Information Sciences 525 (2020) 205–220

Contents lists available at ScienceDirect

Information Sciences

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

A multimodal generative and fusion framework for recognizing faculty homepages

Guanyuan Yu, Qing Li*, Jun Wang, Di Zhang, Yuehao Liu

School of Economic Information Engineering, Southwestern University of Finance and Economics, Chengdu, China

ARTICLE INFO

Article history:
Received 17 May 2019
Revised 10 February 2020
Accepted 2 March 2020
Available online 2 March 2020

Keywords:
Homepages
Multimodal generative adversarial network
Gated fusion network

ABSTRACT

Multimodal data consist of several data modes, where each mode is a group of similar data sharing the same attributes. Recognizing faculty homepages is essentially a multimodal classification problem in which a target faculty homepage is determined from three different information sources, including text, images, and layout. Conventional strategies in previous studies have been either to concatenate features from various information sources into a compound vector or to input them separately into several different classifiers that are then assembled into a stronger classifier for the final prediction. However, both approaches ignore the connections among different feature sets. We argue that such relations are essential to enhance multimodal classification. Besides, recognizing faculty homepages is a class imbalance problem in which the total number of samples of a minority class is far smaller than the sample numbers of other classes. In this study, we propose a multimodal generative and fusion framework for multimodal learning with the problems of imbalanced data and mutually dependent feature modes. Specifically, a multimodal generative adversarial network is first introduced to rebalance the dataset by generating pseudo features based on each mode and combining them to describe a fake sample. Then, a gated fusion network with the gate and fusion mechanisms is presented to reduce the noise to improve the generalization ability and capture the links among the different feature modes. Experiments on a faculty homepage dataset show the superiority of the proposed framework.

1. Introduction

The goals of a faculty search engine are to obtain relevant information on researchers and trace hot research topics. In this research, we have designed and implemented a vertical search engine, Professor++[1], for this purpose. At present, it covers the top 100 universities in the United States, but it will be extended to all universities soon. This faculty-oriented search engine provides query functions in terms of name, university, and research area. It also supports advanced statistical analyses, including analyses of the distributions of faculty members in terms of research interests, ethnicity, and gender. For instance, it can answer questions such as how many researchers are focusing on quantum computing or nanotechnology and What the university distribution of these experts is. Professor++ is able to provide such knowledge from the extracted faculty member database. Such knowledge can be further incorporated into an advanced academic question & answering system. Besides, we have developed a software application called Face++ to help academic conference participants to become

* Corresponding author.

[1] http://www.findingprofessor.com.

https://doi.org/10.1016/j.ins.2020.03.005
0020-0255/© 2020 Published by Elsevier Inc.

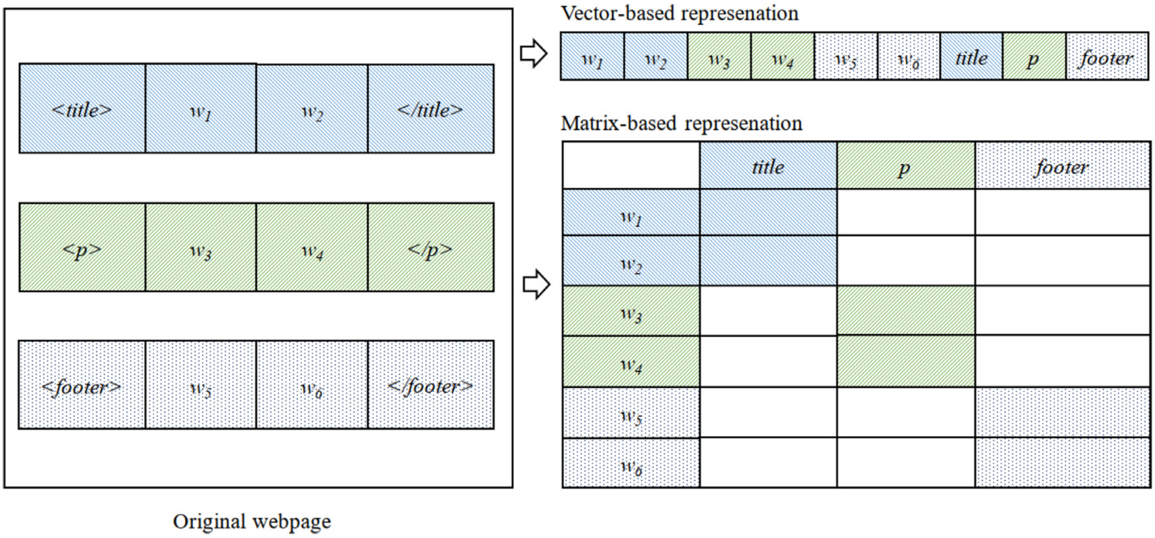


Fig. 1. Modeling multimodal data.

familiar with session speakers. Face++ identifies a professor and provides her/his relevant information to users by taking a photograph and searching the faculty member database of Professor++ via a face recognition technique.

The ability to automatically recognize faculty homepages on university websites is critical for building such a faculty-oriented search engine. This recognition problem is essentially a multimodal classification problem that consists of several data modes, where all data in each mode share the same attributes. Specifically, a target faculty homepage is recognized by evaluating three different information sources, including text, images, and layout. There are two critical challenges in this study.

- **Data imbalance**: Imbalanced training occurs when one class is represented by a large number of samples, while another is represented by only a few [39].
  It has been observed that such an imbalance may degrade the classification accuracy because the classifier will pay less attention to the minority class [14].
  Generally, the number of faculty homepages is much lower than the total number of university webpages.
  A good example is the official website of Yale University, which contains fewer than 5000 faculty and researcher homepages compared with 50,000+ other types of webpages.
- **Multimodality issues**: Generally, the page source of a webpage consists of three parts: text, images, and layout.
  Thus, it is critical to distinguish a target homepage from other types of webpages using these three different feature modes.
  Common strategies in previous studies have been either to concatenate features from the various information sources into one compound vector or to feed them separately into different classifiers, which are then assembled into a stronger classifier for a final decision.
  However, both approaches inevitably ignore the connections among different types of features [34].
  For example, as displayed in Fig. 1, the layout feature set consists of three tags, namely, ⟨*title*⟩, ⟨*p*⟩, and ⟨*footer*⟩.
  Each tag contains some textual information.
  It is evident that the words embedded in the tag ⟨*title*⟩ are more important than those within ⟨*footer*⟩.
  However, such relations are ignored if the tag and text features are concatenated into a compound vector.

In this study, we propose a multimodal generative and fusion framework for recognizing faculty homepages, which is a typical multimodal learning problem with imbalanced data and mutually dependent feature modes. Our solution makes three unique contributions as follows:

- To overcome the class imbalance problem for multimodal data, we design an advanced framework called a multimodal generative adversarial network, which generates fake multimodal features while preserving the connections among them.
- To address the mutual dependence of multimodal data, we propose a fusion neural network.
  This network is achieved by applying the gate mechanism to alleviate noise and redundant features and utilizing the Kronecker product to capture connections among different data modes.
- To the best of our knowledge, Professor++ is the first faculty-oriented search engine to be developed.
  Our source code for recognizing faculty homepages can be accessed via GitHub.[2]

[2] https://github.com/mrspider520/gated_fusion_network.git.

The remainder of this article is organized as follows. Section 2 briefly describes the previous work related to our research. Section 3 presents the design details of the proposed frameworks. Section 4 examines the effectiveness of our approach. Finally, Section 5 concludes the article and offers suggestions for future work.

2. Related work

2.1. Multimodal classification

The problem of recognizing faculty homepages is essentially a classification problem. The typical approach to identifying homepages is to train a binary classifier on webpage features using labeled data and then apply the resulting well-trained classifier to recognize new homepages. However, most previous works on such partitioning of data have relied on webpage features of only one type, such as URLs [3,9,23], text [12], or HTML layout [13]. By contrast, the recognition of faculty homepages is a multimodal data problem with three different data modes (text, images, and layout), each of which carries different information that can be used to support supervised learning. Many studies have found that classifiers using multiple feature sets can achieve better performance than classifiers using a single feature set [4,15,24,33].

Two common strategies have been applied in previous studies for using multiple feature sets. The first strategy is to input the features of different modes separately into different classifiers, which are further assembled into one stronger classifier to reach the final decision via a voting or stacking mechanism [34]. For instance, Joachims et al. trained several support vector machines (SVMs) with different feature sets, including URLs and text, then combined these SVMs into a single strong classifier via a voting mechanism [22]. Glover et al. combined the results from an SVM using extended anchor text features and an SVM using full-text features to achieve better performance [15]. Chen and Hsieh first built an SVM with literal words from webpages and then trained another SVM using semantic information via latent semantic analysis (LSA). A weighting schema was subsequently utilized to assemble these SVMs to make the final decision [6].

The other approach is to concatenate features from various information sources into a single compound vector. For example, Kang and Kim combined feature sets consisting of text, links, and URLs using a weighted sum [24].

However, both of these approaches ignore or dismantle the connections among the different feature sets (modes), although these connections are critical for multimodal supervised learning. In this study, we use three data modes (text, images, and layout) to identify faculty homepages. The connections among these feature modes are essential for identifying homepages. A good example is that the textual information contained in different layout tags has different levels of importance. Destroying these connections results in a loss of valuable information and reduces the classification accuracy. We argue that such relations are critical for multimodal classification. Therefore, we propose a gated fusion network (GFN) to incorporate the relations among different feature modes for multimodal data classification. This is achieved by applying the gate mechanism to alleviate noise and redundant features and utilizing the Kronecker product to capture the connections among different data modes.

2.2. Imbalanced classification

The class imbalance problem refers to the situation in which one class is represented by a large number of examples (the majority class). In contrast, another is represented by only a few examples (the minority class) [39]. Most classical learning algorithms are designed for balanced datasets [11]. Once the dataset becomes imbalanced, the model performance declines because the characteristics of the minority class cannot be learned effectively [17]. The classifier will tend to overrepresent the majority class compared to the minority class. The minority class may be so small that it may be easily ignored, treated as noise, or misidentified as the majority class [18,21,26,32]. Louzada et al. reported that class imbalance data lead to severe deterioration in model performance [30].

The traditional techniques for dealing with the class imbalance problem include the random oversampling (OS), the random undersampling (US), and the synthetic minority oversampling technique (SMOTE) [5].

In the US approach, observations from the majority class are randomly dropped until the remaining number of majority-class samples matches the number of samples in the minority class. This approach results in a loss of valuable information, which inevitably reduces the classification performance [7].

In contrast, the OS approach involves randomly duplicating observations from the minority class until the total number of minority-class samples matches the number of samples in the majority class. However, this approach is prone to overfitting due to the simple replication of samples from the minority class [10,17,38]. In essence, this approach cannot provide additional valuable information for use in classification.

To overcome the weakness of the OS approach, Chawla et al. proposed the SMOTE [5]. This approach increases the number of samples in the minority class by creating virtual samples, each of which is a linear combination of two real samples from the minority class that are located near each other, to rebalance the data. However, these virtual samples are merely linear combinations of local information instead of the overall minority class distribution [10]. Besides, the SMOTE may produce noisy samples, when the boundary between the majority and minority classes is not sufficiently clear [17].

Consequently, it is a critical challenge to generate synthetic samples based on the real minority class distribution. Some researchers have taken the further step of utilizing generative adversarial networks (GANs). Such a framework involves training a generator network and a discriminator network, which compete with each other in a zero-sum game. A well-trained

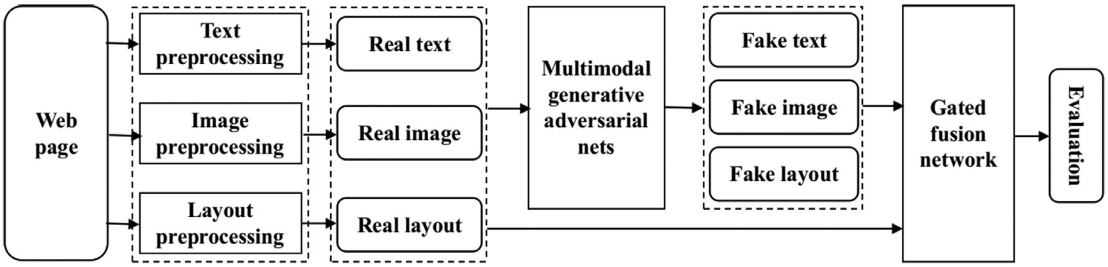


Fig. 2. System framework.

generator can estimate the latent distribution of the real data, and then produces some fake samples based on the global distribution instead of using only local information [16]. For example, Shin et al. utilized a GAN to rebalance medical data by synthesizing abnormal brain tumor MRI images because only limited data from patients with cancer were available [37].

However, previous studies on the application of imbalanced data have focused on unimodal data rather than multimodal data. In this study, we propose a multimodal generative adversarial network (MGAN) for synthesizing samples to address the problem of imbalanced multimodal data. Specifically, the MGAN generates fake faculty homepages by simultaneously using features from three modes(text, images, and layout) in accordance with the global feature distributions and their relations to improve the classification of imbalanced webpages.

3. System design

In this study, we propose a multimodal generative and fusion framework for multimodal learning with the problems of imbalanced data and mutually dependent feature modes. Fig. 2 presents an overview of the proposed generative and fusion framework. Text features, image features, and layout features are first extracted from a set of original webpages and are then preprocessed to form a multimodal dataset. An MGAN is introduced to rebalance the dataset by generating pseudo features for each mode and combining them to describe a fake sample. Then, a GFN with the gate and fusion mechanisms is presented to reduce noise in order to improve the generalization ability and to capture the links among different feature modes.

3.1. Features

The source of a webpage typically consists of three feature sets, including text, images, and layout. These features are described in detail below.

- **Text features**: The text of a webpage can be represented as a word list $\boldsymbol{x}_t \in \mathbb{R}^N$, where N is the number of words on the webpage.
  To enhance the semantic and contextual information it contains, the text can be further represented by word embeddings.
  In this study, we use the Google word vector model[3] to convert $\boldsymbol{x}_t$ into a matrix $\boldsymbol{X}_t \in \mathbb{R}^{N\times E}$, where E is the embedding size.
  Then, we apply convolutional kernels to extract the semantic information from the word embeddings, as suggested by Yih et al. [41].
  Thus, $\boldsymbol{X}_t$ is further transformed into a more abstract feature vector $\boldsymbol{h}_t$.
- **Image features**: Instead of representing images as pixels, we abstract the image features of a webpage as a four-dimensional vector $\boldsymbol{x}_p \in \mathbb{R}^4$.
  The elements of this vector include the number of zero-face images, the number of one-face images, the number of multiple-face images, and the total number of images.
  Whether an image includes one or more faces is determined using a Histograms of Oriented Gradients (HOG) face recognition algorithm [8]. $\boldsymbol{h}_p$ is a high-level feature vector obtained from $\boldsymbol{x}_p$.
- **Layout features**: The layout features of a webpage are represented by a tag vector $\boldsymbol{x}_s \in \mathbb{R}^M$, where M is the total number of tags.
  Each element in this vector reflects the number of HTML tags of a particular type, such as $\langle a\rangle$, $\langle p\rangle$, or $\langle span\rangle$.
  Here, $\boldsymbol{h}_s$ is the corresponding high-level feature vector.

[3] This model comprises 3 million 300-dimensional English word vectors and is accessible at https://code.google.com/archive/p/word2vec/.

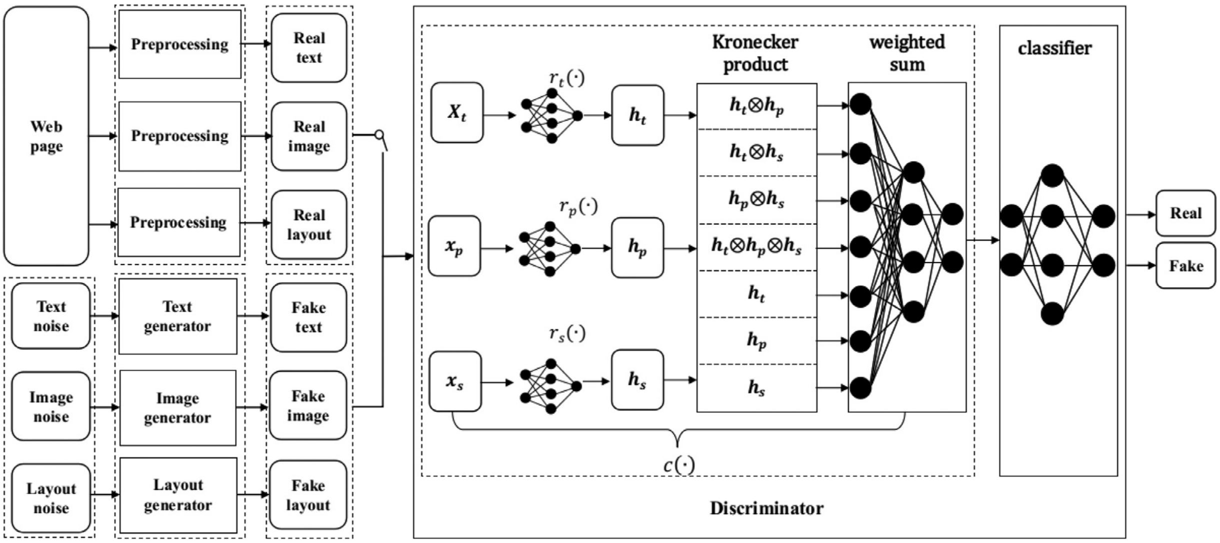


Fig. 3. Multimodal generative adversarial network.

3.2. Multimodal generative adversarial network (MGAN)

To overcome the class imbalance problem for multimodal data, we design a novel architecture called the MGAN to generate artificial samples of the minority class. The MGAN first creates fake features for each feature mode via an iterative adversarial training process. In this way, this procedure can make the fused distribution of the fake features approach the joint distribution of the real features while preserving the characteristics of each feature set and the relationships among them.

Fig. 3 shows an overview of the MGAN framework. It consists of three feature generators, G_t, G_p, and G_s, and one discriminator, D. The three generators forge fake samples $G_t(\boldsymbol{z}_t)$, $G_p(\boldsymbol{z}_p)$, and $G_s(\boldsymbol{z}_s)$ using random samples $\boldsymbol{z}_t$, $\boldsymbol{z}_p$, and $\boldsymbol{z}_s$. These outputs are further fed into a discriminator along with real sample data ($\boldsymbol{X}_t$, $\boldsymbol{x}_p$, and $\boldsymbol{x}_s$) during training. And the discriminator attempts to determine whether the inputs come from the real distribution or the fake one. Essentially, the three generators aim to fool the discriminator, which acts as an anti-fraud agent and provides feedback to improve the fraudulent capabilities of three generators by adjusting their weights. The objective function of the MGAN is defined as follows:

$$\min_{G_t,G_p,G_s}\max_{D} L(G_t,G_p,G_s,D) = \min_{G_t,G_p,G_s}\max_{D}\{\mathbb{E}_{c(\boldsymbol{X}_t,x_p,x_s)\sim p_{x_c}}[logD(c(\boldsymbol{X}_t,x_p,x_s))] \\ +\mathbb{E}_{c(G_t(\boldsymbol{z}_t),G_p(\boldsymbol{z}_p),G_s(\boldsymbol{z}_s))\sim p_{g_c}}[log(1-D(c(G_t(\boldsymbol{z}_t),G_p(\boldsymbol{z}_p),G_s(\boldsymbol{z}_s))))]\}, \tag{1}$$

where $c(\cdot)$ is a fusion function embedded in the discriminator, as illustrated in Fig. 3. This fusion function is defined as follows:

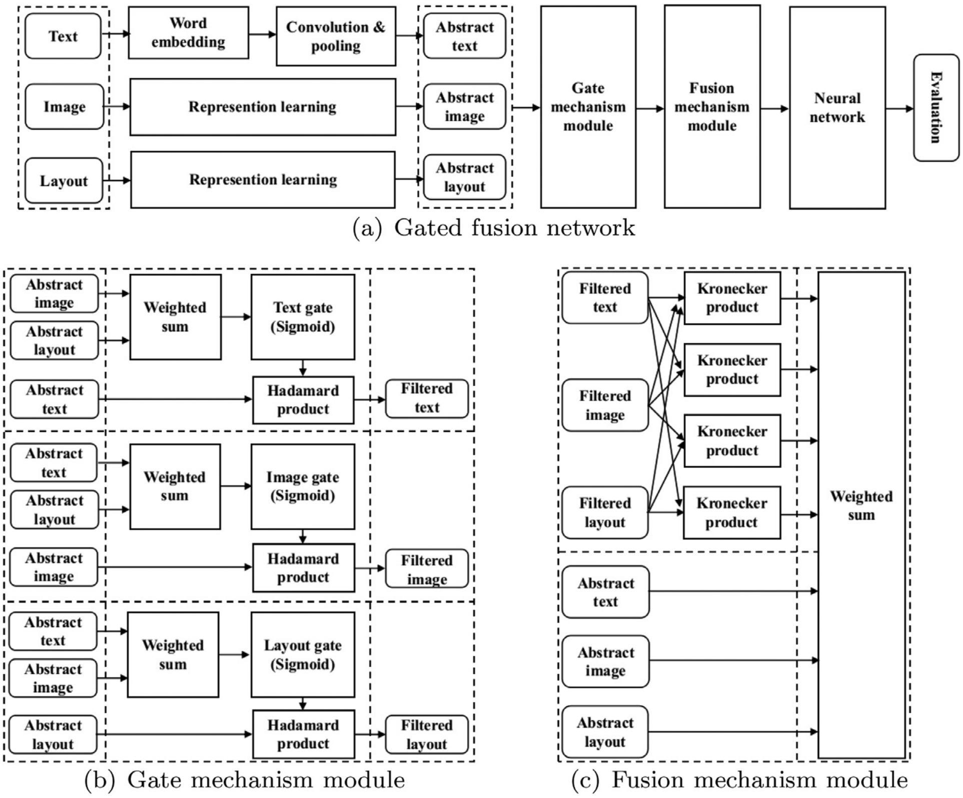
$$c(\boldsymbol{X}_t,\boldsymbol{x}_p,\boldsymbol{x}_s) = \boldsymbol{h}_t\boldsymbol{w}_t + \boldsymbol{h}_p\boldsymbol{w}_p + \boldsymbol{h}_s\boldsymbol{w}_s + \boldsymbol{i}_{tp}\boldsymbol{w}_{tp} + \boldsymbol{i}_{ts}\boldsymbol{w}_{ts} + \boldsymbol{i}_{ps}\boldsymbol{w}_{ps} + \boldsymbol{i}_{tps}\boldsymbol{w}_{tps} + b. \tag{2}$$

In Eq. (2), $\boldsymbol{h}_t = r_t(\boldsymbol{X}_t)$, $\boldsymbol{h}_p = r_p(\boldsymbol{x}_p)$, and $\boldsymbol{h}_s = r_s(\boldsymbol{x}_s)$. $\boldsymbol{h}_t$, $\boldsymbol{h}_p$, and $\boldsymbol{h}_s$ are high-level mappings of $\boldsymbol{X}_t$, $\boldsymbol{x}_p$, and $\boldsymbol{x}_s$ obtained through three subnets r_t, r_p, and r_s, respectively. Here, r and c are partial nets of the neural network D, and $\boldsymbol{w}_t$, $\boldsymbol{w}_p$, $\boldsymbol{w}_s$, $\boldsymbol{w}_{tp}$, $\boldsymbol{w}_{ts}$, $\boldsymbol{w}_{ps}$, $\boldsymbol{w}_{tps}$, and b are network parameters. $\boldsymbol{i}_{tp}$, $\boldsymbol{i}_{ts}$, $\boldsymbol{i}_{ps}$, and $\boldsymbol{i}_{tps}$ fuse the features via the Kronecker product as follows:

$$\begin{aligned} \boldsymbol{i}_{tp} &= \boldsymbol{h}_t \otimes \boldsymbol{h}_p, \\ \boldsymbol{i}_{ts} &= \boldsymbol{h}_t \otimes \boldsymbol{h}_s, \\ \boldsymbol{i}_{ps} &= \boldsymbol{h}_p \otimes \boldsymbol{h}_s, \\ \boldsymbol{i}_{tps} &= \boldsymbol{h}_t \otimes \boldsymbol{h}_p \otimes \boldsymbol{h}_s, \end{aligned} \tag{3}$$

where $\otimes$ denotes the Kronecker product, which measures all possible connections between elements in two feature vectors. Note that to avoid overfitting, a high-level feature vector $\boldsymbol{h}$ could be replaced by a filtered feature vector $\boldsymbol{f}$ via a gate mechanism, as described in Section 3.3.1.

The vector $c(\boldsymbol{X}_t, \boldsymbol{x}_p, \boldsymbol{x}_s)$ is the fused result of the real feature sets $\boldsymbol{X}_t$, $\boldsymbol{x}_p$, and $\boldsymbol{x}_s$. The symbol p_{x_c} denotes its distribution. Similarly, the vector $c(G_t(\boldsymbol{z}_t), G_p(\boldsymbol{z}_p), G_s(\boldsymbol{z}_s))$ is the fused result of the three fake feature sets, and p_{g_c} is its distribution. In the fraud and anti-fraud game between the generators and the discriminator, the goal of three generators is to confuse the discriminator (making $D(c(\boldsymbol{X}_t, \boldsymbol{x}_p, \boldsymbol{x}_s))$ close to 0) while making the discriminator believe that the generated features are real ones (making $D(c(G_t(\boldsymbol{z}_t), G_p(\boldsymbol{z}_p), G_s(\boldsymbol{z}_s)))$ close to 1), for a fixed state of the discriminator. In contrast, the purpose of the discriminator is to distinguish real features from generated ones. That is, the discriminator is trained to make $D(c(\boldsymbol{X}_t, \boldsymbol{x}_p, \boldsymbol{x}_s))$



(a) Gated fusion network

(b) Gate mechanism module

(c) Fusion mechanism module

Fig. 4. Gated fusion network & its two mechanisms.

close to 1 and make $D(c(G_t(\boldsymbol{z}_t), G_p(\boldsymbol{z}_p), G_s(\boldsymbol{z}_s)))$ close to 0, for fixed states of three generators. Solving the above objective function, we obtain $p_{g_c} = p_{x_c}$, which means that the fused distribution of the features generated by the three generators can converge to the joint distribution of the real features. Appendix A.1 presents detailed proof. The pseudocode of the proposed MGAN is shown in Algorithm 1.

Algorithm 1 MGAN training with mini-batch gradient descent.

```
inputs: $\boldsymbol{x}_t \in \mathbb{R}^N$, $\boldsymbol{x}_p \in \mathbb{R}^4$, and $\boldsymbol{x}_s \in \mathbb{R}^M$ with positive label $y_{pos}$
outputs: Three well-trained generators $G_t$, $G_p$, and $G_s$
1: while $k$ steps do
2:     Draw $n$ examples $\boldsymbol{x}_t$, $\boldsymbol{x}_p$, and $\boldsymbol{x}_s$ from the training set
3:     $\boldsymbol{X}_t \in \mathbb{R}^{N\times E} \leftarrow \boldsymbol{x}_t$ is embedded
4:     Draw $n$ noise samples $\boldsymbol{z}_t$, $\boldsymbol{z}_p$, and $\boldsymbol{z}_s$ from a real distribution
5:     $G_t(\boldsymbol{z}_t) \in \mathbb{R}^{N\times E} \leftarrow G_t$ gets $\boldsymbol{z}_t$; $G_p(\boldsymbol{z}_p) \in \mathbb{R}^4 \leftarrow G_p$ gets $\boldsymbol{z}_p$; $G_s(\boldsymbol{z}_s) \in \mathbb{R}^M \leftarrow G_s$ gets $\boldsymbol{z}_s$
6:     Assign negative label $y_{neg}$ to $G_t(\boldsymbol{z}_t)$, $G_p(\boldsymbol{z}_p)$,and $G_s(\boldsymbol{z}_s)$,and use them and real samples totrain the discriminator
7:     Repeat Step 5, assign a positive label $y_{pos}$ to $G_t(\boldsymbol{z}_t)$, $G_p(\boldsymbol{z}_p)$,and $G_s(\boldsymbol{z}_s)$, and use them to train the three generators
8: end while
```

3.3. Gated fusion network (GFN)

In this study, we propose a GFN with the gate and fusion mechanisms, as shown in Fig. 4(a).

3.3.1. Gate mechanism

In the proposed GFN, a gate mechanism is first utilized to reduce noise and strengthen the useful information for multimodal data classification. The gate mechanism was originally proposed along with the long short-term memory (LSTM) architecture to solve long-term dependency problems of recurrent neural networks (RNNs) [19]. Specifically, an input gate controls the inputs by adjusting the amount of valuable information that can access the hidden states. In contrast, an output gate controls the outputs of the hidden states to the next model stage to limit newly generated noise. In this study, we design three gates, namely, a text gate, an image gate, and a layout gate, to preserve the features of each mode that interact with features in another mode, allowing them to be utilized in the fusion mechanism. Fig. 4(b) shows the framework of the proposed gate mechanism, and the mathematical formulas are described as follows:

$$\boldsymbol{g}_t = \sigma(\boldsymbol{h}_p \boldsymbol{w}_p^{(t)} + \boldsymbol{h}_s \boldsymbol{w}_s^{(t)} + b^{(t)}), \tag{4}$$

$$\boldsymbol{g}_p = \sigma(\boldsymbol{h}_t \boldsymbol{w}_t^{(p)} + \boldsymbol{h}_s \boldsymbol{w}_s^{(p)} + b^{(p)}), \tag{5}$$

$$\boldsymbol{g}_s = \sigma(\boldsymbol{h}_t \boldsymbol{w}_t^{(s)} + \boldsymbol{h}_p \boldsymbol{w}_p^{(s)} + b^{(s)}), \tag{6}$$

where $\boldsymbol{h}_t$, $\boldsymbol{h}_p$, and $\boldsymbol{h}_s$ are the preprocessed text, image, and layout feature vectors, respectively. Eqs. (4) to (5) describe the text gate $\boldsymbol{g}_t$, the image gate $\boldsymbol{g}_p$, and the layout gate $\boldsymbol{g}_s$, respectively. σ denotes the sigmoid activation function. The notations $\boldsymbol{w}_p^{(t)}$, $\boldsymbol{w}_s^{(t)}$, $\boldsymbol{w}_t^{(p)}$, $\boldsymbol{w}_s^{(p)}$, $\boldsymbol{w}_t^{(s)}$, and $\boldsymbol{w}_p^{(s)}$, represent weight vectors, and $b^{(t)}$, $b^{(p)}$, and $b^{(s)}$ are the bias terms. Note that there are two approaches for controlling the input information of a target feature mode. One is to control the input information for a target mode (e.g., layout) in terms of all three modes (i.e., text, layout, and image), and the other is to control the input information of a target mode in terms of the different two modes. In our preliminary experiments, we compared both approaches and found that the latter achieved a similar result to the former with fewer computations. Therefore, in this study, to reduce the computational complexity and improve the generalization ability of the model, we control the input information in terms of the other two modes, presented as Eqs. (4) to (5).

Thus, the filtered feature vectors for the fusion mechanism can be obtained as follows:

$$\boldsymbol{f}_t = f(\boldsymbol{g}_t \odot \boldsymbol{h}_t), \tag{7}$$

$$\boldsymbol{f}_p = f(\boldsymbol{g}_p \odot \boldsymbol{h}_p), \tag{8}$$

$$\boldsymbol{f}_s = f(\boldsymbol{g}_s \odot \boldsymbol{h}_s), \tag{9}$$

where $\odot$ denotes the Hadamard product, and $f(\cdot)$ means the activation function.

In Eqs. (4) to (5), $\sigma(\cdot)$ is the sigmoid function. Therefore, the value range of the elements of the vectors $\boldsymbol{g}_t$, $\boldsymbol{g}_p$, and $\boldsymbol{g}_s$ is between 0 and 1. When the ith output value of the sigmoid function approaches 0, the ith gate turns off. Subsequently, the ith element in the feature vector is discarded when multiplied by this gate. Thus, the gate acts to prevent irrelevant information from entering the next stage of the model. Similarly, the gate preserves the corresponding valuable information when the ith output value of the sigmoid function approaches 1. This gate mechanism controls the data flow to obtain the filtered features via parameter learning in the network. To support the fusion mechanism, it reduces the amount of unrelated information and amplifies the influence of the connections among different feature modes.

3.3.2. Fusion mechanism

Essentially, the recognition of faculty homepages is a multimodal problem involving three mutually dependent data modes, namely, text, images, and layout. For example, the textual information contained in different layout tags has different levels of importance. Dismantling these relations results in a loss of valuable information and reduces the classification accuracy. However, previous studies have ignored or destroyed the relations among different feature sets (modes), although these relations are critical for multimodal supervised learning. In the proposed GFN, a fusion mechanism is designed to preserve the connections among the different modes. Fig. 4(c) illustrates the strategy of the fusion mechanism. Intuitively, mutually dependent information is determined by the joint effect of the relevant sources. For example, the impact of an independent variable on a dependent variable can be measured in terms of the magnitudes of other associated independent variables [1]. Therefore, this mutual dependence can be measured by a product if the information sources are scalars or by a Kronecker product if the information sources are vectors [2,40]. Rendle and Steffen adopted the product approach to capture the mutual dependence among features in factorization machines [35].

The feature space that considers the connections among the different information modes can be formally expressed as follows:

$$\boldsymbol{o} = \boldsymbol{h}_t \boldsymbol{w}_t + \boldsymbol{h}_p \boldsymbol{w}_p + \boldsymbol{h}_s \boldsymbol{w}_s + (\boldsymbol{f}_t \otimes \boldsymbol{f}_p)\boldsymbol{w}_{tp} + (\boldsymbol{f}_t \otimes \boldsymbol{f}_s)\boldsymbol{w}_{ts} + (\boldsymbol{f}_p \otimes \boldsymbol{f}_s)\boldsymbol{w}_{ps} + (\boldsymbol{f}_t \otimes \boldsymbol{f}_p \otimes \boldsymbol{f}_s)\boldsymbol{w}_{tps} + b, \tag{10}$$

where $\otimes$ denotes the Kronecker product; $\boldsymbol{h}_t$, $\boldsymbol{h}_p$, and $\boldsymbol{h}_s$ denote the text, image, and layout feature vectors, respectively; and similarly, $\boldsymbol{f}_t$, $\boldsymbol{f}_p$, and $\boldsymbol{f}_s$ mean the filtered text, image, and layout feature vectors, respectively. The reason to apply filtered features instead of the original features is to significantly reduce information considered for fusion processing by removing noise and irrelevant information using the gate mechanism. The detailed gate mechanism is described in Section 3.3.1. $\boldsymbol{f}_t \otimes \boldsymbol{f}_p$, $\boldsymbol{f}_t \otimes \boldsymbol{f}_s$, and $\boldsymbol{f}_p \otimes \boldsymbol{f}_s$ measure the mutual dependence of the text and image features, the text and layout features, and the image and layout features, respectively. $\boldsymbol{f}_t \otimes \boldsymbol{f}_p \otimes \boldsymbol{f}_s$ measures the mutual dependence among the text, image, and layout features. $\boldsymbol{w}_t$, $\boldsymbol{w}_p$, $\boldsymbol{w}_s$, $\boldsymbol{w}_{tp}$, $\boldsymbol{w}_{ts}$, $\boldsymbol{w}_{ps}$, and $\boldsymbol{w}_{tps}$ are the weight vectors, and b is the bias.

3.3.3. Learning

After the features have been preprocessed with the gate and fusion mechanisms, the task becomes a classic binary classification problem. The objective function of the network is defined as follows:

$$L = \frac{1}{n}\sum_{\boldsymbol{x}_t,\boldsymbol{x}_p,\boldsymbol{x}_s}[y \times log(GFN(\boldsymbol{x}_t,\boldsymbol{x}_p,\boldsymbol{x}_s)) + (1-y) \times log(1-GFN(\boldsymbol{x}_t,\boldsymbol{x}_p,\boldsymbol{x}_s))], \tag{11}$$

where $GFN(\boldsymbol{x}_t,\ \boldsymbol{x}_p,\ \boldsymbol{x}_s)$ denotes the output of the network concerning $\boldsymbol{x}_t$, $\boldsymbol{x}_p$, and $\boldsymbol{x}_s$ and, n is the number of samples. To minimize the objective function, we apply the adaptive moment estimation (Adam) [25]. To implement Adam, the derivative of the prediction is defined as follows:

$$\nabla_{\boldsymbol{w},b}\frac{1}{n}\sum[y \times log(GFN(\boldsymbol{x}_t,\boldsymbol{x}_p,\boldsymbol{x}_s)) + (1-y) \times log(1-GFN(\boldsymbol{x}_t,\boldsymbol{x}_p,\boldsymbol{x}_s))]. \tag{12}$$

The pseudocode for the proposed gated fusion network is shown in Algorithm 2.

Algorithm 2 GFN training with mini-batch gradient descent.

```
inputs: Feature sets $\boldsymbol{x}_t \in \mathbb{R}^N$, $\boldsymbol{x}_p \in \mathbb{R}^4$, and$\boldsymbol{x}_s \in \mathbb{R}^M$ and label $y$
outputs: The well-trained GFN
while $k$ steps do
    Draw $n$ examples $\boldsymbol{x}_t$, $\boldsymbol{x}_p$, and $\boldsymbol{x}_s$ from the training set
    $\boldsymbol{X}_t \in \mathbb{R}^{N \times E} \leftarrow \boldsymbol{x}_t$ is embedded
    Map $\boldsymbol{X}_t$ to $\boldsymbol{h}_t$, $\boldsymbol{x}_p$ to $\boldsymbol{h}_p$, and $\boldsymbol{x}_s$ to $\boldsymbol{h}_s$ through a series of operations
    Calculate the three gates $\boldsymbol{g}_t$, $\boldsymbol{g}_p$, and $\boldsymbol{g}_s$ in accordance with Eqs. (4) to (5)
    Calculate the three filtered feature vectors $\boldsymbol{f}_t$, $\boldsymbol{f}_p$, and $\boldsymbol{f}_s$ in accordance with Eqs. (7) to (8)
    Fuse the three filtered feature vectors in accordance with Eq. (10)
    Feed the fused features to the next stage of the network
end while
```

4. Experimental evaluation

This section presents a series of experiments conducted to gauge the effectiveness of the proposed multimodal generative and fusion framework. In particular, the evaluation targets this framework's internal functions for processing the mutually dependent multimodal data and imbalanced data.

4.1. Measures

The standard *accuracy* metric is applied to evaluate model performance [31]:

$$accuracy = \frac{TP + TN}{TP + FN + FP + TN}, \tag{13}$$

where *TP* denotes the number of true positives, representing the faculty homepages that are successfully identified; *TN* denotes the number of true negatives, representing the non-faculty webpages that are successfully identified; *FP* denotes the number of false positives, representing non-faculty webpages that are misclassified as faculty homepages; and *FN* denotes the number of false negatives, representing the faculty homepages that are misclassified as non-faculty webpages.

However, due to the class imbalance problem in the multimodal dataset, this *accuracy* metric alone is unable to provide a comprehensive evaluation of model performance. Consequently, we adopt additional assessment metrics, such as *precision*, *recall*, and *F1*, to evaluate model performance [17]:

$$\begin{aligned} precision &= \frac{TP}{TP + FP}, \\ recall &= \frac{TP}{TP + FN}, \\ F1 &= \frac{2 \times recall \times precision}{recall + precision}. \end{aligned} \tag{14}$$

Specifically, the *precision* is defined as the proportion of true positive faculty homepages among the total number of predicted faculty homepages, while the *recall* measures how many faculty homepages are correctly identified among the total true faculty homepages, and the *F1* quantifies the tradeoff between the *precision* and *recall*.

The experimental platform is a Linux server with 80 CPU cores (Intel(R) Xeon(R) CPU E5-2698 v4 @ 2.20GHz), 500 GB of RAM, and 4 GPUs (NVIDIA Tesla M10).

4.2. Experimental data

In this study, we implemented a distributed web crawler to collect approximately 39,715 webpages from the websites of several universities in the United States. The dataset was then further divided into four tracks to evaluate the system performance for various levels of data imbalance. This dataset can be accessed online along with our source code[4]:

- FW_1: Includes 7943 faculty homepages among a total of 15,886 webpages. The ratio of faculty homepages to non-faculty webpages is 1: 1.
- FW_2: Includes 7943 faculty homepages among a total of 23,829 webpages. The ratio of faculty homepages to non-faculty webpages is 1: 2.
- FW_3: Includes 7943 faculty homepages among a total of 31,645 webpages. The ratio of faculty homepages to non-faculty webpages is 1: 3.
- FW_4: Includes 7943 faculty homepages among a total of 39,715 webpages. The ratio of faculty homepages to non-faculty webpages is 1: 4.

4.3. Parameters

4.3.1. Our framework

We implemented three tricks to alleviate the challenges of non-convergence, mode collapse, and slow training when training the MGAN. In particular,

- we added batch normalization layers only into the discriminator [20] to accelerate and stabilize the training process;
- we chose Adam optimizer as the top-priority solver to accelerate the training process [25];
- we added random noise to both the real and fake samples [36] to alleviate mode collapse.

We conducted a series of preliminary experiments to find the optimal settings for the discriminator and the three generators. The final chosen settings are as follows. The discriminator has a convolutional layer with 250 1D filters (convolutional kernels) with a size of 3 and four fully-connected layers with 250 units. The activation function is the sigmoid function. Among the generators, the text generator consists of five convolutional layers with 300 1D filters, whose size is set to 3. The image and layout generators, each is comprised of five fully-connected layers with different numbers of units; the first has 100 units, and the second has 500 units. The ReLU activation function is used in the generators.

To address the problem of multimodal data with mutually dependent feature sets, our proposed GFN consists of an embedding layer, a convolutional layer, and six fully-connected layers. The activation functions for the gate and fusion mechanisms are the sigmoid and the hyperbolic tangent, respectively. Here, we adopted Adam optimizer to train the GFN. More detailed information can be found by referring to our source code.

4.3.2. Other models

To gauge the overall performance of the proposed framework and its internal functions, we compare this framework with several classic algorithms and frameworks, namely, the support vector machine (SVM) algorithm, the multilayer perceptron (MLP) algorithm, the decision tree (DT) algorithm, the standard convolutional neural network (CNN) algorithm, the unimodal GAN framework, and the GAN-based CNN framework.

- SVM: Includes a radial basis kernel function with $\gamma = 0.01$, a shrinking heuristic, a stopping criterion tolerance of $1e - 3$, and a penalty parameter of $c = 1$.
- MLP: Includes two hidden layers of 2000 units, each with sigmoid activation functions. The maximum number of iterations is 200, and the $L2$ penalty parameter is $1e - 3$. The Adam optimizer, with an initial learning rate of $lr = 0.01$, is chosen as the optimal solver. This optimizer has a first exponential decay rate of $\beta_1 = 0.9$, a second exponential decay rate of $\beta_2 = 0.999$, and a stability of $\epsilon = 1e - 8$.
- DT: Includes the *Gini* splitting criterion, the *best* strategy for choosing the split at each node, a minimum sample number of 2 for splitting an internal node, and a minimum sample number of 1 at a leaf node.
- CNN: Has the same settings as the GFN except for the gate and fusion mechanisms.
- Unimodal GAN: Consists of three groups of GANs, namely, a text GAN, an image GAN, and a layout GAN. The generators in the unimodal GAN have the same settings as the generators in the MGAN. In contrast, the text discriminator in the unimodal GAN framework has a convolutional layer with 300 1D filters with a size of 3 and one fully connected layer with 500 units. The layout discriminator has two fully-connected layers with 500 units each. And the image discriminator has two fully-connected layers with 100 units each.
- GAN-based CNN: Combines the unimodal GAN and CNN models.

[4] https://github.com/mrspider520/gated_fusion_network.git.

Table 1
Comprehensive comparison among different models on the different datasets.

Data	Method	*Precision*	*Recall*	*F*1	*Accuracy*
FW_1	MLP	0.7952	0.9280	0.8565	0.8431
	DT	0.8366	0.8846	0.8599	0.8559
	SVM	0.7059	0.9106	0.7575	0.7621
	GAN-based CNN	0.8594	0.9004	0.8794	0.8747
	Our approach	**0.8530**	**0.9113**	**0.8812**	**0.8765**
FW_2	MLP	0.7209	0.8530	0.7814	0.8349
	DT	0.6312	0.6301	0.6306	0.7654
	SVM	0.5045	0.8961	0.6456	0.7774
	GAN-based CNN	0.8300	0.8058	0.8177	0.8778
	Our approach	**0.9310**	**0.8702**	**0.8946**	**0.8830**
FW_3	MLP	0.6578	0.8927	0.7575	0.7964
	DT	0.7208	0.9265	0.8108	0.8559
	SVM	0.4341	0.9120	0.5882	0.7465
	GAN-based CNN	0.8227	0.7755	0.7984	0.9013
	Our approach	**0.9255**	**0.8617**	**0.8925**	**0.9063**
FW_4	MLP	0.5979	0.8559	0.7040	0.8050
	DT	0.5647	0.8892	0.6907	0.8398
	SVM	0.3551	0.8991	0.5091	0.7930
	GAN-based CNN	0.8045	0.7811	0.7926	0.8960
	Our approach	**0.8055**	**0.8859**	**0.8438**	**0.9173**

4.4. Model comparison

To gauge the overall performance of the proposed framework, we compare this framework with several classical models, including the SVM, MLP, DT, and GAN-based CNN models aforementioned, on four experimental datasets, namely, FW_1, FW_2, FW_3, and FW_4. Table 1 shows the results for comparison in terms of the four assessment metrics.

It can be observed that the proposed approach outperforms the other methods on all four datasets. As the amount of training data increases, the proposed approach becomes more accurate, which indicates the scalability of the proposed approach. As the level of imbalance increases, the MLP, DT, and SVM models become more vulnerable to imbalance effects, while the GAN-based CNN and the proposed approach show robust performance despite the imbalance. Moreover, the proposed framework performs better than GAN-based CNN. Overall, the MGAN generates better samples for dataset rebalancing than the unimodal GAN does.

4.5. Internal functions

In this study, we propose a multimodal generative and fusion framework with two unique modules to address the challenges that arise in learning from imbalanced multimodal data. Notably, to our knowledge, this paper is the first time that an MGAN is introduced to rebalance a dataset by generating pseudo features for each mode and then combining them to describe a fake sample. Then, a GFN with the gate and fusion mechanisms is presented to reduce noise to improve the generalization ability of the model and capture the links among the different feature modes. To gauge the effectiveness of the MGAN and GFN modules, in the rest of this section, we first examine the performance of the MGAN when faced with the class imbalance problem, and then validate the ability of the GFN to capture the relations among different feature modes. Finally, we explore the robustness of the GFN against different data sizes.

4.5.1. MGAN for the class imbalance problem

We first carried out the experiments using the proposed multimodal generative and fusion framework with its MGAN function disabled. Table 2(a) shows the results obtained without addressing the imbalance problem in terms of the four measurement metrics. In fact, we carried out preliminary experiments on each of the four datasets aforementioned to evaluate the performance of the MGAN in different class imbalance situations. Similar results show that MGAN is an efficient mechanism for improving classification performance with different levels of class imbalance. However, due to the page limitations, we only reported the experimental results obtained on dataset FW_4. The *precision* of identifying the non-faculty homepages with 0.9375 is much higher than that of recognizing the faculty homepages with 0.8055.

Previous studies on the application of GANs to imbalanced data have focused on unimodal data rather than multimodal data. Table 2(b) shows the results when using the GAN model, instead of the proposed MGAN, in the proposed framework to generate fake features. The *recall* of identifying faculty homepages is improved from 0.7453 (in the case of no data augmentation) to 0.8387, while the corresponding *precision* declines from 0.8055 to 0.7652. At the same time, the *recall* of the opposite class is reduced by a large margin.

To overcome the class imbalance problem for the case of multimodal data, we enabled the MGAN function in the proposed framework to generate fake samples to augment the minority class (faculty homepages). The MGAN module consists

Table 2
Classification results on different datasets.

(a) Imbalanced dataset

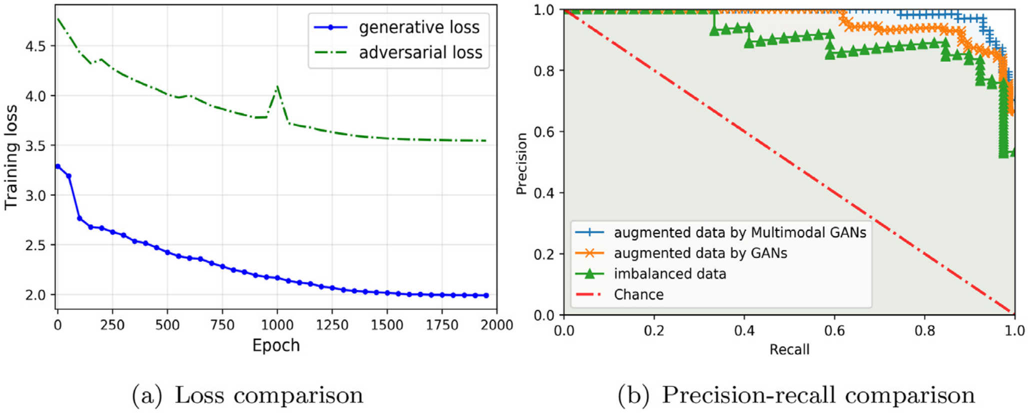
Class	*Precision*	*Recall*	*F1*
Non-faculty	0.9375	0.9550	0.9462
Faculty	**0.8055**	**0.7453**	**0.7742**
Average	0.9111	0.9131	0.9118

(b) GAN-augmented dataset

Class	*Precision*	*Recall*	*F1*
Non-faculty	0.9587	0.9357	0.9470
Faculty	**0.7652**	**0.8387**	**0.8003**
Average	0.9200	0.9163	0.9177

(c) MGAN-augmented dataset

Class	*Precision*	*Recall*	*F1*
Non-faculty	0.9732	0.9310	0.9517
Faculty	**0.8186**	**0.9238**	**0.8680**
Average	0.9232	0.9189	0.9205



(a) Loss comparison

(b) Precision-recall comparison

Fig. 5. Loss & precision-recall comparison.

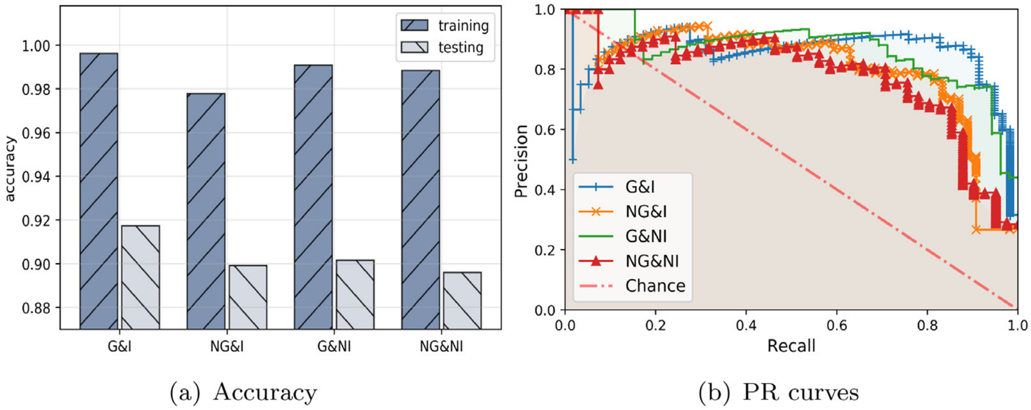
of three generators and one discriminator. The generators create features for each feature mode through iterative interaction with the discriminator, thus causing the fused distribution of the generated data to approach the real distribution over the iterations gradually. Fig. 5(a) shows the adversarial loss (of the discriminator) and the generative loss (of the generators) during each iteration of the network learning process. Both the adversarial and the generative losses initially sharply decrease and then gradually converge to lower values. The generative loss declines very smoothly throughout the entire learning process, while the adversarial loss fluctuates over a relatively broad range at the beginning of the learning process. Both losses converge when the number of iterations exceeds 1750. That is, the fake fused distribution can effectively imitate the real distribution after the fraud and anti-fraud competition between the generators and the discriminator. By applying the MGAN, the imbalanced dataset is converted into an augmented dataset with balanced classes.

Table 2(c) shows the results when using MGAN in the proposed framework to address the class imbalance problem. Compared with Table 2(b), the *precision*, *recall*, and *F*1 for identifying the faculty homepages are further improved from 0.7652 to 0.8186, from 0.8387 to 0.9238, and from 0.8003 to 0.8680, respectively.

Fig. 5(b) presents the precision-recall (PR) curves of the above three methods. Here, a curve that is closer to the upper right corner represents a model with better performance. The results illustrate that the MGAN outperforms the other two approaches. A good explanation of the superior performance of the MGAN over the classical GAN is that its ability to generate the fake features for each feature mode allows it to preserve both the characteristics of each feature set and the relationships among the features.

4.5.2. Gate and fusion mechanisms in the GFN

The task of recognizing faculty homepages is essentially a multimodal classification problem, in which a target faculty homepage is identified based on three different features, including text, images, and layout. To solve this problem, we propose a GFN with the gate and fusion mechanisms. Specifically, the gate mechanism is introduced to reduce the irrelative information to support the fusion mechanism; the fusion mechanism is applied to capture the links among different fea-



(a) Accuracy

(b) PR curves

Fig. 6. Learning curves of the GFN in the four experimental tracks.

ture modes. This section reports a series of experiments carried out to evaluate the effectiveness of these two proposed mechanisms. Specifically, four tracks of experimental evaluations were conducted, as follows:

- *G&I*: The proposed GFN with both gate and fusion mechanisms enabled.
- *G&NI*: The proposed GFN with the gate mechanism enabled and the fusion mechanism disabled.
- *NG&I*: The proposed GFN with the fusion mechanism enabled and the gate mechanism disabled.
- *NG&NI*: The proposed GFN with both the gate and fusion mechanisms disabled.

Fig. 6(a) shows the performance achieved in these four tracks on training and test datasets. We first explore the effectiveness of the gate mechanism. As seen in Fig. 6(a), *G&NI* outperforms *NG&NI*. It achieves a more significant enhancement on the test data than on the training data, while *G&I* outperforms *NG&I* on both training and test datasets by virtue of the gate mechanism. The gate mechanism alleviates overfitting by discarding irrelevant information and improves the generalization ability and *accuracy* of the model. Then, we examine the efficiency of the fusion mechanism. Compared with *NG&NI*, *NG&I* shows only a small improvement on the test data while showing a reduced performance on the training set. There is a significant increase in the *accuracy* of both training and test sets in terms of *G&I*, compared to *G&NI*. A good explanation is that while the fusion mechanism captures the connections among different feature modes, it also introduces noise that deteriorates the performance if no efficient gate mechanism is applied to filter out this noise. Fig. 6(b) shows the PR curves for each of the examined tracks. The fact that *G&I* shows the best performance further supports the above explanation.

In brief, the fusion mechanism captures the relations among different information modes but brings in noise, while the gate mechanism serves to filter out the noise and trivial information. The combination of both methods reaches a tradeoff that improves the performance for multimodal learning with mutually dependent feature modes.

4.5.3. Comparison of the GFN with other models

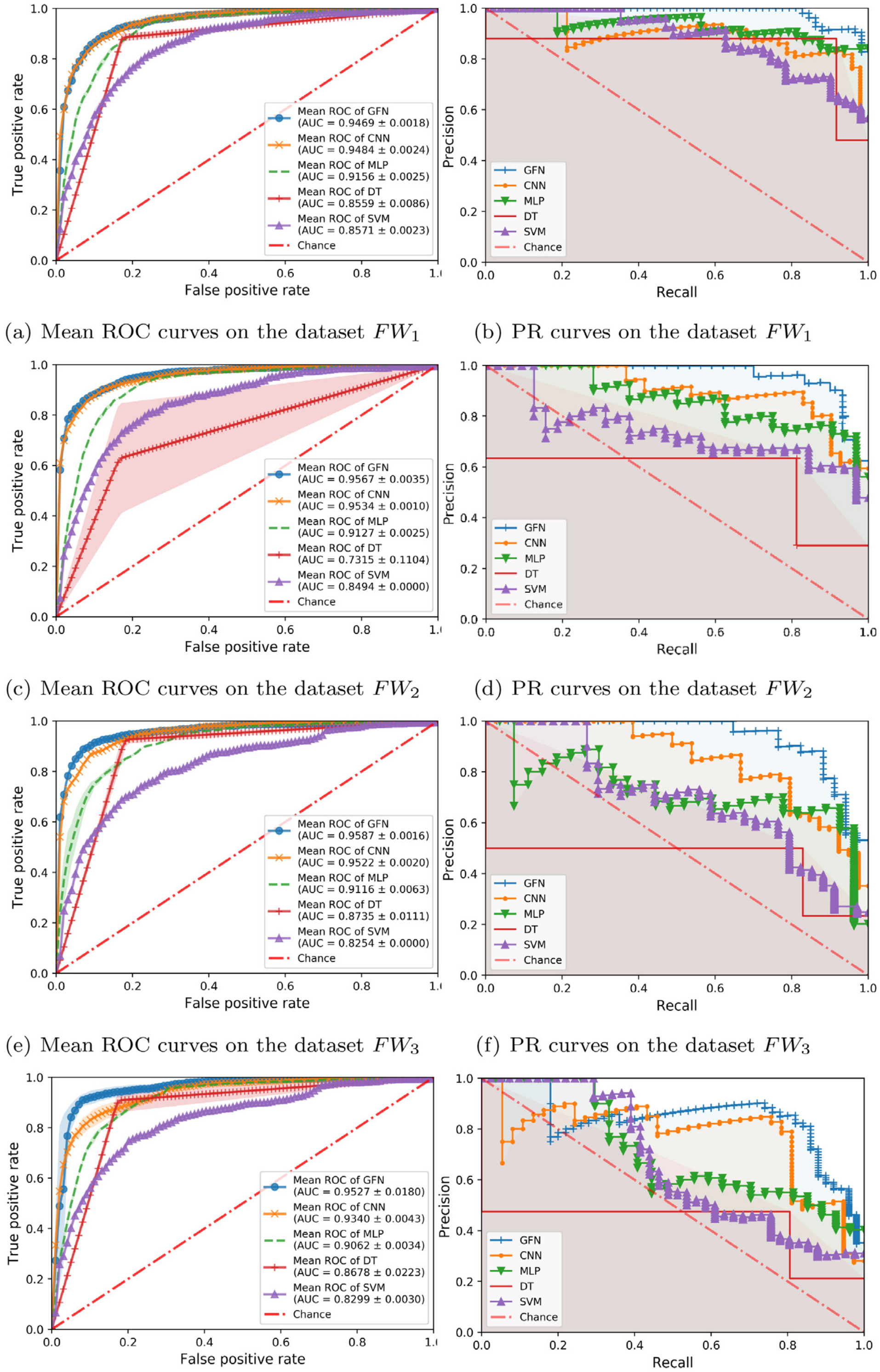
To gauge the overall performance of the proposed GFN, we compare it with several classic algorithms (SVM, MLP, DT, and CNN). In addition, to make the performance comparison more convincing, we compare the GFN with the other models on all four datasets $FW_1 \sim FW_4$ described in Section 4.2.

Table 3 shows the detailed experimental results in terms of the four assessment metrics. Fig. 7(a) ~ (d) show the receiver operating characteristic (ROC) curves, along with the mean and standard deviation of the area under the ROC curve (AUC), and PR curves for these approaches on the datasets FW_1 and FW_2, respectively. The proposed framework achieves the best performance, generally followed (in approximate order of decreasing performance) by the CNN, DT, MLP, and SVM models, as the class imbalance becomes more severe. However, our model does not have an overwhelming advantage over the CNN model when evaluated on the two smaller datasets.

Our model exhibits more obvious advantages when evaluated on FW_3 and FW_4, as shown in Table 3. The values of the four main evaluation metrics further increase. Fig. 7(e) ~ (h) also demonstrate that the GFN achieves a significant performance improvement. Fig. 7(e) and (g) show that GFN has the largest true positive rate, and the lowest false positive rate, compared with all four baselines, which indicates that the proposed approach ensures both the highest probability of identifying faculty homepages and the lowest likelihood of misidentifying non-faculty pages as faculty homepages.

5. Conclusions and future work

The task of recognizing faculty homepages is a typical binary classification problem with multimodal features, namely, text, images, and HTML layout, all of which are related in a complicated fashion. In this study, a fusion mechanism is introduced to capture the intrinsic relations among these multimodal features. This mechanism is achieved by applying the Kronecker product to preserve these relationships among the different feature modes, and then iteratively optimizing them.



(a) Mean ROC curves on the dataset FW_1 (b) PR curves on the dataset FW_1

(c) Mean ROC curves on the dataset FW_2 (d) PR curves on the dataset FW_2

(e) Mean ROC curves on the dataset FW_3 (f) PR curves on the dataset FW_3

(g) Mean ROC curves on the dataset FW_4 (h) PR curves on the dataset FW_4

Fig. 7. ROC (AUC = mean ± standard deviation) and PR curves of different models on the four datasets.

Table 3
Comparison among different models on four datasets.

Data	Method	*Precision*	*Recall*	*F*1	*Accuracy*
FW_1	MLP	0.7952	0.9280	0.8565	0.8431
	DT	0.8366	0.8846	0.8599	0.8559
	SVM	0.7059	0.9106	0.7575	0.7621
	CNN	0.8594	0.9004	0.8794	0.8747
	GFN	**0.8530**	**0.9113**	**0.8812**	**0.8765**
FW_2	MLP	0.7209	0.8530	0.7814	0.8349
	DT	0.6312	0.6301	0.6306	0.7654
	SVM	0.5045	0.8961	0.6456	0.7774
	CNN	0.8227	0.7755	0.7984	0.8966
	GFN	**0.9317**	**0.8091**	**0.8661**	**0.9055**
FW_3	MLP	0.6578	0.8927	0.7575	0.7964
	DT	0.7208	0.9265	0.8108	0.8559
	SVM	0.4341	0.9120	0.5882	0.7465
	CNN	0.7286	0.8764	0.7957	0.8866
	GFN	**0.8068**	**0.8922**	**0.8473**	**0.9110**
FW_4	MLP	0.5979	0.8559	0.7040	0.8050
	DT	0.5647	0.8892	0.6907	0.8398
	SVM	0.3551	0.8991	0.5091	0.7930
	CNN	0.7145	0.8417	0.7729	0.9005
	GFN	**0.7937**	**0.8140**	**0.8037**	**0.9105**

To overcome noise and model complexity, mainly brought in by the fusion mechanism, a gate mechanism is proposed to filter out the noise and trivial information. Recognizing faculty webpages is also a class imbalance problem, in which the total sample number of the minority class (faculty homepages) is far smaller than the total number of the majority class (non-faculty webpages). In this study, we introduce a multimodal generative adversarial network (MGAN) to rebalance the dataset. The MGAN generates fake features for each feature mode during iterative adversarial training. As a result of this procedure, the fused distribution of the counterfeit features approaches the joint distribution of the real features, while the characteristics of each feature set and the relationships among the features are preserved.

The proposed multimodal generative and fusion framework can be generalized to many other multimodal learning problems with class-imbalanced data and mutually dependent feature modes. One good example is the prediction of the media-aware stock movements, in which the market information space consists of several modes, including transaction data, news articles, and investors' moods in bear markets [27–29]. However, the effectiveness of the multimodal generative and fusion framework is yet to be explored in other related fields. We plan to perform such explorations in the near feature.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (NSFC) (71671141 and 71873108), the Fundamental Research Funds for the Central Universities (JBK 171113, JBK 170505, and JBK 1806003), the Sichuan Province Science and Technology Department (2019YJ0250), the Key Laboratory of Internet Natural Language Processing of the Sichuan Provincial Education Department, the Financial Innovation Center of Southwestern University of Finance and Economics, and the Financial Intelligence and Financial Engineering Key Laboratory of Sichuan Province.

Appendix A

A1. MGAN

For simplicity, let

$$\begin{aligned} \boldsymbol{x}_c &= c(\boldsymbol{X}_t, \boldsymbol{x}_p, \boldsymbol{x}_s), \\ \boldsymbol{g}_c &= c(G_t(\boldsymbol{z}_t), G_p(\boldsymbol{z}_p), G_s(\boldsymbol{z}_s)). \end{aligned} \tag{15}$$

Given three generators, the following function is maximized to find the optimal discriminator D:

$$\begin{aligned} L(G_t, G_p, G_s, D) &= \int_{\boldsymbol{x}_c} [p_{x_c log(D(\boldsymbol{x}_c))}] d\boldsymbol{x}_c + \int_{\boldsymbol{g}_c} [p_{g_c} log(1 - D(\boldsymbol{g}_c))] d\boldsymbol{g}_c \\ &= \int_{\boldsymbol{x}_c} [p_{x_c} log(D(\boldsymbol{x}_c)) + p_{g_c} log(1 - D(\boldsymbol{x}_c))] d\boldsymbol{x}_c. \end{aligned} \tag{16}$$

Therefore,

$$\frac{\partial L}{\partial \boldsymbol{x}_c} = \frac{p_{x_c}}{D(\boldsymbol{x}_c)}\frac{\partial D(\boldsymbol{x}_c)}{\partial \boldsymbol{x}_c} - \frac{p_{g_c}}{1 - D(\boldsymbol{x}_c)}\frac{\partial D(\boldsymbol{x}_c)}{\partial \boldsymbol{x}_c}$$

$$\frac{\partial L}{\partial D(\boldsymbol{x}_c)} = \frac{\frac{\partial L}{\partial \boldsymbol{x}_c}}{\frac{\partial D(\boldsymbol{x}_c)}{\partial \boldsymbol{x}_c}} = \frac{p_{x_c}}{D(\boldsymbol{x}_c)} - \frac{p_{g_c}}{1 - D(\boldsymbol{x}_c)} = 0. \tag{17}$$

Consequently, the optimal discriminator is

$$D^*(\boldsymbol{x}_c) = \frac{p_{x_c}}{p_{x_c} + p_{g_c}}. \tag{18}$$

Given three generators G_t, G_p, and G_s, the maximum of the objective function given in Eq. (1) with the optimal discriminator D^* is

$$\begin{aligned}\max_D L(G_t, G_p, G_s, D) &= L(G_t, G_p, G_s, D^*)\\ &= \mathbb{E}_{\boldsymbol{x}_c\sim p_{x_c}}[logD^*(\boldsymbol{x}_c)] + \mathbb{E}_{\boldsymbol{x}_c\sim p_{g_c}}[(1 - logD^*(\boldsymbol{g}_c))]\\ &= \mathbb{E}_{\boldsymbol{x}_c\sim p_{x_c}} log\frac{p_{x_c}}{p_{x_c} + p_{g_c}} + \mathbb{E}_{\boldsymbol{x}_c\sim p_{g_c}} log\frac{p_{g_c}}{p_{x_c} + p_{g_c}}.\end{aligned} \tag{19}$$

Then, $L(G_t, G_p, G_s, D^*)$ is minimized to obtain the optimal generators (G_t, G_p, and G_s):

$$\begin{aligned}&\min_{p_{g_c}} L = \min_{p_{g_c}}\left[\mathbb{E}_{\boldsymbol{x}_c\sim p_{x_c}} log\frac{p_{x_c}}{p_{x_c} + p_{g_c}} + \mathbb{E}_{\boldsymbol{x}_c\sim p_{g_c}} log\frac{p_{g_c}}{p_{x_c} + p_{g_c}}\right],\\ &\frac{\partial L}{\partial p_{g_c}} = \frac{1}{p_{g_c}} - \frac{2}{p_{x_c} + p_{g_c}} = 0,\\ &p_{g_c} = p_{x_c}.\end{aligned} \tag{20}$$

Therefore, the minimum value of $L(G_t, G_p, G_s, D^*)$ is $-\log 4$, and we obtain $p_{g_c} = p_{x_c}$.

$$\begin{aligned}JS(p_{x_c}||p_{g_c}) &= \frac{1}{2}[KL(p_{x_c}||p_{g_c}) + KL(p_{g_c}||p_{x_c})]\\ &= \frac{1}{2}\left[p_{x_c} log\frac{p_{x_c}}{p_{g_c}} + p_{g_c} log\frac{p_{g_c}}{p_{x_c}}\right] = 0,\end{aligned} \tag{21}$$

where KL denotes the Kullback-Leibler divergence, and JS denotes the Jensen-Shannon divergence, both of which measure the similarity between two distributions. A small value indicates only slight differences between the two distributions. A result of $JS = 0$ means that the generators G_t, G_p, and G_s are able to learn the real feature distributions. When D^* is given, $U = \mathbb{E}_{\boldsymbol{x}_c\sim p_{g_c}}[logD^*(\boldsymbol{x}_c)] + \mathbb{E}_{\boldsymbol{x}_c\sim p_{g_c}}[1 - logD^*(\boldsymbol{g}_c)]$ is a convex function with a unique global optimal solution $p_{g_c} = p_{x_c}$. In other words, during iterative network training via the gradient descent, p_{g_c} gradually converges to p_{x_c}.

In short, the generators adjust their network parameters through repeated interactions with the discriminator, such that the fused distribution of the generated data gradually approaches the real fused distribution.

References

[1] C. Ai, E.C. Norton, Interaction terms in logit and probit models, Econ. Lett. 80 (1) (2003) 123–129.

[2] A. Basu, Elementary Statistical Theory in Sociology, vol. 12, Brill Archive, 1976.

[3] E. Baykan, M. Henzinger, L. Marian, I. Weber, Purely URL-based topic classification, in: Proceedings of the 18th International Conference on World Wide Web (WWW), ACM, 2009, pp. 1109–1110.

[4] P. Calado, M. Cristo, E. Moura, N. Ziviani, B. Ribeiro-Neto, M.A. Gonçalves, Combining link-based and content-based methods for web document classification, in: Proceedings of the 12th International Conference on Information and Knowledge Management (CIKM), ACM, 2003, pp. 394–401.

[5] N.V. Chawla, K.W. Bowyer, L.O. Hall, W.P. Kegelmeyer, SMOTE: synthetic minority over-sampling technique, J. Artif. Intell. Res. (JAIR) 16 (2002) 321–357.

[6] R.C. Chen, C.H. Hsieh, Web page classification based on a support vector machine using a weighted vote schema, Expert Syst. Appl. (ESWA) 31 (2) (2006) 427–435.

[7] A. Dal Pozzolo, O. Caelen, R.A. Johnson, G. Bontempi, Calibrating probability with undersampling for unbalanced classification, in: Computational Intelligence, 2015 IEEE Symposium Series, IEEE, 2015, pp. 159–166.

[8] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: Computer Vision and Pattern Recognition (CVPR), 2005. IEEE Computer Society Conference, vol. 1, IEEE, 2005, pp. 886–893.

[9] M.I. Devi, K. Selvakuberan, Fast web page categorization without the web page, in: International Conference on Semantic Web and Digital Libraries (ICSD) 2007, Citeseer, 2007.

[10] G. Douzas, F. Bacao, Effective data generation for imbalanced learning using conditional generative adversarial networks, Expert Syst. Appl. (ESWA) 91 (2018) 464–471.

[11] G. Douzas, F. Bacao, Improving imbalanced learning through a heuristic oversampling method based on k-means and smote, Inf. Sci. 465 (2018) 1–20.

[12] S. Dumais, H. Chen, Hierarchical classification of web content, in: Proceedings of the 23rd Annual International Conference on Research and Development in Information Retrieval (SIGIR), ACM, 2000, pp. 256–263.

[13] F. Esposito, D. Malerba, L. Di Pace, P. Leo, A machine learning approach to web mining, in: Congress of the Italian Association for Artificial Intelligence (AIIA), Springer, 1999, pp. 190–201.

[14] U. Fiore, A. De Santis, F. Perla, P. Zanetti, F. Palmieri, Using generative adversarial networks for improving classification effectiveness in credit card fraud detection, Inf. Sci. 479 (2017) 448–455.

[15] E.J. Glover, K. Tsioutsiouliklis, S. Lawrence, D.M. Pennock, G.W. Flake, Using web structure for classifying and describing web pages, in: Proceedings of the 11th International Conference on World Wide Web (WWW), ACM, 2002, pp. 562–569.
[16] I. Goodfellow, J. Pouge Abadie, M. Mirza, B. Xu, D. Warde Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in: Advances in Neural Information Processing Systems (NIPS), 2014, pp. 2672–2680.
[17] H. He, E.A. Garcia, Learning from imbalanced data, IEEE Trans. Knowl. Data Eng.(TKDE) (9) (2008) 1263–1284.
[18] H. He, X. Shen, A ranked subspace learning method for gene expression data classification, in: International Conference on Artificial Intelligence (ICAI), 2007, pp. 358–364.
[19] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (8) (1997) 1735–1780.
[20] S. Ioffe, C. Szegedy, Batch normalization: accelerating deep network training by reducing internal covariate shift, in: International Conference on Machine Learning (ICML), 2015.
[21] N. Japkowicz, S. Stephen, The class imbalance problem: a systematic study, Intell. Data Anal. (IDA) 6 (5) (2002) 429–449.
[22] T. Joachims, N. Cristianini, J. Shawe Taylor, Composite kernels for hypertext categorization, in: Proceedings of the 8th International Conference on Machine Learning (ICML), vol. 1, 2001, pp. 250–257.
[23] M.Y. Kan, H.O.N. Thi, Fast webpage classification using URL features, in: Proceedings of the 14th ACM International Conference on Information and Knowledge Management (CIKM), ACM, 2005, pp. 325–326.
[24] I.H. Kang, G. Kim, Query type classification for web document retrieval, in: Proceedings of the 26th Annual International Conference on Research and Development in Information Retrieval (SIGIR), ACM, 2003, pp. 64–71.
[25] D.P. Kingma, J.L. Ba, Adam: a method for stochastic optimization, in: International Conference on Learning Representations (ICLR) 2015, 2015.
[26] M. Kubat, R.C. Holte, S. Matwin, Machine learning for the detection of oil spills in satellite radar images, Mach. Learn. 30 (2–3) (1998) 195–215.
[27] Q. Li, Y. Chen, L.L. Jiang, P. Li, H. Chen, A tensor-based information framework for predicting the stock market, ACM Trans. Inf. Syst. (TOIS) 34 (2) (2016) 0–35.
[28] Q. Li, J. Tan, J. Wang, H. Chen, A multimodal event-driven lstm model for stock prediction using online news, IEEE Transactions on Knowledge and Data Engineering(TKDE) (2020). DOI:10.1109/TKDE.2020.2968894
[29] Q. Li, T. Wang, Q. Gong, Y. Chen, Z. Lin, S. Song, Media-aware quantitative trading based on public web information, Decis. Support Syst. (DSS) 61 (2014) 93–105.
[30] F. Louzada, P.H. Ferreira Silva, C.A. Diniz, On the impact of disproportional samples in credit scoring models: an application to a brazilian bank data, Expert Syst. Appl. (ESWA) 39 (9) (2012) 8071–8078.
[31] C.E. Metz, Basic principles of ROC analysis, in: Seminars in Nuclear Medicine, vol. 8, Elsevier, 1978, pp. 283–298.
[32] R. Pearson, G. Goney, J. Shwaber, Imbalanced clustering for microarray time-series, in: Proceedings of the International Conference on Machine Learning (ICML), vol. 3, 2003.
[33] X. Qi, B.D. Davison, Knowing a web page by the company it keeps, in: Proceedings of the 15th ACM International Conference on Information and Knowledge Management (CIKM), ACM, 2006, pp. 228–237.
[34] X. Qi, B.D. Davison, Web page classification: features and algorithms, ACM Comput. Surv. (CSUR) 41 (2) (2009) 12.
[35] S. Rendle, Factorization machines, in: 2010 IEEE International Conference on Data Mining (ICDM), 2010, pp. 995–1000.
[36] K. Roth, A. Lucchi, S. Nowozin, T. Hofmann, Stabilizing training of generative adversarial networks through regularization, in: Advances in Neural Information Processing Systems (NIPS), 2017, pp. 2018–2028.
[37] H.C. Shin, N.A. Tenenholtz, J.K. Rogers, C.G. Schwarz, M.L. Senjem, J.L. Gunter, K.P. Andriole, M. Michalski, Medical image synthesis for data augmentation and anonymization using generative adversarial networks, in: International Workshop on Simulation and Synthesis in Medical Imaging (SASHIMI), Springer, 2018, pp. 1–11.
[38] J. Sun, J. Lang, H. Fujita, H. Li, Imbalanced enterprise credit evaluation with DTE-SBD: decision tree ensemble based on smote and bagging with differentiated sampling rates, Inf. Sci. 425 (2018) 76–91.
[39] C. Tsai, W. Lin, Y. Hu, G. Yao, Under-sampling class imbalanced datasets by combining clustering analysis and instance selection, Inf. Sci. 477 (2019) 47–54.
[40] M.N. Vartak, et al., On an application of Kronecker product of matrices to statistical designs, Ann. Math. Stat. 26 (3) (1955) 420–438.
[41] W. Yih, X. He, C. Meek, Semantic parsing for single-relation question answering, in: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL) (Short Papers), vol. 2, 2014, pp. 643–648.