Fake Detection in Imbalance Dataset by Semi-Supervised Learning with GAN

Jinus Bordbar[1], Saman Ardalan[2], Mohammadreza Mohammadrezaei[3*], Zahra Ghasemi[4]

[1]Department of Computer, Islamic Azad University, Shiraz Branch, Shiraz, Iran.
[2] Institute for Medical Informatics and Statistics, University of Kiel, Kiel, Germany.
[3]Department of Computer, Islamic Azad University, Ramhormoz Branch, Ramhormoz,Iran.
[4]Department of Computer, Islamic Azad University, Dezful Branch, Dezful,Iran.

*Corresponding author(s). E-mail(s): Mohammadrezaei.m.reza@gmail.com;
Contributing authors: Jinus.Bordbar@gmail.com; Ardalan@medinfo.uni-kiel.de;
M855ghasemi22@gmail.com;

Abstract

As social media usage continues to grow across the globe, the detection of fake accounts has become increasingly challenging and crucial for platform integrity. A major obstacle in this domain lies in the complexity of graph-structured data, which contains numerous nodes and unrelated features, often making data analysis and its manipulation harder. Additionally, such data structures frequently suffer from imbalanced classes, which creates further challenges for accurate detection. For our study, we used Twitter's (now "X") social media data, represented as a graph structure, where nodes represent individual users, and edges denote the connections between them, forming a vast and intricate network. This approach of graph representation can effectively address significant issues related to both imbalanced datasets and limited labeled data. Therefore, we implemented an Autoencoder along with a type of neural network technique called a Semi-supervised Generative Adversarial Network (SGAN), which can handle scenarios with few labeled samples. The results of this test ultimately showed that the accuracy reached 81% in detecting fake accounts, even when using only 100 labeled samples.

Keywords: Fake Accounts, Generative Adversarial Networks, Semi-Supervised Learning, Auto-Encoder

1 Introduction

As social media platforms continue to develop, the detection of fake accounts has become a challenge, which has affected user experience. Twitter, formerly one of the most well-known platforms with over 450 million monthly active users [1], has been vulnerable to fake accounts spreading misinformation [2] [3]. However, Twitter has changed significantly in recent years, rebranding as "X" and implementing more restrictive API policies. These changes may reduce the amount of user data accessible to researchers and altering studies reliant on user network data. Consequently, the adaptability of fake account detection methods, even with limited data, has become a major issue.

Our approach utilizes a semi-supervised Generative Adversarial Network (SGAN) model combined with auto-encoders to optimize feature extraction and handle imbalanced datasets, making it effective for scenarios with limited labeled data. By adjusting similarity metrics and user relationship measures, our approach can adapt to the specific data structures and same characteristics in other platforms. Handling large datasets typically demands substantial computational power, like having powerful GPUs,

increasing both time and cost. To address the challenges unique to social media, our study investigates the following specific research questions:

1. How can a semi-supervised GAN framework, integrated with auto-encoders for feature extraction, enhance the accuracy of fake account detection specifically in highly imbalanced and sparsely labeled social network datasets?

2. What specific adaptations within the SGAN framework effectively address data imbalance challenges unique to large-scale social network graphs, and how do these adaptations impact model performance with minimal labeled data?

3. How does the proposed SGAN method compare in accuracy and computational efficiency to other established models for fake account detection, such as traditional GANs or simpler machine learning models?

4. How does incorporating feature extraction techniques, like similarity measures between users, influence the model's ability to detect fake accounts within highly connected social network graphs?

Using similarity criteria to understand about connections between accounts and applying feature extraction techniques to focus on dispersed features in the data matrix, effectively reducing overfitting. These aspects present interesting research directions for future studies. Our proposed method aims to achieve high accuracy in fake account detection through a three-step process:

1. Strengthening Account Connections through Similarity Measures: By analyzing shared characteristics between user accounts—such as common friends [4] and similarity metrics like Adamic Adar [5] similarity—we establish stronger, more influential connections. This step enhances the model's ability to detect patterns indicative of fake accounts.

2. Reducing Dimensionality with Auto-Encoders [6]: To manage the high dispersion of critical features across the data matrix, we use auto-encoders for feature extraction and dimensionality reduction. This process prevents overfitting by focusing on the most relevant features, allowing the model to generalize better.

3. Leveraging SGAN for Optimal Detection: Finally, we employ a semi-supervised GAN (SGAN) framework to classify fake and real accounts. This framework optimizes the use of limited labeled data by using both labeled and unlabeled samples, achieving high accuracy even with imbalanced datasets [7].

The steps are illustrated in Figure 1. In this study, we represent the social media data from Twitter (now "X") as a user network, where each user account is modeled as a node and each interaction or friendship between users is depicted as an edge [8]. This network structure allows us to capture the relationships between users, which is essential for identifying patterns linked to fake accounts. Using this graph-based model, we analyze multiple pre-written similarity measures—such as Adamic Adar similarity and common friends. This structured approach facilitates the application of techniques, including feature extraction via auto-encoders and classification using a semi-supervised GAN (SGAN) framework. By constructing and analyzing this network, our method identifies influential connections and patterns within the graph that contribute to distinguishing real users from fake accounts with high accuracy.

The remaining sections of this paper are organized as follows: Section 2 discusses the Background & related work. Section 3 describes the methodology employed in this study. The evaluation and performance results on the Twitter dataset are presented in Section 4. Section 5 concludes the paper and discusses future work.

2 Background & Related Work

2.1 Background

In recent years, detecting fake accounts on social media has become a significant area of research due to its impact on platform integrity and user experience. This study employs a combination of deep learning techniques, specifically Auto-Encoders (AE) and Generative Adversarial Networks (GAN), to address challenges in fake account detection, such as data imbalance and limited labeled data.

2.1.1 Auto-Encoder (AE)

To address the challenges in fake account detection, this study incorporates multiple advanced techniques. Auto-Encoders (AE)[6] are employed for feature extraction and dimensionality reduction. Auto-Encoders work by compressing data into a simpler representation, preserving essential features while discarding irrelevant information. This process helps to reduce the dimensionality of the dataset,

making the model less prone to overfitting and more computationally efficient. An autoencoder consists of two main components: the Encoder, which compresses and condenses the input data, and the Decoder, which decompresses the compressed input by reconstructing it. Both the Encoder and Decoder components in an autoencoder are comprised of multiple layers of Neural Networks (NN), facilitating the reduction of input size through reconstruction [9].

2.1.2 Semi-Supervised Learning

Semi-supervised learning[7] is a broad category of machine learning techniques that leverage both labeled and unlabeled data. As the name suggests, it combines elements of both supervised and unsupervised learning. The fundamental concept behind semi-supervised learning is to treat datasets differently based on the presence or absence of labels. For labeled datasets, the algorithm employs traditional supervision to update the model weights. On the other hand, for unlabeled data, the algorithm aims to minimize prediction differences among similar training samples.

In this paper, a semi-supervised setting was employed, utilizing both labeled and unlabeled data. The labeled dataset served as the foundation for model predictions, contributing structure and defining the learning problem by specifying the number of classes and comparable clusters. Unlabeled datasets provided contextual information, allowing the model to grasp the overall distribution pattern to the greatest extent possible. By training on both labeled and unlabeled data, the model achieved accurate estimation and improved performance[10].

2.1.3 Generative Adversarial Network (GAN)

GANs are highly effective techniques for training datasets. These deep generative models consist of two sub-models: the discriminator and the generator. The discriminator model is responsible for classifying examples as either real or fake, while the generator model is trained to generate new examples [11]. In a zero-sum game adversarial scenario, the two models are jointly trained until the discriminator model is fooled approximately half of the time, indicating that the generator model is producing plausible examples. While GANs are a fascinating and rapidly evolving field, their ability to generate natural examples is particularly beneficial in addressing a wide range of challenging use cases, including imbalanced datasets.

2.2 Related Work

To identify abnormal nodes in online social networks, Wanda et al.[12] proposed a model that utilizes a dynamic deep-learning architecture to extract extensive features and classify malicious nodes based on node-link information. They introduced a WalkPool pooling function to improve network performance, enabling dynamic deep learning and achieving high classification accuracy with minimal loss in network training. This study highlights the potential of using dynamic deep learning for node classification, which aligns with our approach by demonstrating the effectiveness of advanced architectures for complex network analysis.

Kim et al.[3] focused on anomaly detection within imbalanced datasets, using a GAN-based method with an autoencoder as the generator. Their innovative model employed two separate discriminators for normal and abnormal data, optimizing detection through Patch loss and Anomaly adversarial loss functions. Cause finding anomaly in the dataset is just like finding the minority class. This method is relevant to our study as it demonstrates the potential of GAN and autoencoder in handling data imbalance—a challenge that our SGAN framework addresses directly in fake account detection.

Bordbar et al.[13] introduced a method specifically for fake account detection on Twitter by applying similarity measures and the GAN framework to a large dataset containing 10,000 fake nodes. Their model achieved 75% accuracy in classifying fake accounts in an imbalanced dataset. Although promising, their approach heavily relied on a large labeled dataset, which contrasts with our semi-supervised method that reduces dependency on labeled data, making it more adaptable for scenarios with limited labeling resources.

Kaplan et al.[2] applied the BiGAN [14] algorithm to the KDDCUP99 [15] dataset to detect anomalies, utilizing a one-class anomaly detection approach. They identified and addressed the interdependence between discriminator and generator by proposing separate training steps, ultimately improving BiGAN's performance in anomaly detection. This work supports our approach by underscoring the effectiveness of GAN-based models in identifying anomalies, an aspect we leverage within our SGAN to detect fake accounts.

Mohammadrezaei et al. Finally, [8] used traditional classification algorithms (SVM [16], logistic regression [17], and Gaussian SVM [18]) on Twitter data to identify fake accounts. Despite the Gaussian SVM's superior performance, they encountered limitations due to data imbalance, even with the Synthetic Minority Oversampling Technique. This highlights the importance of addressing data imbalance effectively, which our SGAN method tackles by learning from both labeled and unlabeled data.

3 Proposed Method

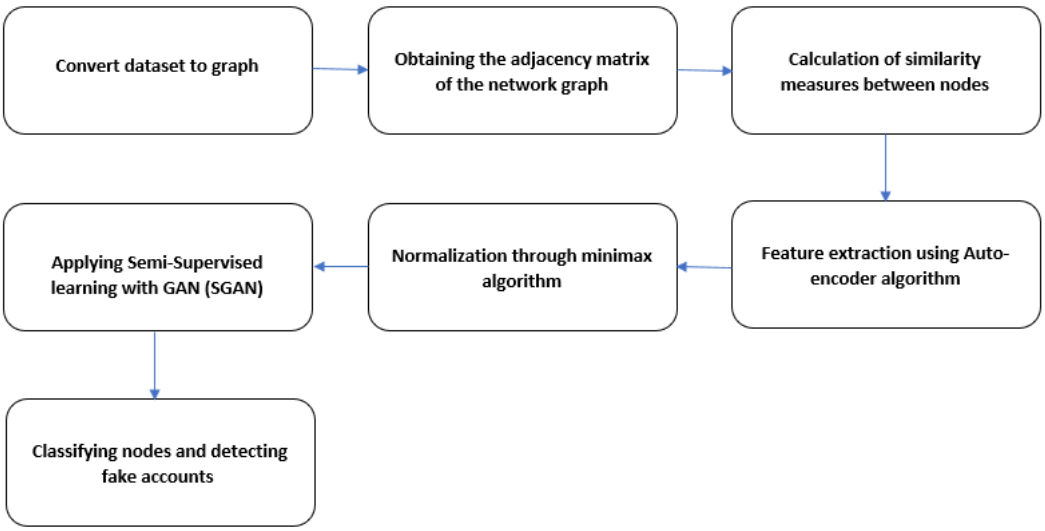

Fig. 1: Steps of proposed method

The Twitter data repository can be accessed on internet [1]. There are 5,384,162 users in this dataset with 16,011,445 links among them. the dataset includes a limited set of labeled examples, where a small subset of data, like 12438, is labeled as Fake. Real data set as 1 and fake ones as 0. This label is crucial for training the SGAN, as it provides ground truth information about which nodes represent fake accounts and which represent real ones. We initially separate a set of 10,000 user nodes, then randomly selected a subset of 50 real labeled examples, 50 labeled as fake in the subset. These labeled examples, crucial for the supervised training component of our SGAN, serve as ground truth data to differentiate between genuine and fraudulent profiles. Measuring criteria for each node were obtained using 10 similarity measures, resulting in 10 matrices of size 10,000*10,000. Subsequently, an autoencoder (AE) was employed for each matrix, with a code size of 10, to compress the matrices into 10 features each. The number of 10 features was determined through experimentation. Finally, all the matrices were concatenated, resulting in a 10,000*100 dimensional matrix, of which 9,900*100 dimensions were utilized in the unsupervised SGAN section. Furthermore, 50 fake labels and 50 real labels were randomly assigned from the dataset, which amounted to a 100*100 (100*10*10) input for the supervised SGAN section.

Supervised learning involves having data along with corresponding labels, while unsupervised learning deals with unlabeled data. Semi-supervised learning falls between these two categories, where there are limited labels available alongside a large amount of unlabeled data. In a regular GAN, the discriminator is trained in an unsupervised manner to distinguish between real and fake data. However, in SGAN, the discriminator is trained not only in an unsupervised manner but also in a supervised mode. The unsupervised discriminator learns features, while the supervised discriminator learns classification. In SGAN, the generator is solely utilized for generating fake nodes relations, thereby improving the discriminators' classification performance. A dataset consisting of ten thousand samples was selected,

[1] https://github.com/kagandi/anomalous-vertices-detection/tree/master/data

including 50 randomly chosen fake nodes and 50 samples of real nodes. Min-Max normalization was applied to normalize the data. The dataset was divided into a 70% training set and a 30% testing set, determined through experimentation. A supervised model was then defined, using the Sigmoid activation function and binary cross-entropy as the loss function. To convert the supervised model into an unsupervised one, an additional layer was added to act as a Sigmoid function, but with an improved activation function calculated as the normalized summation of the exponential outputs [19]. The unsupervised layer used the same binary cross-entropy loss function as the supervised layer. It is worth mentioning that the SGAN approach achieves better accuracy for limited labeled data compared to CNN [10].

3.1 Mapping Social Network's Data into Graph

To analyze the Twitter Dataset, the data was converted into a graph by utilizing similarity measures [20]. During this process, each user was represented as a node, and each relation between users was represented as an edge. Depending on the nature of the relationship in the social network, the graph could be either directional or non-directional. In this particular paper, the graph extracted from the dataset was non-directional.

3.2 Calculation of Adjacency Matrix in Network Graph

For a graph with N users, an N*N square matrix is utilized as the adjacency matrix [4]. If there exists an edge or path connecting vertex i to vertex j, the element at the i-th row and j-th column, as well as the element at the j-th row and i-th column, is assigned a value of one. Otherwise, it is set to zero.

3.3 Calculation of Different Similarity Measures Between Nodes

The information gathered from various papers led to the conclusion that no individual feature alone was sufficient to distinguish between users in the network. Hence, in this method, multiple features were utilized to enhance the accuracy of detecting fake accounts. The objective of defining similarity measures was to optimize and improve the quality of the features extracted from the user's network. Ten features were selected, including Common Friends, Total Friends, Adamic Adar Similarity, and so on.

Friendship Graph

A social network G = (E, N) maps into a graph, so that a set of N nodes represents social network users, while the set of edges $E \subset N \times N$ represents the relationships. In addition, the dot sign was used to refer to a particular component in a graph.
(1) A represents the sparse adjacency matrix for graph G. If (v, u) is an edge in G, then A (v, u) =1. Otherwise, A (v, u) =0.
(2) Friendship graph (FG): Considering the social network graph G and a node $v \in G.N$ the friendship graph is a vertex containing all vertices that are directly connected to that node and are defined in (1). Direct connection of nodes between particular nodes, with no connection to the node by itself, represented as graph G [4]:

$$FG(v).N = \{v\} \cup \{n \in G.N |\ n \neq v, \exists\ e \in G.E, e = \langle v, n\rangle\} \tag{1}$$

$$FG(v).E = \{\langle v, n'\rangle \in G.E |\ n \in FG(v).N\} \cup \{\langle n, n'\rangle \in G > E | n, n' \in FG(v).N\} \tag{2}$$

Common Friends

Nodes which are on the length of 2 between other nodes are common friends of those two nodes [21] [22] :

$$CF(u, v) = |FG(v).N \cap FG(u).N| \tag{3}$$

Total Friends

The number of different friends between two nodes was displayed as follows: [22].

$$Total\ Friends(v, u) = |FG(v).N \cup FG(u).N| \tag{4}$$

Correlation Similarity

This measure is used to calculate the strength of the linear relation in two data samples. [23].

$$Pearson's\ correlation\ coef(v.u) = \frac{covariance(v,u)}{x}(stdv(v) * stdv(u)) \tag{5}$$

Jaccard Similarity

Jaccard coefficient stipulated the ratio of mutual friends of two node neighbors to their total friends [21].

$$Jaccard\,F(v,u) = \frac{\left|FG(u).N \cap FG(v).N\right|}{\left|FG(u).N \cup FG(v).N\right|} \tag{6}$$

Euclidean Similarity

The Euclidean Similarity formula is used to calculate the distance between two points in Cartesian coordinates, which can be seen in Formula 7. Euclidean distance finds the shortest path between two points as the distance [24].

$$D(x,y) = \sqrt{\sum_{n-1}^{i}(x_i - y_i)^2} \tag{7}$$

Adamic Adar Similarity

This similarity measure was originally employed to identify strong connections between web pages and was based on the number of shared features between two pages. In the context of link prediction, this shared connection referred to the common neighbor of two nodes. The similarity between two vertices in this method was determined by the following relation, where Z represented the common neighbor of vertices U and V.[5].

$$score(v,u) = \sum_{z\epsilon\Gamma(x)\cap\Gamma(y)} \frac{cosine(x,y)}{log(|\Gamma|)} \tag{8}$$

L1_Norm Similarity

[25]

$$L1\_norm(v.u) = \frac{|FG(v).N \cap FG(u).N|}{|FG(v).N|.|FG(u).N|} \tag{9}$$

Cosine Similarity

Another measure of similarity between graph nodes was Cosine Similarity. Cosine Similarity computes the resemblance between two product vectors [4]

$$Cos(v.u) = \frac{|FG(v).N \cap FG(u).N|}{\sqrt{|FG(v).N|.|FG(u).N|}} \tag{10}$$

Edge Weight Measure:

Edge Weight similarity was first calculated as two distinct features for each of the two vectors [26].

$$w(v) = \frac{1}{\sqrt{1 + FG(v).N}} \tag{11}$$

$$w(u) = \frac{1}{\sqrt{1 + FG(u).N}} \tag{12}$$

Now the Edge Weight between two vertices U and V can be estimated in the next two ways.

Summation of the weights: The summation of the weights was identical to the two weights of u and v which are added jointly:

$$W(v,u) = w(v) + w(u) \tag{13}$$

Coefficient of weights: The coefficient of weights was equal to the product of the two weights. It was defined as below :

$$W(v,u) = w(v) * w(u) \tag{14}$$

In this section, for each similarity measure, a matrix with a diagonal of zero would be determined.

3.4 Distribution modification in Imbalance Data

The term "Imbalanced Data" [27] refers to a situation where a dataset exhibits an unequal distribution, with the "majority classes" accounting for a significant portion of the dataset, while the minority classes represent a smaller percentage. The dataset utilized in this paper encountered an imbalanced data problem. As mentioned earlier, due to the high dimensionality and dispersion associated with the matrix, the data dimension was reduced using an AE (Auto-encoder) that was trained with the 10,000 samples. After obtaining the final matrix, as explained in previous sections, SGAN (Semi-supervised Generative Adversarial Network) was employed to address the issue of imbalance. In SGAN, the unsupervised layer compensated for the limited availability of labeled data by leveraging the available unlabeled dataset.

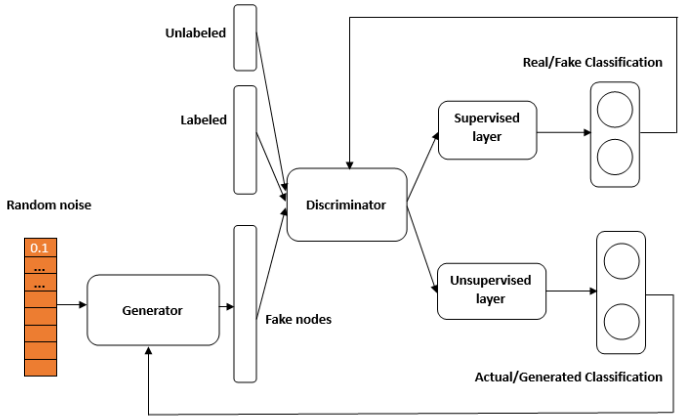

Fig. 2: Structure of Proposed Method

3.5 Training SGAN and Fake Accounts Detection

To address the challenges associated with imbalanced datasets and limited labeled data, this study utilizes a Semi-Supervised Generative Adversarial Network (SGAN). The SGAN approach is designed to enhance classification accuracy, particularly when identifying fake accounts with minimal labeled data. This algorithm allowed learning of the fake nodes format with only a small amount of labeled data. Initially, a subset of 10,000 data points was separated from the dataset. Two sets consisting of 50 randomly chosen nodes were set aside, with one set labeled as fake and the other as real. The generator played a crucial role in enhancing the classification accuracy of the discriminator. It took random noise as input and generated nodes that closely resembled the actual dataset. A base discriminator was subsequently defined, which shared weights between the supervised and unsupervised discriminators. The base discriminator served as the foundation for both the supervised and unsupervised discriminators, differing only in the last layer and activation functions. At this stage, an unsupervised discriminator was specified, functioning as a binary classifier to classify fake and real nodes with a custom activation function represented by D(x) = Z(x) / (Z(x) + 1), where Z(x) = sum(exp(l(x))) * l(x). Finally, during the inference stage, the trained supervised discriminator, as a component of SGAN, was employed to classify actual data or generated dataset by the Tanh activation function.

4 Evaluation

In the field of deep learning, there are several evaluation criteria used to assess and analyze the performance of algorithms. One commonly used technique is the Confusion matrix, which is a square matrix

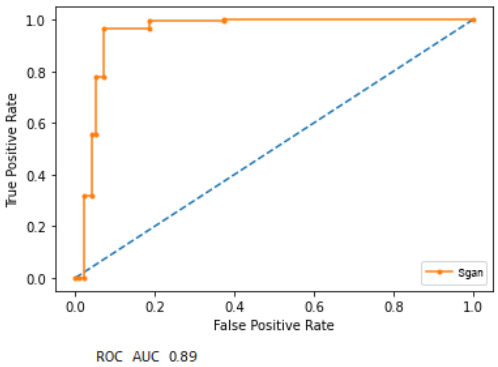

Fig. 3: AUC of proposed method

with dimensions equal to the number of classification classes [28]. Another criterion that provides a graphical representation of classification performance is the ROC curve [29]. The ROC curve plots the true positive rate (detection of the correct rate for positive classes) against the false positive rate (detection of the wrong rate for negative classes). The area under the ROC curve (AUC) is a measure of the model's ability to classify the data accurately. An AUC value of one indicates that the model can classify the data as accurately as possible. Figure 4 presents the confusion matrix of the proposed method, which provides a visual representation of the classification performance. Furthermore, Figure 3 illustrates that the proposed method achieved an AUC of 89%. Given the significant class imbalance (1:100 real-to-fake ratio), we used the weighted F1 score to provide a balanced evaluation of the model's performance across both classes.

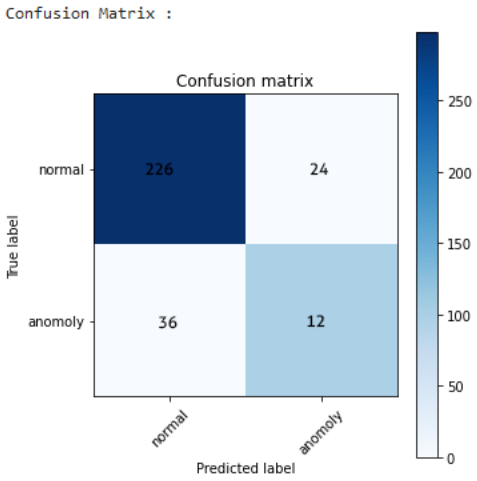

Fig. 4: Confusion matrix of proposed method

4.1 Results Comparison

The semi-supervised GAN model (SGAN) extends the traditional GAN architecture by incorporating both a generator model and two discriminators—one supervised and one unsupervised. This design enables SGAN to generate new examples while also effectively classifying data. In previous work by Bordbar et al. [13], a traditional GAN was employed to analyze fake accounts with a large dataset. The dataset was divided into separate fake and real classes, and the GAN algorithm was used to analyze fake accounts. This analysis involved 10,000 fake nodes from a dataset of 1,000,000, a considerable data volume that required extensive computational resources. Since our dataset is highly imbalanced, with about 100 real nodes for every 1 fake node, we used the weighted F1 score to evaluate performance. This metric adjusts for the uneven class sizes, giving a fairer view of how well the model detects both real and fake nodes. Using the weighted F1 score helps us accurately assess performance on the smaller, minority class of fake nodes.

In addition to traditional GAN, we compared SGAN with a Semi-Supervised SVM (S3VM) as an additional baseline. S3VM uses both labeled and unlabeled data to refine its classification boundary, extending traditional SVM to improve accuracy in cases with limited labeled data. However, unlike SGAN, S3VM does not generate synthetic data to address class imbalance. In highly imbalanced datasets, such as those containing a smaller number of fake accounts, S3VM's performance is limited by its reliance on available labeled data alone.

These metrics give a comprehensive view of the SGAN model's suitability for various real-world applications. The model achieved an AUC value of 89%, underscoring its potential effectiveness in fake account detection. The study also highlights how SGAN can reduce the demand for labeled data, improving both efficiency and scalability in detecting fake accounts.

By comparing SGAN with both traditional GAN [13] and a Semi-Supervised SVM (S3VM), we demonstrate SGAN's superior capability to address class imbalance and reduce dependency on labeled data. Our findings indicate that SGAN achieves higher accuracy, recall, and AUC scores across both majority and minority classes, making it more suitable for real-world applications where labeling resources are limited. These results highlight SGAN's adaptability and cost-effectiveness, as it reduces the demand for large labeled datasets without compromising classification performance.

In real-world scenarios, however, it is common to have a large amount of unlabeled data and only a limited number of labeled instances. This research aims to address that challenge by reducing dependency on labeled data while maintaining high classification accuracy. Additionally, this method minimizes the need for powerful hardware by optimizing the use of a smaller labeled dataset, making it more practical and cost-effective for large-scale applications.

Table 1: Table of comparison

Modules	Accuracy	Recall	Precision	AUC	F1(Fake)	F1(Real)	Weighted F1
Semi-Supervised GAN	%81	%80	%76.3	%89	%74	%86	%85
Traditional GAN	%80	%82	%84	%75	%65	%83	%77
Semi-supervised SVM	%65	%67	%68.5	%60	%52	%74	%69

5 Conclusion & Further Work

By using SGAN, researchers aim to optimize the utilization of this small number of labeled examples and minimize reliance on powerful hardware. The SGAN model consists of a generator model, an unsupervised discriminator, and a supervised discriminator, which are trained simultaneously. This approach allows for the generation of new examples by the generator model and effective classification of unseen examples by the supervised discriminator. By applying this method, it is possible to detect anomalies, identify fake accounts, or address other challenging use cases in real-world scenarios. Node classification in social media plays a crucial role in detecting anomalies within a network. This approach introduces a Semi-supervised generative deep-learning classifier, aiming to address the challenges associated with handling large-scale data and imbalanced classes. This method achieved a commendable AUC score of 89%. The corresponding percentages for each criterion are presented in Table 2. These results lead to the conclusion that SGAN can serve as an effective solution for handling minority classes and big data challenges and also, the impact of this approach on fake account detection could be significant as it aims to improve the accuracy and effectiveness of identifying fake accounts in social media. Using Generative Adversarial Networks (GAN) is promising for fake account detection because it allows for the creation of synthetic data that closely resembles real data. This is achieved through the use of two neural networks: a generator network that creates fake data and a discriminator network that tries to distinguish between real and fake data. By training these networks together in an adversarial manner, the generator network learns to create synthetic data that is difficult for the discriminator network to distinguish from real data. By leveraging deep learning techniques and utilizing network-based measures along with advanced algorithms like Generative Adversarial Networks (GAN), the approach has the potential to enhance the reliability of fake account detection. This, in turn, can lead to increased trust and security on social media platforms, helping to reduce the spread of false information, and other malicious activities associated with fake accounts. Furthermore, future

research can explore the potential of improving the classifier by exploring different similarity criteria, adjusting the number of layers in the discriminator or generator architecture, and modifying the learning rate. These aspects present interesting research directions for future studies.

6 Acknowledgement

We acknowledged there is a preprint version of this manuscript that exists on arxiv.org and Research square.

7 Funding Information

This research was conducted without external funding. The authors declare that no specific grants, financial support, or sponsorships were received for the design, implementation, or publication of this study.

8 Comments on Ethics

In our proposed method, we represent each user as a node in the social network, characterized by a set of numbers, with connections between users depicted as edges indicating which numbers are interconnected. We acknowledge the importance of ethical considerations in handling user data within social networks. We adhere to ethical standards by ensuring the confidentiality and privacy of user information. The data used in this study is anonymized and does not include personally identifiable information. Additionally, any potential implications for user privacy have been carefully considered, and steps have been taken to mitigate any adverse effects. We are committed to conducting our research with the utmost respect for ethical guidelines and standards.

References

[1] Agrawal, A., Hamling, T.: Sentiment analysis of tweets to gain insights into the 2016 us election (2020) https://doi.org/10.52214/cusj.v11i.6359

[2] Kaplan, M.O., Alptekin, S.E.: An improved bigan based approach for anomaly detection. Procedia Computer Science **176**, 185–194 (2020) https://doi.org/0.1016/j.procs.2020.08.020

[3] Kim, J., Jeong, K., Choi, H., Seo, K.: Gan-based anomaly detection in imbalance problems. In: European Conference on Computer Vision, pp. 128–145 (2020). https://doi.org/10.1109/IJCNN.2011.6033365 . Springer

[4] Akcora, C.G., Carminati, B., Ferrari, E.: User similarities on social networks. Social Network Analysis and Mining **3**(3), 475–495 (2013) https://doi.org/10.1007/s13278-012-0090-8

[5] Lü, L., Zhou, T.: Link prediction in weighted networks: The role of weak ties. EPL (Europhysics Letters) **89**(1), 18001 (2010) https://doi.org/10.1209/0295-5075/89/18001

[6] Baldi, P.: Autoencoders, unsupervised learning, and deep architectures. In: Proceedings of ICML Workshop on Unsupervised and Transfer Learning, pp. 37–49 (2012). JMLR Workshop and Conference Proceedings. https://api.semanticscholar.org/CorpusID:10921035

[7] Learning, S.-S.: Semi-supervised learning. CSZ2006. html (2006)

[8] Mohammadrezaei, M., Shiri, M.E., Rahmani, A.M.: Identifying fake accounts on social networks based on graph analysis and classification algorithms. Security and Communication Networks **2018** (2018) https://doi.org/10.1155/2018/5923156

[9] Meng, Q., Catchpoole, D., Skillicom, D., Kennedy, P.J.: Relational autoencoder for feature extraction. In: 2017 International Joint Conference on Neural Networks (IJCNN), pp. 364–371 (2017). https://doi.org/10.1109/IJCNN.2017.7965877 . IEEE

[10] Odena, A.: Semi-supervised learning with generative adversarial networks. arXiv preprint arXiv:1606.01583 (2016) https://doi.org/10.1109/ISBI.2018.8363749

[11] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial networks. Communications of the ACM **63**(11), 139–144 (2020) https://doi.org/10.1145/3422622

[12] Wanda, P., Jie, H.J.: Deepfriend: finding abnormal nodes in online social networks using dynamic deep learning. Social Network Analysis and Mining **11**(1), 1–12 (2021) https://doi.org/10.1007/s13278-021-00742-2

[13] Bordbar, J., Mohammadrezaie, M., Ardalan, S., Shiri, M.E.: Detecting fake accounts through generative adversarial network in online social media. arXiv preprint arXiv:2210.15657 (2022)

[14] Donahue, J., Krähenbühl, P., Darrell, T.: Adversarial feature learning. arXiv preprint arXiv:1605.09782 (2016)

[15] Tavallaee, M., Bagheri, E., Lu, W., Ghorbani, A.A.: A detailed analysis of the kdd cup 99 data set. In: 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications, pp. 1–6 (2009). Ieee

[16] Hearst, M.A., Dumais, S.T., Osuna, E., Platt, J., Scholkopf, B.: Support vector machines. IEEE Intelligent Systems and their applications **13**(4), 18–28 (1998)

[17] Hosmer Jr, D.W., Lemeshow, S., Sturdivant, R.X.: Applied Logistic Regression. John Wiley & Sons, ??? (2013)

[18] Keerthi, S.S., Lin, C.-J.: Asymptotic behaviors of support vector machines with gaussian kernel. Neural computation **15**(7), 1667–1689 (2003)

[19] Salimans, T., Goodfellow, I.J., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training gans. ArXiv **abs/1606.03498** (2016)

[20] Jouili, S., Tabbone, S., Valveny, E.: Comparing graph similarity measures for graphical recognition. In: International Workshop on Graphics Recognition, pp. 37–48 (2009). https://doi.org/10.1007/978-3-642-13728-0_4 . Springer

[21] Santisteban, J., Tejada-Cárcamo, J.: Unilateral jaccard similarity coefficient (2015)

[22] Dong, L., Li, Y., Yin, H., Le, H., Rui, M.: The algorithm of link prediction on social network. Mathematical problems in engineering **2013** (2013) https://doi.org/10.1155/2013/125123

[23] Benesty, J., Chen, J., Huang, Y., Cohen, I.: Pearson correlation coefficient (2009) https://doi.org/10.1007/978-3-642-00296-0_5

[24] Elmore, K.L., Richman, M.B.: Euclidean distance as a similarity metric for principal component analysis. Monthly weather review **129**(3), 540–549 (2001) https://doi.org/10.1007/978-3-642-00296-0_5

[25] Kwak, N.: Principal component analysis based on l1-norm maximization. IEEE transactions on pattern analysis and machine intelligence **30**(9), 1672–1680 (2008) https://doi.org/10.1109/TPAMI.2008.114

[26] Cukierski, W., Hamner, B., Yang, B.: Graph-based features for supervised link prediction. In: The 2011 International Joint Conference on Neural Networks, pp. 1237–1244 (2011). https://doi.org/10.1109/IJCNN.2011.6033365 . IEEE

[27] He, H., Garcia, E.A.: Learning from imbalanced data. IEEE Transactions on knowledge and data engineering **21**(9), 1263–1284 (2009) https://doi.org/10.1109/TKDE.2008.239

[28] Stehman, S.V.: Selecting and interpreting measures of thematic classification accuracy. Remote sensing of Environment **62**(1), 77–89 (1997) https://doi.org/10.1016/S0034-4257(97)00083-7

[29] Davis, J., Goadrich, M.: The relationship between precision-recall and roc curves. In: Proceedings

of the 23rd International Conference on Machine Learning, pp. 233–240 (2006). https://doi.org/10.1145/1143844.1143874