. GATED RECURRENT UNIT

In practice, because of the vanishing or exploding gradients, the basic RNN [30] cannot learn long-distance temporal dependencies with gradient-based optimization (Bengio *et al.*, 1994) [31]. One way to deal with this is to make an extension that includes “memory” units to store information over long periods, commonly known as LSTM unit (Hochreiter and Schmidhuber, 1997) [32], (Graves, 2013) [33] and gated recurrent unit (GRU). Here, we briefly introduce the GRU. Unlike an RNN unit, GRU has gating units that modulate the flow of the content inside the unit. The following equations are used for a GRU layer:

$$z_t = \sigma(x_t \mathbf{U}_z + h_{t-1} \mathbf{W}_z) \quad (6)$$

$$r_t = \sigma(x_t \mathbf{U}_r + h_{t-1} \mathbf{W}_r) \quad (7)$$

$$\tilde{h}_t = \tanh(x_t \mathbf{U}_h + (h_{t-1} \cdot r_t) \mathbf{W}_h) \quad (8)$$

$$h_t = (1 - z_t) \cdot h_{t-1} + z_t \cdot \tilde{h}_t \quad (9)$$

where  $\mathbf{W}_z$ ,  $\mathbf{W}_r$ ,  $\mathbf{W}_h$ ,  $\mathbf{U}_z$ ,  $\mathbf{U}_r$ ,  $\mathbf{U}_h$  are parameters.  $\sigma$  is a logistic sigmoid function.  $z_t$  and  $r_t$  are update gate and reset gate of GRU, respectively.  $\tilde{h}_t$  denotes the candidate activation of the hidden state  $h_t$ .

### C. WORD2VEC

Natural language is a special composition of characters produced by human beings for communication. For a computer to understand the natural language, words in the natural language need to be encoded. In the early days of NLP, words were often converted into discrete individual symbols according to the order in which the word appeared

in a corpus. This encoding method was called one-hot-encoding. However, such an encoding method does not reflect the relationship between words. To overcome this problem, Bengio *et al.* (2003) [34] proposed the concept of word embedding for the first time in 2003. They assumed that each word in the glossary corresponds to a continuous eigenvector. This idea has since been widely applied to various NLP models, including word2vec. In word2vec, the two most important models are the CBOW model (continuous bag-words-model) and skip-gram model. The basic idea of CBOW and skip-gram is to make the vector represent the information contained in the word as fully as possible, while keeping the dimensions of the vector within a manageable range (between 25 and 1,000 dimensions, if appropriate). The CBOW model learns the expression of word vectors from the prediction of context to the target word. Mathematically, the CBOW model is equivalent to the embedding matrix of a word bag model multiplied by a successive embedding matrix. Conversely, the skip-gram model learns word vectors from the target word's prediction of context.

#### D. ATTENTION MECHANISM

The attention mechanism, as the name suggests, is a technique that enables models to focus on important information. It is not a complete model but a technique that can be used in deep learning models. The mechanism was first proposed in the field of visual images. Mnih *et al.* (2014) [35] used the attention mechanism on the RNN model to classify images. Then, Bahdanau *et al.* (2014) [36] applied an attention mechanism to NLP that combined translation and alignment in machine translation tasks. In NLP, different words in a sentence have different importance, such as I hate this movie. In an emotional analysis, it is obvious that the word “hate” plays a more important role than the other words, which means that the model should concentrate on that word. Further, the attention mechanism is widely used in various NLP tasks based on neural network models, such as RNN/CNN. In 2017, the Google machine translation team made extensive use of the self-attention mechanism to learn textual representation [37]. The mechanism of self-attention has also become a recent research hotspot and has been explored in various NLP tasks.

#### E. GRAPH ATTENTION NETWORKS

Graph attention network (GAT) [39] is a kind of GNN. GNNs are especially deep learning-based methods that operate on non-Euclidean structures data domain. Due to their convincing performance and high interpretability, GNN is the widely applied graph analysis method recently [38].

There are several variants of GNNs, of which GAT is the most commonly used. GATs are proposed by Velickovic *et al.* (2018) [39]. This kind of neural network incorporates the attention mechanism into the graph propagation step. It computes the hidden states of each node by attending over its neighbours, following a self-attention strategy. Velickovic *et al.* (2018) [39] proposed a single graph attentional

layer and constructed arbitrary graph attention networks by stacking this layer. The layer computes the coefficients in the attention mechanism of the node pair  $(i, j)$  by:

$$a_{ij} = \frac{\exp(\text{LeakyReLU}(a^T [\mathbf{W}\vec{h}_i || \mathbf{W}\vec{h}_j]))}{\sum_{k \in N_i} \exp(\text{LeakyReLU}(a^T [\mathbf{W}\vec{h}_i || \mathbf{W}\vec{h}_k]))} \quad (10)$$

where  $a_{ij}$  is the attention coefficient of node  $j$  to  $i$ ,  $N_i$  represents the neighbourhoods of the  $i$ th node in the graph.  $\mathbf{W}$  is the weight matrix of a shared linear transformation, which is applied to every node.  $a$  is the weight vector of a single-layer feedforward neural network.

#### IV. PROBLEM STATEMENT

Let  $\Phi = \{s_1, s_2, \dots, s_{|\Phi|}\}$  be the set of source tweets (short-text) and  $\Psi = \{g_1, g_2, \dots, g_{|\Psi|}\}$  be the set of propagation structure graph. Each source tweet  $s_i \in \Phi$  corresponds to a propagation structure graph  $g_i \in \Psi$ . When a Twitter post  $s_i$  is published, other users will retweet it. The propagation structure graph  $g_i \in \Psi$  of each source tweet is composed of its retweet users  $u_j$ .  $g_i = \{u_1, u_2, \dots, u_j\}$  and we denote  $F = \{f_1, f_2, \dots, f_{|F|}\}$  as the set of user features. Given a source tweet  $s_i$ , along with the corresponding propagation  $g_i$  containing users  $u_j$  who retweet  $s_i$ , as well as their feature vectors  $F_j$ , the goal of our model is to classify  $s_i$  as ‘true’ or ‘fake’. The classifier performs learning through labelled information, i.e.,  $C_i : s_i \rightarrow y_i$ . In addition, we require our model to highlight some users  $u_j \in u_i$  who retweet  $s_i$  and some words  $q_i \in s_i$  that can interpret why  $s_i$  is predicted as a true or fake one.

#### V. THE PROPOSED MVAN MODEL

We propose a novel neural network model, multi-view attention networks (MVAN), to detect fake news based on the source tweet and its propagation structure. As can be seen from Fig. 2, the proposed model MVAN consists of three components. The first is text semantic attention networks. Its role is to obtain the semantic representation of the source tweet text information. The second is propagation structure attention networks. It captures the hidden information in the propagation structure of a tweet. The last is the prediction module. It generates the final detection result by concatenating text semantic representation and propagation structure representation.

##### A. TEXT SEMANTIC ATTENTION NETWORKS

In this work, to correctly represent the information contained in the source tweet text and capture the key clue words in the source tweet text, we propose text semantic attention networks to process the source tweet text. We define  $x_i \in \mathbb{R}^d$  as the  $d$ -dimensional word embedding corresponding to the  $i$ -th word in the source tweet. Because the length of each source tweet is different, we perform zero-padding here by setting a maximum length  $L$ . Let  $E = [e_1, e_2, \dots, e_l] \in \mathbb{R}^d$  be the input vector of the source tweet, in which  $e_l$  is the embedding vector of the word, when the  $i$ th word is the ⟨pad⟩ token, its embedding vector  $e_l$  is a vector filled with 0. We use

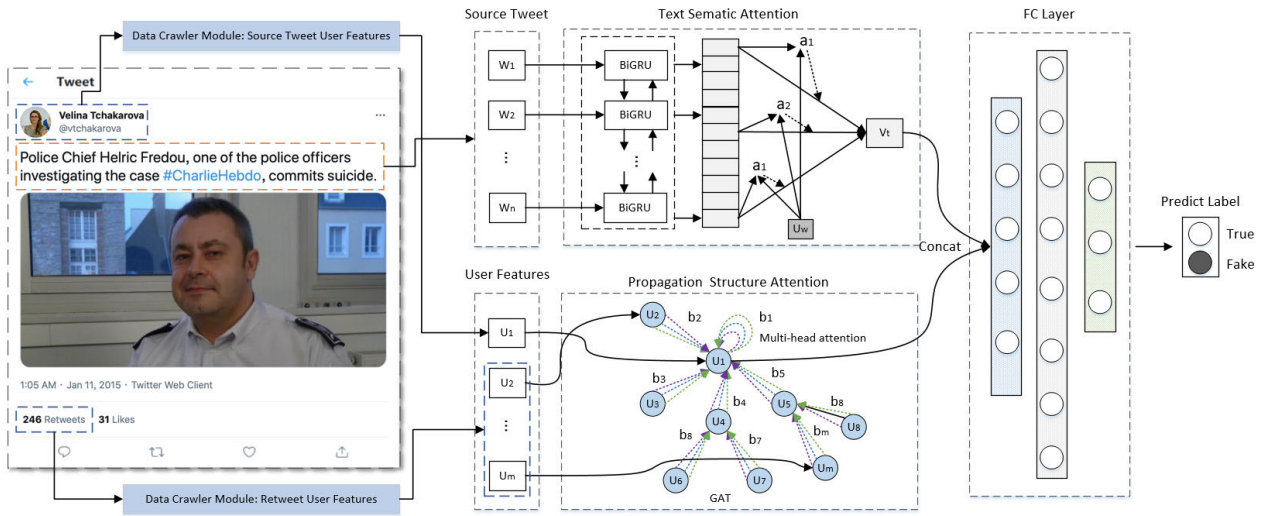


FIGURE 2. Architecture of the MVAN model. The two attention mechanisms represent two different kinds of information.

word2vec to encode the words of the source tweet. Moreover, a deep bi-directional gated recurrent unit (BiGRU) is used to capture the relationships among words and generate the source tweet representation:

$$h_t = \text{BiGRU}(e_t, h_{t-1}) \quad (11)$$

where  $h_t$  is the final hidden state of BiGRU,  $h_t \in \mathbb{R}^{d_t}$ .  $e_t$  is the word embedding vector of the source tweet.

Since each word has a different role in detecting fake news, we believe that the model could focus on the keywords and reduce the role of other irrelevant words. We use a layer of the fully connected network to map the output vector  $h_t$  of BiGRU to a matrix-vector  $u_t$ .

$$u_t = \tanh(\mathbf{W}_w h_t + b_w) \quad (12)$$

where  $\mathbf{W}_w \in \mathbb{R}^{n \times l}$  and  $b_w$  are the weight and bias of attention and  $\tanh$  is the activation function mapping the value between  $[-1, 1]$ .  $u_t \in \mathbb{R}^m$ , and  $n$  is the number of neural units in the fully connected layer. Then we calculate the attention coefficient of each word, which is the final weight of each word.

$$a_t = \text{softmax}(u_t u_w) = \frac{\exp(u_t^T u_w)}{\sum_i \exp(u_i^T u_w)} \quad (13)$$

where  $u_w$  is the weight matrix,  $\cdot^T$  represents transposition and  $a_t \in \mathbb{R}^l$ .

Finally, each word vector  $h_t$  and attention coefficient  $a_t$  are weighted and summed to obtain the representation of the source tweet:

$$S = \sum_t a_t h_t, \quad S \in \mathbb{R}^{o \times l} \quad (14)$$

where  $o$  is dimensional of output layer. Through text semantic attention networks, we get a representation vector containing text semantics and the attention weight of each word.

## B. PROPAGATION STRUCTURE ATTENTION NETWORKS

Research shows that fake news and real news have different propagation structures [8]. Therefore, we propose propagation structure attention networks to capture the clues implicit in the propagation structure of news. In this part, we discuss how to encode the propagation structure into the node representation for fake news detection. Inspired by GAT [39], we apply the attention mechanism to learn a distributed representation of each user node (retweet user) in the propagation structure graph by attending over its neighbours (set of user who retweeted that user's post).

The input to the propagation structure attention layer is a set of user features,  $u = \{\vec{u}_1, \vec{u}_2, \dots, \vec{u}_N\}$ ,  $\vec{u}_i \in \mathbb{R}^F$ , where  $N$  is the number of user nodes and  $F$  is the number of features in each user node. The output of the propagation structure attention layer is a new set of user node features,  $p = \{\vec{p}_1, \vec{p}_2, \dots, \vec{p}_N\}$ ,  $\vec{p}_i \in \mathbb{R}^{F'}$ . Note that  $F'$  may not be equal to  $F$ .

To obtain enough presentation power to transform the original user features into higher-level features, at least one learnable linear transformation is required. For this purpose, as an initial step, a shared linear transformation, parametrized by a weight matrix,  $\mathbf{W} \in \mathbb{R}^{F' \times F}$ , is applied to every user node. Then self-attention mechanism  $\text{att}$  is used on each user node:  $\mathbb{R}^{F'} \times \mathbb{R}^{F'} \rightarrow \mathbb{R}$  computes attention coefficients that indicate the importance of user node  $j$ 's features to user node  $i$ .

$$c_{ij} = \text{att}(\mathbf{W}\vec{u}_i, \mathbf{W}\vec{u}_j) \quad (15)$$

We inject the propagation structure into the mechanism by performing masked attention, we only compute  $c_{ij}$  for nodes  $j \in U_i$ , where  $U_i$  is some neighbourhood of user node  $i$  in the propagation structure graph. In our experiment, each user node calculates the attention coefficients of all its first-order neighbours. To prevent the information of user node  $i$  from being forgotten, we regard user node  $i$  as its first-order

neighbour. To make the attention coefficients easy to compare between different nodes, we use the softmax function to normalize them among all the choices of  $j$ :

$$a_{ij} = \text{softmax}(c_{ij}) = \frac{\exp(c_{ij})}{\sum_{k \in U_i} \exp(c_{ik})} \quad (16)$$

In this paper, the propagation structure attention mechanism is a fully connected layer, parametrized by a weight vector  $\vec{a} \in \mathbb{R}^{2F'}$ , using the LeakyReLU function nonlinearity. Fully expanded out, the coefficient calculated by the attention mechanism can be expressed as:

$$a_{ij} = \frac{\exp(\text{LeakyReLU}(\vec{a}^T [\mathbf{W}\vec{u}_i \parallel \mathbf{W}\vec{u}_j]))}{\sum_{k \in U_i} \exp(\text{LeakyReLU}(\vec{a}^T [\mathbf{W}\vec{u}_i \parallel \mathbf{W}\vec{u}_k]))} \quad (17)$$

where means transposition and  $\parallel$  is the concatenation operation.

$$\vec{u}'_i = \parallel_{h=1}^H \text{ELU}(\sum_{j \in U_i} a_{ij}^h \mathbf{W}^h \vec{u}_j) \quad (18)$$

where  $\parallel$  represents concatenation,  $a_{ij}^h$  are normalized attention coefficients calculated by the  $h$ -th attention mechanism and  $\mathbf{W}^h$  is the corresponding input liner transformation's weight matrix. ELU is the activation function.

Finally, in the output layer of propagation structure attention networks, we replace the previous *concatenation* with the average and use ReLU instead of ELU as the activation function. The process can be expressed as:

$$\vec{p}_i = \text{ReLU}(\frac{1}{H} \sum_{h=1}^H \sum_{j \in U_i} a_{ij}^h \mathbf{W}^h \vec{u}'_j) \quad (19)$$

Through propagation structure attention networks, we get a representation vector of the propagation structure of news and the attention weight of each user node and its neighbour nodes.

### C. PREDICTION MODULE

The prediction module is a multi-layer feedforward neural network module. Based on the output of the text semantic attention networks and propagation structure attention networks, we use a softmax function in the output layer to predict the label of the Twitter news:

$$\hat{y} = \text{softmax}(\text{ReLU}([V_t \parallel V_p] \mathbf{W}_{tp} + b_{tp})) \quad (20)$$

where  $\parallel$  represents concatenation,  $\mathbf{W}_{tp}$  and  $b_{tp}$  are parameters in the output layer and  $V_t$  and  $V_p$  are the output vectors of text semantic attention networks and propagation structure attention networks, respectively.

For each training process, we use the cross-entropy loss function to minimize the deviation between the predicted label and the real label:

$$\mathfrak{L}(\Theta) = -y \log(\hat{y}_t) - (1 - y) \log(1 - \hat{y}_f) \quad (21)$$

where  $\Theta$  is the model parameter to be estimated,  $y$  is the real label and  $\hat{y}_t$  and  $\hat{y}_f$  are the two labels predicted by the model: true and fake.

TABLE 2. Statistics of the datasets.

Statistic	Twitter 15	Twitter 16
# of users	190,868	115,036
# of source tweets	742	410
# of true	372	205
# of fake	370	205
Avg. retweets per story	292.2	308.7
Avg. words per source tweet	13.25	12.81

TABLE 3. A summary of user features in the two datasets.

No.	Features	Type
1	user_contributors_enabled	Binary
2	user_default_profile	Binary
3	user_default_profile_image	Binary
4	user_favourites_count	Integer
5	user_follow_request_sent	Binary
6	user_followers_count	Integer
7	user_following	Binary
8	user_friends_count	Integer
9	user_geo_enabled	Binary
10	user_has_extended_profile	Binary
11	user_listed_count	Integer
12	user_profile_use_background_image	Binary
13	user_protected	Binary
14	user_statuses_count	Integer
15	user_verified	Binary

## VI. EXPERIMENT AND RESULTS

### A. DATASETS

Two well-known fake news datasets, Twitter15 and Twitter16,<sup>1</sup> were used to evaluate our MVAN model. The statistics of the datasets are shown in Table 2. Each dataset contains a collection of source tweets,<sup>2</sup> along with their corresponding sequences of retweet user IDs. We chose only “true” and “fake” labels as the ground truth. And we balance the three words “obama”, “Paul”, and “Sydney” that were strongly unbalanced in the datasets. Because the original dataset did not include user profiles, we used user IDs to crawl user feature information using Twitter API.<sup>3</sup> Some users have been deleted or abolished during the crawling process. In the Twitter15 and Twitter16 datasets, the total number of retweets is 190,868 and 115,036 respectively. The number of user information we collected through API is 177,049 and 108,801 respectively, and the corresponding percentages are 92.76% and 94.58%. It can be found that the missing users account for only a small part of the total. For missing user information, we use the mean value of other user features in the same propagation structure to fill it. Through the API, we crawled a total of 38 user features. Based on these

<sup>1</sup> <https://www.dropbox.com/s/7ewzdrbelpmrnxu/rumdetect2017.zip?dl=0>

<sup>2</sup> Hashtags is a part of the textual information in the source tweet.

<sup>3</sup> <https://dev.twitter.com/rest/public>

previous works [2], [9], [44], 15 common user features that are available on Twitter, which is summarized in Table 3.

## B. EXPERIMENTAL SETUP

### 1) MODEL COMPARISON

We compare our proposed models with other models, including some of the current state-of-the-art models.

- **SVM-BOW** [40]: an SVM classifier using bag-of-words and N-gram (e.g., 1-gram, bigram and trigram) features [22].
- **BiLSTM** [41]: a bidirectional RNN-based tweet model that considers the bidirectional contexts between targets and tweets [42].
- **TextCNN** [42]: a convolutional neural network model for obtaining the representation of each tweet and their classifications using a softmax layer [43].
- **CSI** [43]: a state-of-the-art fake news detection model that captures the temporal pattern of user activities and scores users based on their behaviour [44].
- **CRNN** [9]: A Hybrid Deep Model for Fake News Detection: A model combining RNN and CNN, which learns the text representations through the characteristics of users in the Twitter propagation path [17].
- **dEFEND** [19]: a state-of-the-art co-attention-based fake news detection model that learns the correlation between the source article's sentences and user profiles [20].
- **GCAN** [25]: a state-of-the-art model that uses graph-aware co-attention networks (GCAN) to detect fake news [26].
- **G-SAGE** [26]: a state-of-the-art detecting fake news by modeling conversation structure as a graph using GraphSAGE and BiLSTM [27].
- **MVAN**: Our new deep neural network model, which uses both text semantic attention and propagation structure graph attention to detect fake news.

### 2) PARAMETER SETTING

In the text processing stage, we first cleaned the text information by removing useless expressions and symbols, uniform case, etc. We used GoogleNews pre-trained word2vec data with 300 dimensions for word embedding and set the maximum vocabulary to 250,000. The hidden size of BiGRU is 300, and the number of layers is 2. The batch size of the model was 64. The 15 user features shown in Table 3 were used as input data in the training process of the propagation structure attention network. We used 5-head attention in the graph attention network, and the number of graph attention network layers is 2. Moreover, we used the LeakyReLU nonlinearity with a negative input slope  $\alpha = 0.3$ . In the training phase, we used Adam with a 0.001 learning rate to optimize the model, with the dropout rate set to 0.5.

### 3) EVALUATION METRICS

For a fair comparison, we adopted the same evaluation metrics used in previous work. Therefore, the Accuracy,

Precision Recall, and F1-measure (F1) were adopted for evaluation, which as described in the following equations:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (22)$$

$$Precision = \frac{TP}{TP + FP} \quad (23)$$

$$Recall = \frac{TP}{TP + FN} \quad (24)$$

$$F1\_Measure = \frac{2 * Recall * Precision}{Recall + Precision} \quad (25)$$

where  $TP$  are the true positive,  $TN$  are the true negative,  $FP$  the false positive and  $FN$  the false negative predictions.

We followed GCAN [25] to split the datasets. In this paper, 70% of the data were randomly selected for training and the remaining 30% is used for testing. The results of the experiment are an average of ten times.

## C. EXPERIMENTAL RESULTS

The main experimental results are shown in Table 4. The proposed MAVN model is significantly better than the state-of-the-art model in all the evaluation criteria in the two public datasets. Compared with the state-of-the-art model G-SEGA, the accuracy of Twitter15 and Twitter16 is improved by about 3.06% and 2.03%, respectively. Compared with the machine learning model SVM-BOW, the accuracy of our proposed model on both Twitter15 and Twitter16 datasets is improved by about 25%. As shown in Table 5, we performed statistical tests on all models, and reported the average accuracy  $\pm$  standard deviation with confidence levels of 0.90, 0.95 and 0.98 respectively. The results in Table 5 show that our model has a significant performance improvement on the two public datasets. MVAN can better represent text and propagation structural information, thereby improving the accuracy of fake news detection.

## D. ABLATION STUDY

To determine the relative importance of each module of the MVAN, we conducted a series of ablation studies on key parts of the model. The brief introduction of each model used for ablation research is as follows:

- **MVAN-TSA**: Both text and propagation structure information are used, but the text semantic attention mechanism is removed from the original MVAN model, and only BiGRU is used to encode the text.
- **MVAN-PSA**: Both text and propagation structure information are used but the MVAN model removes propagation structure attention mechanism and Node2vec is used to encode the propagation graph structure directly.
- **TSAN**: Text semantic attention network only uses text data to classify news.
- **PSAN**: The propagation structure attention network only uses propagation structure data to classify news.

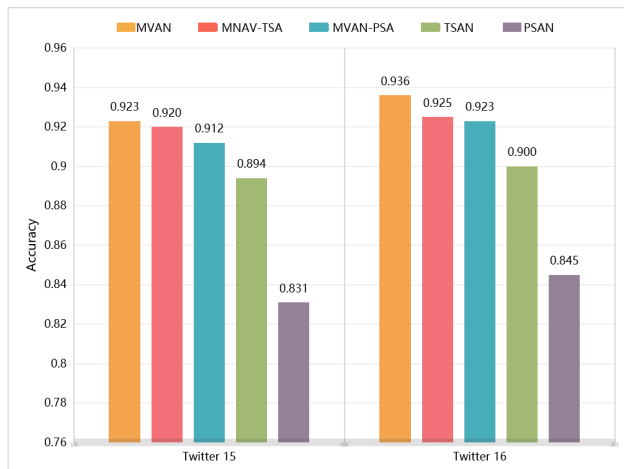
The experimental comparison results are shown in Fig. 3. We found that when the MVAN model removed the text

**TABLE 4.** Comparison of multi-view attention networks with other models. The best model and the best competitor are highlighted in bold and underline. The results reported are the average of 10 runs. Note: The results of the model GCAN in the table are directly taken from the results shown in the original paper.

Method	Twitter 15				Twitter 16			
	Acc	Pre	Rec	F1	Acc	Pre	Rec	F1
SVM-BOW	0.6694	0.6357	0.6243	0.6292	0.6786	0.6783	0.6725	0.6747
BiLSTM	0.7870	0.7370	0.780	0.7650	0.7850	0.7510	0.7730	0.7620
TextCNN	0.7950	0.7630	0.7910	0.7770	0.7940	0.7370	0.7770	0.7590
CSI	0.8573	0.8435	0.8596	0.8517	0.8465	0.8360	0.8557	0.8458
CRNN	0.8626	0.8564	0.8644	0.8604	0.8708	0.8674	0.8669	0.8672
dEFEND	0.8721	0.8669	0.8745	0.8707	0.8802	0.8785	0.8789	0.8787
GCAN	0.8767	0.8257	0.8295	0.8250	0.9084	0.7594	0.7632	0.7593
G-SEGA	0.8928	0.8914	0.8990	0.8952	0.9162	0.9034	0.9167	0.9093
MVAN	<b>0.9234</b>	<b>0.9264</b>	<b>0.9159</b>	<b>0.9224</b>	<b>0.9365</b>	<b>0.9324</b>	<b>0.9348</b>	<b>0.9336</b>

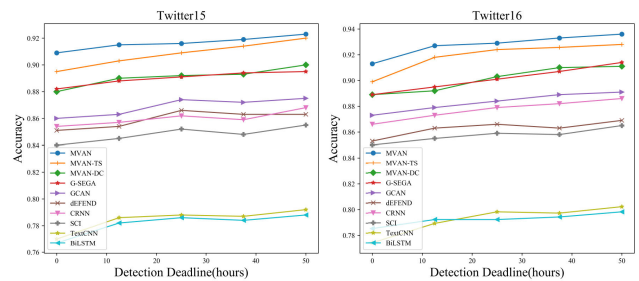
**TABLE 5.** Test accuracy on rumor detection with different models. Each model we ran 10 times and report the mean  $\pm$  standard deviation. Our model significantly outperforms all the baseline based on t-tests(Confidence Level: 0.90, 0.95, 0.98).

Method	Twitter 15			Twitter 16		
	90%	95%	98%	90%	95%	98%
SVM-BOW	0.6694 $\pm$ 0.052	0.6694 $\pm$ 0.062	0.6694 $\pm$ 0.073	0.6786 $\pm$ 0.069	0.6786 $\pm$ 0.083	0.6786 $\pm$ 0.098
BiLSTM	0.7870 $\pm$ 0.045	0.7870 $\pm$ 0.054	0.7870 $\pm$ 0.064	0.7850 $\pm$ 0.061	0.7850 $\pm$ 0.073	0.7850 $\pm$ 0.086
TextCNN	0.7950 $\pm$ 0.044	0.7950 $\pm$ 0.053	0.7950 $\pm$ 0.063	0.7940 $\pm$ 0.060	0.7940 $\pm$ 0.071	0.7940 $\pm$ 0.085
CSI	0.8573 $\pm$ 0.038	0.8573 $\pm$ 0.046	0.8573 $\pm$ 0.055	0.8465 $\pm$ 0.053	0.8465 $\pm$ 0.064	0.8465 $\pm$ 0.076
CRNN	0.8626 $\pm$ 0.038	0.8626 $\pm$ 0.045	0.8626 $\pm$ 0.054	0.8708 $\pm$ 0.050	0.8708 $\pm$ 0.059	0.8708 $\pm$ 0.070
dEFEND	0.8721 $\pm$ 0.037	0.8721 $\pm$ 0.044	0.8721 $\pm$ 0.052	0.8802 $\pm$ 0.048	0.8802 $\pm$ 0.057	0.8802 $\pm$ 0.068
GCAN	0.8767 $\pm$ 0.036	0.8767 $\pm$ 0.043	0.8767 $\pm$ 0.051	0.9084 $\pm$ 0.043	0.9084 $\pm$ 0.051	0.9084 $\pm$ 0.061
G-SEGA	0.8928 $\pm$ 0.034	0.8928 $\pm$ 0.041	0.8928 $\pm$ 0.048	0.9162 $\pm$ 0.041	0.9162 $\pm$ 0.049	0.9162 $\pm$ 0.058
MVAN	0.9234 $\pm$ 0.029	0.9234 $\pm$ 0.035	0.9234 $\pm$ 0.041	0.9365 $\pm$ 0.036	0.9365 $\pm$ 0.043	0.9365 $\pm$ 0.051



**FIGURE 3.** MVAN ablation analysis in accuracy. The results reported are the average of 10 runs.

semantic attention mechanism or propagation structure attention mechanism, the performance dropped by about 1%. This shows that these two attention mechanisms have a certain effect on our model performance. When the model used TSAN only, the performance of the model dropped by 2.9% to 3.6%, because the model loses very important propagation



**FIGURE 4.** Early fake news detection results.

structure information. In addition, if only PSAN was used, the performance of the model dropped by about 9% on both data sets, because the model does not even read the text content of the news itself. However, the performance of PSAN on Twitter15 and Twitter16 data sets reached 8.31% and 8.45%, respectively, which proves that numerous clues can be used to detect fake news in the propagation structure.

### E. EARLY DETECTION PERFORMANCE

It is very important to detect fake news in the early stages of propagation so that preventive measures can be taken as quickly as possible. In the early detection tasks, all user information after the detection deadline is invisible during the



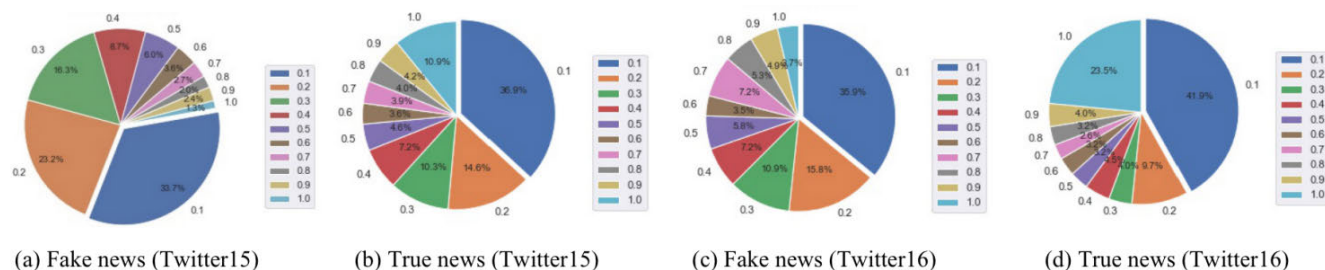


FIGURE 5. Distribution of statistics of word attention weight value.

test. The earlier the deadline, the less propagation information is available.

While comparing the previous comparison models, we add ST-GCN [45] and DCRNN [46] models for comparison. These two models are neural network models for processing temporal sequence graphs. For the convenience of comparison, we replaced the propagation structure attention modules in our model with ST-GCN and DCRNN, which we named MVAN + ST and MVAN + DC, respectively. As can be seen in Fig. 4, our model can achieve an accuracy of approximately 91% in the earliest stage. DCRNN is a combination of GCN and RNN. It only supports the timing input of the isomorphic graph model. Therefore, we build a corresponding DCRNN model for each time window for testing, but because each graph corresponds to For each structure, the RNN module in the model training process is equivalent to only one output, so the performance of this model is not fully utilized. ST-GCN can be used to convolve information in both spatial and temporal dimensions at the same time. Therefore, the effect of ST-GCN is quite good, and there is a great possibility of improvement afterwards. Moreover, from the broken line diagram, the curve of MVAN is very stable, indicating that our model has good robustness and high performance in early fake news detection.

F. INTERPRETABILITY ANALYSIS

The text semantic attention assigns relative weight values to the words in the source tweet (the weight value is between 0 and 1). We analyzed the weight distribution of the words in the two datasets Twitter15 and Twitter16. Regardless of whether it is fake news or true news, most words had a small weight of about 0.1 (Fig. 5), implying that most words have little effect in distinguishing the truth of the news. Additionally, we found that the proportion of words with an attention weight equal to 1 in real news was much higher than that in fake news. For example, in Twitter15 and Twitter16, words with an attention weight of 1 accounted for 1.3% and 3.7%, respectively, in fake news and 10.9% and 23.5%, respectively, in true news.

Therefore, according to the word weights given by the text semantic attention mechanism in the model, we visualized high-weight words in the two data sets with word clouds. These words had more weight on real news, “confirmed”,

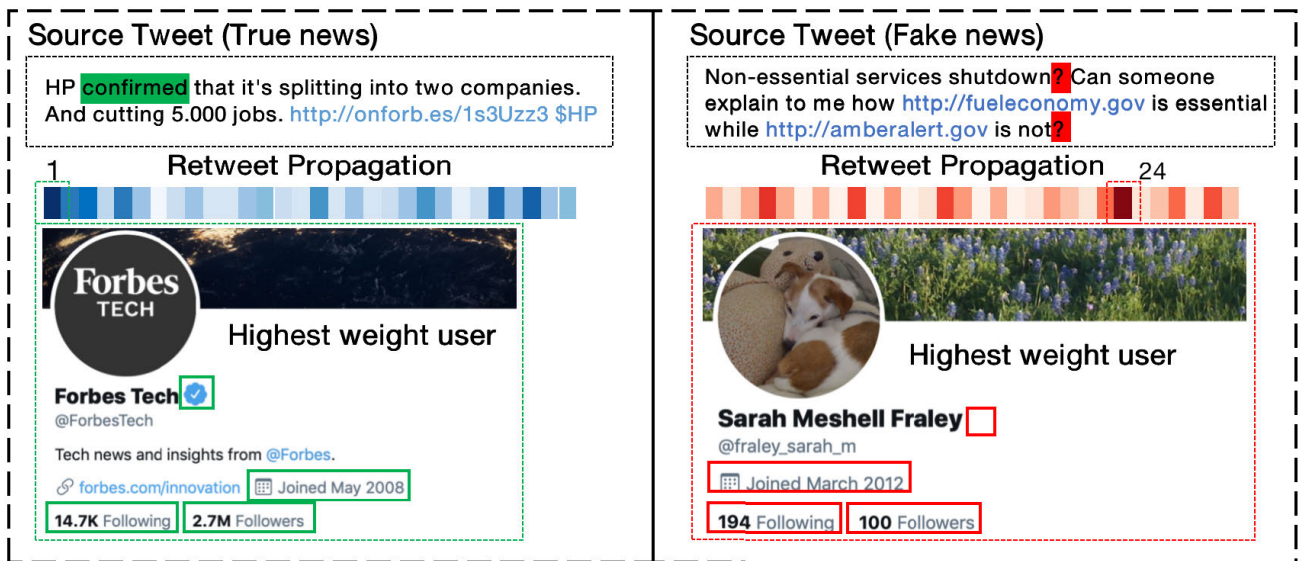


FIGURE 6. Evidence words through word cloud visualization. A larger font indicates a higher text semantic attention weight.

“Sugarhill”, “reportedly”, “hostages”, etc. (Fig. 6). However, in the fake news, “question mark”, “investigating”, “Obama”, “announces”, “future”, etc., these words were more weighted. Furthermore, true news contains more certain and authoritative words, while fake news may be vague and suspicious.

To further analyze the interpretability of the model, we selected two examples from the real data set for analysis. As shown in Fig. 7, the left one is true news and the right one is fake news. We used the text semantic attention mechanism to highlight the evidence words. We thought that if the word “confirmed” was included in the news, then the news is more likely to be true. Contrarily, if there were more “?” in the news, which represents doubtful information, then the news may be fake.

In addition, we used the propagation structure attention mechanism to find the most weighted key clue retweet users in the propagation structure. We randomly select a fake and a real source tweet, and draw their weights according to the propagation structure attention, as shown in Fig. 7, where the



**FIGURE 7.** Real case analysis of true news and fake news. The key clue words in the source tweet are highlighted by text semantic attention weights. Visualization of attention weights for retweet propagations of a fake and a true source tweets. From left to right is retweet order, and dark colors refer to higher attention weights. The user with the highest weight in the retweet propagation is marked, and the main characteristics of the user are displayed.

horizontal direction from left to right represents the order of retweets. As shown in the Fig. 7, in the retweet propagation structure of the real news, the first user has the highest weight, and the 24th user of the fake news has the highest weight. The results show that to determine whether a news is fake, one should first check the features of users who early retweet the source tweet. In the propagation structure of fake news, user weights are more evenly distributed. The features of key Twitter users in the propagation structure of true news and fake news were markedly different (Fig. 7). The user accounts in the true news were created earlier, with authentication icons, profiles and followed by many users and were followed by many users. Such users are generally more authoritative official news accounts. However, the user accounts in fake news are created late, with no certification, no profile and without many followers. General, such a user account perhaps spreads fake news.

## VII. CONCLUSION

We propose a new deep learning model for fake news detection, MVAN, which combines two attention mechanisms, text semantic attention and propagation structure attention, to simultaneously capture the important hidden clues and information in the source tweet text and the propagation structure. The evaluation results using two public data sets show that MVAN has strong performance and reasonable interpretation ability. In addition, MVAN can provide early detection of fake news with satisfactory performance. We believe that MVAN can be used not only for fake news detection but also for other text classification tasks on social media, such as sentiment classification, topic classification, insult detection, etc.

In future work, the users' reply information will be added to further improve the performance of the model. Subsequently, GNN combined with the attention mechanism will be used to capture the key information hidden in the conversation structure graph composed of the source tweet and its replies. We will start from a real-world perspective and detect fake news from more different perspectives.

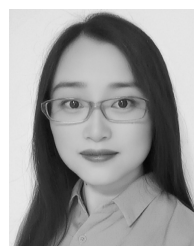
## REFERENCES

- [1] F. A. Ozbay and B. Alatas, "Fake news detection within online social media using supervised artificial intelligence algorithms," *Phys. A, Stat. Mech. Appl.*, vol. 540, Feb. 2020, Art. no. 123174.
- [2] C. Castillo, M. Mendoza, and B. Poblete, "Information credibility on Twitter," in *Proc. 20th Int. Conf. World Wide Web (WWW)*, 2011, pp. 675–684.
- [3] V. Qazvinian, E. Rosengren, D. Radev, and Q. Mei, "Rumor has it: Identifying misinformation in microblogs," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2011, pp. 1589–1599.
- [4] K. Popat, "Assessing the credibility of claims on the web," in *Proc. 26th Int. Conf. World Wide Web Companion (WWW) Companion*, 2017, pp. 735–739.
- [5] F. Yang, Y. Liu, X. Yu, and M. Yang, "Automatic detection of rumor on Sina Weibo," in *Proc. ACM SIGKDD Workshop Mining Data Semantics (MDS)*, 2012, pp. 1–7.
- [6] F. Jin, E. Dougherty, P. Saraf, Y. Cao, and N. Ramakrishnan, "Epidemiological modeling of news and rumors on Twitter," in *Proc. 7th Workshop Social Netw. Mining Anal. (SNAMDD)*, 2013, pp. 1–9.
- [7] J. Sampson, F. Morstatter, L. Wu, and H. Liu, "Leveraging the implicit structure within social media for emergent rumor detection," in *Proc. 25th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2016, pp. 2377–2382.
- [8] J. Ma, W. Gao, and K. F. Wong, "Detect rumors in microblog posts using propagation structure via kernel learning," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, 2017, pp. 708–717.
- [9] Y. Liu and Y.-F. B. Wu, "Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 1–8.
- [10] C. Song, C. Yang, H. Chen, C. Tu, Z. Liu, and M. Sun, "CED: Credible early detection of social media rumors," *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 8, pp. 3035–3047, Aug. 2021.

- [11] J. Ma, W. Gao, P. Mitra, S. Kwon, B. J. Jansen, K. F. Wong, and M. Cha, "Detecting rumors from microblogs with recurrent neural networks," in *Proc. 25th Int. Joint Conf. Artif. Intell.*, 2016, p. 3818.
- [12] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, "A comprehensive survey on graph neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 1, pp. 4–24, Jan. 2021.
- [13] K. Wu, S. Yang, and K. Q. Zhu, "False rumors detection on Sina Weibo by propagation structures," in *Proc. IEEE 31st Int. Conf. Data Eng.*, Apr. 2015, pp. 651–662.
- [14] B. Rath, W. Gao, J. Ma, and J. Srivastava, "From retweet to believability: Utilizing trust to identify rumor spreaders on Twitter," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining*, Jul. 2017, pp. 179–186.
- [15] F. Yu, Q. Liu, S. Wu, L. Wang, and T. Tan, "A convolutional approach for misinformation identification," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Aug. 2017, pp. 3901–3907.
- [16] T. Chen, X. Li, H. Yin, and J. Zhang, "Call attention to rumors: Deep attention-based recurrent neural networks for early rumor detection," in *Proc. Pacific-Asia Conf. Knowl. Discovery Data Mining.*, 2018, pp. 40–52.
- [17] O. Ajao, D. Bhowmik, and S. Zargari, "Fake news identification on Twitter with hybrid CNN and RNN models," in *Proc. 9th Int. Conf. Social Media Soc.*, Jul. 2018, pp. 226–230.
- [18] F. Yu, Q. Liu, S. Wu, L. Wang, and T. Tan, "Attention-based convolutional approach for misinformation identification from massive and noisy microblog posts," *Comput. Secur.*, vol. 83, pp. 106–121, Jun. 2019.
- [19] K. Shu, L. Cui, S. Wang, D. Lee, and H. Liu, "DEFEND: Explainable fake news detection," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2019, pp. 395–405.
- [20] J. Li, S. Ni, and H.-Y. Kao, "Meet the truth: Leverage objective facts and subjective views for interpretable rumor detection," 2021, *arXiv:2107.10747*. [Online]. Available: <http://arxiv.org/abs/2107.10747>
- [21] J. Ma, W. Gao, and K. F. Wong, *Rumor Detection On Twitter With Tree-Structured Recursive Neural Networks*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2018.
- [22] F. Monti, F. Frasca, D. Eynard, D. Mannion, and M. M. Bronstein, "Fake news detection on social media using geometric deep learning," 2019, *arXiv:1902.06673*. [Online]. Available: <http://arxiv.org/abs/1902.06673>
- [23] T. T. Nguyen, "Graph-based rumour detection for social media," *Tech. Rep.*, 2019. [Online]. Available: <http://infoscience.epfl.ch/record/265745>
- [24] T. Bian, X. Xiao, T. Xu, P. Zhao, W. Huang, Y. Rong, and J. Huang, "Rumor detection on social media with bi-directional graph convolutional networks," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 1, 2020, pp. 549–556.
- [25] Y.-J. Lu and C.-T. Li, "GCAN: Graph-aware co-attention networks for explainable fake news detection on social media," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 505–514.
- [26] J. Li, Y. Sujana, and H.-Y. Kao, "Exploiting microblog conversation structures to detect rumors," in *Proc. 28th Int. Conf. Comput. Linguistics*, 2020, pp. 5420–5429.
- [27] J. Li, S. Ni, and H.-Y. Kao, "Birds of a feather rumor together? Exploring homogeneity and conversation structure in social media for rumor detection," *IEEE Access*, vol. 8, pp. 212865–212875, 2020.
- [28] Z.-H. Zhou and J. Feng, "Deep forest," *Nat. Sci. Rev.*, vol. 6, no. 1, pp. 74–86, 2019.
- [29] S. Ni and H. Y. Kao, "PSForest: Improving deep forest via feature pooling and error screening," in *Proc. Asian Conf. Mach. Learn. (PMLR)*, 2020, pp. 769–781.
- [30] L. R. Medsker and L. C. Jain, *Recurrent Neural Networks. Design and Applications*, vol. 5, 2001, pp. 64–67.
- [31] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994.
- [32] S. Hochreiter and J. Schmidhuber, "LSTM can solve hard long time lag problems," in *Proc. Adv. Neural Inf. Process. Syst.*, 1997, pp. 473–479.
- [33] A. Graves, A.-R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2013, pp. 6645–6649.
- [34] Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin, "A neural probabilistic language model," *J. Mach. Learn. Res.*, vol. 3, pp. 1137–1155, Feb. 2003.
- [35] V. Mnih, N. Heess, and A. Graves, "Recurrent models of visual attention," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 2204–2212.
- [36] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2014, *arXiv:1409.0473*. [Online]. Available: <http://arxiv.org/abs/1409.0473>
- [37] A. Vaswani, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.
- [38] J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, "Graph neural networks: A review of methods and applications," 2018, *arXiv:1812.08434*. [Online]. Available: <http://arxiv.org/abs/1812.08434>
- [39] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," in *Proc. ICLR*, 2018, pp. 1–12.
- [40] J. Ma, W. Gao, and K.-F. Wong, *Rumor Detection on Twitter With Tree-Structured Recursive Neural Networks*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2018.
- [41] I. Augenstein, T. Rocktäschel, A. Vlachos, and K. Bontcheva, "Stance detection with bidirectional conditional encoding," 2016, *arXiv:1606.05464*. [Online]. Available: <http://arxiv.org/abs/1606.05464>
- [42] Y.-C. Chen, Z.-Y. Liu, and H.-Y. Kao, "IKM at SemEval-2017 task 8: Convolutional neural networks for stance detection and rumor verification," in *Proc. 11th Int. Workshop Semantic Eval. (SemEval-)*, 2017, pp. 465–469.
- [43] N. Ruchansky, S. Seo, and Y. Liu, "CSI: A hybrid deep model for fake news detection," in *Proc. ACM Conf. Inf. Knowl. Manage.*, Nov. 2017, pp. 797–806.
- [44] H. Guo, J. Cao, Y. Zhang, J. Guo, and J. Li, "Rumor detection with hierarchical social attention network," in *Proc. 27th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2018, pp. 943–951.
- [45] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Proc. AAAI*, Apr. 2018, vol. 32, no. 1, pp. 7444–7452.
- [46] Y. Huang, Y. Weng, S. Yu, and X. Chen, "Diffusion convolutional recurrent neural network with rank influence learning for traffic forecasting," in *Proc. 18th IEEE Int. Conf. Trust, Secur. Privacy Comput. Commun./13th IEEE Int. Conf. Big Data Sci. Eng. (TrustCom/BigDataSE)*, Aug. 2019, pp. 678–685.



**SHIWEN NI** (Student Member, IEEE) was born in Tianmen, Hubei, China, in 1996. He received the B.S. degree in mechanical design manufacturing machine automation from Henan Agricultural University, Zhengzhou, China, in 2017, and the M.S. degree in mechanical engineering from Wuhan University, Wuhan, China, in 2019. He is currently pursuing the Ph.D. degree in computer science and information engineering (CSIE) with the National Cheng Kung University, Tainan, Taiwan. His research interests include machine learning, deep learning, NLP, data mining, and rumor detection.



**JIAWEN LI** received the B.S. and M.S. degrees in management science and engineering from Dalian Maritime University, in 2014 and 2017, respectively. She is currently pursuing the Ph.D. degree in computer science with the Department of Computer Science and Information Engineering (CSIE), National Cheng Kung University. Her research interests include machine learning, deep learning, and social network analysis, mainly working on graph neural networks and NLP.



**HUNG-YU KAO** (Member, IEEE) received the B.S. and M.S. degrees in computer science from the National Tsing Hua University, in 1994 and 1996, respectively, and the Ph.D. degree from the Electrical Engineering Department, National Taiwan University, in July 2003. From 2003 to 2004, he was a Postdoctoral Fellow of the Institute of Information Science (IIS), Academia Sinica. He is currently the Director of the Department of Computer Science and Information Engineering (CSIE), National Cheng Kung University. He has published more than 60 research articles in refereed international journals and conference proceedings. His research interests include web information retrieval/extraction, search engine, knowledge management, data mining, social network analysis, and bioinformatics. He is a member of ACM. He is also the Chair of the IEEE CIS Tainan Chapter.

...