International Journal of Innovative Research and Scientific Studies, 8(1) 2025, pages: 2295-2304

IJIRSS
International Journal of Innovative Research & Scientific Studies

ISSN: 2617-6548

URL: www.ijirss.com

AI-enhanced cybersecurity: Machine learning classification application for APT malware attribution

Grozdan Hristov

Department of Computer Systems, Faculty of Computer Systems and Technologies Technical University of Sofia 8 Kliment Ohridski blvd., 1000 Sofia, Bulgaria.

(*Email: grozdan.hristov@dir.bg*)

Abstract

As technology becomes ever more integrated into daily tasks, the possibilities for conducting attacks against it increase as well. This gives rise to a number of challenges in the cybersecurity and technological fields. One such challenge is malware attribution, especially when it comes to determining the source and related threat actor of complex assaults. This article proposes a new machine learning-based method for Advanced Persistent Threat (APT) attribution that uses a dual-classifier system to predict the malware sample's nation of origin as well as the APT organization that is responsible for it. For the purpose of the research, the chosen dataset consists of roughly 3,500 tagged state-sponsored malware samples gathered from a variety of threat intelligence sources, containing information on malware hash values, malware family, connected country, etc. The model leverages static features extracted from the malware, including cryptographic hash values (MD5, SHA1, SHA256) and malware family labels, to build robust Random Forest classifiers. The choice of static analysis allows for efficient and scalable feature extraction, making the approach well-suited for large-scale datasets and real-time applications. The experimental results show an achievement for APT accuracy reaching 100% or very close to 100%, while the country accuracy was around 70%.

Keywords: Artificial intelligence, Attribution, APT, Cybersecurity, Machine learning, Malware, Random forest.

DOI: 10.53894/ijirss.v8i1.4955
Funding: This study received no specific financial support.
History: Received: 9 January 2025 / **Revised:** 10 February 2025 / **Accepted:** 17 February 2025 / **Published:** 26 February 2025

Competing Interests: The author declares that there are no conflicts of interests regarding the publication of this paper.
Transparency: The author confirms that the manuscript is an honest, accurate, and transparent account of the study; that no vital features of the study have been omitted; and that any discrepancies from the study as planned have been explained. This study followed all ethical practices during writing.
Publisher: Innovative Research Publishing

1. Introduction

With the increasing ease of access to artificial intelligence models and application and their growing popularity people have been finding new and innovative ways to use them and integrated them in their everyday activities. According to a publication from the European parliament in 2020 use cases range from online shopping and advertisement to machine translation and personal assistants [1]. And as the usage and interaction between AI and people increases something

interesting has been observed – people are starting to form relationships or invest a higher emotional percentage with AI systems [2]. Sadly, not every use of these amazing new systems and application is with good intent.

Artificial intelligence can and already is used in the field of cybersecurity both offensively and defensively. Uses of AI techniques in offensive operations can be quite broad. They can be used in social engineering attacks, phishing attacks, disinformation spreading, leveraging deepfakes, malware creating and enhancement, etc. [3]. This makes it very difficult to develop a one hit all solution and only emphasizes the statement about the constant race between sword and shield development.

The increased everyday use of AI will lead to many opportunities and ease of life, but like all things it will create different problems, one of which will be the increase of cyberattacks and probability of using high tech solutions with malicious intend other than a direct cyberattack, both by soloing cybercriminals and larger entities like hacker groups, government agencies and whole countries.

The aim of this paper is to propose a method which to help in the defensive cyber operations, specifically malware attribution to a specific actor. This way an opportunity would arise which if leveraged correctly could help in trend analysis and be of use in system hardening and prevention of malicious attacks to software products.

The paper is structured as follows: Section 2 describes the different uses of AI in the field of cybersecurity. Section 3 focuses on the work methodology and specifics used in the creation of the proposed method. Section 4 describes the system architecture and workflow. Section 5 analyses the test result. Section 6 discusses the strengths and limitations of the proposed solution. Section 7 is a conclusion.

2. AI Algorithms in the Field of Cybersecurity

As broad as the fields of artificial intelligence and cybersecurity are an interesting merging between the two can be observed. If, for example, the implementation and use of antivirus software is examined in most case one would find some sort of algorithm or method, that utilizes principles from AI. Currently, heuristic, data mining, agent, and artificial neural network techniques are some of the main artificial intelligence methods used in antivirus detection. The usage of these principles in antivirus detection aside from enhancing the effectiveness of antivirus detection systems is thought to encouraging the development of novel artificial intelligence algorithms, with which to combat the ever growing and increasing in complexity threat of malicious programs [4]. Some of the defensive advantages include automatic threat response and zero-day attack mitigation, with the latter being highly valuable when it comes to assistance in dealing with undocumented vulnerabilities and their exploitation [5].

Now logically arises the question “Are AI principles only used for defensive purposes?” The short answer is sadly, no. Artificial intelligence (AI) methods are unavoidably being used by cybercriminals to avoid detection and inflict more harm in cyberspace. What is referred to be "offensive AI" would enable cybercriminals to launch targeted attacks at previously unheard-of speed and scale while eluding conventional, rule-based detection methods [6]. Thus, the future of cybersecurity will be impacted by the emergence of a new generation of cybercriminals who are cunning and covert [7].

2.1. Defensive

In a bit more detailed view defensive use of AI in cybersecurity could range from data analysis to implementation of sophisticated AI methods in specialized defense software and hardware solutions. These days, it is impossible to anticipate an effective defense with just human labor without the aid of automated tools, given the pace of procedures and volume of data utilized in cyber security. However, it is challenging to create software using traditional fixed methods for effective security against constantly developing network threats. Artificial intelligence techniques that offer adaptability and learning potential can be used to do this. There is a good chance that enhanced defense system intelligence will lead to the development of cyber defense capabilities. It is evident from the real-world issues with cyber defense that many of these issues can only be effectively resolved by applying artificial intelligence techniques [8]. One field that is benefiting, at least to some degree, is Threat Intelligence. Conventional methods of analyzing and predicting cyber threats frequently can't keep up with the quickly changing strategies used by bad actors. Because of this, businesses are looking to AI-powered Threat Intelligence (AITI) as a viable way to improve and automate their cybersecurity initiatives. AITI gathers, examines, and interprets enormous volumes of data in real time by utilizing artificial intelligence algorithms. AITI gives enterprises the ability to more precisely identify risks and anticipate possible assaults before they happen by utilizing machine learning algorithms and advanced analytics. The creation of more potent mitigation techniques is made possible by this proactive strategy, which eventually strengthens an organization's overall cybersecurity posture [9]. Of course, this method is not fool proof, but it does shine a new and interesting aspect to examination of malicious programs.

2.2. Offensive

While it is true that technologies based on artificial intelligence (AI) are being actively employed for cyber protection, they are increasingly more implemented in offensive and malicious cyber methods. The utilized assaults range from creating adversarial traffic signals to compromising the safety of autonomous cars to manipulating medical photos using adversarial machine learning to falsely identify cancer [10]. And AI weaponization is not limited to cyber space only. The use of AI as a weapon, especially in nuclear, toxic, and chemical materials, is well-documented and has also been examined in relation to space exploration and climate change. In Burton and Soare [11] the authors stress that AI is a dual-use technology, and like other dual-use technologies, its characteristics dictate how likely it is to proliferate in the military or the civilian world. Currently, narrow AI, such as reactive and restricted memory AI, is the most common type used in the military industry. Numerous military platforms, systems, and procedures have integrated these types of technology. An

example is given in Burton and Soare [11] where AI is in use in logistics and training; augmented reality systems, for example, are already in use in the Royal New Zealand Navy for training engineers to work on naval platforms.

3. Cybersecurity, AI and APTs

Advanced Persistent Threats (APTs) pose a serious risk to enterprises worldwide in the dynamic field of cybersecurity. These sophisticated threats exploit system flaws over an extended period of time, making them challenging to identify and neutralize. AI has emerged as a crucial instrument for enhancing techniques for identifying and addressing APTs but also as a means to be utilized in the attack software and methods of various malicious threat actors [12].

But what exactly are APTs? As stated in Sternberg [13] APTs are typically a stealthy threat actor, usually a state or state-sponsored organization, that obtains unauthorized access to a computer network and stays hidden for a long time. An example of known APTs is APT29, which is attributed to Russia's Foreign Intelligence Service (SVR) [14]. They have operated since at least 2008, often targeting government networks in Europe and NATO member countries, research institutes, and think tanks [15]. Another example is the APT known as Mustang Panda, which is a threat actor for cyber espionage with headquarters in China that was initially discovered in 2017 but may have been active since at least 2014. Mustang Panda has targeted a variety of non-governmental groups, including government agencies, NGOs, religious institutions, and others in the United States, Europe, Mongolia, Myanmar, Pakistan, and Vietnam [16].

Of course, APTs are not limited to the aforementioned countries only. They have been observed using information and communication infrastructure in many different countries all over the world. Also, APT utilized a very broad range of tactics, techniques, and procedures (TTPs), which can range from basic denial of service (DoS) or distributed denial of service (DDoS) to building and exploiting custom malicious software. The specific of the broad range continues to stick when analyzing the potential goals for the attacks, again ranging from simple disruption of the normal workflow (which could become a major problem, depending on the victim and the attacked system) to very sophisticated intrusion, espionage and system destruction.

Generally, APTs have a 5 stage set up when conduction operations against an adversary, with Figure 1 showing a graphical representations of the steps: [17]

- Access gain – In order to introduce malware onto a target network, attackers typically enter through a network, an infected file, spam email, or an app weakness.
- Foothold establishment – In order to move about in systems without being noticed, cybercriminals install malware that makes it possible to create a network of backdoors and tunnels. To assist hackers, hide their tracks, the virus frequently uses strategies like altering code.
- Access deepening – Once inside, hackers employ methods like password cracking to obtain administrator privileges, which allows them to take control of further system components and obtain even more access.
- Lateral movement – With administrator privileges, hackers have more freedom to roam around the system. They can also try to get into other servers and other areas of the network that are protected.
- Persistence – Hackers may get any information they choose since they have complete knowledge of the system's weaknesses and how it operates while they are within it. Hackers have the option to try to continue this procedure forever or to stop it after achieving a certain objective. They frequently leave a back door open so they may later enter the system.

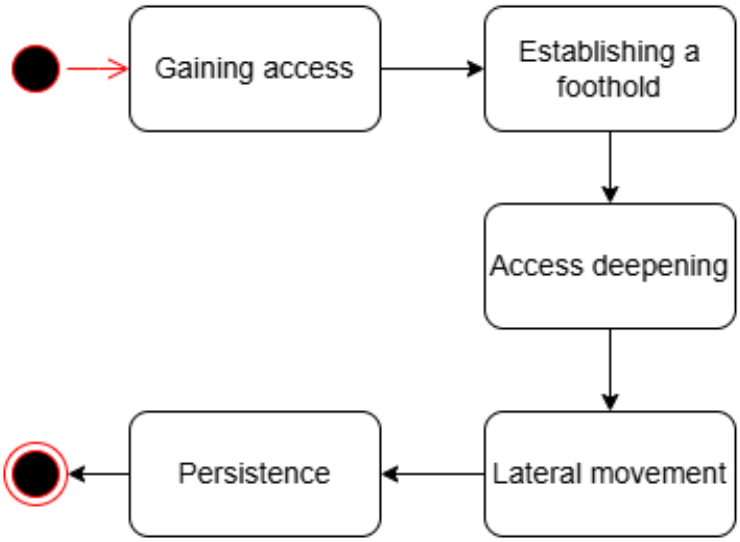

Figure 1.
Basic 5 stages of APT cyber operations.

With the growing reliance on technological solutions and their integration in state and human level everyday activities the blocking or exploitation of APTs arises exceptionally. One such case can be given in the usage of APTs and cyberattacks preluding and during the currently ongoing Russo-Ukrainian conflict. Numerous cyberattacks against Ukraine and a few against Russia were documented during the lead-up to and the actual Russian invasion of Ukraine. On January

14, 2022, more than a dozen Ukrainian government websites were taken down in the first significant cyberattack [18]. About 70 government websites, including those of the National Security and Defense Council (NSDC), the Cabinet of Ministers, and the Ministry of Foreign Affairs, were targeted, according to Ukrainian officials [19]. And it's not only the aforementioned conflict that is fought to some extend in the cyber domain except in the physical one. So, this raises the question are cyberattacks and APT usage useful when considering the whole scale and tendency of the armed conflicts? The short answer is probably no, since hackers on both sides of the Russo-Ukraine conflict have had difficulty responding to battlefield events, much less shaping them. Similar findings emerge from an examination of the dynamics of the Syrian conflict: the timing of cyberattacks is unrelated to ground combat. If we take the findings in Fortinet [20] it can be seen that it's stress that cyberattacks are not (as of yet) successful as instruments of coercion in war, which might have major ramifications. Even so the cyber front has proven to be quite tricky and demanding in resources, both human and technological, with little regard if the opposing sides are in an official war, have complicated diplomatic relationships or one side just wants to have an edge on the others. Also, the integration of AI methods in the TTPs of malicious actors has increased the skill level of APTs and proven that more and swift and precise research and development for counter measures needs to take place.

One such possible counter measure is the main topic of this paper – creating of an application using AI methods, which could help in the discovery and attribution of malicious APT software.

4. Work Methodology

4.1. Data Choice and Specifics

In any project or decision-making process, the initial phase of data gathering could be considered as one of the most, if not the most, important one. If for nothing else, then at least for the hard learned truth that not having enough data to make an informed decision is bad, but having too much data is also bad. The same principle is applied in the field of computing. For the creation and assessment of machine learning models, choosing the right dataset is essential, especially in cybersecurity applications where model performance is directly impacted by data quality. Data selection for this study was informed by a number of important factors, such as representativeness, feature variety, and compatibility with actual malware attribution problems.

For the purpose of the research the chosen dataset consists of roughly around 3500 tagged state-sponsored malware samples gathered from a variety of threat intelligence sources, such as commercial databases, open-source repositories, and internal cybersecurity logs. A variety of attributes are linked to each sample, including comprehensive details on its properties and possible place of origin. The primary features and labels selected for analysis include: [21]

- Family: The malware family, which indicates the type or lineage of the sample. This feature helps in identifying common patterns across similar types of malwares.
- Status: Indicates the activity status (e.g., active or dormant) of the malware sample, which can be indicative of recent threats or legacy attacks.
- Cryptographic Hashes (MD5, SHA1, SHA256): These features provide unique identifiers for each malware sample. While not directly indicative of origin, certain patterns in hash values can reflect specific compilation tools or libraries used by different threat actors.
- Country: The country label represents the presumed origin of the malware based on intelligence reports and prior attributions. This label is used as the target variable for the country classification task.
- APT Group: The APT group label identifies the specific threat actor or group linked to the malware. This serves as the target variable for the APT group classification task.

The chosen features and labels are intended to record a thorough profile of every malware sample, including contextual data (such family and status) as well as static attributes (like cryptographic hashes). To ensure that the proposed model can generalize across various threat actors, the dataset was carefully selected to include a wide range of samples ascribed to various nations and APT organizations. Samples from well-known APT organizations, such APT 28 (Russia), APT 10 (China), and Equation Group (USA), are included to improve the model's prediction power by teaching it unique patterns linked to particular threat actors. The unequal distribution of classes is one of the main issues with cybersecurity datasets. Because of their extensive activity, some nations (such China and Russia) and APT groups (including APT 28 and APT 10) are overrepresented, while others are underrepresented. To mitigate this as much as possible a combination of resampling and class weighting was used, which will be covered more broadly in the following subsections. Also, for training purposes the majority of the data was used (around 95%), and for testing a small random sample was selected from unseen data by the model.

4.2. Algorithm Selection

Another important phase during the conducted research and creation of a proposed solution was the artificial intelligence algorithm selection. The Random Forest classifier, an ensemble machine learning technique renowned for its resilience and adaptability in managing classification problems, especially in the fields of malware research and cybersecurity, is the main algorithm chosen for this investigation. Because of its capacity to handle intricate, high-dimensional datasets, control feature interactions, and provide probabilistic predictions—all of which are critical for making secure decisions regarding malware attribution—Random Forest was selected [22, 23].

In its core the algorithm generates a single outcome by combining the output of many decision trees. Since the random forest model is made up of multiple decision trees, a quick description of what they represent is necessary. "Should I go

out?" is an example of a simple question that decision trees begin with. To get a response, you can then ask a series of questions, such "Is it cold?" or "Is the wind blowing offshore?" These inquiries serve as a way to divide the data and constitute the decision nodes in the tree. Every inquiry aids a person in reaching a conclusion, which is represented by the leaf node. The "Yes" branch will be followed by observations that meet the requirements, while the opposite path will be followed by those that don't. The Classification and Regression Tree (CART) technique is commonly used to train decision trees, which look for the optimal split to subset the data. Despite being popular supervised learning algorithms, decision trees can have issues including bias and overfitting. However, the random forest method predicts more accurate outcomes when several decision trees create an ensemble, especially when the individual trees are uncorrelated [24].

By combining feature randomness and bagging to produce an uncorrelated forest of decision trees, the random forest algorithm is a continuation of the bagging technique. By creating a random subset of features, feature randomness—also referred to as feature bagging or "the random subspace method"—ensures that decision trees have little association with one another. This is one of the main distinctions between random forests and decision trees. Each tree in the ensemble of decision trees that make up the random forest method is composed of a bootstrap sample, which is a data sample taken from a training set with replacement. The forecast will be determined differently depending on the type of difficulty. A majority vote, or the most common categorical variable, will determine the projected class for a classification job, whereas the average of the individual decision trees will be used for a regression work [24].

The specifics of the former tend to create problems with overrepresentation, as stated in the previous subsection. To mitigate this, a mixture of undersampling for majority classes and oversampling for minority classes was used to resample the dataset. This was done with the goal of limiting, as much as possible, the biasing of the model towards the most frequent classes and providing a balanced training set. To further ensure that the classifier assigns the proper weight to underrepresented classes, class weights were used during model training in addition to resampling. This modification enhances the model's capacity to identify less prevalent but noteworthy malware sources.

Even though the explained shortcomings of the Random Forest algorithm it was still chosen for the proposed solution for its numerous advantages, which make it well suited for the task. Some of those advantages, include: [25]

- Handling High-Dimensional Data: Malware datasets often contain a large number of features (e.g., cryptographic hashes, categorical labels), and Random Forest can effectively manage high-dimensional data without significant feature engineering.
- Resilience to Overfitting: By averaging the predictions of multiple trees, Random Forest reduces the risk of overfitting, which is especially beneficial in cybersecurity tasks where data can be noisy or incomplete.
- Interpretability and Feature Importance: Random Forest provides insights into feature importance, helping analysts understand which features are most indicative of certain countries or APT groups. This is valuable for validating the model’s decision-making process.
- Probabilistic Predictions: The algorithm produces a probability distribution over class labels, allowing the application to quantify uncertainty and provide confidence scores for each prediction.

It was assed that the reviewed AI algorithm features would help immensely with the set goal, while still keeping a relatively simple level for development and later user interaction.

5. Language Choice and System Architecture

5.1. Programming Language Selection

One of the most popular programming languages for writing algorithms and models in artificial intelligence is Python [26]. The proposed model though was written in Java and before getting in to details of its working, first an explanation will be given for why, some may say an unpopular, programming language was used. Now Java is both a platform and a programming language. Java is a powerful, multi-platform, object-oriented, secure, and a high-level programming language. The language has been extremely popular amongst developers thanks to its high quality and easily accessible learning resources, active community support, etc. One key point in system building with Java is the offered robust ecosystem of built-in features and libraries for creating a variety of applications, which help to reduce the development time. Another one is its platform independence. Any underlying platform, including Windows, Linux, iOS, and Android, may run Java code without requiring any changes. Of course, there are some drawbacks. Java can be quiet resource (e.g. memory) hungry. Another key disadvantage is the performance. When Java applications are compared against natively built C or C++ programs, it is evident that Java programs tend to be slower [27, 28].

When AI is concerned, a field of endless possibilities for creating intelligent apps is demonstrated by combining Java's resilience and AI's cognitive power. Thanks to the aforementioned advantages, the language develops a versatility that allows for the development of intelligent systems across a range of sectors, such as banking, healthcare, and e-commerce. To facilitate the creation of such systems, Java has several different AI frameworks: Weka, Deeplearning4j, and Apache Spark MLlib, to name a few. For the implementation of the proposed machine learning application, the Weka framework, a full set of Java-based machine learning tools, was used. Clustering, classification, and feature selection are among the machine learning techniques offered by Weka (Waikato Environment for Knowledge Analysis) for data mining applications. Numerous pre-implemented machine learning techniques, such as Random Forest, Support Vector Machines (SVM), decision trees, and others, are available in Weka.

This simplified the process of testing out several algorithms and choosing the top-performing model for the predefined research goals [29]. A key consideration in selecting Weka was the Random Forest classifier's availability, which was finally chosen because of its resilience and capacity to process high-dimensional data. Additionally, Weka has built-in preprocessing tools for data, including filters for managing missing values, normalization, resampling, and discretization.

The research benefited greatly from this flexibility as it made it possible to handle unbalanced data and do the required preprocessing procedures prior to model training. Given the heterogeneous nature of our dataset—which included categorical labels, cryptographic hashes, and other metadata—the framework's support for both nominal (categorical) and numerical (real-valued) features was crucial. Last, but not least, Weka is a freely accessible and highly adaptable framework that is open-source under the GNU General Public License (GPL). Because the framework is open-source, it may be customized to meet particular needs without requiring further funding if needed. Other frameworks were also considered, e.g., Deeplearning4j and Apache Spark MLlib, but were either paid, optimized for a different problem or made the process and code overly complex [30].

5.2. System Architecture

The malware classification application is designed to streamline the process of analyzing malware samples, extracting relevant features, and predicting the probable country of origin and associated Advanced Persistent Threat (APT) group. Utilizing a modular design, the system combines output production, machine learning classification, feature extraction, and data preparation. Flexibility, scalability, and ease of maintenance are guaranteed by this modular architecture. The system architecture consists of the following main components:

- Data Ingestion and Preprocessing Module – is responsible for loading and getting ready for analysis the raw malware dataset. The raw dataset is read by the system from CSV files that include malware samples. Cryptographic hash values (MD5, SHA1, SHA256), family labels, status, nation, and APT group labels are all included in each record. Duplicate entry, missing value, and incorrect data issues are then addressed. To guarantee precise model training and prediction, data quality is essential. Using label encoding or hashing methods, the module converts cryptographic hash values and category information (such as Family and Status) into numerical forms. The data can be efficiently processed by the machine learning algorithms thanks to this conversion. To address class imbalance, the module applies resampling techniques, such as oversampling minority classes and undersampling majority classes. This ensures a balanced training dataset and improves model performance.
- Feature Extraction Module – extracts significant characteristics for model training from the cleaned and encoded data. From the malware samples, static features including Family, Status, and encoded hash values (MD5, SHA1, SHA256) are extracted using Static Feature Extraction. These characteristics were picked because they are essential to spotting trends in virus activity. In feature engineering, the most relevant features are transformed and chosen. The resultant feature set may be fed into machine learning models as it is a structured dataset with both nominal and numerical information.
- Model Training Module – has the mail goal of using the preprocessed and feature-extracted dataset to train the machine learning classifiers. Two distinct models are used by the system: one for the nation classifier, which predicts the malware sample's country of origin using a Random Forest classifier, and another for the APT group linked to the malware sample using a Random Forest classifier.
- Prediction Module – handles the real-time processing of new malware samples. This module takes in fresh data, extracts features, and uses the learned models to provide predictions. The workflow includes data input and preprocessing, where new malware samples are ingested in the form of a CSV file (new_features.csv). In the same way as the training data, the data is preprocessed and encoded. The next step is country prediction, in which the preprocessed data is sent into the Country Classifier, which produces a probability distribution across all potential nation classes as well as a projected country label. The APT Group Classifier then processes the same data to provide a projected APT group label and related probability scores for the APT group prediction.
- Output Generation and Analysis Module – compiles the predictions and generates a user-friendly report. The module performs tasks for prediction compilation, where for each malware sample, the predicted country, APT group, and associated probability distributions are compiled into a summary report; output file generation, where the results are written to an output file (prediction_output.txt), which includes the original input features of the malware sample, the predicted country of origin, and the predicted APT group. Each prediction's probability score is shown in the IDE console, providing transparency and confidence levels.

The architecture of the proposed application is designed to facilitate a seamless workflow, visualized in Figure 2, from data ingestion to prediction output.

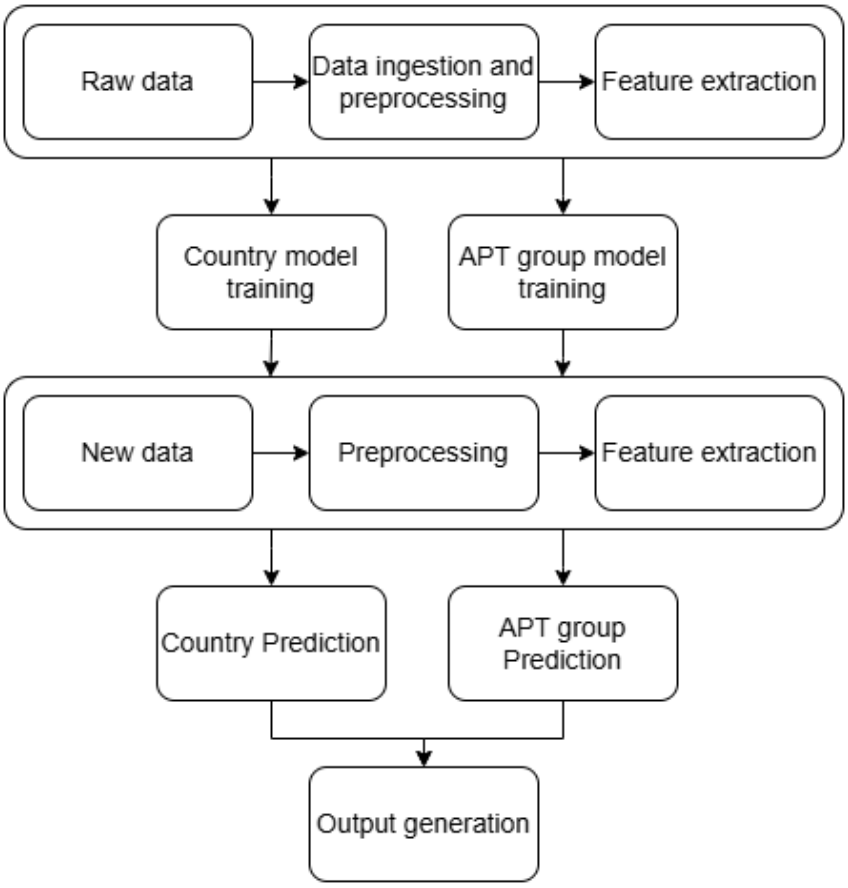

Figure 2.
System flow diagram.

An initial implementation of the proposed solution can be found in the following link: https://drive.google.com/drive/folders/1Jf37XnunxSBsNw4q2GY_IsMWPzBLWk7Z?usp=drive_link

6. Results and Evaluation

As was already highlighted the used dataset consists of roughly around 3500 tagged state-sponsored malware samples gathered from a variety of threat intelligence sources. After doing an analysis on the dataset it was concluded the distribution of APT groups was relatively even and when doing a test, the results were almost always correct. This was not the case for the country distribution. Initially the predominant country in the set was China, which lead to heavy biasing of the results. To mitigate this as much as possible resampling was used in order to even out the spread of each country in the used data. Figure 3 shows the initial and resampled distribution of countries in the dataset.

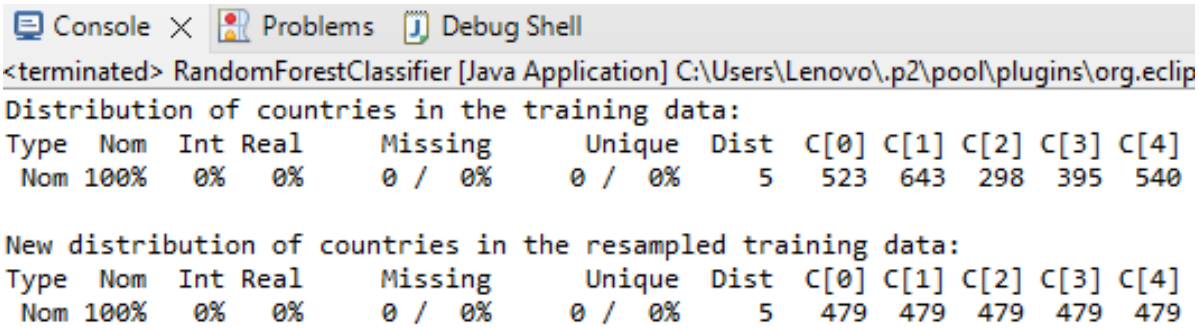

Figure 3.
Initial and resampled distribution of countries.

The result from using resampling increased greatly the level of accuracy when evaluating the country predictions. A test 5 malware sample data, that haven't been part of the training data, was used. The data was purposely chosen to represent different countries and APT groups. The results of the test are shown on Figure 4.

```
e6ecb146f469d243945ad8a5451ba1129c5b190f7d50c64580dbad4b8246f88e,China,APT 10 predicted country: China, predicted APT-group: APT 10
f37940a7b52fad1b54a96abc767cb329d9bcd4bafc7bfa9a5e07b0aaeb8ebff1,China,APT 10 predicted country: North-Korea, predicted APT-group: Dark Hotel
170e5eb004357dfce6b41de8637e1dbeb87fa58e8b54a2031aac33afb930f3c8,China,APT 10 predicted country: China, predicted APT-group: Energetic Bear
c8087186a215553d2f95c68c03398e17e67517553f6e9a8adc906faa51bce946,China,APT 10 predicted country: China, predicted APT-group: APT 28
14b6f5f3a04a8f6ba5dce1a71c48d28d8c5f9ba7e2615ab55e92e85fe9db610e,China,APT 10 predicted country: Pakistan, predicted APT-group: Gorgon Group
```

Figure 4.
APT and country predictions from the model.

After an analysis of the results was done it was concluded that when considering APT prediction all 5 samples where correctly attributed, but even after the resampling country attribution was still problematic with 3 out of 5 samples having correct values. Also, the biasing towards a specific country (i.e., China) was present, but compared to the initial set greatly reduces. For the final test 10 malware sample data were used. To the previous 5 samples 5 more were added at random. The results confirmed the intermediate analysis with APT accuracy reaching a 100% or very close to a 100%, while the country accuracy was around 70%, shown in Figure 5. The cause of the reduced country accuracy is attributing APTs wrongly, which is most probably caused by a still somewhat uneven dataset. Model hyperparameters could also have some influence, although the same values for APT groups are used (i.e., 100 trees and a maximum depth of 10) and the accuracy there is excellent. Most probably a combination of training data and hyperparameters customization will be needed in order to improve country accuracy.

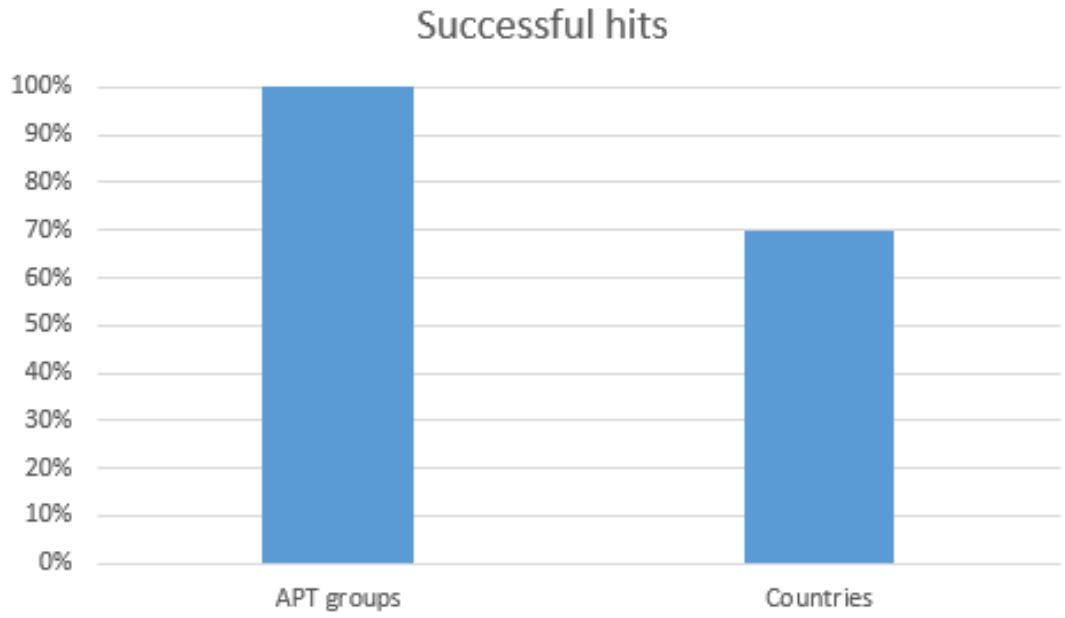

Figure 5.
Graphical representation of APT and country predictions from the model.

7. Discussions – Comparison to Existing Solutions

The proposed AI model is not the first that was created and used with the intention of analysis and/or attributing malware. In one such example [31] constructed an application for malware analysis using genetic information. The main point in their article is that since malicious software or parts of it tends to be reused and have modification patterns, which are geared towards achieving the APT group's goal, principles from linguistics and biology could be used to analyze it. The proposed solution in Pfeffer [31] initially examines the "genetics" of malware, or reverse-engineered versions of the original software, which provide important details about the program that cannot be found by merely examining the executable version. After that in to account is take the malware's evolutionary process, or how it changes from one species to another, might provide information about the malware's lineage, the traits of the attacker, and the potential sources and appearances of future assaults. Also, when applied to malware characterization, functional linguistics—the study of the intent behind communicative acts—can aid in the investigation of the intent underlying malicious activities.

Another example is given from Zhang, et al. [32] with the introduction of sophisticated approach for assigning particular threat actors to APT malware. The main point of the approach is the approach's key component is the creation of an event behavior graph that records host system execution traces by using API commands and associated actions. This method improves the knowledge of malware behavior by modeling the complex interactions between events using Graph Neural Networks (GNNs). The study also employs Image CNTM to analyze opcode images, capturing both local spatial correlations and continuous long-term dependencies. This method makes it easier to classify malware more precisely by giving a visual depiction of its operating patterns. A multi-feature, multi-input deep learning model is proposed by the researchers by combining behavior and word frequency information. By combining several malware traits, this fusion strategy improves the accuracy of attribution.

The common point between the two studies is the use of certain software specifics for the attribution of the malware. The first study adopts a genetic information-based approach, treating software code as genetic material to detect code reuse

and establish lineage between malware samples. The second study introduces a multi-feature fusion model combining behavior traces and opcode image analysis, leveraging complex graph-based and deep learning techniques. In contrast, the model proposed in this article takes a different, and largely underexplored, approach for APT attribution. Instead of relying on dynamic analysis, behavior graphs or deep learning models it focuses on static features derived directly from the malware sample, which could be extracted without high levels of programming skills or complex programming function. Cryptographic hash values (MD5, SHA1, SHA256) are used as characteristics to give each malware sample a distinct fingerprint. Hash values may be easily retrieved without the need for intricate dynamic analysis, in contrast to behavior-based features. Large-scale malware datasets benefit greatly from this method's ability to handle data more quickly and with less computing overhead. The malware sample's family label is incorporated into the analysis when it is accessible. Like the genetic analogy in the first study, this feature records lineage information, but instead of needing a complete lineage reconstruction, it makes use of established categories. Even if the sample demonstrates code repetition or obfuscation, the model may identify patterns suggestive of certain APT organizations by integrating the malware family. This makes the model accessible and efficient, enabling quicker deployment in real-time cybersecurity environments. Also the test in [32] were done with the majority of the same dataset as in this article. Comparing the accuracy of the models with regards to APT group attribution the two are almost identical and with extremely high accuracy percentage. The here proposed solution does have drawbacks when it comes to country attribution, but this is negated by the accuracy of the APT group attribution, because this information, in certain cases, is more valuable and can be used for crosschecking different source to pinpoint the originating country, if there is one.

8. Conclusion

With the increasing operations of APT groups, discovering effective countermeasures and attribution is becoming highly important in the world of cybersecurity. For that reason, in this article, a novel machine learning-based approach for malware attribution, aimed at identifying both the country of origin and the associated APT group, is proposed. The methodology leverages static features extracted from malware samples, including cryptographic hash values (MD5, SHA1, SHA256) and family information, to build two separate Random Forest classifiers. By focusing on static analysis and leveraging interpretable machine learning techniques, this approach offers a valuable tool for cybersecurity analysts, aiding in the rapid identification of threat actors and enhancing overall threat intelligence capabilities. Future enhancements and possible integration into cyber-defense systems could prove valuable for cybersecurity teams when facing challenges regarding the analysis and fighting of sophisticated cyberattacks.

References

[1] European Parliament, "What is artificial intelligence and how is it used?, European Parliament," Retrieved: https://www.europarl.europa.eu/topics/en/article/20200827STO85804/what-is-artificial-intelligence-and-how-is-it-used. 2020.

[2] H. Melissa, "Here's how people are actually using AI, MIT Technology Review," Retrieved: https://www.technologyreview.com/2024/08/12/1096202/how-people-actually-using-ai/. 2024.

[3] Lucia.stanham, "AI-powered cyberattacks, Crowdstrike," Retrieved: https://www.crowdstrike.com/en-us/cybersecurity-101/cyberattacks/ai-powered-cyberattacks/. 2024.

[4] X. B. Wang, G. Y. Yang, Y. C. Li, and D. Liu, "Review on the application of artificial intelligence in antivirus detection system," presented at the 2008 IEEE Conference on Cybernetics and Intelligent Systems, IEEE, 2008.

[5] Mary-Ann Hut, "The role of artificial intelligence in modern antivirus software, IronTree," Retrieved: https://www.irontree.co.za/artificial-intelligence-in-modern-antivirus-software/. 2023.

[6] DarkTrace, "The next paradigm shift AI-Driven cyber-attacks, DarkTrace Research White Paper," Retrieved: https://www.oixio.ee/sites/default/files/the_next_paradigm_shift_-_ai_driven_cyber_attacks.pdf. 2021.

[7] B. Guembe, A. Azeta, S. Misra, V. C. Osamor, L. Fernandez-Sanz, and V. Pospelova, "The emerging threat of ai-driven cyber attacks: A review," *Applied Artificial Intelligence*, vol. 36, no. 1, p. 2037254, 2022. https://doi.org/10.1080/08839514.2022.2037254

[8] Ş. Ensar, *Use of artificial intelligence techniques / applications in cyber defense*. Cornell University arxiv. https://doi.org/10.48550/arXiv.1905.12556, 2019.

[9] R. Shad, B. Peter, and P. Kaledio, "AI-powered threat intelligence: Automating cyber threat analysis and prediction," Retrieved: https://easychair.org/publications/preprint/pKRp/download. 2024.

[10] M. M. Yamin, M. Ullah, H. Ullah, and B. Katt, "Weaponized AI for cyber attacks," *Journal of Information Security and Applications*, vol. 57, p. 102722, 2021. https://doi.org/10.1016/j.jisa.2020.102722

[11] J. Burton and S. R. Soare, "Understanding the strategic implications of the weaponization of artificial intelligence," presented at the 11th International Conference on Cyber Conflict (CyCon), Tallinn, Estonia. https://doi.org/10.23919/CYCON.2019.8756866, 2019.

[12] Akitra, "Leveraging AI for advanced persistent threat (APT) detection and mitigation, medium," Retrieved: https://medium.com/@akitrablog/leveraging-ai-for-advanced-persistent-threat-apt-detection-and-mitigation-c5089863d0fb. 2024.

[13] R. J. Sternberg, "Human intelligence", Encyclopedia Britannica," Retrieved: https://www.britannica.com/science/human-intelligence-psychology. 2023.

[14] White House, "Imposing costs for harmful foreign activities by the russian government, White House," Retrieved: https://www.whitehouse.gov/briefing-room/statements-releases/2021/04/15/fact-sheet-imposing-costs-for-harmful-foreign-activities-by-the-russian-government/. 2021.

[15] Mitre, "Mitre ATT&CK, APT29, Mitre ATT&CK," Retrieved: https://attack.mitre.org/groups/G0016/. 2017.

[16] M. Adam, "Meet crowdStrike's adversary of the month for June: MUSTANG PANDA, Crowd Strike Blog," Retrieved: https://www.crowdstrike.com/en-us/blog/meet-crowdstrikes-adversary-of-the-month-for-june-mustang-panda/. 2018

[17] Kaspersky, "What is an advanced persistent threat (APT)?, Kaspersky Resource Center," Retrieved: https://www.kaspersky.com/resource-center/definitions/advanced-persistent-threats. 2024.
[18] BBC, "Ukraine cyber-attack: Russia to blame for hack, says Kyiv, BBC News," Retrieved: https://www.bbc.com/news/world-europe-59992531. 2022.
[19] P. Pavel and H. Steve, "Cyberattack hits Ukraine as U.S. warns Russia could be prepping for war, Reuters," Retrieved: https://www.reuters.com/world/europe/expect-worst-ukraine-hit-by-cyberattack-russia-moves-more-troops-2022-01-14/. 2022.
[20] Fortinet, "Artificial intelligence (AI) in cybersecurity," Retrieved: https://www.fortinet.com/resources/cyberglossary/artificial-intelligence-in-cybersecurity. 2023.
[21] Cyber-Research, "APTMalware, GitHub," Retrieved: https://github.com/cyber-research/APTMalware. 2019.
[22] L. Breiman, "Random forests," *Machine Learning*, vol. 45, pp. 5-32, 2001. https://doi.org/10.1023/A:1010933404324
[23] G. Biau and E. Scornet, "A random forest guided tour," *Test*, vol. 25, pp. 197-227, 2016. https://doi.org/10.1007/s11749-016-0481-7
[24] IBM, "What is random forest?, IBM," Retrieved: https://www.ibm.com/topics/random-forest. 2024.
[25] S. Alex, "What are the advantages and disadvantages of random forest?, Pickl.AI," Retrieved: https://www.pickl.ai/blog/advantages-and-disadvantages-random-forest/. 2024.
[26] Alex Ryabtsev, "8 reasons why python is good for AI and ML, Djangostars," Retrieved: https://djangostars.com/blog/why-python-is-good-for-artificial-intelligence-and-machine-learning/. 2024.
[27] J. Farrel, *Java™ programming*. USA: Cengage Learning Inc, 2018.
[28] Dmitry Nazarevich, "Benefits and drawbacks of Java, Innowise," Retrieved: https://innowise.com/blog/benefits-and-drawbacks-of-java/. 2021.
[29] E. Frank, M. A. Hall, and I. H. Witten, *The Weka workbench, online appendix for data mining: Practical machine learning tools and techniques*, 4th ed. USA: Morgan Kaufmann, 2016.
[30] Eclipse Deeplearning4j Development Team, "Deeplearning4j: Open-source distributed deep learning for the JVM, Apache Software Foundation License 2.0," Retrieved: https://deeplearning4j.konduit.ai/en-1.0.0-beta7. 2024.
[31] A. Pfeffer, "Malware analysis and attribution using genetic information " presented at the 2012 7th International Conference on Malicious and Unwanted Software, Fajardo, PR, USA. https://doi.org/10.1109/MALWARE.2012.6461006, 2012.
[32] J. Zhang, S. Liu, and Z. Liu, "Attribution classification method of APT malware based on multi-feature fusion," *Plos One*, vol. 19, no. 6, p. e0304066, 2024. https://doi.org/10.1371/journal.pone.0304066