Towards Trustworthy Keylogger detection: A Comprehensive Analysis of Ensemble Techniques and Feature Selections through Explainable AI

Monirul Islam Mahmud
Dept. of Computer & Information Science
Fordham University
New York, United States
mim9@fordham.edu
ORCID: 0009-0005-2812-8553

***Abstract*—Keylogger detection involves monitoring for unusual system behaviors such as delays between typing and character display, analyzing network traffic patterns for data exfiltration. In this study, we provide a comprehensive analysis for keylogger detection with traditional machine learning models (SVC, Random Forest, Decision Tree, XGBoost, AdaBoost) and advanced ensemble methods including stacking, blending and voting. Moreover, feature selection approaches such as information gain, gain ratio, correlation coefficient and Fisher score are thoroughly assessed to improve predictive performance and lower computational complexity. The Keylogger Detection dataset [1] from publicly available Kaggle website will be used in this project. In addition to accuracy-based classification, this study implements the approach for model interpretation using Explainable AI (XAI) techniques — namely SHAP (Global) and LIME (Local) — to deliver finer explanations for how much each feature contributes in assisting or hindering the detection process. It is intended to provide robust and interpretable detection models which can be suitable for real-world Cybersecurity applications. To evaluate the models result, we will use AUC score, sensitivity, Specificity, Accuracy and F1 score. Insights from experimental evaluations in conjunction with Machine Learning models, such as the ensemble method with default subset of features from both methods that offers high accuracy while allowing transparency in classification.**

***Index Terms*—Keylogger, Ensemble, Feature Selection, Explainable AI**

I. Introduction

One type of malware termed as keylogger is used to record user keystroke attempts. Cybercriminals typically use keyloggers to watch the target's keystrokes and steal data from their computer or other computing devices. In general, there are two types of keyloggers. The first kind of keyloggers is inserted into the user device to steal the information through the keystrokes. In that instance, the primary focus of hackers was on acquiring the login credentials, such as user IDs and passwords, that users enter to access any website or application. The keyloggers capture each keystroke and transmit it to the fraudsters. Cybercriminals utilize the keylogger in this procedure to obtain private data. Malware often operates in secret, and the user won't realize they have an infection until the computer has been harmed [2]. Cybercriminals can use ransomware to encrypt a computer and demand money to unlock it. If the user disagrees with their specifications, they have the power to delete the data or harm the hardware [3]. Programs known as keyloggers are placed on computers and record each keystroke. As a result, the numerous websites a user visits provide credentials, including usernames and passwords, to hackers. It is advised to avoid using public devices for sensitive transactions or to enter usernames and passwords because there is a genuine chance that keyloggers could be installed on them [4]. There are different approaches to detect Keylogger using Machine Learning, but little research has been done with comparison to different feature selection and Ensemble approached with Explainable AI to effeciently detect Keylogger with less computational power. To fill the research gaps, our proposed novel approaches are given below:

- We will compare traditional machine learning models and advanced ensemble models including stacking, blending and voting.
- Feature selection approaches such as information gain, gain ratio, correlation coefficient and Fisher score will be assessed to improve predictive performance and lower computational complexity.
- Explainable AI techniques - SHAP and LIME will be visualized to make the models more interpretable and trustworthy.

II. Related Works

Keyloggers are incredibly harmful tools that monitor all of our computer activity. Every key the user pushes can be recorded by the keylogger, which can then store the data in a log file and email the file to the designated IP address. The financial system, which is used for everyday business operations, is seriously at risk from it. The types, functions, and characteristics of several keyloggers are described in detail [5]. Pillai showed how to detect keyloggers installed

Identify applicable funding agency here. If none, delete this.

on a computer using a modified SVM-based architecture. Eight open source keyloggers were installed on their system [6]. Brown introduced well-known techniques for Android keylogger classification, including XOR, GEFeS, and SDM [16]. Wen L. et al. introduced the unsupervised dimensionality reduction approach and the supervised learning classifier SVM and PCA-RELIEF [7]. A novel data-collecting method based on a unified activity list was used to obtain the novel dataset shown in [8], which was collected in a realistic environment. The overall average accuracy was 79% for the binary class, which has two target variables, and 77% for the multi-class, which has more than two target variables. The breadth of the monitoring included all dataset aspects, including keyloggers, OS activity, microphone access, phone calls, and social media access. According to the findings of [9], random forest performs better in malware detection than deep neural network models. Random Forest attained the maximum accuracy at 99.78%. Using four classifiers—ID3, K-Nearest Neighbors, Decision Tree, C4.5 Decision Tree, and linear SVM, the authors of [10] demonstrated a supervised classification method for identifying Android malware (SVM).

III. Methodology

In this study, we propose a comprehensive methodology for detecting keyloggers using machine learning techniques. These include data acquisition, data preprocessing, feature selection, Models and Exaplianble AI analysis. For this purpose, we use a dataset called the "Keylogger Detection" [1] from Kaggle, consisting of system process data characterized as benign or malicious.

A. Data Collection and Preprocessing

First, the dataset is checked for completeness and missing or inconsistent data are properly dealt with. These categorical features are one-hot encoding and numerical ones are scaled to provide consistency between scales (Normalization). We will use SMOTE (Synthetic Minority Over-sampling Technique) to balance class imbalance by generating synthetic samples of the minority classes in the dataset.

B. Feature Selection

We utilize filter techniques such as Information Gain, Gain Ratio, Correlation Coefficient, and Fisher Score for feature selection to determine precise and informative feature subset. Because these methods assess the usefulness of individual features on the target variable without relying on any learning algorithm, we can rank features and retain the top contributors for our keylogger-detection effort.

C. Model Development

We trained and evaluated traditional machine learning models such as Decision Trees, Support Vector Machines, Random Forest, k-Nearest Neighbours as well as ensemble approaches: Stacking, Blending and Voting classifiers. Each preprocessed and balanced dataset is used to train the models and then hyperparameters are adjusted using grid search and cross-validation to improve the performance.

D. Model Evaluation

Evaluating each model with Accuracy, Sensitivity, Specificity, F1-Score, and Area Under the Receiver Operating Characteristic Curve (ROC-AUC) These metrics encompass a broad range of how effectively the models detect the keyloggers, with particular weight given to the minority class (benign processes).

E. Explainable AI (XAI)

As such, we make use of Explainable AI methods (SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-agnostic Explanations)) to interpret the predictions and know the contribution of each feature. Through these techniques, we can understand how specific features of the samples are affecting the output of the models leading to an improved trust and transparency of the detection.

References

[1] "Keylogger Detection," Kaggle, Sep. 17, 2021. https://www.kaggle.com/datasets/subhajournal/keylogger-detection.

[2] Javaheri D; Hosseinzadeh M; Rahmani AM. (2018). Detection and elimination of spyware and ransomware by intercepting kernel-level system routines. IEEE Access. 78321-32.

[3] Oz H; Aris A; Levi A; Uluagac AS. (2022,). A survey on ransomware: Evolution, taxonomy, and defence solutions. ACM Computing Surveys (CSUR). 1-37.

[4] Thakur KK; Nair NR; Sharma M. (2022). Keylogger: A Boon or a Bane. Trinity Journal of Management; IT & Media (TJMITM). 145-53.

[5] Ahmed YA; Maarof MA; Hassan FM; Abshir MM. (2014). Survey of Keylogger technologies. International journal of computer science and telecommunications. 5(2).

[6] Pillai D; Siddavatam I. (2019). A modified framework to detect keyloggers using machine learning algorithm. International Journal of Information Technology. 707-12.

[7] Wen L; Yu H. (2017). An Android malware detection system based on machine learning. AIP conference proceedings. (Vol. 1864, No. 1, p. 020136).

[8] Qabalin MK; Naser M; Alkasassbeh M. (2022). Android Spyware Detection Using Machine Learning: A Novel Dataset. Sensors. 22(15):5765.

[9] Rathore H; Agarwal S; Sahay SK; Sewak M. (2018). Malware detection using machine learning and deep learning. International Conference on Big Data Analytics. pp. 402-411.

[10] Aafer Y; Du W; Yin H. (2013). Droidapiminer: Mining API-level features for robust malware detection in Android. International conference on security and privacy in communication systems. pp. 86-103.