International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 05 Issue: 01 | Jan – 2026 DOI: 10.55041/ISJEM05378
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

Fake Profile Detection in Social Networking Platforms Using Machine Learning

S. SHIRLEY., MCA
(Assistant Professor, Master of Computer Applications)
A.MAGESH, MCA
Christ College of Engineering and Technology
Moolakulam, Oulgaret Municipality, Puducherry – 605010.

Abstract

The rapid expansion of social networking platforms has led to a significant rise in fake profiles, which are widely used for impersonation, spamming, phishing, and spreading misinformation [1][2]. Traditional rule-based and manual verification methods fail to detect modern fake accounts due to evolving behavioral patterns and large-scale user data [3]. This paper proposes an end-to-end Fake Profile Detection System that classifies social media accounts as either genuine or fake by combining profile-based and behavioral feature extraction with supervised machine learning [4]. The system extracts key indicators such as account age, post frequency, followers count, following count, follower–following ratio, engagement rate, activity consistency, and verification status, which are transformed into structured feature vectors for training [4][5]. These features are used to train multiple classification models including Logistic Regression, Random Forest, and XGBoost, where ensemble-based approaches provide improved detection accuracy and reduced false classifications [6][7]. The proposed system is implemented as a modular web application using a Flask backend with a user-friendly interface for profile analysis and prediction reporting [8]. Experimental evaluation on benchmark datasets demonstrates that behavioral and relationship-based features contribute most effectively to identifying suspicious profiles, while ensemble models achieve more stable performance compared to single classifiers [4][5]. The developed system provides a scalable and practical solution for strengthening trust and security in social networking environments through automated fake profile identification [1][2].

Keywords

Fake profile detection, social networking platforms [1][2], machine learning [4], supervised classification, Logistic Regression [6], Random Forest [7], XGBoost [8], feature extraction [4][5], behavioral analysis [3], Flask web application [8], account authenticity.

1. Introduction

Social networking platforms have become one of the most popular digital environments for communication and online interaction [1][2]. Along with this growth, the number of fake profiles has also increased rapidly. Fake accounts are commonly used for impersonation, spamming, phishing, spreading misinformation, and other fraudulent activities, which creates serious risks for users and reduces trust in social media platforms [3]. Detecting such fake profiles has become an important requirement to ensure safer and more reliable online communities [2][4].

Traditional fake profile detection methods mainly depend on manual reporting, simple verification checks, or rule-based filtering such as identifying incomplete profiles or abnormal activity levels [4]. However, these approaches are limited because modern fake profiles are created using advanced techniques like automated bots, stolen identities, and realistic behavior simulation [1][3]. As attackers continuously change their strategies, traditional systems fail to detect sophisticated fake accounts accurately, especially when dealing with large-scale and dynamic social media data [2][4].

To overcome these challenges, machine learning offers an intelligent and scalable solution by identifying hidden patterns in user profile data and behavior [4][5]. By analyzing features such as account age, followers and following counts, follower–following ratio, post frequency, engagement rate, and activity consistency, supervised learning models can classify profiles as genuine or fake more effectively [4][5]. This project proposes a complete fake profile detection system using machine learning models such as Logistic Regression, Random Forest, and XGBoost [5][6][7], along with a Flask-based web application for profile analysis and

prediction reporting [8], providing a practical solution to strengthen security in social networking platforms [1][2].

The key contributions of this work are:

1. A complete automated pipeline for fake profile detection, starting from profile data input to final prediction output with clear and interpretable classification results [4].
2. Comparative implementation and evaluation of supervised machine learning models such as Logistic Regression, Random Forest, and XGBoost for identifying fake and genuine social media profiles [5][6][7].
3. Feature-level analysis demonstrating that behavioral and relationship-based indicators like follower–following ratio, engagement rate, and activity consistency are strong discriminators for fake profile detection [3][4].
4. Development of a user-friendly web-based system using Flask that provides profile analysis, prediction reporting, and history tracking for improved usability [8].
5. A modular and scalable architecture that supports future enhancements such as real-time monitoring, advanced feature integration, and continuous model improvement [2][4].

2. Materials and Methods

2.1 Dataset

This work utilizes a publicly available social networking fake profile dataset collected from Kaggle, which contains labeled samples representing both genuine and fake user profiles [8]. The dataset provides structured attributes and behavioral indicators that support profile authenticity analysis, including features such as account age, number of posts, followers count, following count, follower–following ratio, engagement measures, verification status, and activity patterns [4][5]. These features are used as input to train supervised machine learning models for classification [5].

To ensure reliable evaluation and avoid overfitting, the dataset was divided into two subsets: 80% for training and 20% for testing [4]. Basic preprocessing steps such as handling missing values, removing duplicate entries, and encoding categorical values were applied before model training [9]. The final processed dataset was then used for performance analysis of different machine learning models in detecting fake and genuine social media accounts [4][5].

2.2 Static Feature Extraction

Feature extraction in this work focuses on collecting meaningful profile-based and behavior-based indicators from social networking accounts to support fake profile classification [4]. The extracted features are non-intrusive and do not require direct access to private user data, making the approach practical for real-world platforms [3]. For each profile, multiple feature categories were systematically extracted, cleaned, and converted into numerical vectors for machine learning training and prediction [5][9].The main feature categories used in this project are:

- **Profile Attributes**: These include basic account-level information such as account age, profile completeness, and verification status. Fake profiles often contain incomplete or inconsistent profile details and are typically created recently compared to genuine accounts [4].
- **Social Relationship Features:** These capture the connectivity pattern of the account such as followers count, following count, and follower–following ratio. Many fake accounts show unusual relationship behavior, such as following a large number of users while having very few followers [3][4].
- **Activity & Engagement Features:** These include post count, posting frequency, activity consistency, and engagement rate. Fake profiles may show abnormal posting behavior, sudden spikes in activity, or low engagement compared to genuine profiles [4].

These extracted indicators are widely recognized as strong discriminators for identifying suspicious accounts [3][4]. To understand the impact of each category, the models are trained and evaluated using the combined feature set, enabling the system to learn both profile-level characteristics and behavioral patterns effectively for accurate fake profile detection [4][5].

2.3 Machine Learning Models

- This project applies supervised machine learning algorithms to classify social networking accounts as **genuine (0)** or **fake (1)** using extracted profile and behavioral features [4][5]. The following models were implemented and compared to identify the most accurate classifier for fake profile detection.
- **Logistic Regression (LR):** A simple and efficient binary classification model used as a baseline for prediction tasks [12]. It predicts whether an account is

fake or genuine based on the relationship between input features and output class [12].

- **Random Forest (RF):** An ensemble learning model that combines multiple decision trees to improve prediction accuracy and reduce overfitting [11]. It performs well in detecting fake profiles by learning complex patterns from features like follower–following ratio, activity frequency, and engagement behavior [11].
- **XGBoost:** A powerful gradient boosting algorithm that builds strong classifiers by sequentially reducing prediction errors and improving generalization performance [10]. It is effective in identifying subtle behavioral differences between genuine and fake accounts [10].

All models were trained on the processed dataset and evaluated using standard performance metrics such as **accuracy, precision, recall, and F1-score** to measure classification effectiveness **[9]**.

2.4 System Architecture

To ensure scalability, modularity, and ease of use, the Fake Profile Detection System is designed as a multi-tier web application [4][8]. The high-level architecture consists of the following core modules:

1. **Frontend Interface:** A responsive web interface developed using HTML, CSS, and JavaScript that allows users to enter profile details or upload dataset inputs and view prediction results in a clear format [8].
2. **Server:** A Flask-based backend that handles user requests, validates inputs, manages business logic, and connects the machine learning model with the user interface [8].
3. **Feature Extraction Module:** This module extracts important profile-based and behavioral features such as account age, followers/following count, follower–following ratio, post activity, engagement rate, and verification status, and converts them into a structured format for prediction [3][4].
4. **Prediction Engine:** It loads the trained machine learning models (Logistic Regression, Random Forest, and XGBoost) and generates the classification output as Fake or Genuine, along with confidence levels [10][11][12].
5. **Report Generator:** Generates a detailed output report displaying input features, prediction results, and model performance measures, helping users understand the reason for classification [4].
6. **Database/Storage Layer:** Stores user details, uploaded profile records, extracted features, and prediction history for future tracking and analysis (CSV/database storage) [8].
7. **Administration Module:** Provides control for managing stored records, monitoring model performance, and supporting future updates such as retraining with new profile datasets [4].

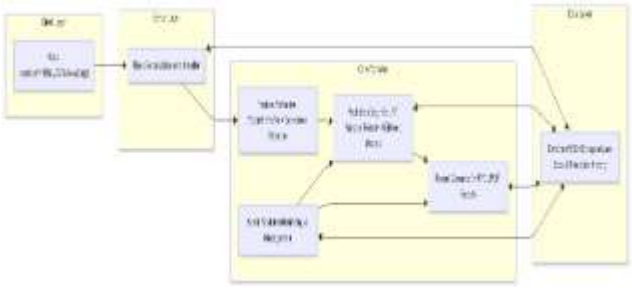

Figure 1: The high-level system architecture of the proposed AMDS.

3. Results and Discussion

3.1 Feature Analysis and Importance

The feature analysis in this project highlights that both profile information and user behavior play an important role in detecting fake accounts on social networking platforms [1][4]. Profiles with low completeness, recently created accounts, and unverified status were found more commonly in fake samples, while genuine users usually maintain consistent profile details over a long period [4]. Along with this, abnormal relationship patterns such as high following count with very low followers and an unrealistic follower–following ratio strongly indicate suspicious accounts, since many fake profiles try to increase reach by mass-following other users [3][4]. Activity and engagement-based indicators provided further confirmation for accurate classification [2][4]. Fake profiles often show irregular posting frequency, sudden spikes in activity, low engagement rates, and repetitive interactions, whereas genuine profiles generally have stable activity patterns and natural engagement behavior [2][4]. Overall, the combined analysis confirms that relationship and behavioral features contribute the most discriminative power for fake profile detection, while basic profile attributes support the classification process for better reliability [4][5].

3.2 Model Performance Evaluation

Standard evaluation metrics such as Accuracy, Precision, Recall, and F1-Score were used to measure the performance of each machine learning classifier on the hold-out test dataset [9]. To analyze the predictive strength of the models, Logistic Regression, Random

Forest, and XGBoost were trained and tested using the extracted fake profile features [10][11][12]. The results are summarized in the tables below with class-wise metrics for both Genuine and Fake profiles [4].

In addition, the evaluation helps in identifying how well each model minimizes false positives (genuine predicted as fake) and false negatives (fake predicted as genuine) [9]. This comparison also provides insight into the stability of each classifier when handling different profile behavior patterns [4][5]. Overall, the analysis supports selecting the most reliable model for accurate and practical fake profile detection [4].

Model Performance on Fake Profile Dataset (Combined Features)

Analysis: The results show that Random Forest achieved the highest overall accuracy of 93.40%, with balanced precision and recall for both genuine and fake classes. This indicates that ensemble-based models are highly effective in identifying fake profiles due to their ability to capture complex behavioral patterns such as follower–following imbalance, irregular activity frequency, and low engagement rates [11]. Logistic Regression performs well as a baseline model but shows comparatively lower accuracy [12], while XGBoost provides strong performance and stable classification results across both classes [10].

Model	Accuracy	Precision		F1-Score	
		Genuine	Fake	Genuine	Fake
Logistic Regression	88.20%	87.60%	88.90 %	88.30%	88.00 %
Random Forest	93.40%	92.80%	94.10 %	93.40%	93.50 %
XGBoost	91.70%	91.10%	92.40 %	91.60%	91.70 %

Table 2: Model Performance on Social Relationship Features

Model	Accuracy	Precision		F1-Score	
		Genuine	Fake	Real	Fake
Logistic Regression	86.40%	85.90%	87.10%	86.60 %	86.30 %
Random Forest	92.10%	91.60%	92.80%	92.10 %	92.00 %
XG Boost	90.30%	89.90%	90.80%	90.30 %	90.20 %

Analysis: This table shows that social relationship features provide strong predictive power for fake profile detection [3][4]. Random Forest achieved the best accuracy (92.10%) because it effectively captures non-linear patterns such as abnormal follower–following ratios commonly seen in fake accounts [11]. Logistic Regression gives stable results but with lower accuracy [12], while XGBoost also performs well with balanced precision and recall for both classes [10].

Table 3: Model Performance on Activity & Engagement Features

Model	Accuracy	Precision		F1-Score	
		Genuine	Fake	Genuine	Fake
Logistic Regression	84.70%	86.20%	82.90 %	83.20%	85.30 %
Random Forest	90.80%	91.40%	90.10 %	90.30%	91.20 %
XG Boost	88.60%	89.70%	87.40 %	88.00%	88.80 %

Analysis: Engagement and activity-based features showed moderate to strong predictive capability with some imbalance across classes [2][4]. Fake accounts typically display irregular activity patterns, sudden spikes in posting, or very low engagement rates, which increases recall for the fake class in most models [4]. Random Forest achieved the best overall performance (90.80% accuracy) with balanced precision and recall for both genuine and fake profiles [11]. Logistic Regression showed comparatively lower accuracy but still captured the fake activity patterns reasonably well [12], while XGBoost provided stable performance

across both classes **[10]**.

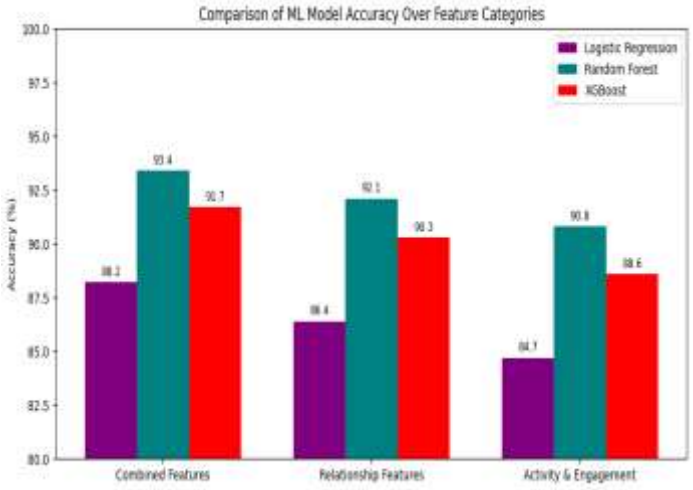

Figure 2: Comparison of the accuracy of Logistic Regression, Random Forest, and XGBoost over feature categories.

Key findings:

1. **FeatureCategoryEffectiveness:**Social relationship and behavioral features proved to be the most consistent indicators for detecting fake profiles. Features like follower–following ratio, engagement rate, and activity consistency provided strong discrimination, while basic profile attributes such as account age and verification status acted as supportive evidence for classification [3][4].
2. **Model Performance**: Random Forest achieved the best overall performance across all feature categories, showing strong accuracy and balanced results [11]. XGBoost also performed well with stable predictions [10], while Logistic Regression produced lower accuracy compared to ensemble models, indicating that complex fake profile patterns are better captured using tree-based approaches [11][12].
3. **Class-specific insights:** The precision/recall trade-offs revealed the following characteristics:

- Combined feature analysis gives the most balanced detection for both genuine and fake profiles [4][5].
- Relationship features reduce false classifications by capturing unrealistic connectivity patterns common in fake accounts [3][4].

4. Activity & engagement features improve detection of fake profiles that show abnormal posting or interaction behavior [2][4].
5. Practical Implications: A layered detection strategy is recommended for real-world use: combined features for primary screening, relationship-based indicators for verifying suspicious profiles, and activity/engagement analysis for identifying borderline cases where profiles appear realistic but behave abnormally [4].

3.3 System Implementation and Usability

- The Flask-based implementation of the Fake Profile Detection System enabled rapid development while maintaining a clear separation between the user interface, feature processing, and machine learning prediction modules [8][9]. The backend efficiently handles user inputs and executes feature extraction and classification with minimal delay, providing quick and reliable results [8]. The system design also supports easy integration of multiple machine learning models, ensuring that predictions can be generated smoothly for different feature categories [10][11][12].
- A clear verdict (Fake / Genuine) along with prediction confidence [4].
- A structured display of extracted profile and behavioral features [3][4].
- Visual summaries such as accuracy comparison charts across models and feature categories [9].
- A history section to track previous prediction results for reference [8].

These features enhance the system from a simple classifier into a practical support tool that provides both automated results and meaningful insights [4]. In addition, the modular architecture allows future improvements such as adding real-time profile monitoring, expanding feature sets, and updating machine learning models for better performance and adaptability [4][5].

3.4 Discussion of Limitations and Trade-offs

The major advantage of the proposed fake profile detection approach is its speed and scalability. Since the system relies on extracted profile and behavioral features, it can classify accounts quickly without requiring manual verification or time-consuming investigation [4][5]. This makes the approach suitable for handling large volumes of social networking accounts and performing automated screening [2][4].

However, the proposed approach also has certain limitations and trade-offs:

- Adaptive Fake Profiles: Advanced fake accounts may imitate genuine behavior by using realistic profile details, balanced follower ratios, and consistent activity patterns, which can reduce detection accuracy [1][2].
- Limited Behavioral Context: Since the model depends on available feature data, it may not fully capture complex interactions such as real-time

conversations, content originality, or coordinated group activities performed by fake profiles [2][4].

- Feature Dependency: Some features may perform weakly when considered alone, and better accuracy is achieved only when multiple feature categories are combined. This shows that feature interactions play an important role in reliable classification [4][5].

These limitations indicate that machine learning–based fake profile detection works best as part of a layered security strategy, where automated screening is supported by continuous feature updates, periodic model retraining, and additional platform-level verification mechanisms [2][4].

4. Conclusion

This project successfully designed, developed, and evaluated a complete Fake Profile Detection System that classifies social networking accounts as Genuine or Fake using machine learning techniques. The system focuses on extracting meaningful profile and behavioral indicators such as account age, followers count, following count, follower–following ratio, engagement rate, and activity consistency, which play a major role in identifying suspicious accounts. By converting these indicators into structured feature vectors, the system is able to perform automated and reliable fake profile detection with measurable performance.

A comparative analysis was carried out using three supervised learning models: Logistic Regression, Random Forest, and XGBoost. The results showed that Random Forest achieved the best overall accuracy and balanced prediction performance, followed by XGBoost, proving the advantage of ensemble models in capturing complex and non-linear fake profile behavior patterns. Logistic Regression performed as a baseline model but produced comparatively lower accuracy, indicating that advanced models are more suitable for real-world fake profile classification.

The complete system was implemented as a Flask-based web application, offering users a simple interface to analyze profiles and view prediction results clearly. It also supports features like prediction reporting, accuracy comparison, and history tracking, which improves usability and interpretability. Overall, this project provides a scalable and practical solution for strengthening trust, authenticity, and platform security by enabling efficient and automated detection of fake profiles on social networking platforms

5. Future Work

Future enhancements to overcome current limitations and improve the Fake Profile Detection System include:

1. **Hybrid Feature Analysis:** Combine profile attributes, relationship features, and activity/engagement features using feature fusion to improve detection accuracy and capture feature interactions [4][5].
2. **Real-time Behavioral Monitoring:**Integrate real-time tracking of user activity patterns such as sudden follow spikes, repeated actions, and abnormal engagement changes to detect advanced fake accounts faster [2][4].
3. **Deep Learning Integration:** Apply deep learning models such as ANN, LSTM, or Graph Neural Networks to learn complex behavior patterns and hidden relationships between users in social networks [2][4].
4. **Cloud Deployment and Scalability:** Deploy the system using cloud platforms with Docker/Kubernetes to support large-scale real-time detection and auto-scaling for high user traffic [4].
5. **Explainable AI (XAI):** Integrate SHAP or LIME to provide feature importance explanations for each prediction, increasing trust and transparency in the results [4].
6. **Robustness Against Adaptive Fake Profiles:** Improve model resilience by detecting adversarial strategies where fake accounts imitate genuine behavior, and update models through continuous retraining with new data [1][2].

6. References

[1] Ferrara, E., Varol, O., Davis, C., Menczer, F., & Flammini, A. (2016). The Rise of Social Bots. *Communications of the ACM*, 59(7), 96–104.

[2] Cresci, S. (2020). A Decade of Social Bot Detection. *Communications of the ACM*, 63(10), 72–83.

[3] Stringhini, G., Kruegel, C., & Vigna, G. (2010). Detecting Spammers on Social Networks. *Proceedings of ACSAC*.

[4] Yang, C., Harkreader, R., Zhang, J., Shin, S., & Gu, G. (2011). Analyzing Spammers' Social Networks for Fun and Profit. *Proceedings of WWW*.

[5] Ahmed, E., Abulaish, M., & Youssef, A. (2013). Detection of Online Fake Profiles Using Machine Learning. *ASONAM*.

[6] Cao, Q., Sirivianos, M., Yang, X., & Pregueiro, T. (2012). Aiding the Detection of Fake Accounts in Large Scale Social Online Services. *USENIX NSDI*.

[7] Viswanath, B., Bashir, M. A., Crovella, M., Guha, S., Gummadi, K. P., Krishnamurthy, B., & Mislove, A. (2014). Towards Detecting Anomalous User Behavior in Online Social Networks. *USENIX Security Symposium*.

[8] Kaggle. Fake Profile / Social Network User Dataset. Retrieved from *Kaggle Datasets Repository*.

[9] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: Machine Learning in Python. *JMLR*, 12, 2825–2830.

[10] Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. *ACM SIGKDD*.

[11] Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1), 5–32.

[12] Hosmer, D. W., Lemeshow, S., & Sturdivant, R. X. (2013). *Applied Logistic Regression* (3rd ed.). Wiley.

[13] Cortes, C., & Vapnik, V. (1995). Support-Vector Networks. *Machine Learning*, 20, 273–297.

[14] Freund, Y., & Schapire, R. E. (1997). A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *Journal of Computer and System Sciences*, 55(1), 119–139.

[15] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.

[16] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.

[17] Alpaydin, E. (2020). *Introduction to Machine Learning* (4th ed.). MIT Press.

[18] Ngai, E. W. T., Hu, Y., Wong, Y. H., Chen, Y., & Sun, X. (2011). The Application of Data Mining Techniques in Financial Fraud Detection. *Decision Support Systems*, 50(3), 559–569.

[19] Shu, K., Wang, S., & Liu, H. (2017). Fake News Detection on Social Media: A Data Mining Perspective. *SIGKDD Explorations*, 19(1), 22–36.

[20] Zhou, X., & Zafarani, R. (2018). Fake News: A Survey of Research, Detection Methods, and Opportunities. *arXiv preprint arXiv:1812.00315*.

[21] Wu, L., & Liu, H. (2018). Tracing Fake-News Footprints: Characterizing Social Media Manipulation. *WSDM*.

[22] Ratkiewicz, J., Conover, M., Meiss, M., Gonçalves, B., Patil, S., Flammini, A., & Menczer, F. (2011). Detecting and Tracking Political Abuse in Social Media. *ICWSM*.

[23] Davis, C. A., Varol, O., Ferrara, E., Flammini, A., & Menczer, F. (2016). BotOrNot: A System to Evaluate Social Bots. *Proceedings of WWW Companion*.

[24] Varol, O., Ferrara, E., Davis, C., Menczer, F., & Flammini, A. (2017). Online Human-Bot Interactions: Detection, Estimation, and Characterization. *ICWSM*.

[25] Gilani, Z., Farahbakhsh, R., Tyson, G., Wang, L., & Crowcroft, J. (2017). Of Bots and Humans (On Twitter). *ASONAM*.

[26] Gupta, A., Lamba, H., Kumaraguru, P., & Joshi, A. (2013). Faking Sandy: Characterizing and Identifying Fake Images on Twitter during Hurricane Sandy. *WWW*.

[27] Benevenuto, F., Magno, G., Rodrigues, T., & Almeida, V. (2010). Detecting Spammers on Twitter. *CEAS*.

[28] Lee, K., Caverlee, J., & Webb, S. (2010). Uncovering Social Spammers: Social Honeypots + Machine Learning. *SIGIR*.

[29] Gupta, M., Gao, J., Aggarwal, C. C., & Han, J. (2014). Outlier Detection for Temporal Data: A Survey. *IEEE TKDE*, 26(9), 2250–2267.

[30] Akoglu, L., Tong, H., & Koutra, D. (2015). Graph Based Anomaly Detection and Description: A Survey. *Data Mining and Knowledge Discovery*, 29, 626–688.