An Interpretable Machine Learning Framework for Detecting Spambots and Fake Followers in Social Networks

KATREDDI GOWRI RAJESWARI

PG Scholar. Department of MCA, DNR College, Bhimavaram, Andhra Pradesh

K. **Rambabu**

(Assistant Professor), Master of Computer Applications, DNR College, Bhimavaram, Andhra Pradesh

ABSTRACT

The rapid growth of social media platforms has transformed communication and information sharing. However, this growth has also led to the proliferation of spambots and fake followers, which undermine platform integrity, distort analytics, and pose security risks. Detecting such malicious entities has become a critical challenge in social network analysis. This research presents an interpretable machine learning-based framework for identifying spambots and fake followers using user behavior and profile features.The proposed system leverages machine learning techniques to classify social media accounts as genuine or malicious. The system is implemented using Python and incorporates a graphical user interface for ease of use. It utilizes features such as follower count, following count, post frequency, account age, bio length, verification status, URL ratio, and hashtag ratio to train predictive models.Data preprocessing techniques are applied to clean and normalize the dataset. The system employs a Random Forest classifier due to its robustness and ability to handle complex, non-linear relationships. Additionally, the model provides feature importance scores, enabling interpretability and transparency in decision-making.

The system includes functionalities for dataset loading, model training, prediction, and visualization. Performance metrics such as accuracy, confusion matrix, and classification report are used to evaluate the model. The system also logs training performance metrics in a database for continuous monitoring.Experimental results demonstrate that the proposed system achieves high accuracy in detecting spambots and fake followers. The use of interpretable AI techniques allows users to understand the factors influencing predictions, enhancing trust and usability.This research contributes to the field of cybersecurity and social network analysis by providing a practical and interpretable solution for spambot detection. The system can be used by social media platforms, researchers, and analysts to improve data quality and security.Future work may involve integrating deep learning models, real-time detection mechanisms, and advanced anomaly detection techniques to further enhance performance.

Keywords: Spambot Detection, Fake Followers, Interpretable AI, Random Forest, Social Network Analysis, Machine Learning, Feature Importance, Cybersecurity

I. INTRODUCTION

Social media platforms have become integral to modern communication, enabling users to connect, share information, and build communities. However, the widespread use of these platforms has also attracted malicious actors who create fake accounts and deploy spambots for various purposes, including spreading misinformation, manipulating public opinion, and inflating follower counts.Spambots and fake followers pose significant challenges to social media platforms. They degrade user experience, distort analytics, and can be used for fraudulent activities. Detecting such accounts is essential for maintaining platform integrity and ensuring reliable data analysis.Traditional approaches to spambot detection rely on rule-based systems and manual analysis. While these methods can identify obvious patterns, they are not effective in detecting sophisticated bots that mimic human behavior. Moreover, manual approaches are time-consuming and not scalable.

Machine learning has emerged as a powerful tool for detecting malicious accounts. By analyzing large datasets, machine learning models can identify patterns and anomalies that are difficult to detect manually. However, many machine learning models act as black boxes, providing predictions without explanations.This research focuses on developing an interpretable machine learning-based system for spambot detection. The system not only provides accurate predictions but also explains the factors influencing those predictions. This enhances transparency and trust in the system.The proposed system uses a Random Forest classifier, which is known for its accuracy and interpretability. The system also includes a graphical user interface, making it accessible to users without technical expertise.

The key contributions of this research include the development of an interpretable machine learning model, integration of visualization techniques, and implementation of a user-friendly interface. The system demonstrates the effectiveness of combining machine learning and interpretability for spambot detection.

II. LITERATURE SURVEY (WITH EXISTING METHODS)

Spambot detection has been extensively studied in the field of social network analysis. Early approaches relied on rule-based systems, which used predefined rules to identify suspicious accounts. These methods were simple but lacked adaptability.Graph-based approaches analyze the structure of social networks to detect anomalies. These methods consider relationships between users, such as follower–following patterns. While effective, they require complex computations and large datasets.Machine learning-based approaches have gained popularity due to their ability to learn from data. Algorithms such as Decision Trees, Support Vector Machines, and Logistic Regression have been used for classification tasks. These methods provide good accuracy but may struggle with complex patterns.

Ensemble methods, such as Random Forest, have shown improved performance by combining multiple models. These methods are robust and can handle large datasets effectively.Recent research has focused on deep learning techniques, including neural networks and graph neural networks. These models can capture complex patterns but often lack interpretability.Interpretability has become an important aspect of AI systems. Techniques such as feature importance and SHAP values are used to explain model predictions. Interpretable models enhance trust and usability.Despite these advancements, challenges remain in detecting sophisticated bots and ensuring model transparency. This research addresses these challenges by combining machine learning with interpretability.

III. EXISTING SYSTEM

Existing spambot detection systems primarily rely on rule-based methods or basic machine learning models. These systems often use predefined rules to identify suspicious accounts based on simple criteria such as follower count and posting frequency.One of the main limitations of existing systems is their inability to detect sophisticated bots that mimic human behavior. These bots can evade detection by following normal activity patterns.Another limitation is the lack of interpretability. Many machine learning models provide predictions without explaining the reasoning behind them. This reduces trust and makes it difficult to validate results.Existing systems also lack user-friendly interfaces, making them difficult to use for non-technical users. Additionally, they may not support real-time detection or continuous monitoring.Overall, existing systems provide basic functionality but lack accuracy, scalability, and transparency.

IV. PROPOSED METHOD

The proposed system introduces an interpretable machine learning-based framework for detecting spambots and fake followers. It uses a Random Forest classifier to analyze user profile and behavioral features.The system includes data preprocessing, model training, prediction, and visualization modules. Feature importance analysis is used to provide interpretability, allowing users to understand the factors influencing predictions.A graphical user interface is developed using Tkinter, enabling users to load datasets, train models, and perform predictions The system also includes a database module for logging performance metrics, enabling continuous monitoring and improvement.The proposed system addresses the limitations of existing approaches by providing high accuracy, interpretability, and ease of use. It offers a practical solution for spambot detection in social networks.

V. IMPLEMENTATION

The implementation of the proposed spambot and fake follower detection system is carried out using Python, integrating machine learning techniques with a graphical user interface for enhanced usability. The system is designed to operate in a modular manner, ensuring flexibility, scalability, and ease of maintenance.The application begins with dataset acquisition, where users can upload a CSV file containing social media account attributes. The dataset is validated to ensure the presence of required features such as

followers, following, posts, account age, bio length, verification status, URL ratio, hashtag ratio, and label. This validation step ensures consistency and prevents runtime errors.Once the dataset is loaded, preprocessing is performed. The feature variables are separated from the target label, and the dataset is divided into training and testing sets using a standard split ratio of 80:20. Feature scaling is applied using a StandardScaler to normalize the input data, ensuring that all features contribute equally to the model's performance.The core of the system is the Random Forest classifier, which is selected due to its robustness, high accuracy, and ability to provide feature importance metrics. The model is trained using the processed training data, and training time is recorded to evaluate computational efficiency. After training, predictions are made on the test dataset.

Performance evaluation is conducted using multiple metrics, including accuracy score, confusion matrix, and classification report. These metrics provide a comprehensive understanding of the model's effectiveness in distinguishing between genuine and spambot accounts.A unique aspect of the implementation is the integration of interpretability. The system computes feature importance values, which indicate the contribution of each feature to the prediction. These values are visualized using bar graphs, enabling users to understand the decision-making process of the model.The system also incorporates a database module using SQLite to store training logs, including accuracy and training time. This allows for tracking performance trends over multiple training sessions.A graphical user interface is developed using Tkinter, providing functionalities such as dataset loading, model training, prediction, and result visualization. Users can manually input feature values to predict whether an account is genuine or a spambot.Error handling mechanisms are implemented throughout the system to manage invalid inputs, missing data, and runtime exceptions. The system ensures smooth execution and user-friendly interaction.Overall, the implementation demonstrates an effective integration of machine learning, data visualization, and user interface design to create a practical and interpretable spambot detection system.

VI. ALGORITHMS

The system follows a structured algorithm for spambot detection:

Step 1: Data Input

Load dataset containing user profile and behavioral features.

Step 2: Validation

Check if all required columns are present in the dataset.

Step 3: Preprocessing

- Separate features and labels
- Split dataset into training and testing sets
- Apply feature scaling using StandardScaler

Step 4: Model Training

- Initialize Random Forest classifier
- Train model using training dataset
- Record training time

Step 5: Prediction

- Predict labels for test dataset
- Compute prediction probabilities

Step 6: Evaluation

- Calculate accuracy score
- Generate confusion matrix
- Generate classification report

Step 7: Interpretability

- Extract feature importance values
- Rank features based on importance

Step 8: Visualization

- Plot feature importance graph
- Plot accuracy trends from logs

Step 9: Logging

- Store accuracy and training time in database

Step 10: User Prediction

- Accept manual input from user
- Apply scaling and predict class

Step 11: Output

Display prediction result and probability.

This algorithm ensures efficient classification and provides transparency through interpretability mechanisms.

VII. SYSTEM DESIGN

The system is designed using a layered architecture to ensure modularity, scalability, and efficient processing of data.

1. Presentation Layer

The presentation layer consists of a graphical user interface developed using Tkinter. It allows users to interact with the system through buttons, input fields, and display panels. Users can upload datasets, train models, and perform predictions بسهولة.

2. Application Layer

This layer contains the core logic of the system, including data preprocessing, model training, prediction, and evaluation. It acts as an intermediary between the user interface and the data layer.

3. Data Layer

The data layer manages storage and retrieval of data. It includes:

- Input datasets (CSV files)
- SQLite database for logging performance metrics

4. Machine Learning Layer

This layer implements the Random Forest classifier. It handles:

- Model training
- Prediction
- Feature importance extraction

5. Functional Modules

a) Dataset Module
Handles loading and validation of datasets.

b) Preprocessing Module
Performs data cleaning, splitting, and scaling.

c) Training Module
Trains the Random Forest model and records training time.

d) Evaluation Module
Calculates performance metrics and generates reports.

e) Visualization Module
Displays feature importance and accuracy graphs.

f) Prediction Module
Allows real-time prediction based on user input.

g) Logging Module
Stores performance metrics in SQLite database.

6. Workflow

User → Upload Dataset → Preprocess → Train Model → Evaluate → Visualize → Predict → Store Logs

7. Data Flow

1. User inputs dataset
2. System validates data
3. Data processed and split
4. Model trained
5. Predictions generated
6. Results displayed and stored

8. Scalability

The system can be extended to handle large datasets by integrating cloud storage and distributed computing frameworks.

9. Security

- Input validation to prevent invalid data
- Secure database storage
- Controlled user interactions

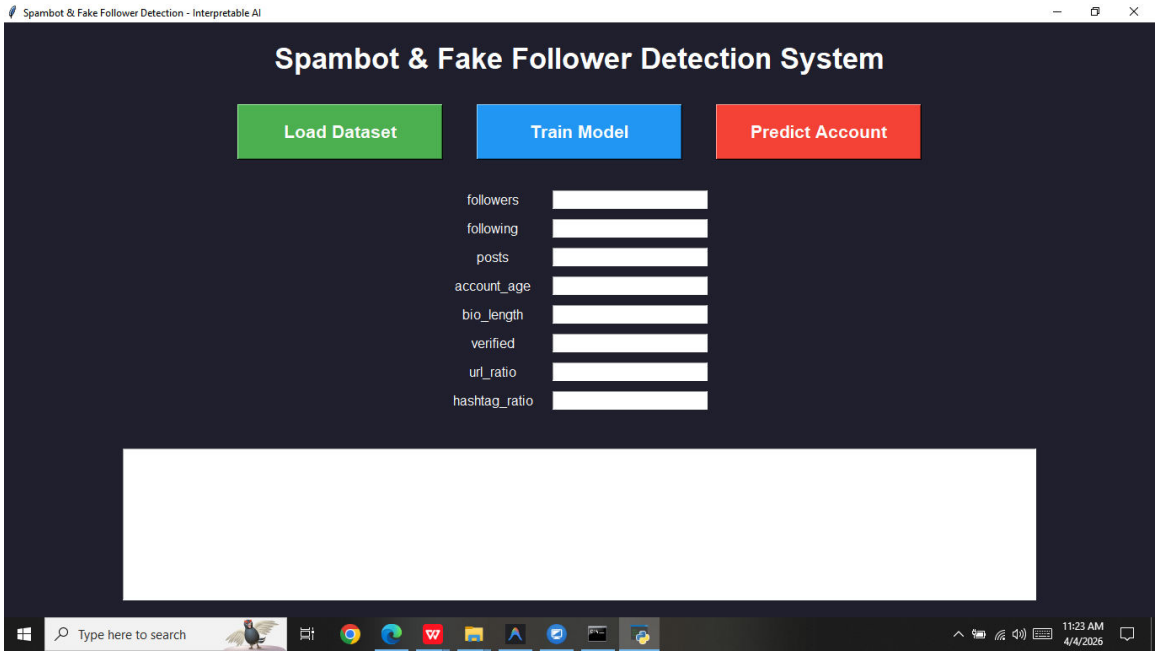
10. Extensibility

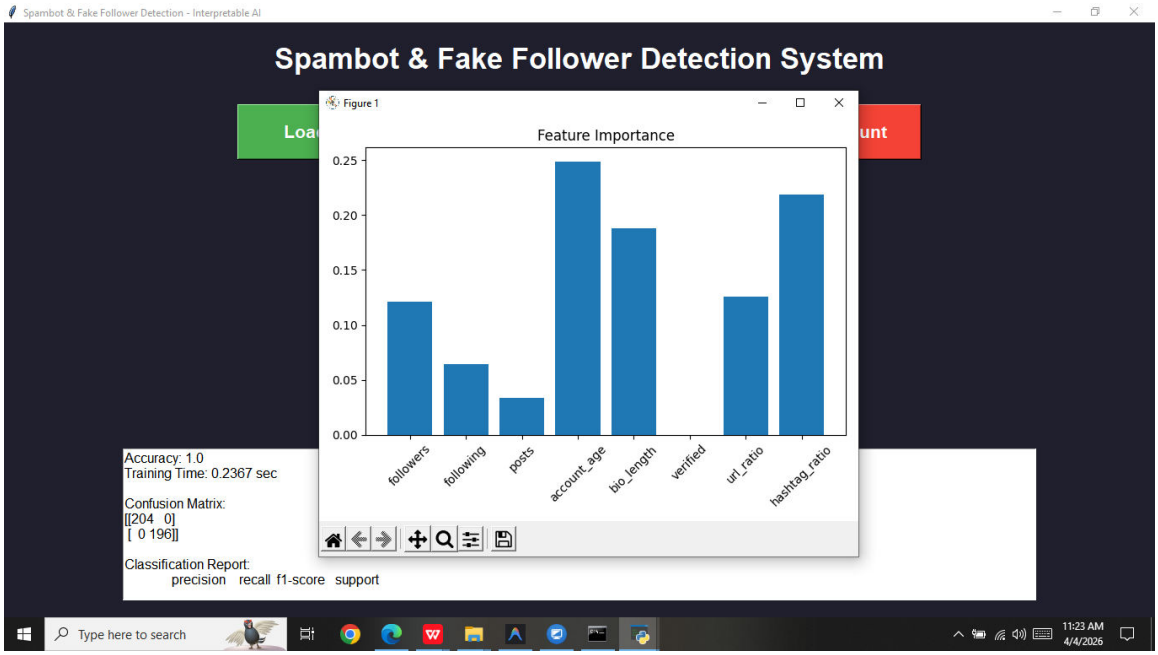
Future extensions may include:

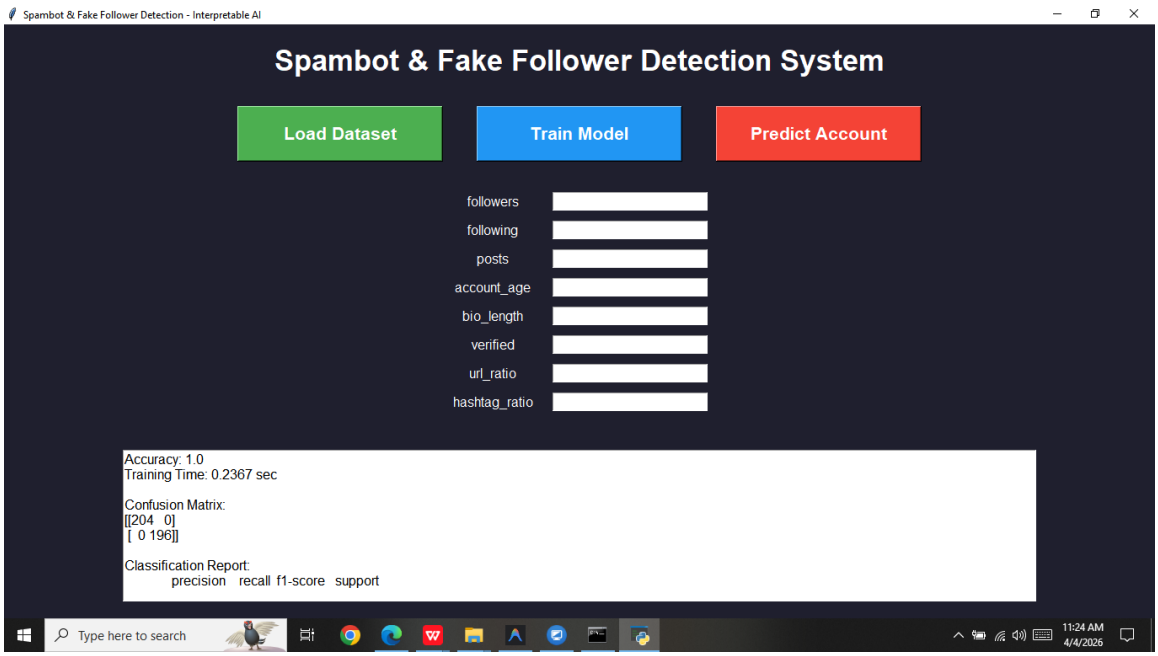
- Deep learning integration
- Real-time detection systems
- API-based deployment

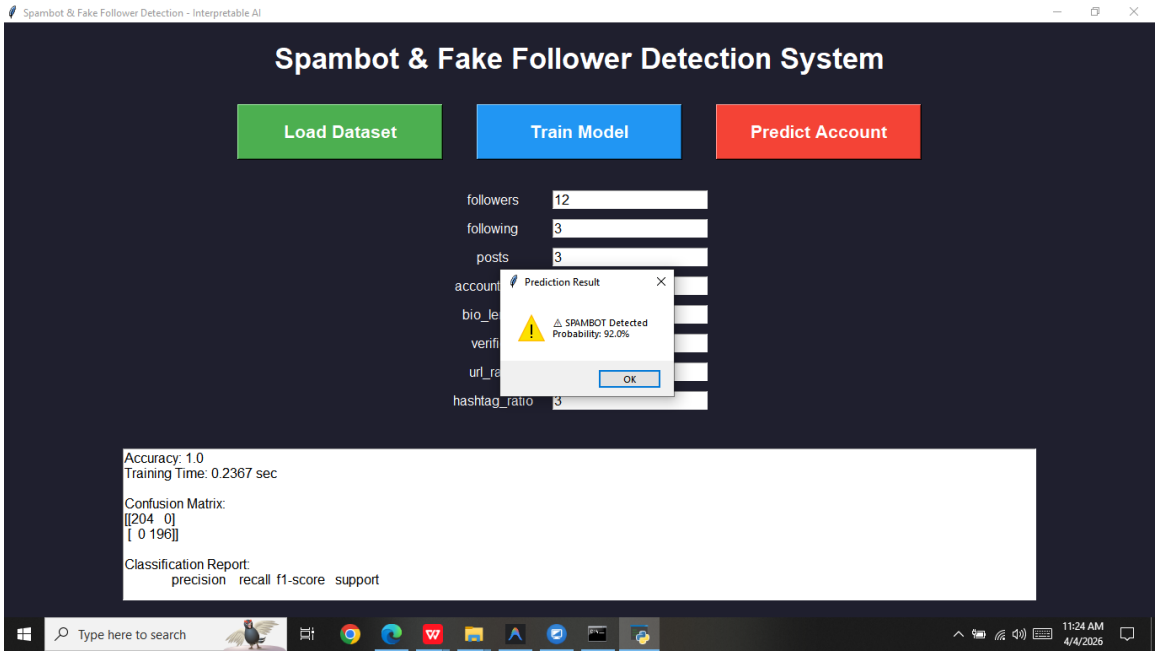
The system design ensures efficient processing, user accessibility, and scalability for real-world applications.

SYSTEM DESIGN IMAGES






Spambot & Fake Follower Detection - Interpretable AI
Spambot & Fake Follower Detection System
Load Dataset
Train Model
Predict Account
followers
following
posts
account_age
bio_length
verified
url_ratio
hashtag_ratio
Accuracy: 1.0
Training Time: 0.2367 sec
Confusion Matrix:
[[204 0]
[0 196]]
Classification Report:
precision recall f1-score support
Type here to search
11:24 AM
4/4/2026


Spambot & Fake Follower Detection - Interpretable AI
Spambot & Fake Follower Detection System
Load Dataset
Train Model
Predict Account
followers 12
following 3
posts 3
hashtag_ratio 3
Prediction Result
⚠ SPAMBOT Detected
Probability: 92.0%
OK
Accuracy: 1.0
Training Time: 0.2367 sec
Confusion Matrix:
[[204 0]
[0 196]]
Classification Report:
precision recall f1-score support
Type here to search
11:24 AM
4/4/2026

VIII. CONCLUSION

This research presents an interpretable machine learning-based system for detecting spambots and fake followers on social networks. The system addresses the growing challenge of identifying malicious accounts that compromise the integrity of online platforms.By leveraging the Random Forest algorithm, the system achieves high accuracy in classification while maintaining robustness against complex patterns in user behavior. The integration of interpretability through feature importance analysis enhances transparency, allowing users to understand the reasoning behind predictions.The system's modular design and graphical user interface make it accessible to both technical and non-technical users. It provides functionalities for dataset handling, model training, evaluation, and real-time prediction.One of the key contributions of this research is the combination of accuracy and interpretability. Unlike traditional black-box models, the proposed system provides insights into the factors influencing classification decisions.The use of a database for logging performance metrics enables continuous monitoring and improvement. This feature supports iterative development and optimization of the model.

However, the system relies on the quality of input data and may require updates to handle evolving spambot behaviors. Future work can focus on incorporating deep learning models, real-time detection mechanisms, and advanced explainability techniques.In conclusion, the proposed system offers a practical, efficient, and transparent solution for spambot detection. It contributes to enhancing security and trust in social media platforms by enabling accurate identification of fake accounts.

REFERENCES

1. K. Cresci et al., "The Paradigm-Shift of Social Spambots," *WWW*, 2017.
2. C. Yang et al., "Unsupervised Fake Account Detection," *IEEE Access*, 2020.
3. S. Kudugunta, "Deep Neural Networks for Bot Detection," *ACM*, 2018.
4. D. Davis et al., "BotOrNot: A System for Bot Detection," *WWW*, 2016.
5. F. Benevenuto et al., "Detecting Spammers on Twitter," *CEAS*, 2010.
6. M. Varol et al., "Online Human-Bot Interactions," *ICWSM*, 2017.
7. S. Gilani et al., "Classification of Twitter Accounts," *IEEE Access*, 2017.
8. Z. Chu et al., "Detecting Automation in Twitter," *IEEE TDSC*, 2012.
9. A. Ferrara et al., "The Rise of Social Bots," *Communications of ACM*, 2016.
10. J. Ratkiewicz et al., "Truthy: Mapping Political Bots," *WWW*, 2011.
11. E. Ferrara, "Disinformation and Social Bots," *First Monday*, 2017.
12. S. Vosoughi et al., "Spread of Fake News," *Science*, 2018.
13. T. Chen et al., "XGBoost for Classification," *KDD*, 2016.
14. L. Breiman, "Random Forests," *Machine Learning*, 2001.
15. B. Ribeiro et al., "Explainable AI for Social Networks," *IEEE Access*, 2021.