Decision Support Systems 151 (2021) 113633

Contents lists available at ScienceDirect

Decision Support Systems

journal homepage: www.elsevier.com/locate/dss

Improving fake news detection with domain-adversarial and graph-attention neural network

Hua Yuan [a], Jie Zheng [a], Qiongwei Ye [b,*], Yu Qian [a], Yan Zhang [a]

[a] School of Management and Economics, University of Electronic Science and Technology of China, Chengdu 611731, China
[b] School of Business, Yunnan University of Finance and Economics, Kunming 650221, China

ARTICLE INFO

Keywords:
Fake news detection
Feature extraction
Adversarial neural network
Graph-attention network

ABSTRACT

With the widespread use of online social media, we have witnessed that fake news causes enormous distress and inconvenience to people's social life. Although previous studies have proposed rich machine learning methods for identifying fake news in social media, the task of detecting fake news in emerging news events/domains remains a challenging problem due to the wide range of news topics on social media as well as the evolution and variation of fake news contents in the web. In this study, we propose an approach which we term "domain-adversarial and graph-attention neural network" (DAGA-NN) model to address the challenge. Its main advantage is that, in a text environment with multiple events/domains, only partial domain sample data are needed to train the model to achieve accurate cross-domain fake news detection in those domains with few (or even no) samples, which makes up for the limitations of traditional machine learning in fake news detection tasks due to news content evolution or cross-domain identification (where there is no sample data). Extensive experiments were conducted on two multimedia datasets of Twitter and Weibo, and the results showed that the proposed model was very effective in detecting fake news across events/domains.

1. Introduction

Online news on the Internet is an important source of information for people when making decisions [1,2] due to the large amount of data it contains about entities [3] events [4,5] and opinion/sentiment [6] related to business activities. However, people will face significant challenges from fake news when they get information for decision making from online news [7,8]. This is because fake news has the widest impact on public opinions, interests and even decisions by changing the way people connect with real information [9].

As witnessed that online fake news is far-reaching into daily lives [10], how to effectively detect them from various news contents has become an important issue of great interest to both academia and industry [8]. To identify fake news efficiently, an important task in previous machine learning-based methods is to find the typical features of fake news, including text, image, creator behavior, and propagation (network) features, from a given training dataset [8,11]. These methods have achieved some success in various fake news identification tasks. However, regardless of whether these features are filtered by statistical methods, manually annotated by experts, or automatically extracted by deep/machine learning methods [11], their performances are heavily affected by the content of the training dataset.

As a result, traditional models face two challenges in identifying fake news in mass media. First, the diversity of news domains in social media limits the efficiency of the trained models in identifying fake news across domains due to insufficient representativeness and veracity of samples (the samples taken may be incomplete) [12]. For example, if an algorithm is good at detecting fake news in the "politics" domain, it will rely on features extracted from the training set that are more or less "political," such as "elections" and "voting." Thus, the trained fake news detection models are hard to be used in detecting fake news in a different domain, e.g., the fake information in the "sport" news on the same social media. Further, for the same reason, they may also fail to identify the fake news in the emerging subthemes of known domains [13] due to the evolutionary nature of fake news (breaking news or a news story may emerge changing contents). Second, some extracted features may lose descriptiveness as fake news are becoming more similar to proper news in writing style [11]. For example, some online fake news creators usually reuse the existing news content [8], which suggests that the structural relationships of news contents may simultaneously facilitate and confound the tasks of detecting fake news. An interesting research question arises as to how to make models trained using partial domain

* Corresponding author.
E-mail addresses: yuanhua@uestc.edu.cn (H. Yuan), zhengjie@std.uestc.edu.cn (J. Zheng), yqw@ynufe.edu.cn (Q. Ye), qiany@uestc.edu.cn (Y. Qian).

https://doi.org/10.1016/j.dss.2021.113633
Received 17 November 2020; Received in revised form 27 May 2021; Accepted 27 June 2021
Available online 6 July 2021

data have the ability to detect fake news across domains and exploit data relationships between true (fake) news within the same domain to achieve high performance in detecting fake news.

Inspired by some recent research about domain adversarial neural network [14,15] and graph attention neural network [16] in the field of natural language processing (NLP), we present a framework, namely the "Domain-Adversarial & Graph-Attention Neural Network" (DAGA-NN) to perform the task of detecting fake news across domains. The contributions of this paper can be summarized as follows:

- The proposed DAGA-NN can learn the domain-invariant features of fake news in various events/domains to achieve the goal of identifying fake news across domains. Synchronously, it retains the relationship between true (fake) samples in the same domain to ensure high performance in the identification process.
- We propose a learning strategy for optimizing graph attention networks, which significantly improve the performance of DAGA-NN in fake news detection by leveraging the positive relationships between two news nodes with the same labels (i.e., effects through true-true or fake-fake links) and blocking the noise relations between nodes (i.e., effects through true-fake links).
- Fake news identification experiments on two real-world social media datasets show that the proposed DAGA-NN can effectively detect fake news across events and domains, and it outperforms state-of-the-art models.

The rest of the paper is organized as follows. Section 2 discusses related work. Section 3 presents our research framework. Section 4 details the proposed method for detecting fake news. Section 5 shows the experimental results using two real datasets. Section 6 discusses the convergence of the model and the role of the components. Finally, Section 7 concludes this study.

2. Related work

2.1. Fake news detection

Previous studies in the field of fake news detection started from exploring the feature of the contents of fake news [11,17]. The features used were mainly derived from the social context and news content [18]. Social context refers to the user engagements of news on social media, which contains news creators, communicators, and propagation paths [8]. Wu et al. [19] proposed a graph-kernel based hybrid support vector machine (SVM) classifier to exploit the propagation structure of news to identify fake news. Ma et al. [20] identified fake news by exploiting the variation of social context features in message propagation over time. Recently, Liu et al. [21] proposed a deep neural network, namely, FNED, which combines multiple deep learning mechanisms and their extensions to solve the fake news early detection problem.

News content refers to the headlines, textual content, images, and videos of news. Textual features have been widely used in fake news research [22–24], and two main methods are knowledge- and style-based analysis. The knowledge-based analysis uses external knowledge to examine true or false news [25]. For example, Wang et al. [26] used external knowledge graphs to provide information to help identify fake news. However, the knowledge-based analysis requires obtaining external knowledge for each news, which increases the complexity of the task of identifying fake news. In addition, it is difficult to obtain relevant knowledge for some unexpected events or new document domains. The style-based analysis uses textual linguistic, semantic features, and writing styles to identify fake news [27,28]. Models in this category range from traditional machine learning methods, such as naive Bayes [29], to more recent deep learning methods, such as RNN [30].

Recently, owing to the development of online multimedia research, several studies have shown that news content derived from more modalities, such as images and videos, help detect fake news [14,26]. For example, Jin et al. [31] found that fake and true/real news have different image distribution patterns. Qi et al. [32] used the RNN method to fuse the visual information of frequency and pixel domains for identifying fake news. Considering the limited feature information available from only one modality of data, many efforts have started fusing information from different modalities before performing fake news identification, which effectively improves the effectiveness of models [14,31,33].

However, the above methods, as they rely heavily on the event and domain features extracted from the training dataset, will be much less effective in identifying fake news once they are applied to a new domain (e.g., a domain lacking training samples). To address this issue, Wang et al. [14] proposed an adversarial learning approach that allows their model to learn event-invariant features. However, their work discarded information about the internal correlation between fake (true) news in a certain domain.

2.2. GNN and text classification

GNNs use deep learning structures to learn graph-structured data [34]. Owing to the excellent performance of GNNs in representing semi-structured and unstructured data, GNNs are also used in text classification tasks. The main contribution of this branch of research is to design a rational graph structure to characterize textual information and then further implement text classification. For example, Huang et al. [35] proposed a new graph-based text classification model, which uses text-level graphs instead of a single graph for an entire corpus. Yao et al. [36] build a single text graph for a corpus based on word co-occurrence and document word relations to learn a text graph convolutional network (TextGCN). TextGCN uses both documents and words as nodes and then uses their relationships to classify the documents. Zhang et al. [37] proposed a graph-based method for inductive text classification; in their work, each text owns its structural graph, and text-level word interactions can be learned. Recently, there have also been some studies that attempted to combine text and related information into graph networks for text classification. For example, Wu et al. [15] proposed an end-to-end, domain-adversarial GNNs for cross-domain text classification. Unlike the usual method, they model the document data directly as graphs and use domain adversarial methods to learn features from these graphs, and this work provides a new idea for the use of graph models in text classification tasks. In [38], Lu et al. proposed VGCN-BERT, which combines the capability of BERT with a vocabulary graph convolutional network. Based on the word co-occurrence, document word relations, and text topic information in a corpus, Ye et al. [39] proposed a short-text graph convolutional network (STGCN) for short text classification.

Recently, the application of GNNs for fake news identification has also attracted the attention of researchers. Lu et al. [40] proposed a graph-aware co-attention networks (GCAN) to predict whether a short-text tweet is fake. Wang et al. [26] used the knowledge graph relationships of entities in text and pictures to construct graphs to identify fake news. Although these studies have made considerable progress in identifying fake news, the focus has been on trying to improve the quality of the features expressed in a graph [41]. We can see that none of these studies are generally designed to address the cross-domain problem of data, or that their data are relatively homogeneous in terms of generation [42]. Clearly, the problem of identifying fake news across domains (identifying fake news with insufficient samples) has not been fully investigated. Therefore, we propose the DAGA-NN model to address this problem in fake news detection.

3. Problem statement and framework overview

In this section, we first formulate the research problem and then present DAGA-NN to address the problem, which is essentially a supervised learning structure.

3.1. Problem statement

Assuming that each news item in our training data has two modal data forms: text (denoted by S) and/or image (denoted by V), then a set of target news whose content needs to be identified as true or fake can be expressed as follows:

$$\mathscr{D} = \{(S, V)\} = \{(S_1, V_1), \ldots, (S_i, V_i), \ldots\}, \quad (1)$$

where S_i is the document for the i-th news consisting of a set of words and V_i is the image for the i-th news. Our problem is to accurately determine whether news i is true or not based on its contents (S_i, V_i). Basically, the core of this study stems from text classification. Unlike the traditional task of data classification, the class label of news (which is used to indicate whether a piece of news is true or fake) is unavailable and it is hard to build a classification model over $\mathscr{D}$ [15].

Consider a training set $\mathscr{D}_M$, where each news in $\mathscr{D}_M$ was added with domain information Y^d (total of K domains) and label information Y^c as follows:

$$\mathscr{D}_M = \{(S, V; Y^d; Y^c)\} = \{(S_1, V_1; Y_1^d; Y_1^c), \ldots, (S_i, V_i; Y_i^d; Y_i^c), \ldots\}, \quad (2)$$

where $Y_i^d \in \mathscr{Y}_d = \{1, \ldots, K\}$ is the domain label and $Y_i^c \in \mathscr{Y}_c = \{0, 1\}$ is the news label ($Y_i^c = 1$ means the news is true, and $Y_i^c = 0$ means the news is fake) of the i-th news, respectively. Since the class labels of news in $\mathscr{D}_M$ are clear and accurate, we can learn a classification model as $F_M : \mathscr{D}_M \rightarrow \mathscr{Y}_c$. Given a dataset $\mathscr{D}_M$ with domain and label information of news, the research problem in this work is how to learn a model F that can be used to identify the labels of news in an unknown dataset $\mathscr{D}$:

$$F : (\mathscr{D} | \mathscr{D}_M) \rightarrow \mathscr{Y}_c. \quad (3)$$

According to relation (3), an intuitive idea is to obtain those "common information" in the dataset $\mathscr{D}$ and $\mathscr{D}_M$ so that the features learned from $\mathscr{D}_M$ can also be applied to the fake news identification task in $\mathscr{D}$. In addition, as we discussed earlier, the structural relationships between the contents of true (fake) news in $\mathscr{D}$ has an impact on the identification of fake news. In the following, we will detail the implementation of the DAGA-NN model on fake news detection tasks.

3.2. Framework overview

Benefiting from the recent development of various advanced machine learning techniques in the field of NLP, DAGA-NN integrates three main components - a multi-modal data *feature extractor*, a *domain discriminator* and a graph-attention-based *fake news classifier*. Fig. 1 describes the workflow of DAGA-NN. The *feature extractor* is used to extract textual feature and image features, respectively, from the text and image data of a piece of news and further fuses these learned feature representations together. Then, the fused features can be used as inputs for the *domain discriminator* and the *fake news classifier*. The *domain discriminator* is used to discriminate the domain information of the input samples. The graph-attention-based *fake news classifier* uses the textual content of news, as well as the features and labels of its neighbors, to determine whether the news is true or fake.

We denote the *feature extractor* as a function of $F_f(\cdot; \boldsymbol{\theta}_f)$ where "·"means the feature representations of the multi-modalities of data D_M and $\boldsymbol{\theta}_f$ means all parameters to be learned in the *feature extractor*. Next, we denote the *domain discriminator* as a function $F_d(F_f; \boldsymbol{\theta}_d)$ where $\boldsymbol{\theta}_d$ means all parameters included, and denote the *fake news classifier* as $F_c(F_f; \boldsymbol{\theta}_c)$ where $\boldsymbol{\theta}_c$ means all parameters included in the classifier.

4. The methodology

In this section, we describe the working principles of each component of DAGA-NN and present the corresponding algorithms.

4.1. Multi-modal data feature extractor

We mainly consider two modalities of text and image of a piece of online news, so the *feature extractor* consists of two parts, i.e., the text and image feature extractors.

As for text feature extracting, the textual content S of news in $\mathscr{D}_M$ is first divided into terms to form a sequence $S = (s_1, \ldots, s_l, \ldots, s_L)$, where s_l stands for the l-th word and L is the length of S. Further, each word $s_l \in S$ is represented as a word vector $\mathbf{s}_l$. Such a vector of $\mathbf{s}_l$ can be obtained by some common methods used in the field of NLP, e.g., the word2vec and BERT [43]. As a result, we can transform the term sequence S into a set of word vectors as $\{\boldsymbol{s}_1, \ldots, \boldsymbol{s}_l, \ldots, \boldsymbol{s}_L\}$.

Further, we input the word vector sequence $\{\boldsymbol{s}_1, \boldsymbol{s}_2, \ldots, \boldsymbol{s}_L\}$ into a Bi-LSTM [44] to extract the textual feature of S. Bi-LSTM is a special kind of LSTM that can process any sequence from the beginning to the end (forward) and the other way from the end to beginning (backward). Let $[\boldsymbol{h}_1, \ldots, \boldsymbol{h}_l, \ldots, \boldsymbol{h}_L]$ represent the states of the LSTM in encoding the text of S, then the output of a Bi-LSTM at each time step is the concatenation

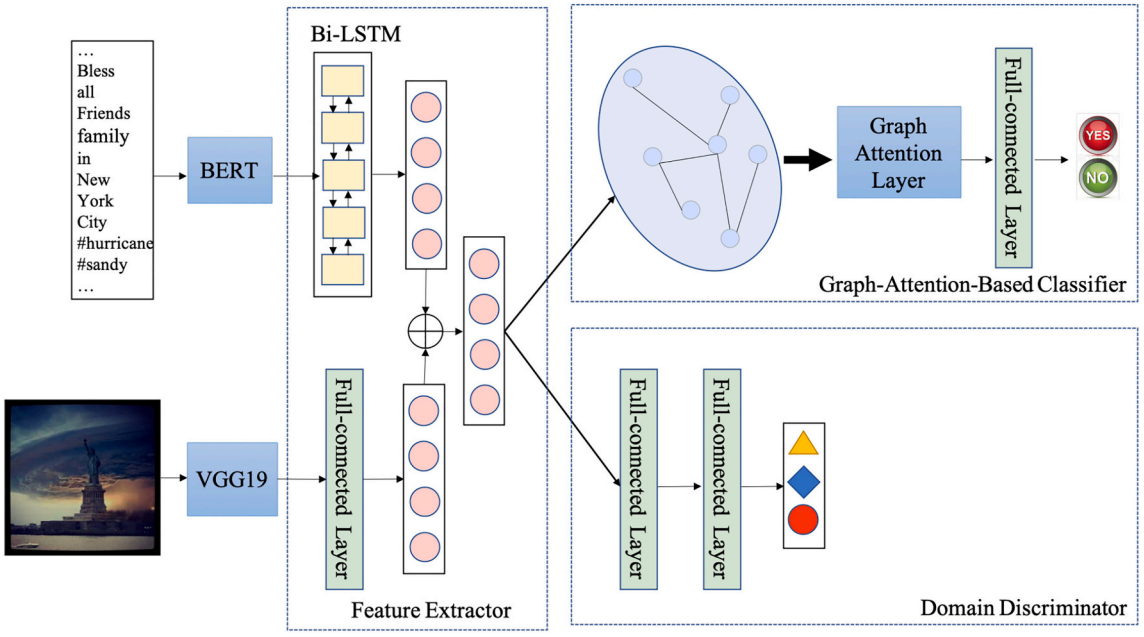


Fig. 1. The framework DAGA-NN.

of both the forward hidden information of $\overleftarrow{\boldsymbol{h}_l} = LSTM\left(\overleftarrow{\boldsymbol{h}_{l-1}}, \boldsymbol{s}_l\right)$ and the backward $\overrightarrow{\boldsymbol{h}_l} = LSTM\left(\overleftarrow{\boldsymbol{h}_{l+1}}, \boldsymbol{s}_l\right)$ as $\left[\overleftarrow{\boldsymbol{h}_l} \parallel \overrightarrow{\boldsymbol{h}_l}\right]$, where $l \in [1, L]$. Accordingly, we can obtain a d_S-dimensional vector representation of S as follows:

$$\boldsymbol{R}_S = Bi - LSTM(\{\boldsymbol{s}_1, \boldsymbol{s}_2, \ldots, \boldsymbol{s}_L\}) \in \mathbb{R}^{d_S}. \tag{4}$$

To extract the image feature associated with news, we employ the pre-trained VGG19 model [45] which is widely used for extracting visual features from massive pictures [46]. To this end, we input picture V of a piece of news into the VGG19 network and obtain its last layer of representation as $\boldsymbol{V}_{vgg}$. By adding a fully connected neural network on top of the VGG19 network [31], we can perform the following calculations to obtain the d_V-dimensional image feature representation of picture V:

$$\boldsymbol{R}_V = \sigma\left(\boldsymbol{W}_{vf} \cdot \boldsymbol{V}_{vgg} + \boldsymbol{b}_{vf}\right) \in \mathbb{R}^{d_V}, \tag{5}$$

where $\boldsymbol{V}_{vgg}$ is the image feature representation obtained from VGG19, $\boldsymbol{W}_{vf}$ is the weight matrix of the fully connected network, $\boldsymbol{b}_{vf}$ is the bias vector and $\sigma(.)$ is the activation function.

Finally, the textual feature representation $\boldsymbol{R}_S$ and the image feature representation $\boldsymbol{R}_V$ will be fused to generate the final data representation:

$$\boldsymbol{R}_{D_M} = (\boldsymbol{R}_S \parallel \boldsymbol{R}_V) \in \mathbb{R}^{d_S + d_V}, \tag{6}$$

where "‖"is the operator for concatenating vectors. Note that $\boldsymbol{R}_{DM}$ will be used as input for the *domain discriminator* and *fake news classifier*. In cases $\boldsymbol{R}_V$ or $\boldsymbol{R}_S$ is missing, we replace it with a $\boldsymbol{0}$ vector.

4.2. Graph-attention-based fake news classifier

It is rational to assume that the multiple modalities of news data in D_M come from various domains, such as music, sports, and politics. We can also assume that the generation and dissemination of fake news generally has its own domain-specific characteristics [42]. For example, fake news in politics has its own domain-specific pattern and this pattern differs from other domains, such as sports. Along this line, in this subsection, we first use the network to construct relationships between news texts in the same domain and then use it to improve the efficiency of the fake news classifier.

4.2.1. Graph structure for news documents

To find out the domain-specific features of fake news, first, our method generates an undirected graph $G^{(k)} = (V, E)$ for domain $k \in [1, K]$ to model the pure semantic relationship between documents within it.

In $G^{(k)}$, variable V represents the vertex set, and we generally regard all news samples in domain k as vertices. Let node $u \in G^{(k)}$ denote news in domain k, we use its corresponding text data S as the main source of features. Therefore, we can obtain a vector representation of node u as follows:

$$\boldsymbol{vec}(u) = \frac{1}{L}\sum_{l=1}^{L}(\boldsymbol{s}_l) \tag{7}$$

Moreover, variable E represents the set of edges between vertices. For any two nodes u and v, the connection between them is constructed based on their textual similarity:

$$e(u, v) = \begin{cases} 1, & \text{if } COS(\boldsymbol{vec}(u), \boldsymbol{vec}(v)) \geq sim_0; \\ 0, & \text{Otherwise.} \end{cases} \tag{8}$$

where $e(u, v)$ denotes the edge between node u and v. $COS(\boldsymbol{vec}(u), \boldsymbol{vec}(v))$ is the cosine similarity between the textual contents of news u and v, and sim_0 is the predefined threshold. Obviously, with the graph structure of $G^{(k)}$, we can capture more information about the interrelationships between the documents in the same domain.

4.2.2. Graph-attention layer

The *graph-attention layer* is operated on graph $G^{(k)}$. It aims to use the semantic relationships between nodes (representing news text) represented on $G^{(k)}$ to help identify whether targeted piece of news is true or fake.

Assuming that there are n_k pieces of news in domain k, the input of the GNN is the feature representations of the n_k news obtained by the *feature extractor* as $\{\mathcal{R}_{S1}^{(k)}, \ldots, \mathcal{R}_{Snk}^{(k)}\}$. Following the idea of graph attention module [16], we first apply a shared linear transformation for every node $u \in G^{(k)}$ as $\boldsymbol{W}\mathcal{R}_u^{(k)}$ where $\boldsymbol{W}$ is a weight matrix. We then compute attention coefficients as follows to show the importance of node v's features to node u:

$$r_{uv} = a\left(\boldsymbol{W}\mathcal{R}_u^{(k)} \parallel \boldsymbol{W}\mathcal{R}_v^{(k)}\right), v \in \mathcal{N}_u, \tag{9}$$

where operator "‖"concatenates the transformed vectors of $\mathcal{R}_u^{(k)}$ and $\mathcal{R}_v^{(k)}$ and inputs them into an unbiased feedforward neural network a; $\mathcal{N}_u$ is some neighborhood of node u in the graph $G^{(k)}$. Further, we normalize r_{uv} using the following function:

$$\alpha_{uv} = \frac{exp(LeakyReLU(r_{uv}))}{\Sigma_{j \in \mathcal{N}_u} exp\left(LeakyReLU\left(r_{uj}\right)\right)}, \tag{10}$$

where *LeakyReLU* is the activation function. Finally, the aggregated features from each neighbor are averaged to obtain the new representation of node u (see Fig. 2):

$$\mathcal{R}_u^{\prime(k)} = \sigma\left(\Sigma_{j \in \mathcal{N}_u} \alpha_{uj} \boldsymbol{W}\mathcal{R}_u^{(k)}\right) \tag{11}$$

4.2.3. Fake news classifier

The determination of whether node u is fake news made by a fully connected network that takes $\mathcal{R}_u^{\prime(k)}$ as the input. The prediction process is as follows:

$$\widehat{\boldsymbol{Y}}_u^c = \sigma\left(\boldsymbol{W}_g \cdot \mathcal{R}_u^{\prime(k)} + \boldsymbol{b}_g\right), \tag{12}$$

where $\boldsymbol{W}_g$ is the weight matrix of the fully connected network, $\boldsymbol{b}_g$ is the bias vector. Accordingly, the expected loss of the *graph attention layer* is as follows:

$$\mathcal{L}_{c_1} = -\mathbb{E}_{(u, Y^c) \sim \left(G^{(k)}, \mathcal{Y}_c^{(k)}\right)}\left[Y_u^c log\left(\widehat{Y}_u^c\right) + \left(1 - Y_u^c\right) log\left(1 - \widehat{Y}_u^c\right)\right], \tag{13}$$

where $G^{(k)}$ represents the relationships between news in domain k, and $\mathcal{Y}_c^{(k)}$ represents the set of labels (ground truth) in domain k.

To obtain a better identification of fake news, the boundaries between the true and fake news in graph $G^{(k)}$ should be as clear as possible. From

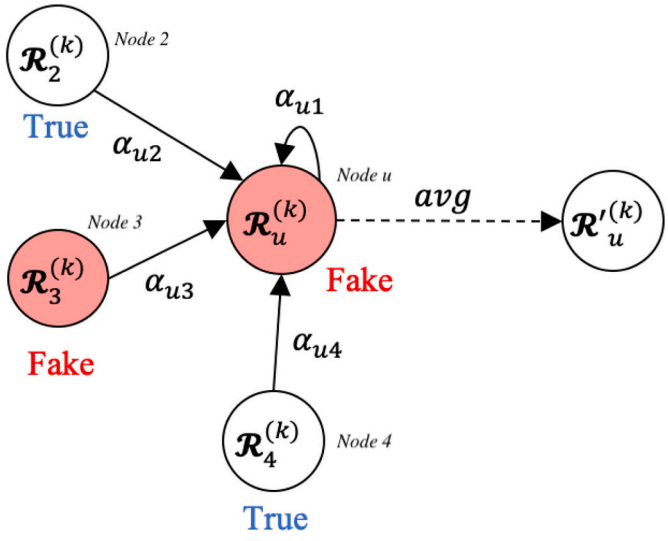


Fig. 2. An illustration of graph attention by node u on its neighborhood.

this perspective, we propose the following strategy to measure the coefficient between node u and its neighbor v: if node v has the same class label with node u (for example, the node 3 in Fig. 2), then we use relation (10) to calculate the weight of the attention coefficient between u and v (i.e., α_{u3}); contrariwise, if v has a different class label from u (for example, the node 2 and 4 in Fig. 2), we should let the weight of α_{uv} (i.e., α_{u2} and α_{u4}) approach 0. To this end, we calculate a "weight loss" as follows:

$$\mathscr{L}_{c_2} = \mathbb{E}_{(u,Y^c)\sim\left(G^{(k)},\mathscr{Y}_c^{(k)}\right)}\left[max\left(0,\left(1-\mathbf{1}_{\left[Y_u^c=Y_v^c\right]}\right)\alpha_{uv}\right)\right], \tag{14}$$

The total loss in predicting fake news can be computed as follows:

$$\mathscr{L}_c\left(\boldsymbol{\theta}_f;\boldsymbol{\theta}_c\right) = \gamma_1\mathscr{L}_{c_1} + \gamma_2\mathscr{L}_{c_2}, \tag{15}$$

where γ_1 and γ_2 are used to balance the individual loss function of $\mathscr{L}_{c_1}$ and $\mathscr{L}_{c_2}$ (In this paper, they are simply set as 1). Accordingly, we can minimize the classification loss by seeking the optimal parameters $\boldsymbol{\theta}_f$ and $\boldsymbol{\theta}_c$, and this can be specified as follows:

$$\left(\widehat{\boldsymbol{\theta}}_f,\widehat{\boldsymbol{\theta}}_c\right) = \underset{\boldsymbol{\theta}_f,\boldsymbol{\theta}_c}{argmin}\ \mathscr{L}_c\left(\boldsymbol{\theta}_f;\boldsymbol{\theta}_c\right) \tag{16}$$

4.3. Domain information discriminator

The *domain discriminator* is used to identify the domain information of a piece of news. To that end, we use a two-layer fully connected neural network to create the *domain discriminator* [14]. The input of the *domain discriminator* is $\boldsymbol{R}_{D_M}$, which is the comprehensive feature representation of the text and picture of news obtained by the *feature extractor*. The output is $\widehat{\boldsymbol{Y}}^d$, which is a K-dimensional vector representing the probability of certain news belonging to each domain (a total of K domains):

$$\widehat{\boldsymbol{Y}}^d = F_d\left(F_f\left(\boldsymbol{R}_{D_M};\boldsymbol{\theta}_f\right);\boldsymbol{\theta}_d\right), \tag{17}$$

where F_d denotes the two-layer fully connected neural network and $\boldsymbol{\theta}_d$ represents its parameters. We use the following cross-entropy to represent the loss function:

$$\mathscr{L}_d\left(\boldsymbol{\theta}_f;\boldsymbol{\theta}_d\right) = -\mathbb{E}_{\left(D_M,Y^d\right)\sim\left(\mathscr{D}_M,\mathscr{Y}_d\right)}\left[\sum_{k=1}^{K}\mathbf{1}_{[k=Y^d]}log\left(F_d\left(F_f\left(\boldsymbol{R}_{D_M};\boldsymbol{\theta}_f\right);\boldsymbol{\theta}_d\right)\right)\right] \tag{18}$$

The parameters of $\boldsymbol{\theta}_d$ in (18) can be estimated by minimizing the function $\mathscr{L}_d$ as $\widehat{\boldsymbol{\theta}}_d = \underset{\boldsymbol{\theta}_d}{argmin}\ \mathscr{L}_d\left(\boldsymbol{\theta}_f;\boldsymbol{\theta}_d\right)$. Further, $\mathscr{L}_d\left(\boldsymbol{\theta}_f;\widehat{\boldsymbol{\theta}}_d\right)$ can be used as the estimation of the dissimilarities of the distribution of different domains. A larger loss implies that the features learned by $F_f(\boldsymbol{R}_{D_M};\boldsymbol{\theta}_f)$ are more domain-invariant [14]. To fully use the known information in dataset D_M in identifying the fake news in D, a feasible method is to obtain the "common information" across all domains; in other words, let the *feature extractor* obtain the domain-invariant features from D_M. In this work, we achieve this by designing the following minmax game [47] between the *feature extractor* and the *domain discriminator*:

$$\left(\widehat{\boldsymbol{\theta}}_f,\widehat{\boldsymbol{\theta}}_d\right) = \underset{\boldsymbol{\theta}_d}{argmin}\ \underset{\boldsymbol{\theta}_f}{max}\ \mathscr{L}_d\left(\boldsymbol{\theta}_f;\boldsymbol{\theta}_d\right) \tag{19}$$

In the above game, the *domain discriminator* strives to discriminate the domain-specific information in the features provided by the feature extractor, while the *feature extractor* strives to maximize the discrimination loss so that the *domain discriminator* is hard to discriminate the domain features of the news.

4.4. Model integration in DAGA-NN

From the structure of DAGA-NN in Fig. 1 and the system's optimization goal relationships (16) and (19), it is known that both the *domain discriminator* and *fake news classifier* may have impacts on the *feature extractor*.

In the training phase of DAGA-NN, the *feature extractor* $F_f(\boldsymbol{R}_{D_M};\boldsymbol{\theta}_f)$ works with the *fake news classifier* $F_c(F_f;\boldsymbol{\theta}_c)$ to improve the classification efficiency of $F_c(.)$. This task is achieved by minimizing the classification loss $\mathscr{L}_c(\boldsymbol{\theta}_f;\boldsymbol{\theta}_c)$. In addition, the *feature extractor* $F_f(\boldsymbol{R}_{D_M};v_f)$ tries to provide domain-invariant representations to the *domain discriminator* by maximizing the loss $\mathscr{L}_d(\boldsymbol{\theta}_f;\boldsymbol{\theta}_d)$. Simultaneously, the *domain discriminator* $F_d(F_f;\boldsymbol{\theta}_d)$ tries to identify the domain information that lies in the feature representations by minimizing the same loss of $\mathscr{L}_d(\boldsymbol{\theta}_f;\boldsymbol{\theta}_d)$. In DAGA-NN, we jointly train the *domain discriminator* and *fake news classifier*. Hence, the total loss can be defined as follows:

$$\mathscr{L}\left(\boldsymbol{\theta}_f;\boldsymbol{\theta}_d;\boldsymbol{\theta}_c\right) = \mathscr{L}_c\left(\boldsymbol{\theta}_f;\boldsymbol{\theta}_c\right) - \lambda\mathscr{L}_d\left(\boldsymbol{\theta}_f;\boldsymbol{\theta}_d\right), \tag{20}$$

where λ is the balance parameter between the objective functions of fake news classification and domain discrimination. The optimal parameters can then be calculated by minimizing the total loss:

$$\begin{cases}\left(\widehat{\boldsymbol{\theta}}_f;\widehat{\boldsymbol{\theta}}_c\right) = \underset{\boldsymbol{\theta}_f,\boldsymbol{\theta}_c}{argmin}\ \mathscr{L}\left(\boldsymbol{\theta}_f;\boldsymbol{\theta}_c;\widehat{\boldsymbol{\theta}}_d\right)\\ \widehat{\boldsymbol{\theta}}_d = \underset{\boldsymbol{\theta}_d}{argmax}\ \mathscr{L}\left(\widehat{\boldsymbol{\theta}}_f;\boldsymbol{\theta}_d\right)\end{cases} \tag{21}$$

The detailed processes are list in Algorithm 4.4 (η denotes the learning rate).

Algorithm 1 DAGA-NN Model Training

1: **Input**: N piece of multi-modal news data $\mathcal{D}_M = \{(S,V;Y^d;Y^c)\}_1^N$;
2: **Output**: Updated parameters $\boldsymbol{\theta}_f, \boldsymbol{\theta}_d, \boldsymbol{\theta}_c$;
3: Initialize weights and biases of DAGA-NN;
4: **for** $\{iteration < max\ epoch\}$ **do**
5: Update the parameters of feature extractor $\boldsymbol{\theta}_f \leftarrow \boldsymbol{\theta}_f - \eta\left(\frac{\partial\mathcal{L}_c}{\partial\boldsymbol{\theta}_f} - \lambda\frac{\partial\mathcal{L}_d}{\partial\boldsymbol{\theta}_f}\right)$;
6: Update the parameters of domain discriminator $\boldsymbol{\theta}_d \leftarrow \boldsymbol{\theta}_d - \eta\frac{\partial\mathcal{L}_d}{\partial\boldsymbol{\theta}_d}$;
7: Update the parameters of fake news classifier $\boldsymbol{\theta}_c \leftarrow \boldsymbol{\theta}_c - \eta\frac{\partial\mathcal{L}_c}{\partial\boldsymbol{\theta}_c}$;
8: **end for**

4.5. Detecting fake news with DAGA-NN

Based on the training results in Algorithm 1, we learn the empirical values of parameters $\boldsymbol{\theta}_f$, $\boldsymbol{\theta}_d$, and $\boldsymbol{\theta}_c$ in DAGA-NN. Next, we can use the well-trained DAGA-NN to identify the authenticity of the news expressed by multi-modal data in the unknown domain $\mathscr{D}$, the process can be expressed in Algorithm 2.

Algorithm 2 DAGA-NN Model Inference

Input: Multi-modal data of a news $\mathcal{D} = \{(S, V)\}$; trained parameters $\boldsymbol{\theta}_f, \boldsymbol{\theta}_d, \boldsymbol{\theta}_c$;

Output: The predicted results $\hat{\boldsymbol{Y}}^c$;

Using trained parameters $\boldsymbol{\theta}_f, \boldsymbol{\theta}_d$, and $\boldsymbol{\theta}_c$ to build the model of DAGA-NN;

Input (S, V) into feature extractor to obtain $\boldsymbol{R}_{D_M} = F_f(\boldsymbol{R}_S, \boldsymbol{R}_V; \boldsymbol{\theta}_f)$;

Input $\boldsymbol{R}_{D_M}$ into fake news classifier to obtain the results of $\hat{\boldsymbol{Y}}^c = F_c(\boldsymbol{R}_{D_M}; \boldsymbol{\theta}_c)$;

return $\hat{\boldsymbol{Y}}^c$.

5. Experimental results

In this section, we first describe the two datasets used in the experiments; they were collected from Weibo (Chinese) and Twitter (English). Then, we show the performance of DAGA-NN compared with some state-of-art approaches for fake news detection.

5.1. Datasets

To evaluate the performance of our method in the classification of fake news, we introduced two datasets consisting of real published news content (with fake news) posted on Twitter and Weibo.

The Twitter dataset used in this work was a part of the 2015 MediaEval Benchmark, which was aimed at detecting manipulation and misuse of Web multimedia content [48]. The dataset has about 15,400 tweets (See Table 1). Each tweet in this dataset contains both image and text content and is labeled as real news or fake news. Note that the dataset has no domain information for the news, but they are collected around 18 specific news events. Therefore, we group the news data that reports on the same "news event" into a "domain" for computation purposes. In the experiments, the dataset is split into two parts - the training set (including 12 news events) and the test set (including 6 news events different from the training set).

The Weibo dataset was collected for the project "Battle from Tech - The Big Data Challenge" to detect fake news on social media during the 2020 COVID-19 pandemic.[1] In the dataset, each news has been verified by Weibo's official rumor debunking system, and about half of the news in the Weibo dataset has both text and image data, whereas the remaining half has only text data. All news in the dataset are categorized into eight different domains (see Table 1). In the experiments, we choose the news from the "entertainment" domain as the test set and news data from the remaining seven domains as the training set.

Table 1
The statistics of the two real-world datasets.

Data	Domain	# fake news	# true news	Total	Average text length	std.
Twitter	Training (12 events)	6742	4921	11,663	12.1	15.5
	Test (6 events)	2546	1209	3755	14.2	8.2
Weibo	Social life	10,203	10,413	20,616	122.9	98.4
	Health care	3321	2999	6320	110.2	62.1
	Science & Tech.	126	155	281	105.0	53.8
	Finance & Business	693	762	1455	123.7	131.5
	Military	151	221	372	109.7	112.1
	Politics	571	670	1241	138.7	183.7
	Education	509	392	901	123.3	45.0
	Test (Entertainment)	1267	1353	2620	119.9	94.0

The detailed statistical information about these two datasets is listed in Table 1. It is worth mentioning that our method is located after the initial vectorization of the text/image data, so it is independent of the size of the text/image in the original news data. Therefore, our algorithm can still maintain a high performance when the news text length changes.

5.2. Baseline methods and evaluation metric

To validate the performance of the proposed framework for fake news detection, we compare DAGA-NN with four categories of baseline models - classic, multi-modal, cross-domain, and GNN-based classification models.

In the comparison experiment with the classic classification model, we chose SVM, LSTM, and XGBoost as comparison models. In general, SVM is considered to be able to achieve substantial improvements over other methods and to behave robustly over a variety of learning tasks [49]. LSTM has a wide range of applications in the field of text classification due to its ability to learn dependencies over larger time lags and its powerful feature selection capabilities [50]. Boosting is a method to ensemble meta-algorithm for primarily reducing bias/variance in machine learning, and convert weak learners to strong ones [51]. Also, we take several state-of-the-art approaches for comparison, including Visual Question Answering (VQA) [33], att-RNN [31], event adversarial neural network (EANN) [14] and TextGCN model [36].

We use the Micro-f1 and Macro-f1 metrics [52] to evaluate the performance of all models. Assume that TP_i, TN_i, FP_i and FN_i are true-positive, true-negative, false-positive, and false-negative counts of class $i = \{1, \ldots, k\}$, respectively. Then, the precision(P_i), recall (R_i) and F1-score ($F1_i$) of class i are defined as follows:

[1] https://www.datafountain.cn/competitions/422.

Table 2
The results of different methods on the datasets of Twitter (T) and Weibo (W).

	Method	P_{fake}	R_{fake}	F_{fake}	P_{true}	R_{true}	F_{true}	Mi-f1	Ma-f1	χ^2 (p-value)
T	SVM	0.6963	0.9863	0.8163	0.7651	0.0943	0.1679	0.6991	0.4921	183.94(<0.01***)
	LSTM	0.8689	0.9733	0.9181	0.9247	0.6907	0.7907	0.8823	0.8544	905.30(<0.01***)
	XGBoost	0.8792	0.9207	0.8995	0.8145	0.7337	0.7720	0.8605	0.8357	733.21(<0.01***)
	VQA	0.8099	0.9540	0.8761	0.8452	0.5285	0.6504	0.8170	0.7632	576.98(<0.01***)
	att-RNN	0.8615	0.9674	0.9114	0.9074	0.6725	0.7724	0.8724	0.8419	847.93(<0.01***)
	EANN	0.8758	0.9776	0.9239	0.9376	0.7080	0.8068	0.8908	0.8653	956.97(<0.01***)
	TextGCN	0.9328	0.7683	0.8426	0.6441	0.8834	0.7450	0.8053	0.7938	31.70(<0.01***)
	DAGA-NN	0.8989	0.9639	0.9303	0.9102	0.7717	0.8353	0.9020	0.8828	
W	SVM	0.9263	0.7040	0.8000	0.7737	0.9475	0.8518	0.8298	0.8259	1073.61(<0.01***)
	LSTM	0.8431	0.6275	0.7195	0.7185	0.8906	0.7954	0.7634	0.7574	1036.64(<0.01***)
	XGBoost	0.9571	0.6867	0.7996	0.7680	0.9712	0.8577	0.8336	0.8287	1069.65(<0.01***)
	VQA	0.8568	0.5998	0.7057	0.7074	0.9061	0.7946	0.7580	0.7501	1046.52(<0.01***)
	att-RNN	0.9058	0.5770	0.7049	0.7044	0.9438	0.8067	0.7664	0.7558	1051.69(<0.01***)
	EANN	0.8811	0.8303	0.8549	0.8492	0.8950	0.8715	0.8637	0.8632	141.16(<0.01***)
	TextGCN	0.5484	0.7539	0.6349	0.5618	0.3370	0.4213	0.5523	0.5281	1049.49(<0.01***)
	DAGA-NN	0.9561	0.9448	0.9504	0.9488	0.9593	0.9541	0.9523	0.9522	

$$P_i = \frac{TP_i}{TP_i + FP_i},\ R_i = \frac{TP_i}{TP_i + FN_i},\ F_i = \frac{2P_i \times R_i}{P_i + R_i}. \tag{22}$$

Further, the Micro-average-precision (P_{Micro}) and Micro-average-recall (R_{Micro}) are computed as follows:

$$P_{Micro} = \frac{\Sigma_{i=1}^{k} TP_i}{\Sigma_{i=1}^{k}(TP_i + FP_i)},\ R_{Micro} = \frac{\Sigma_{i=1}^{k} TP_i}{\Sigma_{i=1}^{k}(TP_i + FN_i)}, \tag{23}$$

the Macro-average-precision (P_{Macro}) and Macro-average-recall (R_{Macro}) are as:

$$P_{Macro} = \frac{\Sigma_{i=1}^{k}\left\{\frac{TP_i}{TP_i+FP_i}\right\}}{k} = \frac{\Sigma_{i=1}^{k} P_i}{k}, R_{Macro} = \frac{\Sigma_{i=1}^{k}\left\{\frac{TP_i}{TP_i+FN_i}\right\}}{k} = \frac{\Sigma_{i=1}^{k} R_i}{k}. \tag{24}$$

The overall quality of a multi-class classification task is usually assessed by Micro-average-F1 (*Mi-f1*) and Macro-average-F1 (*Ma-f1*), which can be specified as follows:

$$Mi\text{-}f1 = \frac{2P_{Micro} \times R_{Micro}}{P_{Micro} + R_{Micro}},\ Ma\text{-}f1 = \frac{2P_{Macro} \times R_{Macro}}{P_{Macro} + R_{Macro}}. \tag{25}$$

We may see that Micro-average-precision and Micro-average-recall are all the same, and equal to the overall accuracy of the classifier, therefore the Micro-average-F1 is just the same as well.

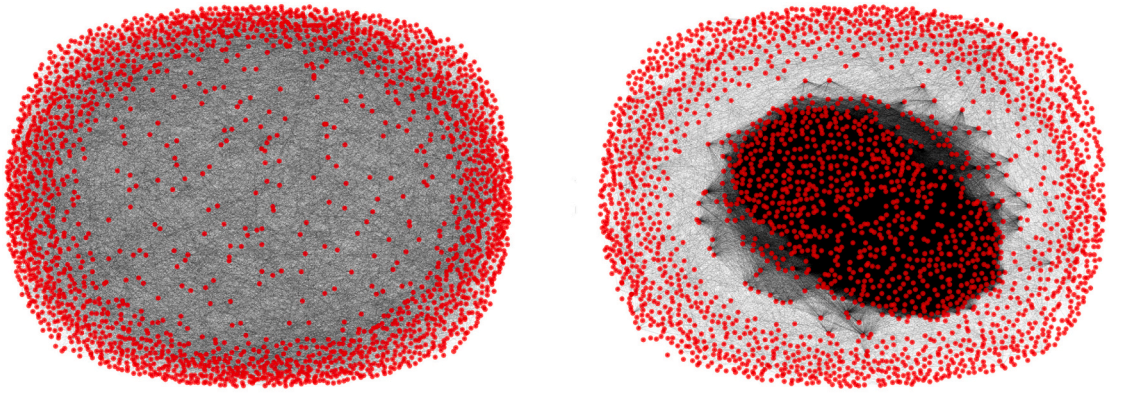
5.3. Performance comparison

Table 2 indicates the experimental results of baselines and the method on the two real datasets. In which, P_{fake}, R_{fake} and F_{fake} denote the performance of precision, recall and F-score for fake news identification; P_{true}, R_{true} and F_{true} denote the performance of precision, recall and F-score for true news identification; and *Mi - F*1 and *Ma - F*1 means the Micro-average F1- and Macro-average F1-score.

To test the significance of DAGA-NN relative to the comparison models in identifying fake news, we have introduced the McNemar's-Test [53] method in the experiments. Statistically based results show that our method performs significantly better than those of comparative methods on the task of identifying fake news (see Table 2, in which, *** indicates significance at the level of 1%, ** at 5%, and * at 10%).

In the task of identifying fake news in the Twitter dataset, we can see that the EANN outperformed the other methods (except DAGA-NN) by a slight margin. Since the strength of EANN is adversarial learning, it can obtain some "common features" (e.g., writing patterns and syntax [42]) of news across domains. However, the above comparison results show that the advantage of EANN is not significant, suggesting that the "common features" of fake news across different Twitter events might be less obvious, which may have limited the efficiency of the adversarial learning approach in identifying fake news in such kind of news data. By contrast, our model, DAGA-NN still achieved good performance for fake news identification when the common features across events/domains were not so obvious, because it introduced the similar relationships between the texts of true (fake) news in the same domain and the information about how these texts interact with each other.

On the Weibo dataset, some observations were similar to those on the Twitter dataset. Notably, almost all models, except SVM and DAGA-NN, were less effective in identifying fake news on the Weibo dataset than



(a) Similarity of texts ("SandyB" event) (b) Similarity of texts (" Entertainment" domain)

Fig. 3. The similarity diagrams of news texts (threshold is 0.9).

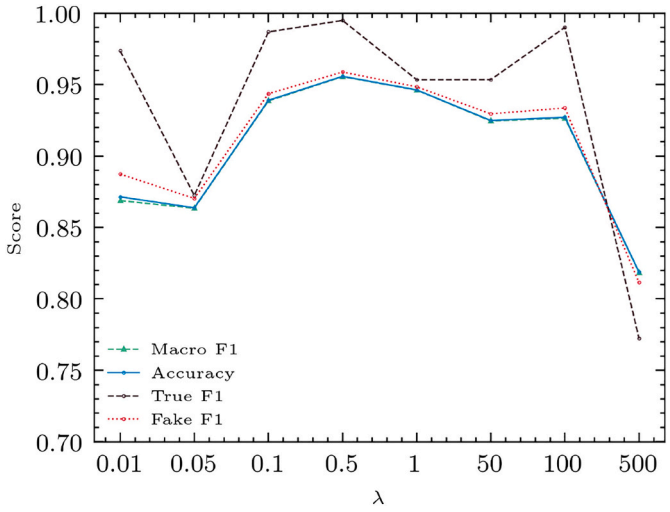


Fig. 4. The effect of parameter λ.

they were on the Twitter dataset. One of the reasons was the incomplete image data on the Weibo dataset.

From the experimental results, DAGA-NN retained higher performance in fake news identification tasks on both datasets because it emphasized the role of the structure of relationships between news texts within the same domain. Fig. 3(a) shows the similarity relationships of news texts about the "SandyB" event in the Twitter dataset. Similarly, Fig. 3(b) shows the similarity relationships of news texts within the "Entertainment" domain in the Weibo dataset. We found that there were more nodes in the Weibo dataset where the textual content of news in the same domain is highly correlated with each other than in the Twitter dataset. This also made DAGA-NN perform better on the task of identifying fake news in the Weibo dataset.

In addition, we have examined the effect of the λ parameters on the performance of DAGA-NN. The results are shown in the Fig. 4 below. We can see that DAGA-NN performs best when λ values are around 0.5. Further, DAGA-NN works well when the λ values vary in the range 0.1–100, where the average F-scores are greater than 0.9.

6. Discussion

To illustrate how DAGA-NN works in fake news identification tasks, in this section, we aim to analyze the effectiveness of each component of the DAGA-NN architecture and the convergence of model training.

6.1. Component effectiveness analysis

To observe the different contributions of each component in DAGA-NN in identifying fake news, we first constructed the simplest *fake news classifier* directly based on the *feature extractor* (FE) and then gradually added the *domain discriminator* (DD) and the *graph attention layer* (GAL) to the classifier. The performance results of these three differently constructed classifiers in identifying fake news in the Weibo and Twitter datasets are shown in the following Table 3.

Table 3
Effectiveness of the components of the DAGA-NN in detecting fake news.

Dataset	Classifier component	Accuracy	Precision	Recall	F1-score
Twitter	FE	0.8854	0.8896	0.8854	0.8806
	FE + DD	0.8901	0.8957	0.8901	0.8844
	FE + DD + GAL	0.9054	0.9050	0.9054	0.9052
Weibo	FE	0.7706	0.8031	0.7706	0.7625
	FE + DD	0.8576	0.8624	0.8576	0.8568
	FE + DD + GAL	0.9450	0.9479	0.9450	0.9448

Based on the experiment results shown in Table 3, first, the FE's contribution is crucial to the task of identifying fake news; second, both the DD and GAL may have impacts on the performance of fake news classifiers. In particular, the proposed GAL, which specifically considers the role of relationships between similar news texts (neighbors on the graph) with the same class label, could make a significant contribution to enhancing the performance of the fake news classifiers.

To further analyze the effectiveness of the proposed GAL, we used the t-SNE method [54] to visualize the features obtained on the Weibo dataset using only the FE and the features obtained using FE + GAL, respectively. The results are shown in Fig. 5; the borderline between fake news (blue) and true news (red) was more pronounced after using the proposed graph attention module.

6.2. Convergence analysis

As an integrated architecture of three complex functional modules, we also analyze the training process of DAGA-NN on the Weibo dataset.

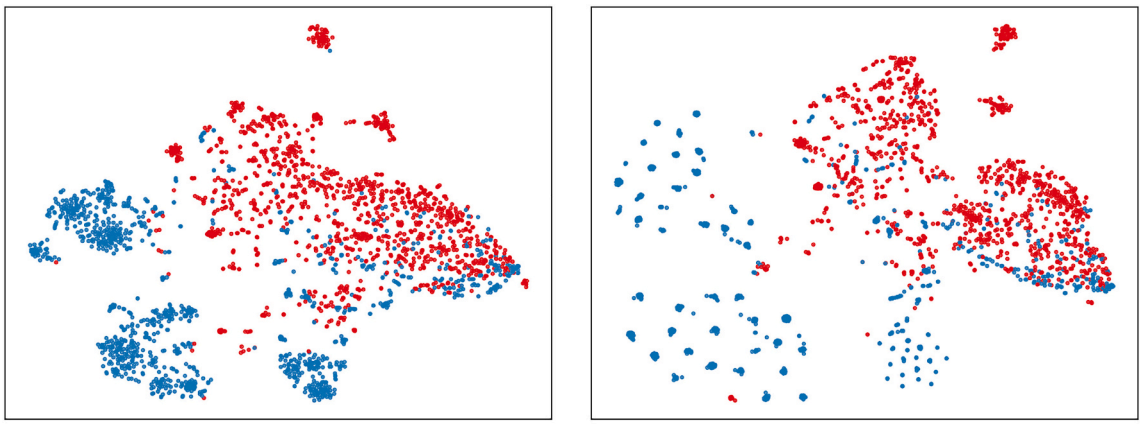
In Fig. 6(a), the loss of *domain discriminator* decreases during the initial training phase and then rises slightly to stabilize at a certain level. This phenomenon indicates that *domain discriminator* can capture the domain information in the feature representation provided by the *feature extractor* easily during the initial phase of training. However, as the minmax game between the *domain discriminator* and *feature extractor* is repeated, the feature information provided by the *feature extractor* becomes progressively domain-invariant, which leads to an increase in training loss of the *domain discriminator*. Eventually, the training loss of the *domain discriminator* converges to a more stable level during the training phase, indicating that the *domain discriminator* and *feature extractor* game has reached some degree of equilibrium. In addition, the changes in *acurracy* in Fig. 6(b) indicates similar trends as that in Fig. 6 (a).

Further, we found a large fluctuation in test losses at the beginning of training, which could be due to the (local) diversity of the graph structure established by news text similarity. Nevertheless, all three types of losses converge smoothly after a period of training (see Fig. 6 (a)). Notably, after multiple rounds of training and testing, the tails of the training and testing losses show small fluctuations due to overfitting. To address the impacts of overfitting, we extract some data from the training set as a validation dataset and stop the training process in time based on the stable performance of DAGA-NN on the validation dataset.

6.3. Multi-class problems in identifying fake news

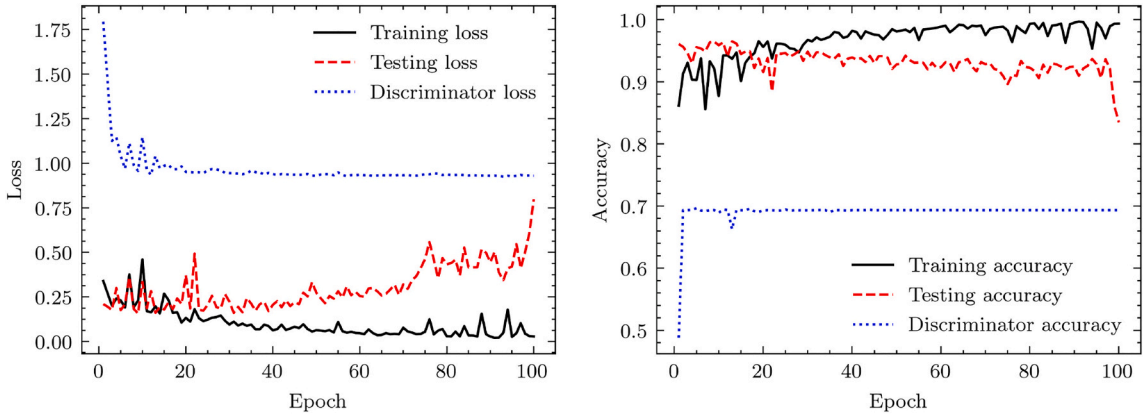
While there are many benefits to viewing fake news identification as a binary class problem. For experts, it makes their perception of news contents clearer and thus the cost of labeling might be reduced. For the average reader, it also makes the identification results easier to understand. However, fake news detection (and labels) in real world is hard to justify as a binary task. Firstly, different fake-news-makers may handle the original content in different ways. As a result, the true and fake content of a news story may be mixed in different degrees. This gives rise to the problem of multiple classifications in fake news identification [55]. Second, fake news contents will evolve during the process of dissemination. In particular, when a fake news becomes widely known, the probability of it being identified increases. Therefore, it will further evolve to keep its content away from absolute fake/true news content that can be easily identified. Finally, online fake news is a combination of multi-modal data (e.g., text, voice and images). Data in different modalities may be tampered with to varying degrees by fake news makers. For example, some news texts are fake, but their accompanying images may be real. These give rise to the importance of studying the problem of multiple classification in fake news identification [55].

To test the performance of DAGA-NN and the comparison algorithms in identifying multi-class labeled fake news, we introduced an additional dataset with news content having six fine-grained labels for the



(a) Features extracted from FE. (b) Features extracted from GAL.

Fig. 5. Visualization of the feature representations obtained on the Weibo dataset.



(a) The three losses. (b) The accuracy.

Fig. 6. Training, testing and *domain discriminator* loss development.

Table 4
The results of different methods on Liar dataset.

Method	PF-F1	F-F1	BF-F1	HT-F1	MT-F1	T-F1	Accuracy	Ma-f1
SVM	0.0000	0.3150	0.0000	0.3344	0.0000	0.0000	0.2278	0.1082
CNN-LSTM	0.1148	0.2691	0.1946	0.3401	0.3278	0.1860	0.2799	0.2387
XGBoost	0.0168	0.2629	0.1931	0.3112	0.3451	0.1773	0.2676	0.2177
VQA	0.0522	0.3129	0.1640	0.2681	0.3680	0.0062	0.2676	0.1952
att-RNN	0.0000	0.2686	0.1598	0.2560	0.3668	0.0000	0.2584	0.1752
EANN	0.1722	0.2806	0.1768	0.3014	0.2910	0.1494	0.2533	0.2286
TextGCN	0.0000	0.0000	0.0000	0.3832	0.0000	0.0000	0.2370	0.0639
DAGA-NN	0.1880	0.3433	0.1141	0.3124	0.3650	0.1739	0.2993	0.2495

truthfulness ratings, namely *pants-fire* (PF), *false* (F), *barely-true* (BT), *half-true* (HT), *mostly-true* (MT), and *true* (T) [55]. The results are presented in Table 4. From the results, overall, our method outperforms the comparison methods on both *Accuracy* (*Mi - f*1) and *Ma - f*1 metrics. Specifically, our method is very good at identifying "very fake" news content (e.g., the *pants-fire* and *false* news). Although our method does not show an absolute advantage in identifying content that is close to "real news" (e.g., the *mostly-true* and *true* news), it also achieves good results. Such results show that our approach (especially the graph-attention model) is very good at learning a clear boundary between fake and true news. However, it is not yet optimal for identifying the truthfulness of news content whose semantics lie near the boundary.

7. Conclusion

Effective identification of fake news is a valuable research topic. Previous studies in this field have followed the patterns and strategies of the general machine learning (deep learning) methods, making the trained models strongly correlated with the domain characteristics (feature space) of the training samples [56]. However, two challenges make these models often less effective in identifying fake news in real-world scenarios than they are in training data contexts. First, the news contents on social media may have numerous domains. Second, the contents of fake news may evolve and mutate during online dissemination.

In this study, we argue that common information across news

domains and semantic relationships between true (fake) news are crucial in fake news identification tasks. Therefore, we propose DAGA-NN to address these problems. DAGA-NN uses a multi-modal *feature extractor* to obtain the basic feature representations of news from the training dataset. To identify fake news across events/domains, DAGA-NN adopts the idea of the domain-adversarial network to introduce a *domain discriminator* to play a minmax game with the *feature extractor* to learn the domain-invariant features. Importantly, DAGA-NN improves the graph attention model to delineate the boundary between the true and fake news texts within the same domain. Experiment results on two real corpora showed that DAGA-NN outperformed state-of-the-art baselines.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (NSFC) [71671027/71572029/91846105]. This research was also supported by Kunming *E*-commerce and Internet Finance R&D Center (KEIRDC[2020]), the Prominent Educator Program (Yunnan [2018]11), Yunnan Province Young Academic and Technical Leader candidate Program (2018HB027) & Yunnan Provincial E-Business Entrepreneur Innovation Interactive Space (2017DS012).

References

[1] N.K. Lankton, C. Speier, E.V. Wilson, Internet-based knowledge acquisition: task complexity and performance, Decis. Support. Syst. 53 (1) (2012) 55–65.
[2] S. Feuerriegel, H. Prendinger, News-based trading strategies, Decis. Support. Syst. 90 (2016) 65–74.
[3] D. Nozza, P. Manchanda, E. Fersini, M. Palmonari, E. Messina, Learningtoadapt with word embeddings: domain adaptation of named entity recognition systems, Inf. Process. Manag. 58 (3) (2021) 102537.
[4] F. Hogenboom, F. Frasincar, U. Kaymak, F. de Jong, E. Caron, A survey of event extraction methods from text for decision support systems, Decis. Support. Syst. 85 (2016) 12–22.
[5] Y.-H. Lee, P.J.-H. Hu, H. Zhu, H.-W. Chen, Discovering event episodes from sequences of online news articles: a time-adjoining frequent itemset-based clustering method, Inf. Manag. 57 (7) (2020) 103348.
[6] S.W. Chan, M.W. Chong, Sentiment analysis in financial texts, Decis. Support. Syst. 94 (2017) 53–64.
[7] C. Zhang, A. Gupta, C. Kauten, A.V. Deokar, X. Qin, Detecting fake news for reducing misinformation risks using analytics approaches, Eur. J. Oper. Res. 279 (3) (2019) 1036–1052.
[8] X. Zhang, A.A. Ghorbani, An overview of online fake news: characterization, detection, and discussion, Inf. Process. Manag. 57 (2) (2020) 102025.
[9] M. Wessel, F. Thies, A. Benlian, The emergence and effects of fake social information: evidence from crowdfunding, Decis. Support. Syst. 90 (2016) 75–85.
[10] S. Talwar, A. Dhir, D. Singh, G.S. Virk, J. Salo, Sharing of fake news on social media: application of the honeycomb framework and the third-person effect hypothesis, J. Retail. Consum. Serv. 57 (2020) 102197.
[11] A. Bondielli, F. Marcelloni, A survey on fake news and rumour detection techniques, Inf. Sci. 497 (2019) 38–55.
[12] M. García Lozano, J. Brynielsson, U. Franke, M. Rosell, E. Tjörnhammar, S. Varga, V. Vlassov, Veracity assessment of online data, Decis. Support. Syst. 129 (2020) 113132.
[13] K. Yan, L. Kou, D. Zhang, Learning domain-invariant subspace using domain features and independence maximization, IEEE Transactions on Cybernetics 48 (1) (2018) 288–299.
[14] Y. Wang, F. Ma, Z. Jin, Y. Yuan, G. Xun, K. Jha, L. Su, J. Gao, 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, KDD '18, ACM, 2018, pp. 849–857.
[15] M. Wu, S. Pan, X. Zhu, C. Zhou, L. Pan, Domain-adversarial graph neural networks for text classification, in: 2019 IEEE International Conference on Data Mining (ICDM), 2019, pp. 648–657.
[16] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Liò, Y. Bengio, Graph attention networks, in: Proceeding of the International Conference on Learning Representations, ICLR, 2018, pp. 1–12.
[17] L. Wu, F. Morstatter, K.M. Carley, H. Liu, Misinformation in social media: definition, manipulation, and detection, SIGKDD Explor. Newsl. 21 (2) (2019) 80–90.
[18] Y. Wang, W. Yang, F. Ma, J. Xu, B. Zhong, Q. Deng, J. Gao, Weak supervision for fake news detection via reinforcement learning, in: The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI Press, 2020, pp. 516–523.
[19] K. Wu, S. Yang, K.Q. Zhu, False rumors detection on sina weibo by propagation structures, in: 2015 IEEE 31st International Conference on Data Engineering, 2015, pp. 651–662.
[20] J. Ma, W. Gao, Z. Wei, Y. Lu, K.-F. Wong, Detect Rumors Using Time Series of Social Context Information on Microblogging Websites, in: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, CIKM '15, 2015, pp. 1751–1754.
[21] Y. Liu, Y.-F.B. Wu, Fned: a deep network for fake news early detection on social media, ACM Trans. Inf. Syst. 38 (3) (2020).
[22] C. Castillo, M. Mendoza, B. Poblete, Information credibility on twitter, in: Proceedings of the 20th International Conference on World Wide Web, WWW '11, 2011, pp. 675–684.
[23] M. Dong, L. Yao, X. Wang, B. Benatallah, Q.Z. Sheng, H. Huang, Dual: A deep unified attention model with latent relation representations for fake news detection, in: H. Hacid, W. Cellary, H. Wang, H.-Y. Paik, R. Zhou (Eds.), Web Information Systems Engineering – WISE, Springer, 2018, pp. 199–209.
[24] J.P. Dickerson, V. Kagan, V.S. Subrahmanian, Using sentiment to detect bots on twitter: Are humans more opinionated than bots?, in: Proceedings of the 2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM '14, 2014, pp. 620–627.
[25] Y. Wu, P.K. Agarwal, C. Li, J. Yang, C. Yu, Toward computational fact-checking, Proc. VLDB Endow. 7 (7) (2014) 589–600.
[26] Y. Wang, S. Qian, J. Hu, Q. Fang, C. Xu, Fake News Detection Via Knowledge-Driven Multimodal Graph Convolutional Networks, in: Proceedings of the 2020 International Conference on Multimedia Retrieval, ICMR '20, ACM, New York, NY, USA, 2020, pp. 540–547.
[27] V.L. Rubin, T. Lukoianova, Truth and deception at the rhetorical structure level, J. Assoc. Inf. Sci. Technol. 66 (5) (2015) 905–917.
[28] M. Potthast, J. Kiesel, K. Reinartz, J. Bevendorff, B. Stein, 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), ACL, Melbourne, Australia, 2018, pp. 231–240.
[29] M. Granik, V. Mesyura, Fake news detection using naive bayes classifier, in: 2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON), 2017, pp. 900–903.
[30] J. Ma, W. Gao, P. Mitra, S. Kwon, B.J. Jansen, K.-F. Wong, M. Cha, Detecting rumors from microblogs with recurrent neural networks, in: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI'16, AAAI Press, 2016, pp. 3818–3824.
[31] Z. Jin, J. Cao, H. Guo, Y. Zhang, J. Luo, Multimodal fusion with recurrent neural networks for rumor detection on microblogs, in: Proceedings of the 25th ACM International Conference on Multimedia, MM '17, ACM, 2017, pp. 795–816.
[32] P. Qi, J. Cao, T. Yang, J. Guo, J. Li, Exploiting multi-domain visual information for fake news detection, in: 2019 IEEE International Conference on Data Mining (ICDM), 2019, pp. 518–527.
[33] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C.L. Zitnick, D. Parikh, Vqa: Visual question answering, in: 2015 IEEE International Conference on Computer Vision (ICCV), 2015, pp. 2425–2433.
[34] T.N. Kipf, M. Welling, Semi-Supervised Classification with Graph Convolutional Networks, in: Proceedings of the 5th International Conference on Learning Representations, ICLR '17, 2017.
[35] L. Huang, D. Ma, S. Li, X. Zhang, H. Wang, Text Level Graph Neural Network for Text Classification, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), ACL, Hong Kong, China, 2019, pp. 3444–3450.
[36] L. Yao, C. Mao, Y. Luo, Graph convolutional networks for text classification, in: The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI, 2019, pp. 7370–7377.
[37] Y. Zhang, X. Yu, Z. Cui, S. Wu, Z. Wen, L. Wang, Every Document Owns its Structure: Inductive Text Classification Via Graph Neural Networks, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL, 2020, pp. 334–339. Online.
[38] Z. Lu, P. Du, J.-Y. Nie, Vgcn-bert: Augmenting bert with graph embedding for text classification, in: J.M. Jose, E. Yilmaz, J. Magalhães, P. Castells, N. Ferro, M. J. Silva, F. Martins (Eds.), Advances in Information Retrieval, Springer, Cham, 2020, pp. 369–382.
[39] Z. Ye, G. Jiang, Y. Liu, Z. Li, J. Yuan, Document and Word Representations Generated by Graph Convolutional Network and BERT for Short Text Classification, in: 24th European Conference on Artificial Intelligence, ECAI, IOS Press, 2020, pp. 2275–2281.
[40] Y.-J. Lu, C.-T. Li, GCAN: Graph-Aware Co-Attention Networks for Explainable Fake News Detection on Social Media, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL, 2020, pp. 505–514. Online.
[41] V.-H. Nguyen, K. Sugiyama, P. Nakov, M.-Y. Kan, Leveraging social context for fake news detection using graph representation, in: Proceedings of the 29th ACM International Conference on Information and Knowledge Management, CIKM '20, ACM, 2020, pp. 1165–1174.
[42] K. Shu, A. Sliva, S. Wang, J. Tang, H. Liu, Fake news detection on social media: a data mining perspective, ACM SIGKDD Explorations Newsletter 19 (2017) 22–36.
[43] X. Qiu, T. Sun, Y. Xu, Y. Shao, N. Dai, X. Huang, Pre-trained models for natural language processing: a survey, SCIENCE CHINA Technol. Sci. 63 (10) (2020) 1872–1897.
[44] M. Schuster, K.K. Paliwal, Bidirectional recurrent neural networks, IEEE Trans. Signal Process. 45 (11) (1997) 2673–2681.
[45] K. Simonyan, A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, in: 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, 2015, pp. 1–14.

[46] A. Kolesnikov, X. Zhai, L. Beyer, Revisiting Self-Supervised Visual Representation Learning, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, 1920–1929.
[47] I.J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative Adversarial Nets, in: Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2, NIPS'14, MIT Press, Cambridge, MA, USA, 2014, pp. 2672–2680.
[48] C. Boididou, K. Andreadou, S. Papadopoulos, D.-T. Dang-Nguyen, G. Boato, M. Riegler, Y. Kompatsiaris, Verifying multimedia use at mediaeval 2015, in: MediaEval 2015 Workshop, 2015.
[49] T. Joachims, Text categorization with support vector machines: Learning with many relevant features, in: C. Nédellec, C. Rouveirol (Eds.), Proceedings of ECML-98, 10th European Conference on Machine Learning, Springer Verlag, Heidelberg, DE, 1998, pp. 137–142.
[50] P. Liu, X. Qiu, X. Huang, Recurrent neural network for text classification with multi-task learning, in: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI'16, AAAI Press, 2016, pp. 2873–2879.
[51] T. Chen, C. Guestrin, Xgboost: A scalable tree boosting system, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16, ACM, 2016, pp. 785–794.
[52] G. Madjarov, D. Kocev, D. Gjorgjevikj, S. Džeroski, An extensive experimental comparison of methods for multi-label learning, Pattern Recogn. 45 (9) (2012) 3084–3104.
[53] T.G. Dietterich, Approximate statistical tests for comparing supervised classification learning algorithms, Neural Comput. 10 (7) (1998) 1895–1923.
[54] L. van der Maaten, G. Hinton, Visualizing data using t-SNE, J. Mach. Learn. Res. 9 (2008) 2579–2605.
[55] W.Y. Wang, liar, liar pants on fire: A new benchmark dataset for fake news detection, in: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Association for Computational Linguistics, Vancouver, Canada, 2017, pp. 422–426.
[56] S.J. Pan, Q. Yang, A survey on transfer learning, IEEE Trans. Knowl. Data Eng. 22 (10) (2010) 1345–1359.

Hua Yuan is currently a full professor of information systems and information management at the University of Electronic Science and Technology of China. His research interests focus on business intelligence, information technology management, social media and networks, and information security management. His research has been published in Decision Support Systems, Information Sciences, The Computer Journal, and Applied Mathematics and Computation, and presented at a number of computer science and information system conferences.

Jie Zheng is currently a master candidate in the EC-lab at the University of Electronic Science and Technology of China. His general research interests are data mining and business intelligence.

Qiongwei Ye is a professor of Electronic Business and vice dean of Business School, Yunnan University of Finance and Economics (Kunming, China). He served as a member of the College Teaching Steering Committee for *E*-Business under the Ministry of Education of China, the vice president of China Information Economics Society and the director the internet economy and cross-border E-Business division of and the director of the Key Laboratory of E-Business Innovation and Entrepreneurship in Yunnan Province. His research interests include E-Business, business intelligence and analytics, and information & experience economics.

Yu Qian is an Associate Professor of management science at the University of Electronic Science and Technology of China (UESTC). Her general research interests include supply chain management, operation management, information economics and e-commerce. Her papers have been published and presented in journals and conferences such as the Flexible Services and Manufacturing Journal, Journal of Systems Science and Systems Engineering, and POMS annual conference.

Yan Zhang is currently a master candidate in the University of Electronic Science and Technology of China. Her general research interests is about data driven business management.