Future Generation Computer Systems 115 (2021) 641–658

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Redundancy Coefficient Gradual Up-weighting-based Mutual Information Feature Selection technique for Crypto-ransomware early detection

Bander Ali Saleh Al-rimy [a,*], Mohd Aizaini Maarof [b], Mamoun Alazab [c], Syed Zainudeen Mohd Shaid [b], Fuad A. Ghaleb [b,*], Abdulmohsen Almalawi [d], Abdullah Marish Ali [d], Tawfik Al-Hadhrami [e]

[a] Faculty of Business and Technology, Unitar International University, 3-01A Level 2, Tierra Crest, Jalan SS 6/3, 47301 Petaling Jaya, Selangor, Malaysia
[b] School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia, 81310 Johor Bahru, Johor, Malaysia
[c] College of Engineering, IT & Environment, Charles Darwin University, Australia
[d] Computer Science Department, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah 21589, Saudi Arabia
[e] School of Science and Technology, Nottingham Trent University, Nottingham, NG11 8NS, United Kingdom

ARTICLE INFO

Article history:
Received 22 May 2020
Received in revised form 28 September 2020
Accepted 3 October 2020
Available online 8 October 2020

Keywords:
Ransomware
Malware
RCGU
Mutual information
Feature selection
Early detection

ABSTRACT

Crypto-ransomware is a type of malware whose effect is irreversible even after detection and removal. Thus, early detection is crucial to protect user files from being encrypted and held to ransom. Several studies have proposed early detection solutions based on the data acquired during the pre-encryption phase of the attacks. However, the lack of sufficient data in the early phases of the attack adversely affects the ability of feature selection techniques in these models to perceive the common characteristics of the attack features, which makes it challenging to reduce the redundant features, consequently decreasing the detection accuracy. Therefore, this study proposes a novel Redundancy Coefficient Gradual Upweighting (RCGU) technique that makes better redundancy–relevancy trade-offs during feature selection. Unlike existing feature significance estimation techniques that rely on the comparison between the candidate feature and the common characteristics of the already-selected features, RCGU compares the mutual information between the candidate feature and each feature in the selected set individually. Therefore, RCGU increases the weight of the redundancy term proportional to the number of already selected features. By integrating the RCGU into the Mutual Information Feature Selection (MIFS) technique, the Enhanced MIFS (EMIFS) was developed. Further improvement was achieved by proposing MM-EMIFS which incorporates the MaxMin approximation with EMIFS to prevent the redundancy overestimation that RCGU could cause when the number of features in the already-selected set increases. The experimental evaluation shows that the proposed techniques achieved accuracy higher than that in related works, which confirms the ability of RCGU to make better redundancy–relevancy trade-offs and select more discriminative pre-encryption attack features compared to existing solutions.

1. Introduction

Since its beginning in the early 1970s, several types of malicious software, also called malware, have been witnessed in the wild, such as Viruses, Worms, Trojans, Spyware and Ransomware [1–3]. Ransomware is a type of malware whose purpose is to hold user data and files to ransom by denying the access to these files [4–9]. Although ransomware history dates back to the late 1980s, it did not gain much popularity among attackers until recently, when some enabling technologies like Ransomware-as-a-Service (RaaS), Internet, cryptography and the difficult-to-trace digital currency, have emerged [10]. These technologies make it easy for even novice attackers to develop and disseminate their own ransomware and get paid without the fear of being caught by the authorities [10,11]. Consequently, the rate of ransomware attacks has increased dramatically in recent years [12–14].

According to Kaspersky, ransomware attacks are now moving towards business and 30% of infections in 2019 were among

* Corresponding authors.
E-mail addresses: bnder321@gmail.com (B.A.S. Al-rimy), aizaini@utm.my (M.A. Maarof), mamoun.alazab@cdu.edu.au (M. Alazab), szainudeen@utm.my (S.Z.M. Shaid), fuadeng@gmail.com (F.A. Ghaleb), balmalowy@kau.edu.sa (A. Almalawi), ammali@kau.edu.sa (A.M. Ali), tawfik.al-hadhrami@ntu.ac.uk (T. Al-Hadhrami).

https://doi.org/10.1016/j.future.2020.10.002
0167-739X/

corporate users instead of individuals [15]. The report also concluded that around 4$ billion of financial loss was caused by WannaCry attacks. This adds to the previous statistics which show that, throughout the world, the losses hit $3 million and $352 million due to ransomware attacks in 2014 and 2015, respectively [16,17]. In 2016, Indiana county alone incurred around $220K to recover from ransomware attacks [17]. In 2017, the estimated loss due to NotPetya and WannaCry ransomware attacks was 8$ billion around the globe [18]. Denying access to data is not the only loss that ransomware victims incur, the damage could also include downtime costs, loss of money and reputation [6]. Based on the severity, ransomware is categorized into locker-ransomware and crypto-ransomware [19]. In contrast to locker-ransomware attacks, whose effect can easily be mitigated, crypto-ransomware attacks are not reversible even after removing the malware. In many cases, the victim has no choice other than paying the ransom to get the decryption key [10]. Therefore, to effectively protect user's digital assets, it is imperative to detect crypto-ransomware attacks early, i.e. before the encryption takes place [10,16,20,21]. The early detection of crypto-ransomware attacks can be achieved by observing its process(es) running in the victim's machine and analysing the runtime data generated during the pre-encryption phase, i.e. the phase in the crypto-ransomware lifecycle that precedes encryption. However, detecting crypto-ransomware at early phases of its attack is challenging, due to insufficient data and attack patterns at this early phase [19,22].

The small amount of data captured during the initial phases of the attack is one of the challenges for the early detection which causes low detection accuracy [23,24]. Even with the availability of many ransomware samples, the runtime data acquired during the pre-encryption phase of the attack is small compared to the entire runtime data that can be collected from each sample if we wait until the end of the attack. This small amount of data contains only a few attack patterns, if any, which are not enough for the model to decide whether this process is normal or malicious. Consequently, the pre-encryption data lack sufficient attack patterns that the detection model needs to make accurate decisions. This data insufficiency also prevents the feature selection technique from identifying the important features that distinguish the ransomware behaviour from the normal behaviour. With the insufficient data collected during the early phases of the attack, the feature selection technique cannot estimate the features' significance accurately. This challenge exacerbates due to high dimensional features generated by feature extraction methods like n-gram, adopted by most detection solutions [20,25–28]. That is, the number of features extracted by n-gram increases exponentially with the size of n, which renders the detection models prone to overfitting [20,25–27,29–31]. Many of those features are either too common or too specific which makes the information they carry about the attacks of little use [16]. In addition, many of those features are redundant and highly correlated due to the dependency between the API calls used by ransomware's running process, which makes these APIs always appear together [32–34]. The redundant features cause a degradation in detection accuracy, as they add no relevant information about the ransomware attack [19]. More importantly, including these redundant features in the selected set comes at the cost of discarding other informative features that the feature selection technique could exclude when exceeding the pre-defined number of required features.

Several Ransomware and malware detection solutions as well as many other solutions incorporate feature selection techniques to reduce data dimensionality and remove redundant features [20,25,35,36]. It turns out that features' redundancy and relevancy are the main factors that govern the performance of any feature selection technique [37]. These techniques try to filter out the redundant and irrelevant features and keep only the informative ones. However, redundancy and relevancy are not always orthogonal. These features are conflicting in nature, as some relevant features might also be redundant [37]. For example, BCryptDeriveKey employed for deriving the key from secret agreement value is always accompanied by BCryptSecretAgreement responsible for creating hSharedSecret handle used as a parameter for the BCryptDeriveKey. Another example is BCryptEncrypt function used for encrypting a block of data usually comes with BCryptGenerateSymmetricKey, BCryptGenerateKeyPair, or BCryptImportKey employed to obtain the hKey handle which is used as an input parameter for BCryptEncrypt. Therefore, the redundancy–relevancy trade-off is needed during the selection process. As such, it is necessary that the feature selection technique can make this redundancy–relevancy trade-off effectively.

The information theory-based feature selection techniques are superior when it comes to the trade-off between redundancy and relevancy, as they make no assumptions about the distribution of the underlying data [29,38]. This is important for ransomware early detection, as it relies on sparse and incomplete attack patterns whose clear distribution has yet to be observed [19]. The redundancy–relevancy trade-off is carried out by adjusting the values of redundancy coefficients, which changes the belief in the redundancy term at each iteration in a way that is inversely proportional to the current size of the selected features set [38]. Although this approach works well for data with full observations about the attacks, it generates a suboptimal feature set when dealing with data that lack sufficient attack patterns [39,40]. This is due to the reliance on the calculation of mutual information between the candidate feature and the common characteristics of all already-selected features in the selected set [29]. Such common characteristics are difficult to perceive from incomplete data acquired during the pre-encryption phase of crypto-ransomware attacks. Consequently, the selected set could include redundant and irrelevant features, given the limited amount of attack patterns, as is the case in the early detection where the entire characteristics of ransomware attack have not yet been observed [39,41]. Therefore, an improvement to the mutual information technique is needed that overcomes the challenge of pre-encryption data insufficiency and estimates features' significance more accurately.

To this end, this paper is devoted to address this issue and proposes a Redundancy Coefficient Gradual Upweighting (RCGU) technique that estimates the features significance accurately even with insufficient attack patterns, as is the case in the early (pre-encryption) phase of crypto-ransomware attacks. By incorporating the proposed RCGU into the feature selection technique, the redundancy between the candidate feature and each feature in the selected set is individually calculated at every iteration of the feature selection process. Unlike existing feature significance techniques that decrease the weight of the redundancy term in the goal function when the number of features in the already-selected set increases, the proposed RCGU proportionally increases the weight of the redundancy term when the number of those features increases.

The key idea is that, instead of comparing the characteristics of the candidate feature with the common characteristics of all features in the selected set (which is very difficult to perceive from the limited amount of pre-encryption data collected at the beginning of a ransomware attack), RCGU (individually) compares between the candidate feature and each feature in the already-selected set. This individual comparison will help to discover redundancy even with the insufficient runtime data collected during the early phases of ransomware attacks. The intuition is that, by comparing the candidate feature with each feature in the

selected set individually, the chance that the candidate feature is redundant with one or more of those features increases with the growth of the selected set's size [40]. With this approach, the need to extract the difficult-to-perceive common characteristics of the features in the already-selected set becomes unnecessary. Consequently, the belief in the redundancy term increases when more features are added to the selected set. As such, the proposed RCGU can make better redundancy–relevancy trade-off when dealing with limited amount of data as it is the case of the data collected during the pre-encryption phase of crypto-ransomware attacks' lifecycle. The contribution of this paper is four-fold.

1- A Redundancy Coefficient Gradual Up-weighting (RCGU) technique is proposed and incorporated into the redundancy term of the goal function of the mutual information feature selection technique to improve the calculation of the relevancy–redundancy trade-off, which in turns helps in selecting a more informative features set.
2- RCGU is incorporated with the Maximum of Minimum (MaxMin) approximation technique to prevent the redundancy overestimation that RCGU could cause when the size of the selected set increases.
3- We have shown that the redundancy term plays a major role in the accuracy of the selected features and is better than the involvement of conditional redundancy in the calculation.
4- An extensive experimental evaluation was conducted to show the efficacy and significance of the improvement that the proposed techniques contributed to.

For the purpose of this study, crypto-ransomware and ransomware are used interchangeably unless stated otherwise. The rest of this paper is organized as follows. Section 2 gives an overview of the related work. Section 3 provides preliminaries about the mutual information-based feature selection techniques. Section 4 details the methodology followed to design and develop the proposed techniques. Section 5, presents the experimental results, which are analysed and discussed and compared with related works. The paper concludes with a summary of the methods and results as well as suggestions for future work, in Section 6.

2. Related works

Like other cyberattacks, ransomware attacks target a variety of systems and networks, including Personal Computers (PCs), mobile devices, Wireless Sensor Networks (WSN), Vehicular Ad-Hoc Networks (VANETs), and the Internet of Things (IoT) [42–46]. Several studies have been conducted to detect crypto-ransomware attacks. These studies can be categorized into data-centric and process-centric approaches. The data-centric approach monitors the user data and files subject to attack and raises the alarm when it detects a suspicious change in those files. Several techniques such as decoy, entropy and similarity have been employed to monitor the file structure before and after access [10,12,47–49]. However, this approach does not distinguish between the changes carried out by benign programs from those caused by crypto-ransomware, which leads to high rate of false alarms [11,50,51]. More importantly, this approach does not fully protect user data from being held to ransom, as it sacrifices part of the data (which could be more valuable to the victim than the remaining data) before detection [50,52]. Thus, the data-centric approach is not effective for crypto-ransomware early detection.

The process-centric approach monitors the behaviour of the running process to discover suspicious patterns. Several studies, such as those by Cohen and Nissim [17], Shahriari [48], Chen, et al. [53], Chen and Bridges [54] have employed this approach and acquired different types of behavioural data to train machine learning classifiers like Random Forest and Naïve Bayes. However, most of those studies followed the malware detection approach that relies on the entire runtime data, which include pre-encryption and post-encryption data, to detect the attacks [23,55]. Such approaches assume the availability of the entire data at detection time [23]. Thus, they are not suitable for crypto-ransomware early detection where the data of the malicious process in question are not fully available.

Another type of process-centric approach is to monitor the computational resources used by ransomware processes. That is, one or more resources in the user machine, such as CPU, network, I/O buffer and memory are observed, and the alarm is raised when some events related to ransomware and/or cryptography are encountered. Maltester, proposed by Cabaj, et al. [56], is one such solution aimed to detect the infection chain of the Cryptowall ransomware family via introspecting the network traffic. Similarly, Cabaj, et al. [57], Cusack, et al. [58] have proposed detection solutions based on monitoring the network traffic between the infected devices and the ransomware's command and control (C&C) server. In another study, Kharraz, et al. [12] propose UNVEIL, which observes I/O access patterns and file system activities. Similarly, Song, et al. [47] put forward a model that monitors CPU, I/O and the device's memory in order to detect the suspicious activities caused by ransomware. However, the reliance on ad-hoc events leads to high rate of false alarms, as those events are not mutually exclusive to crypto-ransomware and some benign programs raise such events as well [51]. Additionally, those events could happen after the encryption takes place, which renders this approach ineffective for early detection [12]. To be effective, it is essential that the detection takes place during early phases, before the attack starts the main sabotage, which is the encryption in the case of crypto-ransomware. Table 1 provides a summarized comparison between the data-centric and process-centric detection approaches adopted by existing ransomware detection research.

To detect crypto-ransomware early, several approaches [16,19–21,23,59] use the data collected during the pre-encryption phase of the crypto-ransomware lifecycle, before the encryption takes place. The collected data are then used to train different machine learning algorithms to classify the programs into benign and ransomware. However, the lack of sufficient data during the early phases of the attack adversely affects the ability of the goal function of feature selection techniques in these models to perceive the common characteristics of attack features. This makes it challenging to reduce the redundant features during the feature selection process. When the number of redundant features increases, the data dimensionality increases and detection accuracy decreases. This paper addresses this issue by proposing the RCGU technique, which improves the ability of the feature selection techniques to make better redundancy–relevancy trade-offs, thus improving the estimation of feature significance. This improved estimation facilitates the selection of important features and overcomes the insufficiency in attack patterns collected during the early phases of crypto-ransomware attacks.

3. Preliminaries

For two discrete variables, the mutual information (MI) criterion is the amount of information that these variables share about each other [39]. This criterion is calculated according to Eq. (1) as follows.

$$I(X;Y) = \sum_{y\epsilon Y}\sum_{x\epsilon X} p(x,y)\log\frac{p(x,y)}{p(x)p(y)} \quad (1)$$

where $I(X;Y)$ denotes the mutual information between X and Y, $P(x)$ and $p(y)$ are the marginal distribution of x and y and $p(x,y)$ is

Table 1
Summary comparison between data centric and process centric detection approaches.

Approach	Description	Studies	Advantages	Disadvantages
Data Centric	– Monitors user data and files subject to attack. – Raises the alarm when it detects a suspicious change in those files. – Employs a set of techniques such as decoy, entropy and similarity to monitor the file structure before and after access.	[10,12,47–49]	– Can identify any data-related activities instantaneously.	– Does not distinguish between the changes carried out by benign programs from those caused by crypto-ransomware. – Sacrifices part of the data (which could be more valuable to victim than the remaining data) before detection.
Process centric	– Behaviour of the running process to discover the suspicious patterns.	[17,48,53,54]	– Can detect the attacks at the runtime.	– Relies on the entire runtime data (pre-encryption and post-encryption), which is not suitable for the early detection.

the joint distribution. According to Li, et al. [29], Brown, et al. [38], Eq. (2) represents the general formula of the framework, which is referred to as a criterion by linear combinations of Shannon information terms.

$$J(X_k) = I(X_k; Y) - \beta \sum_{X_j \epsilon S} I\left(X_j; X_k\right) + \gamma \sum_{X_j \epsilon S} I\left(X_j; X_k | Y\right) \quad (2)$$

where $I(X_k; Y)$ is the mutual information between the candidate feature X_k and the class label Y; $I(X_j; X_k|Y)$ is the conditional mutual information between the candidate feature X_k and the feature X_j in the selected set S, given the class label Y, while β and γ are parameters (coefficients) with values between 0 and 1. It can be noticed that Eq. (2) consists of two parts, namely a relevancy term represented by expression (3) and a redundancy term represented by expression (4). Furthermore, the redundancy term consists of two sub-terms, namely the marginal redundancy, represented by expression (5), and the conditional redundancy, represented by expression (6).

$$I(X_k; Y) \quad (3)$$

$$\beta \sum_{X_j \epsilon S} I\left(X_j; X_k\right) + \gamma \sum_{X_j \epsilon S} I\left(X_j; X_k | Y\right) \quad (4)$$

$$\beta \sum_{X_j \epsilon S} I\left(X_j; X_k\right) \quad (5)$$

$$\gamma \sum_{X_j \epsilon S} I\left(X_j; X_k | Y\right) \quad (6)$$

The information theory-based feature selection techniques try to optimize the trade-off between the relevancy and redundancy terms. In this context, two types of techniques can be distinguished, based on whether they include the conditional redundancy term. The first type has only two terms, namely a relevancy term and a redundancy term, and is calculated using Eq. (2) by considering only the marginal redundancy coefficient β, while $\gamma = 0$. Mutual Information Features selection MIFS and Minimum Redundancy Maximum Relevance (mRMR) are examples of this type. The second type considers both redundancy terms, namely marginal redundancy and conditional redundancy. This is achieved by giving both coefficients β, γ values between 0 and 1 in Eq. (2). Joint Mutual information (JMI) is an example of this type. It turned out that the calculation of the relevancy term is same in all techniques and involves calculating the relevancy between the candidate feature X_k and the class label Y. Thus, the difference in the performance between those techniques is determined by the redundancy calculation.

As shown by Eq. (2), the values of the coefficients β and γ play an important role in the relevancy-redundancy trade-off which determines the feature's significance. Concretely, a small value of β contributes to decreasing the effect of the redundancy, which, consequently, increases the feature's significance, whereas a small value of γ decreases such significance.

4. The methodology

In this section, the design of the proposed RCGU technique that accurately estimates the features' significance based on the data collected during the pre-encryption phase of crypto-ransomware is elaborated. In addition, the integration of RCGU with the Mutual Information Feature Selection (MIFS) technique is detailed. This integration takes place in the redundancy term of the goal function. Moreover, the performance of the proposed RCGU is improved by incorporating the MaxMin approximation approach into the goal function of the MIFS to mitigate the effect of redundancy overestimation that RCGU could cause when the number of features within the already-selected set grows. Accordingly, two feature selection techniques were built, namely the Enhanced Mutual Information Feature Selection (EMIFS) and the Maximum of Minimum Enhanced Mutual Information Feature Selection (MM-EMIFS). Both techniques were used to extract the features from the early runtime data extracted during the pre-encryption phase of the crypto-ransomware lifecycle. Before the design of proposed techniques is detailed, we start by describing the attack model which illustrates the different phases of crypto-ransomware attacks.

4.1. Crypto-ransomware attack model

Ransomware's lifecycle starts from the moment when the malicious code is disseminated and lasts until the financial claim is shown to the victim. During this lifecycle, several actions are conducted in order to successfully hijack the user's files and resources. According to [8,18,21] and [34–39], ransomware attacks go through several essential phases, as illustrated in Fig. 1 and summarized below.

1- Distribution: During this phase, the ransomware is packed and delivered into the victim's system using different exploitation techniques such as email attachment or drive-by download.
2- Installation: In this phase, crypto-ransomware installs itself in the victim's machine. Such installation also involves exploring the running environment and collecting information about the victim's device, such as platform type, OS version, and already-installed programs.
3- Encryption key generation: Crypto-ransomware retrieves the encryption key from the C&C server or generates it locally.
4- File search: Ransomware starts looking for targeted files.
5- Encryption: Based on the attack approach, Crypto-ransomware starts encrypting the targeted files, either one-by-one concurrently with the files' search process or waits until listing all the files then encrypting them all at once.
6- Post encryption: Once the encryption process is finished, the original files are either deleted or moved to another location with new names.

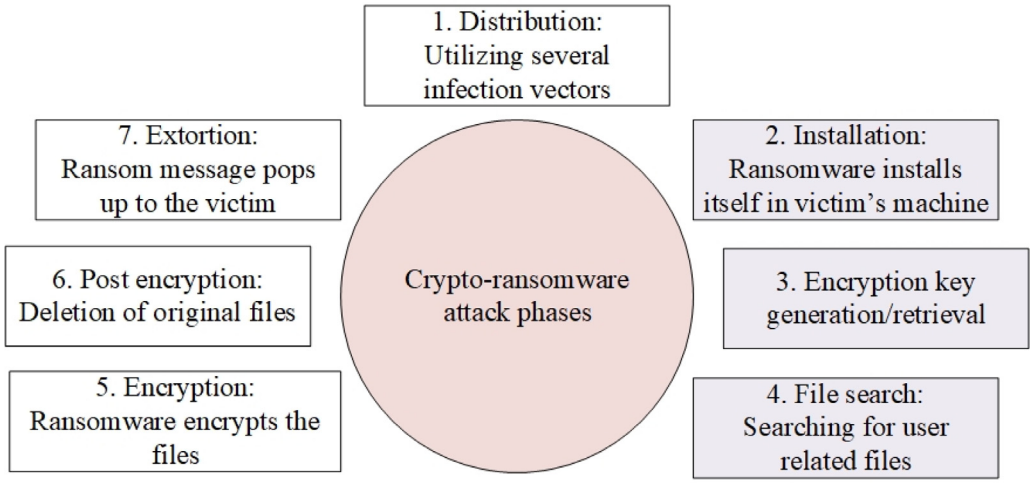


Fig. 1. Crypto-ransomware attack model.

7- Extortion: After encrypting and deleting/moving all files, the extortion message is shown to the victim asking for a ransom accompanied by payment instructions.

Among the stages listed above, the pre-encryption phase of crypto-ransomware attacks lifecycle involves (2) installation, (3) encryption key generation and (4) file search.

4.2. Pre-encryption feature extraction

As in our previous publications [19,59], early detection (also called pre-encryption) is measured from the moment we run the crypto-ransomware sample until we capture the first cryptography-related API call. The cryptography-related APIs are the API functions in the victim's OS that are explicitly or implicitly related to the encryption process. Unlike the fixed time thresholding approach employed by existing crypto-ransomware early detection solutions, our approach uses these APIs to determine the boundary that separates the pre-encryption phase from the encryption phase in the crypto-ransomware lifecycle. The pre-encryption boundary is a set of cryptography-related APIs. These APIs separate the API pre-encryption phase in crypto-ransomware from the subsequent attack phases. The intuition is that the calling of any of cryptography-related APIs for the first time indicates an imminent encryption activity which might be related to an actual attack against the user files and data. The cryptography-related APIs are stored in a vector called the pre-encryption boundary vector, such that the first call to any of its entries represents the borderline between the end of the pre-encryption phase and the beginning of the encryption phase of the crypto-ransomware attacks. This vector is used as a cut-off during data acquisition to stop recording the API calls into trace file when encountering the first call of any of the vector's entries. Table 2 shows a subset of the pre-encryption boundary vector with a description of each entry.

After collecting the API calls during the dynamic analysis of the crypto-ransomware samples, the feature extraction is carried out. The data recorded in the trace files of crypto-ransomware samples are the names of the API calls with their input parameters and returned values. As the purpose of this study is to determine whether a program is a crypto-ransomware, regardless of the family, the input parameters and returned values are removed from the trace files and only the API calls are kept. As the data in trace files are the names of API calls, they are in a textual form. Therefore, it is necessary to transform (vectorize) these textual data into a numerical form so that they become ready for modelling. Following our previous study [19], the Term Frequency-Inverse Document Frequency (TF-IDF) was used to convert the textual data in trace files into the numerical form and extract the features from these data. Concretely, each API name was used as a feature and its TF-IDF value was calculated based on its local and global frequency. The local frequency of a feature was determined by calculating its frequency in each trace file individually while the global frequency was determined by calculating the number of trace files in which this API occurred. Both local and global frequencies were then combined to calculate the TF-IDF value of that feature as shown in Eq. (7).

$$w\left(api_k^j\right) = tf\left(api_k^j\right) \cdot \log \frac{N}{idf(api_k)} \tag{7}$$

where api_k denotes the kth API; $tf(api_k^j)$ is the term frequency, which calculates how many times the api_k was called by the ransomware instance r_j in the subset. Similarly, $idf(api_k)$ is the inverse document frequency, which calculates how many ransomware instances, r_j, there are in the subset called api_k, at least one, while N denotes the total number of ransomware instances in the subset. Fig. 2 shows the general design of crypto-ransomware early detection model which incorporates the RCGU into the feature selection goal function.

4.3. The proposed redundancy Coefficient Gradual Up-weighting Technique for an enhanced MIFS

In this subsection we discuss the design and implementation of the proposed Redundancy Coefficient Gradual Upweighting (RCGU) technique and its integration with MIFS to produce the Enhanced Mutual Information Feature Selection (EMIFS) method used for selecting the discriminative features from the dataset collected during the pre-encryption phase of crypto-ransomware attacks. EMIFS adopts the approach used in the mRMR technique and calculates the mutual information according to Eq. (8).

$$J\left(x_k\right) = MI\left(x_k, y\right) - \beta \sum_{s_j \epsilon S} I\left(x_k, x_j\right) \tag{8}$$

where x_k denotes the candidate feature; $s_j \in S = \{s_1, s_2, \ldots, s_m\}$ is the jth feature in the already-selected features set S, and β is a non-negative parameter (coefficient) between 0 and 1. The left term in the equation represents feature relevancy while the right term represents feature redundancy. The value of β determines the strength of belief in the redundancy term. Fig. 3 shows the integration between the proposed RCGU with both EMIFS and MM-EMIFS.

Table 2
Examples of cryptography-related APIs.

Cryptography-related API	Description
CryptAcquireContexta	Gets a handle to a particular key container within the cryptographic service provider (CSP) in order to call CryptoAPI functions.
CryptCreateHash	For hashing initiation of a stream data.
CryptUnprotectData	For integrity checking and decryption of the data in a DATA_BLOB structure.
CryptHashData	Adds data to a specified hash object.
CertGetNameStringW	Obtains the subject or issuer name from a certificate CERT_CONTEXT structure and converts it to a null-terminated character string.
CryptDecrypt	Decrypts the data encrypted the CryptEncrypt function.
CryptExportKey	Exports a cryptographic key or a key pair from the CSP
CryptGetObjectUrl	Retrieves the URL of the remote object.
CryptGetHashParam	Retrieves data related to a hash object.
CryptGenKey	Generates the session key or the public–private pair
CryptDecodeObjectEx	Decodes the data in a structure determined by the lpszStructType parameter
EncryptMessage	Encrypts a message using the chosen cryptographic algorithms.
CryptReleaseContext	Releases the handle of the CSP with the key container.

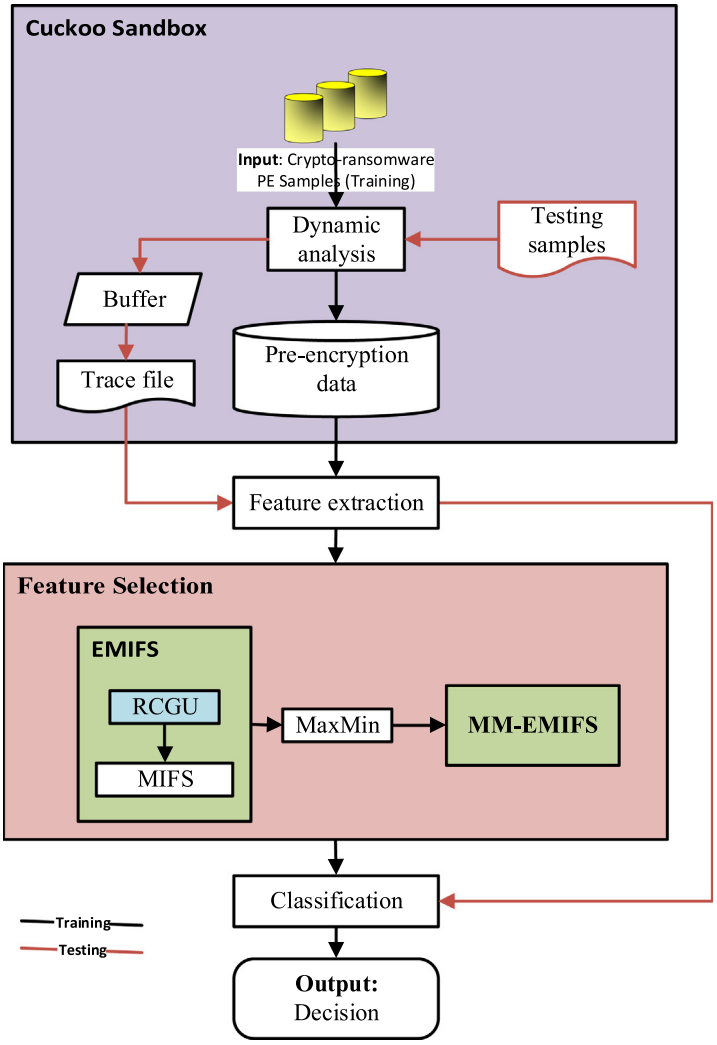


Fig. 2. Ransomware early detection using the improved RCGU-Based feature selection techniques.

4.4. Redundancy Coefficient Gradual Upweighting Technique

In contrast to existing MI techniques that calculate the redundancy coefficient inversely proportional to the size of already-selected feature set, the proposed Redundancy Coefficient Gradual Upweighting (RCGU) technique increases the weight of the redundancy term in proportion to the size of the selected set. Consequently, the belief in the redundancy term increases when the number of features in the selected set increases. Intuitively, by comparing the candidate feature with each feature in the selected set individually, the chance that the candidate feature is redundant with one or more of those features increases when

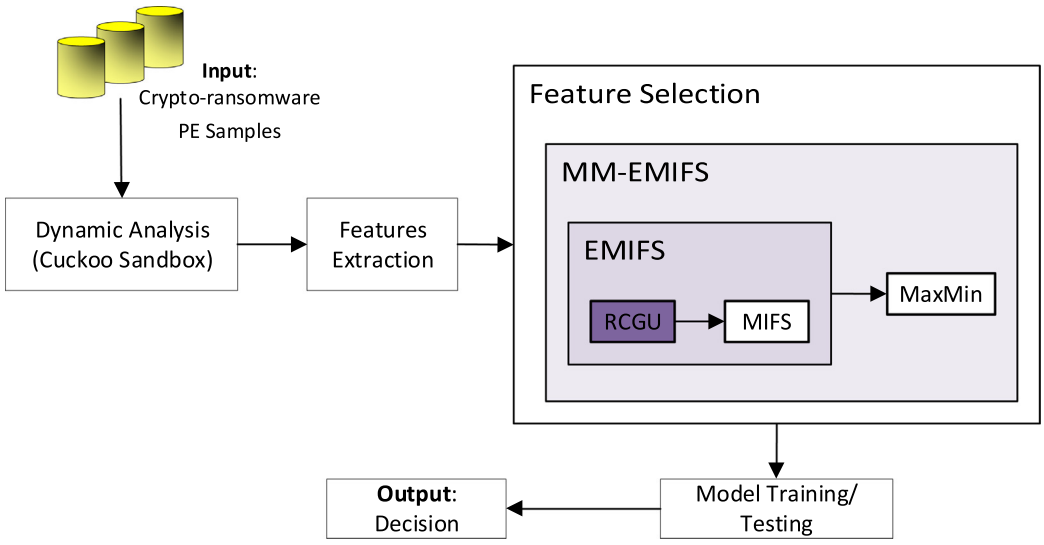


Fig. 3. The general architecture of the crypto-ransomware early detection model including the proposed RCGU-based Mutual Information Feature Selection Technique.

the number of features in the selected set increases [40]. In this way, the RCGU can make better redundancy–relevancy trade-offs when dealing with the limited amount of data collected during the pre-encryption phase of a crypto-ransomware attack's life-cycle, where the common characteristics of the already-selected features are yet to be observed. Consequently, the ability of the feature selection method to estimate feature significance and select the features related to crypto-ransomware attacks is improved.

Unlike existing MI techniques that calculate the coefficient β according to Eq. (9), the RCGU technique calculates it according to Eq. (10). Thus, the value of β starts low at the beginning of selection process, corresponding to the size of the already-selected set S that also starts low. The value of β then increases gradually with the increase in size of S. The value of the denominator does not change throughout the selection process and maintains a fixed value equal to the size of the original features set, F.

$$\beta = \frac{1}{|S|} \tag{9}$$

$$\beta = \frac{|S|}{|F|} \tag{10}$$

where $|S|$ and $|F|$ denote the number of features in the selected and original set, respectively. Therefore, EMIFS selects the informative features according to Eq. (11).

$$J(x_k) = MI(x_k, y) - \frac{|S|}{|F|} \sum_{s_j \epsilon S} I(x_k, x_j) \tag{11}$$

Fig. 4 shows the pseudo code of EMIFS. Given the original features vector. F, EMIFS selects the informative features according to Eq. (11). As shown by Fig. 5, $F = \{f_1, f_2, f_3, \ldots.\}$, is the original features vector with n number of features; V is a temporary set that holds the features whose MI values have already been calculated; $S = \{s_1, s_2, \ldots, s_\tau\}$ is the selected set with τ number of features. EMIFS starts by initializing the empty sets V and S and calculating the MI value for each feature, f_i, in F. Based on the MI value, those features are ranked and stored in the set V. The feature v_k in V with $\max(V, MI)$ is simultaneously removed from V and added into S. The next feature s_p is chosen according to Eq. (12).

$$s_p = \underset{v_j \epsilon V}{\operatorname{argmax}} [MI(v_j; C) - \frac{|S|}{|F|} \sum_{s_j \epsilon S} I(v_k; s_j)] \tag{12}$$

4.5. Maximum of minimum-based enhanced mutual information features selection technique

The RCGU proposed in the previous section has the limitation of overestimating the redundancy term when the number of features in the already-selected set becomes high. To overcome this problem, an improvement to EMIFS is introduced in this section, which employs the maximum of minimum approach for features-to-vector approximation. This approach extends the calculation of $MI(x_i, s_j)$ to $MI(x_i, S)$. As mentioned by Che, et al. [39], the overestimation of the redundancy term weakens the relevancy term. Therefore, MM-EMIFS applies the maximum of minimum approximation on the redundancy term to mitigate the issue of redundancy overestimation caused by RCGU. This approximation relaxes the redundancy calculation without compromising the relevancy term. As such, the integration of maximum of minimum into EMIFS prevents RCGU from overestimating the redundancy term when the size of the already selected set grows. Consequently, in this study, MM-EMIFS was able to produce better estimation for features' relevancy based on the incomplete data collected during the pre-encryption phase of crypto-ransomware lifecycle.

At the beginning of the selection process, the mutual information value for each feature in the original features set is calculated. The feature with higher MI value is then stored in the selected set. The subsequent features are chosen according to the weight calculated by Eq. (13) using RCGU for significance estimation and MaxMin for feature to vector approximation. More specifically, for each feature in the candidate set, the mutual information is calculated with every feature in the selected set according to Eq. (13). The result is then added into a temporary list, L. This value is kept in the list unless a lower MI value between the same candidate feature and another feature in the selected list was found. If a lower MI value is found, the existing value in L is replaced with the new (lower) MI value. The process is repeated for each candidate feature in F so that one minimum value is added into L at the end of each iteration. When all features in F are exhausted, the list L will contain a number of minimum MI values equal to the number of features in F. The candidate feature corresponding to the highest value in L is then simultaneously added into S and removed from F.

Fig. 5 shows the pseudo code of the proposed MM-EMIFS technique. Let $F = \{f_1, f_2, f_3, \ldots .f_{n-1}, f_n\}$ be the original features vector with n number of features; V is the candidate set that holds

Pseudo Code 1: EMIFS Technique

Input: $F = \{f_1, f_2, \ldots, f_n\}$ original features vector; C class label, τ required features.
Output: $S = \{s_1, s_2, \ldots., s_\rho\}$ the final features set.

1: $V \leftarrow \emptyset$; $S \leftarrow \emptyset$
2: for each feature $f_i \in F$:
3: $v_i = MI(f_i; C)$
4: $V \leftarrow V \cup v_i$
5: $v_k \leftarrow \max(V, MI)$
6: $S \leftarrow v_k$; $V \leftarrow V \backslash \{v_k\}$
7: for $\forall (v_j, s_m)$ with $v_j \in V$ and $s_m \in S$
8: compute $MI(C; s_m|v_j)$
9: $s_p = \underset{v_j \in V}{\text{argmax}}[MI(v_j; C) - \frac{|S|}{|F|}\sum_{s_j \epsilon S} I(v_k; s_j)]$:
10: $V \leftarrow V \backslash \{s_p\}$
11: $S \leftarrow S \cup \{s_p\}$
12: Repeat 8 – 11 while $length(S) \leq \tau$

Fig. 4. Pseudo code of the proposed EMIFS technique.

Pseudo Code 2: MM-EMIFS Technique

Input: $F = \{f_1, f_2, \ldots., f_n$ original features vector; C is the class label, τ is the number of required features.
Output: $S = \{s_1, s_2, \ldots., s_\rho\}$ the final features set.

1: $V \leftarrow \emptyset$; $S \leftarrow \emptyset$
2: for each feature $f_i \in F$:
3: $v_i = MI(f_i; C)$
4: $V \leftarrow V \cup v_i$
5: $v_k \leftarrow \max(V, MI)$
6: $S \leftarrow v_k$; $V \leftarrow V \backslash \{v_k\}$
7: for $\forall (v_j, s_m)$ with $v_j \in V$ and $s_m \in S$
8: compute $MI(C; s_m|v_j)$
9: $s_p = arg\max_{v_j \in V} \min_{s_r \in S}\{EMIFS(C; v_j|s_r)\}$
10: $V \leftarrow V \backslash \{s_p\}$
11: $S \leftarrow S \cup \{s_p\}$
12: Repeat 8 – 11 while $length(S) \leq \tau$

Fig. 5. Pseudo code of MM-EMIFS technique.

the features whose MI values have been already calculated; $S = \{s_1, s_2, \ldots, s_\tau\}$ is the selected set with τ number of features. As shown in lines 1 to 6, the process starts by initializing the empty sets V and S, and calculating the MI value for each feature f_i in F. Based on the MI value, those features are ranked and stored in the set V. The feature v_k in V with max (V, MI) is then simultaneously removed from V and added into S. As shown in the lines from 7 to 11, at each iteration for each feature in the candidate set V, the mutual information between that feature and the class label given each feature in the already-selected set S is calculated according to Eq. (13). The selected feature is then simultaneously added into the already-selected set S and removed from the candidate set V. The process continues until satisfying the number of required features in the already-selected set S.

$$s_p = \arg\max_{v_j \in V} \min_{s_r \in S}\{EMIFS(C; v_j|s_r)\} \quad (13)$$

5. Results analysis and discussion

This section describes the implementation and experimental evaluation of the proposed EMIFS and MM-EMIFS techniques. It starts with an explanation of the dataset used by this study. The experimental results of each technique and the comparison with the related works are then presented and discussed.

5.1. The dataset and experimental environment

To collect the attack's behavioural data at the pre-encryption phase of the crypto-ransomware lifecycle, dynamic analysis was used [60,61]. The rationale for this is that dynamic analysis overcomes the obfuscation and packing techniques that advanced malware uses to resist analysis and evade detection [25,62–64]. The dynamic analysis was carried out in a controlled environment built on top of the sandbox technology [65,66]. The sandbox environment was configured according to [66]. The architecture of the analysis environment consisted of the Cuckoo sandbox as a host installed on Virtual Box. Within the host, an MS Windows X86 32-bit guest machine was created and used as a victim machine that the ransomware would target during the analysis. All the crypto-ransomware samples in the dataset were run one sample at a time. The processes created for the running sample were then hooked by the agent in the guest operating system to capture the APIs called by the ransomware during the runtime and record them into an independent trace file dedicated for that ransomware sample. Therefore, each ransomware sample had its own trace file containing all API calls used by that sample. These APIs contained the behavioural patterns of the ransomware, which could be used to introspect the attack characteristics and extract the latent features by analysing the usage (calls) to certain API functions [20,23,67]. This approach

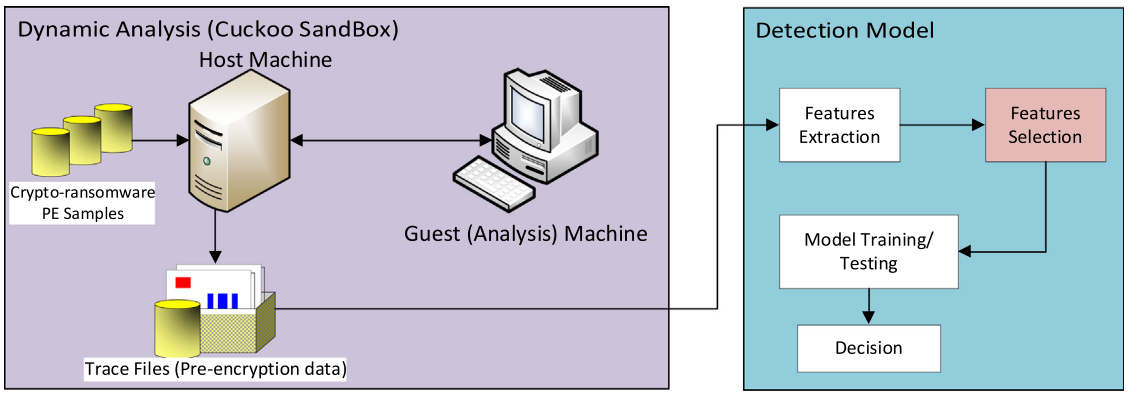


Fig. 6. The dynamic crypto-ransomware analysis and detection process.

Table 3
Crypto-ransomware families used by this study.

Family	Year	Techniques	Target	# samples
Cryptolocker	2013	RSA	User files	6450
Cryptowall	2014	RSA 2048-bit	User files	3309
Cryrar	2012	RAR-sfx	User files	1743
Locky	2016	RSA-2048 + AES-128 cipher with ECB mode	User files	2661
Petya	2016	AES-128	MBR, user files	2846
Reventon	2012	N/A	user files	2798
Teslactypt	2015	ECC	Games and multimedia files	1670
Wannacry	2017	RSA 2048 bit	User files	1899
Cerber	2016	RSA-2048	User files	1203
Filecryptor	2013	2048-bit RSA	User files	3428
Crypt	N/A	N/A	User files	3672
CTB_Locker	2014	ECC	User files	2701
Satana	2016	256-bit AES in ECB	User files	1258
CryptXXX	2016	RSA4096	User files	2934
Sage	2016	AES 256 and RSA 1024	User files	806

is commonly used by many dynamic ransomware and malware detection studies [20,59,60,68–70]. Fig. 6 shows the general architecture of the crypto-ransomware dynamic analysis and detection process.

The ransomware samples used in the experimental evaluation of the techniques proposed in this study were downloaded from virusshare.com, the widely-used public repository of malware [20,23,53,71,72]. We collected 39,378 crypto-ransomware samples. Table 3 shows the full list of ransomware families used by this study, with additional information about each family, including the year of release, encryption technique and the number of samples in each family. Additionally, 16057 benign programs were collected from informer.com, the well-known Windows software repository [20,53,73,74]. Several types of benign applications were downloaded. These types include file tools (e.g. office, developers, paint and multimedia tools), utility programs (e.g. compression, encryption, editing), games, browsers and drivers. Both benign and malicious programs were checked using VirusTotal, the popular malware scanning web service, which uses 56 anti-viruses to confirm the maliciousness of each sample [5,23,75,76]. As in [19,23], the samples that were labelled as malicious by less than 5 Anti-Viruses were excluded. The benign samples were used if they were labelled as trusted by all 56 Anti-Viruses; otherwise they were discarded.

To emulate a real environment, around 925 files, including MS Word documents, Excel, PPT, Visio, PDF, JPG and short video files were stored in different locations within the local storage of the guest machine. These files were attacked during the dynamic analysis when crypto-ransomware samples were executed within the Sandbox environment. The dynamic analysis of both the ransomware and benign programs took place by running them one at a time on the analysing machine (guest machine) in the Cuckoo Sandbox. Once a program is submitted for analysis, the sandbox runs it on the guest machine and hooks the process it creates and records the APIs into a trace file dedicated for that program. During the recording, each API is compared against the pre-encryption boundary vector such that the analysis process will be terminated when a match is found. This is to ensure that all trace files in the dataset contain only the pre-encryption APIs, i.e. the APIs called from the beginning of analysis process until encountering the first cryptography-related API. After each run, the analysis system (guest machine) was reverted back to its original, clean state. The data in trace files constitute the pre-encryption dataset (DS-pre), which was used for crypto-ransomware early detection. Descriptive statistics were used to evaluate the distribution of the pre-encryption data using skewness and kurtosis metrics. For a normally distributed data, the skewness and kurtosis need to be close to zero and three, respectively. In the pre-encryption dataset, the values of skewness and kurtosis were 11.15 and 129.81, respectively. As such, these high values indicate that the data is not normally distributed, which advocates our choice of the information theory-based approach for feature selection for the pre-encryption data, as it does not involve any assumption regarding the distribution of the data.

DS-pre was used to evaluate the accuracy, detection rate and false positive rate of the proposed techniques. The proposed techniques were also evaluated using three additional crypto-ransomware pre-encryption datasets, i.e. DS1, DS2 and DS3, which were built following [20,23,77], respectively. The experimental work, modelling, evaluation and analysis were conducted using several Python-based packages including Sklearn, Pandas, Numpy and SkFeature. The detection accuracy (ACC), detection rate (DR) and False Positive Rate (FPR) were calculated

according to Eqs. (14), (15) and (16), respectively.

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \quad (14)$$

$$DR = \frac{TP}{TP + FN} \quad (15)$$

$$FPR = \frac{FP}{FP + TN} \quad (16)$$

where TP, TN, FP and FN denote true positive, true negative, false positive and false negative, respectively.

5.2. Experimental results of the EMIFS technique

Based on the pre-encryption dataset (DS-pre), EMIFS was used to select the most informative features for the pre-encryption phase of crypto-ransomware lifecycle. The experiments were conducted using several feature sets with different numbers of features, i.e. 5, 10, 15, 20, 25, 30, 35, 40, 45, and 50 features. The dataset was divided into a training set and a testing set using 10-fold cross validation. Several machine learning algorithms were used in this evaluation, including Support Vector Machine (SVM), Logistic Regression (LR), Decision Tree (DT), K-Nearest Neighbour (KNN), Random Forest (RF), AdaBoost, and Multi-Layer Perceptron (MLP). These algorithms were chosen as they are commonly used by the related studies, such as Sgandurra, et al. [20] and Homayoun, et al. [16] and Rhode, et al. [23]. The testing set then was used to determine the classification accuracy of those classifiers according to Eq. (14). To confirm the suitability of the proposed EMIFS technique for use with the other early crypto-ransomware datasets, the same procedures were repeated for the datasets DS1, DS2 and DS3. Each of those datasets was divided into training and testing sets using 10-fold cross validation in the same way that DS-pre was divided. The same set of classifiers used with DS-pre was also used with these datasets. Those classifiers were trained using the training set of each dataset and the classification performance was then measured by detection accuracy, detection rate and FPR, using the testing set.

Tables 4, 5, and 6 show the results of the EMIFS technique on the pre-encryption dataset in terms of detection accuracy, detection rate and False Positive Rate (FPR) of each classifier. Each row in the tables corresponds to one feature set used to train different classifiers. It can be observed from Table 4 that, for most classifiers, the detection accuracy increases with the increase in the size of the feature set, especially for the sets with less than 30 features. When the size of the feature set exceeds 30, the increase becomes less gradual and sometimes the classifiers experience a slight accuracy drop. Likewise, the results in Table 5 show that the detection rate increases proportionally to the number of features until the number of features reaches 25. The detection rate then starts fluctuating. Similar fluctuation could also be noticed with FPR, as shown in Table 6. Furthermore, the averaged (Avg.) accuracy, detection rate and FPR of all feature sets per classifier ranges between 0.9183∼0.9708, 9241∼0.9758, and 0.0764∼0.0247, respectively.

These results emphasize the efficacy of the proposed Redundancy Coefficient Gradual Upweighting (RCGU) used in EMIFS to make better redundancy–relevancy trade-offs than those of related mutual information-based feature selection techniques (see Fig. 7) when dealing with a limited number of attack patterns as is the case in pre-encryption data. This is attributed to the ability of the proposed RCGU to overcome the unavailability of the common characteristics of the already-selected features by individually estimating the redundancy between the candidate feature and each feature in the selected set. Furthermore, the high detection accuracy and detection rate as well as the low FPR that EMIFS starts with indicates that feature relevancy plays the main role at the early stages of the selection process. This is attributed to the low effect that redundant features have due to the small number of features in the selected set at the beginning of the selection process, which decreases the likelihood that the candidate feature is redundant with any of them. This advocates the efficacy of the individual redundancy calculation that RCGU has been built based on. This effectiveness is also evidenced by the way the improvement in classification performance (in terms of accuracy, detection rate and FPR) decelerates when the number of selected features increases, This indicates that the redundant information becomes more influential on the MI calculation when more features are added into the selected feature set, which again supports the assumption that the influence of redundant features starts low then increases gradually when more features are added into the selected set. As such, the proposed RCGU technique deals with this situation more effectively.

Table 4
Detection Accuracy of the EMIFS on pre-encryption dataset (DS-pre) with different sizes of feature sets used to train several classifiers.

DS-Pre	LR	SVM	DT	RF	KNN	AdaBoost	MLP
5	0.9039	0.9286	0.9643	0.9714	0.9511	0.9386	0.9107
10	0.9168	0.9493	0.9661	0.9711	0.9518	0.9432	0.9207
15	0.9203	0.9518	0.9654	0.9707	0.9518	0.9432	0.9239
20	0.9203	0.9514	0.9664	0.9682	0.9518	0.9432	0.9239
25	0.9203	0.9514	0.9654	0.9722	0.9522	0.9432	0.9221
30	0.9203	0.9514	0.9661	0.9707	0.9522	0.9432	0.9207
35	0.9203	0.9514	0.9675	0.9725	0.9518	0.9432	0.9228
40	0.92	0.9514	0.9679	0.9707	0.9518	0.9432	0.9207
45	0.92	0.9514	0.9654	0.9718	0.9511	0.9432	0.9211
50	0.9203	0.9507	0.965	0.969	0.9515	0.9389	0.9243
Avg.	0.9183	0.9489	0.9659	0.9708	0.9517	0.9423	0.9211

Table 5
Detection rate of the proposed EMIFS on pre-encryption dataset (DS-pre) with different sizes of feature sets used to train several classifiers.

DS-Pre	LR	SVM	DT	RF	KNN	AdaBoost	MLP
5	0.9053	0.9353	0.9732	0.9755	0.9520	0.9415	0.9173
10	0.9243	0.9501	0.9664	0.9721	0.9522	0.9496	0.9223
15	0.9249	0.9518	0.9678	0.9738	0.9573	0.9473	0.9282
20	0.9258	0.9594	0.9676	0.9721	0.9523	0.9498	0.9316
25	0.9288	0.9573	0.9709	0.9805	0.9586	0.9520	0.9276
30	0.9278	0.9538	0.9744	0.9773	0.9579	0.9438	0.9209
35	0.9261	0.9580	0.9695	0.9807	0.9574	0.9490	0.9253
40	0.9270	0.9567	0.9762	0.9742	0.9547	0.9485	0.9280
45	0.9283	0.9524	0.9677	0.9745	0.9522	0.9440	0.9267
50	0.9226	0.9522	0.9651	0.9777	0.9528	0.9420	0.9278
Avg.	0.9241	0.9527	0.9699	0.9758	0.9547	0.9468	0.9256

Table 6
False Positive Rate (FPR) of the EMIFS on pre-encryption dataset (DS-pre) with different sizes of feature sets used to train several classifiers.

DS-Pre	LR	SVM	DT	RF	KNN	AdaBoost	MLP
5	0.0951	0.0654	0.0270	0.0249	0.0485	0.0588	0.0835
10	0.0765	0.0500	0.0343	0.0279	0.0478	0.0505	0.0784
15	0.0752	0.0489	0.0325	0.0270	0.0436	0.0529	0.0718
20	0.0749	0.0410	0.0329	0.0283	0.0481	0.0509	0.0687
25	0.0718	0.0432	0.0297	0.0204	0.0422	0.0489	0.0725
30	0.0724	0.0471	0.0256	0.0235	0.0423	0.0569	0.0797
35	0.0743	0.0426	0.0306	0.0200	0.0429	0.0516	0.0749
40	0.0738	0.0435	0.0244	0.0263	0.0460	0.0519	0.0725
45	0.0725	0.0481	0.0329	0.0258	0.0479	0.0566	0.0741
50	0.0774	0.0487	0.0352	0.0227	0.0474	0.0587	0.0725
Avg.	0.0764	0.0479	0.0305	0.0247	0.0457	0.0538	0.0749

5.3. Experimental results of maximum of minimum-based Mutual Information Feature Selection Technique

To evaluate the accuracy of the proposed Maximum of Minimum-based Mutual Information Feature Selection (MM-

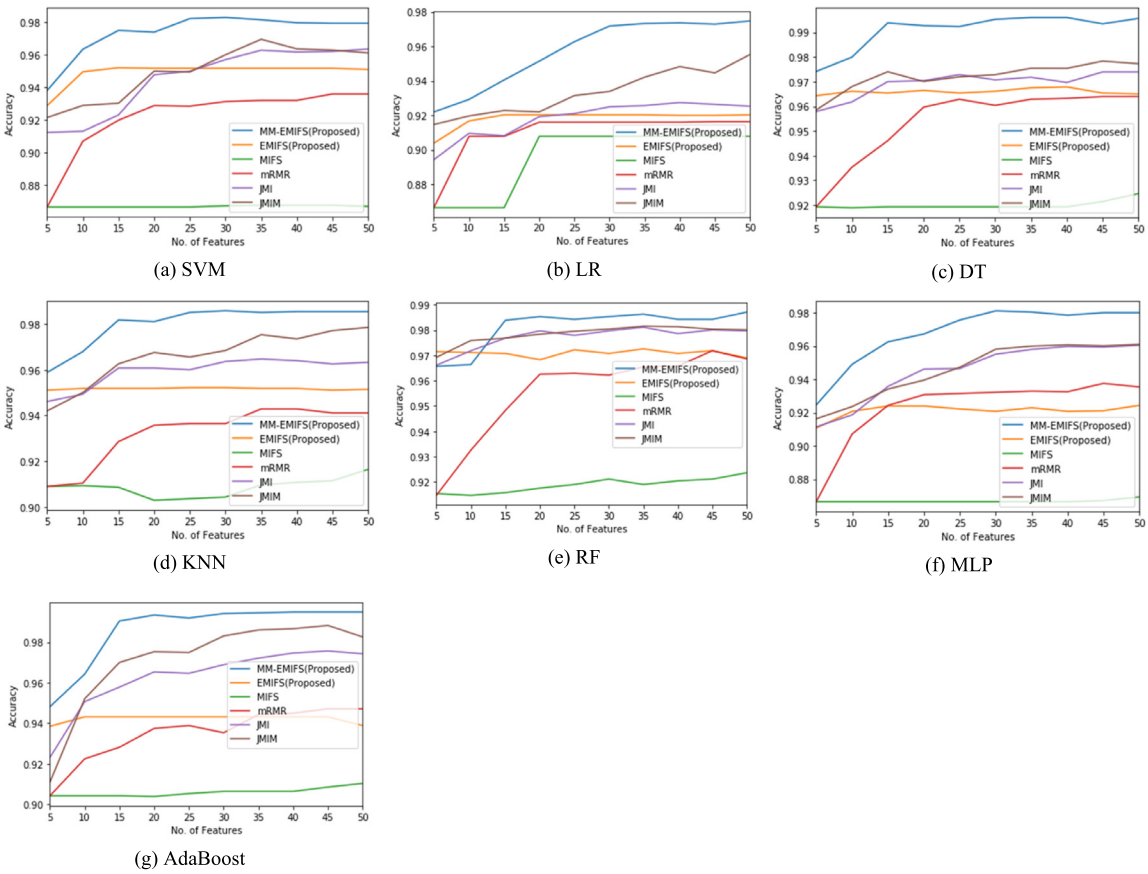


Fig. 7. Comparison of the detection accuracy of the proposed EMIFS and MM_EMIFS with that of the techniques used by related works using different ML algorithms: (a) SVM, (b) LR, (c) DT, (d) KNN, (e) RF, (f) MLP, (g) AdaBoost.

EMIFS) technique, an experimental evaluation was conducted by applying the proposed technique on the pre-encryption dataset (DS-pre). The experiments were carried out using different machine learning classifiers including LR, SVM, DT, RF, KNN, AdaBoost and MLP. Those classifiers are same as those used to evaluate EMIFS in the previous section. Furthermore, several feature sets with different numbers of features ranging between 5 and 50 features were used. The number of features in those sets was similar to those used to evaluate EMIFS. In addition, the dataset was divided into training and testing sets using a 10-fold cross-validation approach. Detection accuracy, detection rate, and FPR were used to measure the performance of the proposed technique.

To confirm the suitability of the proposed MM-EMIFS technique for use with the other crypto-ransomware datasets, the same procedures were repeated for the datasets DS1, DS2 and DS3. Each dataset was divided into training and testing sets using 10-fold cross validation. The same set of classifiers used with DS-pre was also used with those datasets. The classifiers were trained using the training set of each dataset and the accuracy was then measured by detection accuracy, using the test set.

Tables 7, 8, and 9 show the experimental results of the MM-EMIFS technique on the pre-encryption dataset in terms of detection accuracy, detection rate and false positive rate (FPR). It can be observed that, on average, the detection accuracy, detection rate and FPR of all features sets per classifier ranges between 0.9573∼0.9909, 0.9621∼0.9940, and 0.0384∼0.0068, respectively. Furthermore, the classification performance improves proportionally to the number of features until reaching a certain number of features (which varies among the classifiers) before it starts fluctuating. These results show that the integration between the proposed RCGU and the MaxMin technique in MM-EMIFS produced features with better classification accuracy, detection rate and FPR than that of EMIFS. This indicates that such integration helps in maintaining the balance between redundancy overestimation and underestimation. On the one hand, by enhancing the calculation of minimum MI value between the candidate feature and the individual features in the already-selected set, the proposed RCGU improved the maximum of minimum approximation. Such improvement helps to mitigate the effect of redundancy underestimation that the MaxMin approach suffers from [39]. On the other hand, the integration between RCGU and MaxMin prevents the redundancy overestimation that RCGU could cause when the number of features in the already-selected set increases. As such, the feature corresponding to the maximum value of the minimum MIs was more informative than the one selected by the conventional MaxMin technique. This can also be noticed in Fig. 7 which shows that the proposed MM-EMIFS outperformed the conventional maximum of minimum technique. It is worth noting that the fluctuation of MM-EMIFS happens only when the number of selected features used for training the detection model exceeds certain limits. This is related to the overfitting that happens when we train the detection model using a higher number of features.

5.4. Comparison with the related techniques

To show the improvement achieved by the proposed EMIFS and MM-EMIFS techniques over related works, the results were compared with two well-known and widely-used mutual information-based feature selection techniques, Mutual Information Features Selection (MIFS) and Minimum Redundancy Maximum Relevance (mRMR) techniques [78]. The reason for this

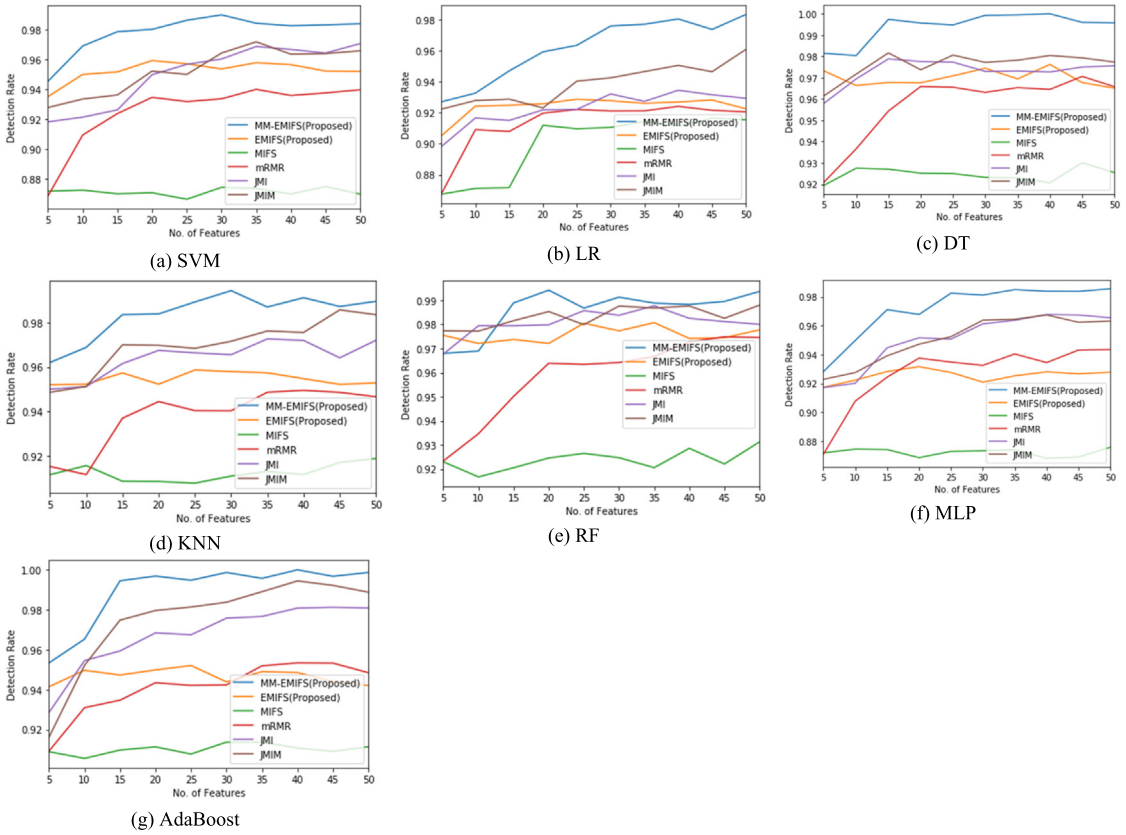


Fig. 8. Comparison of the detection rates of the proposed EMIFS and MM_EMIFS with those of the techniques used by related works using different ML algorithms: (a) SVM, (b) LR, (c) DT, (d) KNN, (e) RF, (f) MLP, (g) AdaBoost.

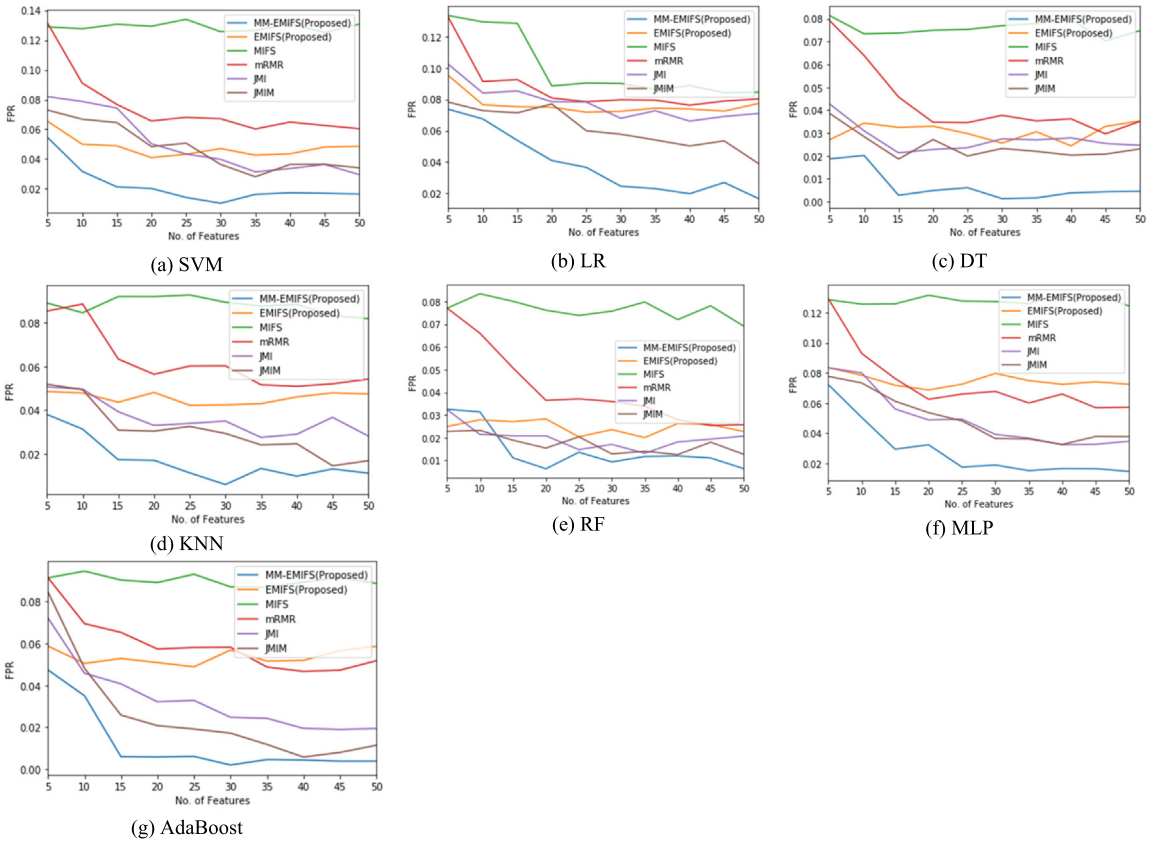


Fig. 9. Comparison of the False Positive Rates (FPRs) of the proposed EMIFS and MM_EMIFS with those of the techniques used by related works using different ML algorithms: (a) SVM, (b) LR, (c) DT, (d) KNN, (e) RF, (f) MLP, (g) AdaBoost.

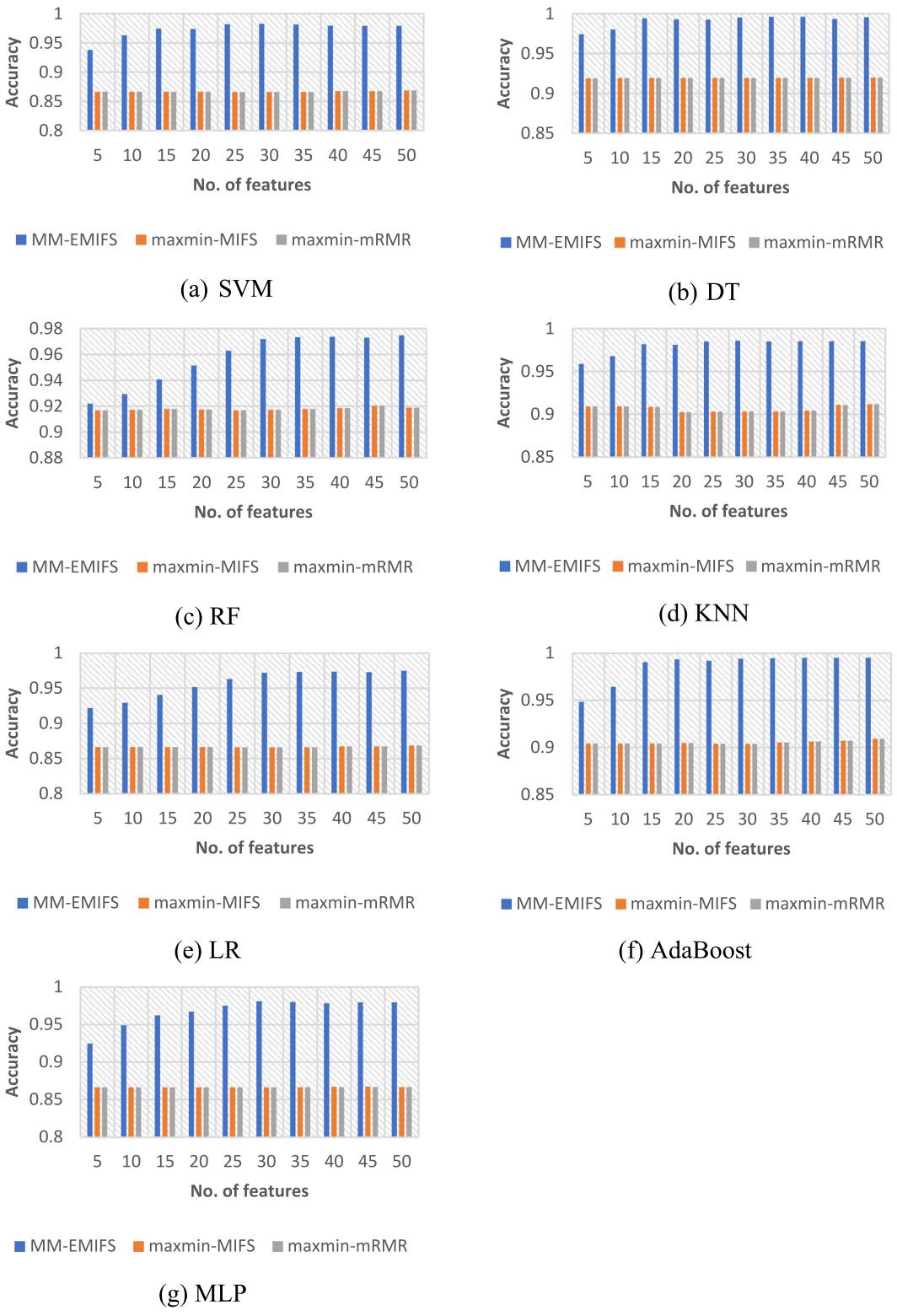


Fig. 10. Comparison of the detection accuracy of the proposed MM-EMIFS with that of MIFS and mRMR implemented using the MaxMin technique.

choice was that the proposed EMIFS and MM-EMIFS resemble these techniques as they do not involve a conditional redundancy calculation. In addition, these techniques were used by several ransomware early detection models [19,20,59].

Several machine learning classifiers were used in this comparison, i.e. LR, SVM, DT, RF, KNN, AdaBoost and MLP. In addition, the experiments were conducted using different sizes of feature sets, ranging from 5 to 50 and incremented by 5 features between each two subsequent sets. Classification accuracy, detection rate and FPR were used to measure classification performance. The comparison results in Figs. 7, 8, and 9 show that EMIFS outperformed MIFS and mRMR, which suggests that the proposed RCGU employed by EMIFS to calculate the value of the redundancy coefficient is more effective than the inversely proportional

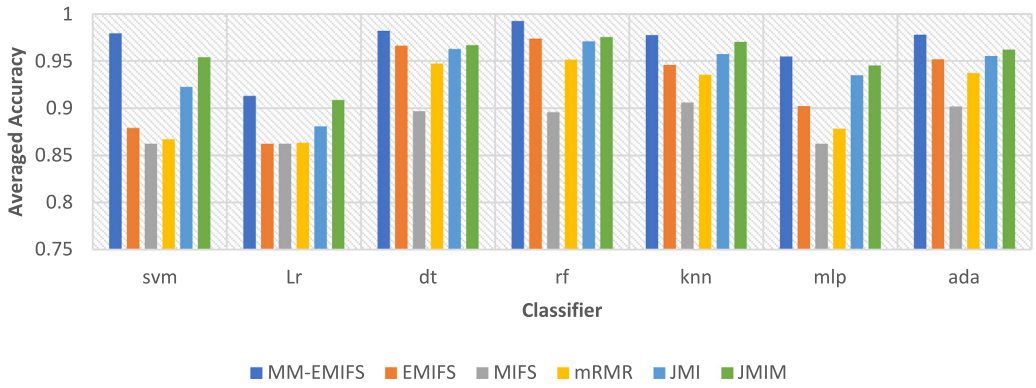


Fig. 11. Comparison of proposed EMIFS and MM-EMIFS with the related techniques in terms of the accuracy averaged over all data subsets (DS1, DS2 and DS3).

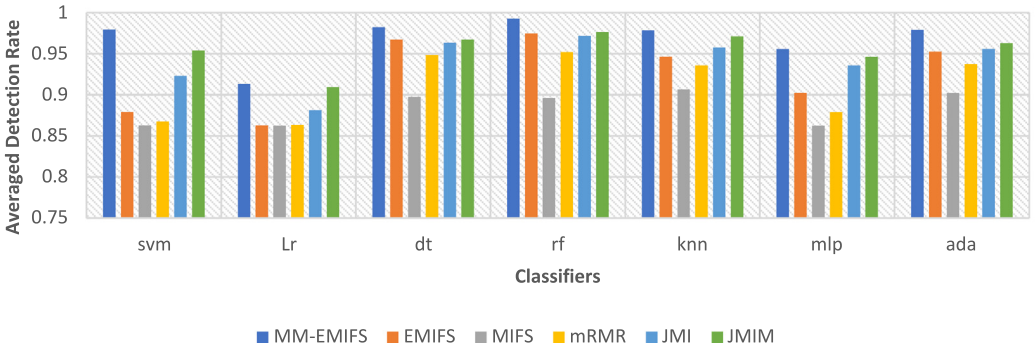


Fig. 12. Comparison of proposed EMIFS and MM-EMIFS with the related techniques in terms of the detection rate averaged over all data subsets (DS1, DS2 and DS3).

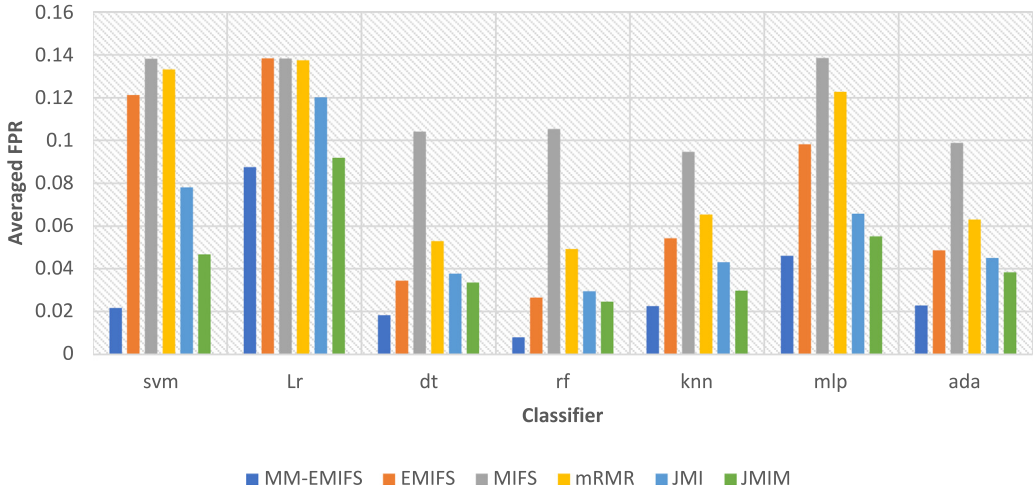


Fig. 13. Comparison of proposed EMIFS and MM-EMIFS with the related techniques in terms of the FPR averaged over all data subsets (DS1, DS2 and DS3).

approach utilized by these previous techniques. This is attributed to the way the RCGU calculates the redundancy term's coefficient such that, at each iteration, it increases the weight of the redundancy term proportionally to the number of features in the selected set as opposed to the inversely-proportional approach employed by related techniques. In addition, the comparison shows that, with most classifiers, EMIFS outperformed JMI when small numbers of features (5 and 10 features) were used. It can also be noticed that the improvement in the detection accuracy of EMIFS starts to either fluctuate or becomes less gradual when the number of features in the already-selected set exceeds a certain limit (which varies among classifiers). This is due to the redundancy overestimation that RCGU could cause when the size of already-selected set grows. This issue of such redundancy overestimation has been alleviated by the proposed MM-EMIFS which integrates the proposed RCGU with the MaxMin technique to balance the redundancy estimation when the number of features in the already-selected set increases.

In addition to MIFS and mRMR, Figs. 7, 8, and 9 also show the comparison of the accuracy, detection rate and FPR of the proposed MM-EMIFS technique with that of the Joint Mutual Information (JMI) and Joint Mutual Information Maximization (JMIM) techniques. JMI was chosen for this comparison as it has been reported as the most accurate and stable mutual information-based feature selection technique [38,40]. Likewise, the JMIM was

Table 7
Detection accuracy for the detection accuracy of MM-EMIFS on pre-encryption dataset (DS-pre) with different sizes of feature sets used to train several classifiers.

DS-Pre	LR	SVM	DT	RF	KNN	AdaBoost	MLP
5	0.922	0.9379	0.9741	0.9656	0.9588	0.9482	0.9245
10	0.9293	0.9631	0.9799	0.9663	0.9679	0.9643	0.949
15	0.9406	0.9747	0.9938	0.9838	0.9818	0.9905	0.9624
20	0.9515	0.9736	0.9927	0.9853	0.981	0.9934	0.9672
25	0.9628	0.9819	0.9923	0.9842	0.9851	0.992	0.9756
30	0.9719	0.9826	0.9952	0.9853	0.9858	0.9942	0.981
35	0.9734	0.9812	0.996	0.9863	0.985	0.9945	0.9803
40	0.9737	0.9794	0.996	0.9842	0.9854	0.9949	0.9785
45	0.973	0.979	0.9934	0.9842	0.9854	0.9949	0.9799
50	0.9748	0.979	0.9956	0.9871	0.9854	0.9949	0.9799
Avg.	0.9573	0.97324	0.9909	0.98123	0.98016	0.98618	0.96783

Table 8
Detection rate of the MM-EMIFS on pre-encryption dataset (DS-pre) with different sizes of feature sets used to train several classifiers.

DS-Pre	LR	SVM	DT	RF	KNN	AdaBoost	MLP
5	0.9272	0.9455	0.9815	0.9680	0.9620	0.9534	0.9281
10	0.9327	0.9692	0.9804	0.9690	0.9688	0.9653	0.9498
15	0.9471	0.9787	0.9974	0.9889	0.9835	0.9945	0.9711
20	0.9594	0.9803	0.9957	0.9942	0.9838	0.9969	0.9678
25	0.9636	0.9864	0.9948	0.9868	0.9892	0.9948	0.9825
30	0.9760	0.9900	0.9993	0.9913	0.9942	0.9987	0.9812
35	0.9772	0.9844	0.9995	0.9888	0.9869	0.9957	0.9850
40	0.9806	0.9828	1.0000	0.9883	0.9911	1.0000	0.9838
45	0.9739	0.9834	0.9960	0.9895	0.9871	0.9968	0.9837
50	0.9835	0.9841	0.9958	0.9937	0.9895	0.9987	0.9856
Avg.	0.9621	0.9785	0.9940	0.9858	0.9836	0.9895	0.9719

Table 9
False Positive Rate (FPR) of the MM-EMIFS on pre-encryption dataset (DS-pre) with different sizes of feature sets used to train several classifiers.

DS-Pre	LR	SVM	DT	RF	KNN	AdaBoost	MLP
5	0.0736	0.0545	0.0187	0.0325	0.0380	0.0474	0.0723
10	0.0675	0.0317	0.0202	0.0314	0.0313	0.0350	0.0504
15	0.0538	0.0213	0.0027	0.0111	0.0173	0.0058	0.0294
20	0.0410	0.0202	0.0048	0.0063	0.0170	0.0057	0.0323
25	0.0367	0.0142	0.0060	0.0136	0.0113	0.0060	0.0175
30	0.0246	0.0103	0.0013	0.0093	0.0060	0.0019	0.0189
35	0.0231	0.0162	0.0016	0.0117	0.0133	0.0044	0.0152
40	0.0199	0.0173	0.0038	0.0120	0.0097	0.0043	0.0166
45	0.0270	0.0170	0.0043	0.0111	0.0131	0.0037	0.0165
50	0.0167	0.0165	0.0045	0.0064	0.0112	0.0037	0.0146
Avg.	0.0384	0.0219	0.0068	0.0145	0.0168	0.0118	0.0284

chosen as it is the state-of-the-art mutual information-based feature selection technique that employs the MaxMin approximation for the redundancy overestimation problem [40]. The comparison results show that the proposed MM-EMIFS outperforms MIFS, mRMR, JMI and JMIM. This confirms the ability of the proposed RCGU technique to overcome the data insufficiency in the pre-encryption phase of crypto-ransomware attacks and calculate the redundancy term more accurately. The results also show that MM-EMIFS outperformed the EMIFS and generated higher classification accuracy, higher detection rate and lower FPR even when the number of features in the already-selected set increased. This is attributed to the integration between the RCGU technique and the MaxMin in the MM-EMIFS, which addressed the redundancy overestimation caused by RCGU in EMIFS when the number of features in the already-selected set increased. Thus, it enabled the MaxMin to make better feature-to-vector approximation, which in turn balanced the redundancy estimation with RCGU.

It is worth noting that the classification accuracy, detection rate and FPR of the proposed MM-EMIFS were better than those

Table 10
Comparison of significance test (t-test) results of the proposed MM-EMIFS technique and the related techniques (JMI and JMIM) with different classification algorithms.

Classifiers	MM-EMIFS With JMI	MM-EMIFS With JMIM
LR	3.91E−07	1.79E−05
SVM	4.03E−05	2.42E−05
DT	2.58E−09	1.86E−08
RF	0.005	0.05
KNN	4.69E−09	2.66E−06
adaBoost	1.22E−07	0.00024
MLP	1.27E−07	7.24E−07

of the JMI and JMIM techniques, which involve the conditional redundancy term in the MI calculation. This confirms that the insufficiency of data and attack patterns at the early (pre-encryption) phase of the crypto-ransomware lifecycle degrades the ability of the conditional redundancy term (used in JMI and JMIM) to calculate accurately the MI between several variables (the candidate feature, the already-selected features and the class label). Fig. 10 shows that the employment of only the MaxMin approximation has no effect on the accuracy of the feature selection techniques. However, the accuracy improved significantly once this approximation had been integrated with the proposed RCGU. This confirms the ability of RCGU to overcome the data insufficiency challenge during the early phases of the crypto-ransomware lifecycle and make better redundancy–relevancy trade-offs, which improves the performance of detection model.

As the proposed EMIFS and MM-EMIFS exclude the conditional redundancy term from the mutual information calculation, the time complexity is similar to that of the MIFS, which is expressed in the Big O notation as $O(N \log N)$ [79]. Concretely, the proposed EMIFS and MM-EMIFS calculate the mutual information between the feature and class label (for feature relevance) and the mutual information between the candidate feature and the features in the already-selected set (for redundancy) in the same step. Accordingly, the time complexity of both EMIFS and MM-EMIFS is $O(N \log N)$, as well. Therefore, the proposed RCGU technique improves the performance of the early detection of attacks while maintaining the same time complexity as existing techniques.

The accuracy performance of the proposed EMIFS and MM-EMIFS techniques was also evaluated against the related works using the datasets DS1, DS2 and DS3 used in the related works [20,23,77]. Figs. 11–13 show the comparisons of the accuracy, detection rate and FPR averaged over the three data sets. The comparison shows that MM-EMIFS achieved better accuracy, detection rate and FPR with all classifiers compared to the related techniques. This confirms the ability of MM-EMIFS to select the features with the highest discriminative power, even when attack's data and patterns are not sufficient, thanks to the proposed RCGU technique that makes an accurate redundancy–relevancy trade-off. Moreover, the significance test results in Table 10 using the t-test show that, with all classifiers, the p-value was less than or equal to 0.05 (the standard value), which confirms that the improvement in the detection performance achieved by the proposed RCGU was statistically significant compared to the related techniques.

The future improvements that could be added into the proposed RCGU is the inclusion of the conditional redundancy term into the feature's significance calculation within the goal function. Such a calculation might improve the estimation of feature significance even further, as it involves comparison of the candidate feature with the already-selected features with respect to the target class label. Therefore, the decision to include or exclude the feature will be based on how much it contributes to the class label.

6. Conclusion

In this paper, the Redundancy Coefficient Gradual Upweighting (RCGU) technique was proposed for better estimation of the significance of a crypto-ransomware feature when the amount of data is insufficient during the early phases of the attack lifecycle. The proposed RCGU was integrated into the goal function of the Mutual Information Features Selection and two improved feature selection techniques for early detection of crypto-ransomware were proposed: EMIFS and MM-EMIFS. The integration of the RCGU technique improved the redundancy–relevancy trade-off. The detection accuracy of the RCGU-aided feature selection techniques (EMIFS and MM-EMIFS) were higher than those of the related techniques. The results show the efficacy of the proposed RCGU for crypto-ransomware early detection. This technique could be applied for early detection of other attacks such as malware detection and intrusion detection. One limitation of the proposed RCGU technique is the lack of consideration of the conditional redundancy term when calculating feature significance. Currently, we are working on applying a similar approach on the conditional redundancy term to further improve the feature selection process and enhance detection accuracy.

CRediT authorship contribution statement

Bander Ali Saleh Al-rimy: Conceptualization, Methodology, Data curation, Writing - original draft, Software, Visualization, Investigation, Writing- review & editing, Formal analysis. **Mohd Aizaini Maarof:** Supervision. **Mamoun Alazab:** Writing - review & editing, Visualization. **Syed Zainudeen Mohd Shaid:** Supervision. **Fuad A. Ghaleb:** Validation, Conceptualization, Investigation. **Abdulmohsen Almalawi:** Project administration. **Abdullah Marish Ali:** Funding acquisition. **Tawfik Al-Hadhrami:** Writing - review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This project was funded by the Deanship of Scientific Research (DSR), King Abdulaziz University, Jeddah, Saudi Arabia, under grant No. (DF-274-611-1441). The authors, therefore, gratefully acknowledge DSR technical and financial support.

References

[1] D. Vasan, M. Alazab, S. Wassan, H. Naeem, B. Safaei, Q. Zheng, IMCFN: Image-based malware classification using fine-tuned convolutional neural network architecture, Comput. Netw. 171 (2020) 107138, http://dx.doi.org/10.1016/j.comnet.2020.107138.

[2] B.A.S. Al-rimy, M.A. Maarof, S.Z.M. Shaid, Ransomware threat success factors, taxonomy, and countermeasures: A survey and research directions, Comput. Secur. 74 (2018) 144–166.

[3] A. Azab, R. Layton, M. Alazab, J. Oliver, Mining malware to detect variants, in: 2014 Fifth Cybercrime and Trustworthy Computing Conference, 24–25 Nov. 2014, 2014, pp. 44–53, http://dx.doi.org/10.1109/CTC.2014.11.

[4] I. Yaqoob, et al., The rise of ransomware and emerging security challenges in the internet of things, Comput. Netw. 129 (2017) 444–458, http://dx.doi.org/10.1016/j.comnet.2017.09.003 (in English).

[5] J. Chen, C.H. Wang, Z.M. Zhao, K. Chen, R.Y. Du, G.J. Ahn, Uncovering the face of android ransomware: Characterization and real-time detection, IEEE Trans. Inf. Forensics Secur. 13 (5) (2018) 1286–1300, http://dx.doi.org/10.1109/Tifs.2017.2787905.

[6] A. Azmoodeh, A. Dehghantanha, M. Conti, K.-K.R. Choo, Detecting crypto-ransomware in IoT networks based on energy consumption footprint, J. Ambient Intell. Humaniz. Comput. 9 (4) (2017) 1141–1152, http://dx.doi.org/10.1007/s12652-017-0558-5.

[7] S.D. Yalew, G.Q. Maguire, S. Haridi, M. Correia, Hail to the thief: Protecting data from mobile ransomware with ransomsafedroid, in: 2017 IEEE 16th International Symposium on Network Computing and Applications, Vol. 2017-January, NCA 2017, 2017, pp. 1–8, http://dx.doi.org/10.1109/NCA.2017.8171377, [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85046532213&doi=10.1109%2fNCA.2017.8171377&partnerID=40&md5=5d13566de417b1241f43c27f7373b315.

[8] M. Alazab, M. Alazab, A. Shalaginov, A. Mesleh, A. Awajan, Intelligent mobile malware detection using permission requests and API calls, Future Gener. Comput. Syst. 107 (2020) 509–521, http://dx.doi.org/10.1016/j.future.2020.02.002.

[9] N. Etaher, G.R.S. Weir, M. Alazab, From ZeuS to zitmo: Trends in banking malware, in: 2015 IEEE Trustcom/BigDataSE/ISP, Vol. 1, 20–22 Aug. 2015, 2015, pp. 1386–1391, http://dx.doi.org/10.1109/Trustcom.2015.535.

[10] J.A. Gomez-Hernandez, L. Alvarez-Gonzalez, P. Garcia-Teodoro, R-Locker: Thwarting ransomware action through a honeyfile-based approach, Comput. Secur. 73 (2018) 389–398, http://dx.doi.org/10.1016/j.cose.2017.11.019 (in English).

[11] R. Moussaileb, B. Bouget, A. Palisse, H. Le Bouder, N. Cuppens, J.L. Lanet, Ransomware's early mitigation mechanisms, in: presented at the 13th International Conference on Availability, Reliability and Security, ARES 2018, 2018, Conference Paper. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85055288709&doi=10.1145%2f3230833.3234691&partnerID=40&md5=fd3fa38ed1fb15bb45641bb3db029589.

[12] A. Kharraz, S. Arshad, C. Mulliner, W. Robertson, E. Kirda, UNVEIL: A large-scale automated approach to detecting ransomware, in: Proceedings of the 25th Usenix Security Symposium, 2016, pp. 757–772. [Online]. Available: <Go to ISI>://WOS:000385263000045.

[13] C. Everett, Ransomware: To pay or not to pay? Comput. Fraud Secur. 2016 (4) (2016) 8–12, http://dx.doi.org/10.1016/S1361-3723(16)30036-7, (in English).

[14] A. Kharraz, W. Robertson, D. Balzarotti, L. Bilge, E. Kirda, Cutting the gordian knot: A look under the hood of ransomware attacks, in: 12th International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment, Vol. 9148, DIMVA 2015, 2015, pp. 3–24.

[15] Kaspersky, Ransomware 2018–2020, 2018, https://media.kasperskycontenthub.com/wp-content/uploads/sites/100/2020/05/12075747/KSN-article_Ransomware-in-2018-2020-1.pdf. (Accessed 11 July 2020).

[16] S. Homayoun, A. Dehghantanha, M. Ahmadzadeh, S. Hashemi, R. Khayami, Know abnormal find evil: Frequent pattern mining for ransomware threat hunting and intelligence, IEEE Trans. Emerg. Top. Comput. (2017).

[17] A. Cohen, N. Nissim, Trusted detection of ransomware in a private cloud using machine learning methods leveraging meta-features from volatile memory, Expert Syst. Appl. 102 (2018) 158–178, http://dx.doi.org/10.1016/j.eswa.2018.02.039 (in English).

[18] E. Berrueta, D. Morato, E. Magaña, M. Izal, A survey on detection techniques for cryptographic ransomware, IEEE Access 7 (2019) 144925–144944, http://dx.doi.org/10.1109/ACCESS.2019.2945839.

[19] B.A.S. Al-rimy, M.A. Maarof, S.Z.M. Shaid, Crypto-ransomware early detection model using novel incremental bagging with enhanced semi-random subspace selection, Future Gener. Comput. Syst. 101 (2019) 476–491.

[20] D. Sgandurra, L. Muñoz González, R. Mohsen, E.C. Lupu, Automated dynamic analysis of ransomware: Benefits, limitations and use for detection, 2016, arXiv preprint arXiv:1609.03020.

[21] S. Homayoun, et al., DRTHIS: Deep ransomware threat hunting and intelligence system at the fog layer, Future Gener. Comput. Syst. Int. J. Esci. 90 (2019) 94–104, http://dx.doi.org/10.1016/j.future.2018.07.045 (in English).

[22] M. Alam, S. Bhattacharya, D. Mukhopadhyay, A. Chattopadhyay, RAPPER: Ransomware prevention via performance counters, 2018, arXiv preprint arXiv:1802.03909.

[23] M. Rhode, P. Burnap, K. Jones, Early-stage malware prediction using recurrent neural networks, Comput. Secur. 77 (2018) 578–594, http://dx.doi.org/10.1016/j.cose.2018.05.010 (in English).

[24] B.A.S. Al-rimy, M.A. Maarof, Y.A. Prasetyo, S.Z.M. Shaid, A.F.M. Ariffin, Zero-day aware decision fusion-based model for crypto-ransomware early detection, Int. J. Integr. Eng. 10 (6) (2018).

[25] Y. Ye, T. Li, D. Adjeroh, S.S. Iyengar, A survey on malware detection using data mining techniques, ACM Comput. Surv. 50 (3) (2017) 1–40, http://dx.doi.org/10.1145/3073559.

[26] H. Peng, J. Wei, W. Guo, Micro-architectural features for malware detection, in: Conference, Springer, 2016, pp. 48–60.

[27] J. Stiborek, T. Pevny, M. Rehak, Multiple instance learning for malware classification, Expert Syst. Appl. 93 (2018) 346–357, http://dx.doi.org/10.1016/j.eswa.2017.10.036 (in English).

[28] A. Azab, M. Alazab, M. Aiash, Machine learning based botnet identification traffic, in: 2016 IEEE Trustcom/BigDataSE/ISPA, 23–26 Aug. 2016, 2016, pp. 1788–1794, http://dx.doi.org/10.1109/TrustCom.2016.0275.

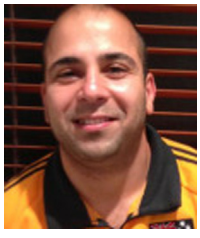
[29] J. Li, et al., Feature selection: A data perspective, ACM Comput. Surv. 50 (6) (2017) 1–45, http://dx.doi.org/10.1145/3136625.

[30] S. Fallahpour, E.N. Lakvan, M.H. Zadeh, Using an ensemble classifier based on sequential floating forward selection for financial distress prediction problem, J. Retailing Consum. Serv. 34 (2017) 159–167, http://dx.doi.org/10.1016/j.jretconser.2016.10.002.
[31] T. Reineking, Active classification using belief functions and information gain maximization, Internat. J. Approx. Reason. 72 (Suppl. C) (2016) 43–54, http://dx.doi.org/10.1016/j.ijar.2015.12.005.
[32] M.H. Aghdam, P. Kabiri, Feature selection for intrusion detection system using ant colony optimization, Int. J. Netw. Secur. 18 (3) (2016) 420–432.
[33] S.S. Hansen, T.M.T. Larsen, M. Stevanovic, J.M. Pedersen, An approach for detection and family classification of malware based on behavioral analysis, in: 2016 International Conference on Computing, Networking and Communications, ICNC, 15–18 Feb. 2016, 2016, pp. 1–5, http://dx.doi.org/10.1109/ICCNC.2016.7440587, [Online]. Available: http://ieeexplore.ieee.org/ielx7/7430154/7440540/07440587.pdf?tp=&arnumber=7440587&isnumber=7440540.
[34] N. Nissim, Y. Lapidot, A. Cohen, Y. Elovici, Trusted system-calls analysis methodology aimed at detection of compromised virtual machines using sequential mining, Knowl. Based Syst. 153 (2018) 147–175, http://dx.doi.org/10.1016/j.knosys.2018.04.033 (in English).
[35] Y. Wang, J. Wang, H. Liao, H. Chen, An efficient semi-supervised representatives feature selection algorithm based on information theory, Pattern Recognit. 61 (Suppl. C) (2017) 511–523, http://dx.doi.org/10.1016/j.patcog.2016.08.011.
[36] H. Liu, J. Sun, L. Liu, H. Zhang, Feature selection with dynamic mutual information, Pattern Recognit. 42 (7) (2009) 1330–1339, http://dx.doi.org/10.1016/j.patcog.2008.10.028.
[37] H. Zhou, Y. Zhang, Y. Zhang, H. Liu, Feature selection based on conditional mutual information: minimum conditional relevance and minimum conditional redundancy, Appl. Intell. 49 (3) (2019) 883–896, http://dx.doi.org/10.1007/s10489-018-1305-0.
[38] G. Brown, A. Pocock, M.J. Zhao, M. Luján, Conditional likelihood maximisation: A unifying framework for information theoretic feature selection, J. Mach. Learn. Res. 13 (2012) 27–66 (in English). [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84863403768&partnerID=40&md5=40bf62f9b175be25ba88410ea4584e9d.
[39] J. Che, Y. Yang, L. Li, X. Bai, S. Zhang, C. Deng, Maximum relevance minimum common redundancy feature selection for nonlinear data, Inform. Sci. 409 (Suppl. C) (2017) 68–86, http://dx.doi.org/10.1016/j.ins.2017.05.013.
[40] M. Bennasar, Y. Hicks, R. Setchi, Feature selection using joint mutual information maximisation, Expert Syst. Appl. 42 (22) (2015) 8520–8532, http://dx.doi.org/10.1016/j.eswa.2015.07.007.
[41] S. Das, Y. Liu, W. Zhang, M. Chandramohan, Semantics-based online malware detection: Towards efficient real-time protection against malware, IEEE Trans. Inf. Forensics Secur. 11 (2) (2016) 289–302, http://dx.doi.org/10.1109/tifs.2015.2491300.
[42] C. Benzaid, K. Lounis, A. Al-Nemrat, N. Badache, M. Alazab, Fast authentication in wireless sensor networks, Future Gener. Comput. Syst. 55 (2016) 362–375, http://dx.doi.org/10.1016/j.future.2014.07.006.
[43] A. Mamoun, V. Sitalakshmi, W. Paul, A. Moutaz, Information security governance: The art of detecting hidden malware, in: M. Daniel, S. Luis Enrique, F.-M. Eduardo, G.P. Mario (Eds.), IT Security Governance Innovations: Theory and Research, IGI Global, Hershey, PA, USA, 2013, pp. 293–315.
[44] F.A. Ghaleb, M.A. Maarof, A. Zainal, B.A.S. Al-Rimy, F. Saeed, T. Al-Hadhrami, Hybrid and multifaceted context-aware misbehavior detection model for vehicular ad hoc network, IEEE Access 7 (2019) 159119–159140, http://dx.doi.org/10.1109/ACCESS.2019.2950805.
[45] F.A. Ghaleb, M.A. Maarof, A. Zainal, B.A.S. Al-rimy, A. Alsaeedi, W. Boulila, Ensemble-based hybrid context-aware misbehavior detection model for vehicular ad hoc network, Remote Sens. 11 (23) (2019) 2852, [Online]. Available: https://www.mdpi.com/2072-4292/11/23/2852.
[46] F.A. Ghaleb, et al., Misbehavior-aware on-demand collaborative intrusion detection system using distributed ensemble learning for VANET, Electronics 9 (9) (2020) 1411.
[47] S. Song, B. Kim, S. Lee, The effective ransomware prevention technique using process monitoring on android platform, Mobile Inf. Syst. 2016 (2016) http://dx.doi.org/10.1155/2016/2946735, Artn 2946735 (in English).
[48] M.M.A.H.R. Shahriari, 2entFOX: A framework for high survivable ransomwares detection, in: Presented At the Information Security and Cryptology (ISCISC), 2015 12th International Iranian Society of Cryptology Conference on Information Security and Cryptology, University of Guilan, Rasht, Iran, 2016.
[49] F. Mbol, J.-M. Robert, A. Sadighian, An efficient approach to detect torrentlocker ransomware in computer systems, in: S. Foresti, G. Persiano (Eds.), Cryptology and Network Security: 15th International Conference, CANS 2016, Milan, Italy, November 14–16, 2016, Proceedings, Springer International Publishing, Cham, 2016, pp. 532–541.
[50] N. Scaife, H. Carter, P. Traynor, K.R. Butler, CryptoLock (and Drop It): Stopping ransomware attacks on user data, in: Presented at the Distributed Computing Systems (ICDCS), 2016 IEEE 36th International Conference on, 2016.
[51] D. Morato, E. Berrueta, E. Magaña, M. Izal, Ransomware early detection by the analysis of file sharing traffic, J. Netw. Comput. Appl. 124 (2018) 14–32, http://dx.doi.org/10.1016/j.jnca.2018.09.013.
[52] M.A. Sotelo Monge, J.M. Vidal, L.J. García Villalba, A novel self-organizing network solution towards crypto-ransomware mitigation, in: 13th International Conference on Availability, Reliability and Security, ARES 2018, Association for Computing Machinery, 2018, p. 48, http://dx.doi.org/10.1145/3230833.3233249, [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85055254545&doi=10.1145%2f3230833.3233249&partnerID=40&md5=9084df3e1dd9e3ce2a2cd658ca263229.
[53] Z.-G. Chen, H.-S. Kang, S.-N. Yin, S.-R. Kim, Automatic ransomware detection and analysis based on dynamic API calls flow graph, in: Presented at the Proceedings of the International Conference on Research in Adaptive and Convergent Systems, Krakow, Poland, 2017.
[54] Q. Chen, R.A. Bridges, Automated behavioral analysis of malware a case study of wannacry ransomware, 2017, arXiv preprint arXiv:1709.08753.
[55] S. Mehnaz, A. Mudgerikar, E. Bertino, RWGuard: A real-time detection system against cryptographic ransomware, LNCS, in: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 11050, 2018, pp. 114–136.
[56] K. Cabaj, P. Gawkowski, K. Grochowski, D. Osojca, Network activity analysis of CryptoWall ransomware, Prz. Elektrotech. 91 (11) (2015) 201–204, http://dx.doi.org/10.15199/48.2015.11.48 (in English).
[57] K. Cabaj, M. Gregorczyk, W. Mazurczyk, Software-defined networking-based crypto ransomware detection using HTTP traffic characteristics, Comput. Electr. Eng. 66 (2017) 353–368, http://dx.doi.org/10.1016/j.compeleceng.2017.10.012.
[58] G. Cusack, O. Michel, E. Keller, Machine Learning-Based Detection of Ransomware Using SDN, in: Presented at the Proceedings of the 2018 ACM International Workshop on Security in Software Defined Networks & Network Function Virtualization, Tempe, AZ, USA, 2018.
[59] B.A.S. Al-rimy, M.A. Maarof, S.Z.M. Shaid, A 0-day aware crypto-ransomware early behavioral detection framework, in: F. Saeed, N. Gazem, S. Patnaik, A.S. Saed Balaid, F. Mohammed (Eds.), Recent Trends in Information and Communication Technology: Proceedings of the 2nd International Conference of Reliable Information and Communication Technology, IRICT 2017, Springer International Publishing, Cham, 2017, pp. 758–766.
[60] S. Maniath, A. Ashok, P. Poornachandran, V.G. Sujadevi, A.U.P. Sankar, S. Jan, Deep learning LSTM based ransomware detection, in: 2017 Recent Developments in Control, Automation & Power Engineering, RDCAPE, 26–27 Oct. 2017, 2017, pp. 442–446, http://dx.doi.org/10.1109/RDCAPE.2017.8358312.
[61] B.A.S. Al-rimy, et al., A pseudo feedback-based annotated TF-IDF technique for dynamic crypto-ransomware pre-encryption boundary delineation and features extraction, IEEE Access (2020) 1, http://dx.doi.org/10.1109/ACCESS.2020.3012674.
[62] B. Yu, Y. Fang, Q. Yang, Y. Tang, L. Liu, A survey of malware behavior description and analysis, Front. Inf. Technol. Electron. Eng. 19 (5) (2018) 583–603, http://dx.doi.org/10.1631/Fitee.1601745 (in English).
[63] Y.A. Ahmed, B. Koçer, S. Huda, B.A. Saleh Al-rimy, M.M. Hassan, A system call refinement-based enhanced minimum redundancy maximum relevance method for ransomware early detection, J. Netw. Comput. Appl. 167 (2020) 102753, http://dx.doi.org/10.1016/j.jnca.2020.102753.
[64] Y.A. Ahmed, B. Koçer, B.A.S. Al-rimy, Automated analysis approach for the detection of high survivable ransomware, KSII Trans. Internet Inf. Syst. (TIIS) 14 (5) (2020) 2236–2257.
[65] S.M. Bidoki, S. Jalili, A. Tajoddin, Pbmmd: A novel policy based multi-process malware detection, Eng. Appl. Artif. Intell. 60 (2017) 57–70, http://dx.doi.org/10.1016/j.engappai.2016.12.008 (in English).
[66] C. Rossow, et al., Prudent practices for designing malware experiments: Status quo and outlook, in: 2012 IEEE Symposium on Security and Privacy, 20–23 May 2012, 2012, pp. 65–79, http://dx.doi.org/10.1109/SP.2012.14.
[67] N. Hampton, Z. Baig, S. Zeadally, Ransomware behavioural analysis on windows platforms, J. Inf. Secur. Appl. 40 (2018) 44–51, http://dx.doi.org/10.1016/j.jisa.2018.02.008 (in English).
[68] D.B. Prelipcean, A.S. Popescu, D.T. Gavrilut, Improving malware detection response time with behavior-based statistical analysis techniques, in: Proceedings - 17th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, SYNASC 2015, 2015, pp. 232–239, http://dx.doi.org/10.1109/SYNASC.2015.44, [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84964801508&partnerID=40&md5=54432560c8d0b3b1e23fe25c22a081a9.
[69] H.S. Galal, Y.B. Mahdy, M.A. Atiea, Behavior-based features model for malware detection, J. Comput. Virol. Hacking Tech. 12 (2) (2016) 59–67, http://dx.doi.org/10.1007/s11416-015-0244-0.
[70] D. Vasan, M. Alazab, S. Wassan, B. Safaei, Q. Zheng, Image-based malware classification using ensemble of CNN architectures (IMCEC), Comput. Secur. 92 (2020) 101748, http://dx.doi.org/10.1016/j.cose.2020.101748.

[71] C. Le Guernic, A. Legay, Ransomware and the legacy crypto API, in: Risks and Security of Internet and Systems: 11th International Conference, CRiSIS 2016, Roscoff, France, September 5–7, 2016, Revised Selected Papers, Vol. 10158, Springer, 2017, p. 11.
[72] J.B. Christensen, N. Beuschau, Ransomware detection and mitigation tool, 2017.
[73] A. Ioanid, C. Scarlat, G. Militaru, The effect of cybercrime on Romanian SMEs in the context of wannacry ransomware attacks, in: 12th European Conference on Innovation and Entrepreneurship, ECIE 2017, 2017, p. 307.
[74] S.K. Pandey, B.M. Mehtre, Performance of malware detection tools: A comparison, in: 2014 IEEE International Conference on Advanced Communication, Control and Computing Technologies, ICACCCT 2014, Institute of Electrical and Electronics Engineers Inc., 2014, pp. 1811–1817, http://dx.doi.org/10.1109/ICACCCT.2014.7019422, [Online]. Available: http://www.scopus.com/inward/record.url?eid=2-s2.0-84923314531&partnerID=40&md5=7be1f1c3299819f9ca69e7547333e299 http://ieeexplore.ieee.org/ielx7/6996691/7019129/07019422.pdf?tp=&arnumber=7019422&isnumber=7019129.
[75] H.Q. Zhang, X. Xiao, F. Mercaldo, S.G. Ni, F. Martinelli, A.K. Sangaiah, Classification of ransomware families with machine learning based on N-gram of opcodes, Future Gener. Comput. Syst. Int. J. Esci. 90 (2019) 211–221, http://dx.doi.org/10.1016/j.future.2018.07.052 (in English).
[76] A. Zimba, Z.S. Wang, H.S. Chen, Multi-stage crypto ransomware attacks: A new emerging cyber threat to critical infrastructure and industrial control systems, Ict Express 4 (1) (2018) 14–18, http://dx.doi.org/10.1016/j.icte.2017.12.007 (in English).
[77] S. Homayoun, et al., DRTHIS: Deep ransomware threat hunting and intelligence system at the fog layer, Future Gener. Comput. Syst. 90 (2019) 94–104, http://dx.doi.org/10.1016/j.future.2018.07.045.
[78] J. Wang, J.M. Wei, Z. Yang, S.Q. Wang, Feature selection by maximizing independent classification information, IEEE Trans. Knowl. Data Eng. 29 (4) (2017) 828–841, http://dx.doi.org/10.1109/TKDE.2017.2650906.
[79] P.A. Estevez, M. Tesmer, C.A. Perez, J.M. Zurada, Normalized mutual information feature selection, IEEE Trans. Neural Netw. 20 (2) (2009) 189–201.

Bander Ali Saleh Al-rimy is a senior lecturer at UNITAR International University. He received the B.Sc. degree in computer engineering from the Faculty of Engineering, Sana'a University, Yemen, in 2003, the M.Sc. degree in information technology from OUM, Malaysia, in 2013, and the Ph.D. degree in computer science (information security) from the Faculty of Engineering, Universiti Teknologi Malaysia (UTM), in 2019. His research interests include but not limited to Malware, IDS, IoT, network security, and routing technologies. Dr. Al-Rimy was a recipient of several academic awards and recognitions including UTM Alumni Award, UTM Best Student Award, UTM Merit Award, UTM Excellence Award, OUM Distinction Award, and the Best Research Paper Award.

Mohd Aizaini Maarof received the B.Sc. degree in computer science from Western Michigan University, Kalamazoo, MI, USA, the M.Sc. degree in computer science from Central Michigan University, Mount Pleasant, MI, USA, and the Ph.D. degree in IT security from Aston University, Birmingham, U.K. He is currently a Professor with the School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia (UTM). He is also the head of UTM-CSM Cyber Threat Intelligence Lab (CTIL) and a member of the Information Assurance and Security Research Group (IASRG), UTM. His research interest includes information system security.

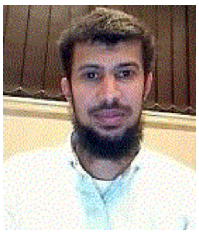


Mamoun Alazab is an Associate Professor in the College of Engineering, IT and Environment, Charles Darwin University, Australia. He is a cyber security researcher and practitioner with industry and academic experience. His research is multidisciplinary that focuses on cyber security and digital forensics of computer systems including current and emerging issues in the cyber environment like cyber–physical systems and internet of things, by taking into consideration the unique challenges present in these environments, with a focus on cybercrime detection and prevention. A/Prof Alazab received his Ph.D. degree in Computer Science and has more than 100 research papers. He presented at many invited keynotes talks and panels, at conferences and venues nationally and internationally (22 events in 2018 alone). He is a Senior Member of the IEEE. He is an editor on multiple editorial boards including Associate Editor of IEEE Access and Editor of the Security and Communication Networks Journal.

Syed Zainudeen Mohd Shaid is a lecturer at Universiti Teknologi Malaysia (UTM) where he teaches subjects like Penetration Testing, Security Programming, Exploitation and other security related subjects. He received his B.Sc, M.Sc, and Ph.D. (Computer Science) from Universiti Teknologi Malaysia (UTM). A member of the Information Assurance & Security Research Group (IASRG), he is active in Malware Research. He also does training and consultancy on Web Security, Secure Coding, Android, and embedded systems. He loves gadgets and enjoys exploring new things related to security

Fuad A. Ghaleb received the B.Sc. degree in computer engineering from the Faculty of Engineering, Sana'a University, Yemen, in 2003, and the M.Sc. and Ph.D. degrees in computer science (information security) from the School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia (UTM), Johor, Malaysia, in 2014 and 2018, respectively. From 2004 to 2012, he was a Lecturer of network and computer engineering with the Sana'a Community College, Yemen. He is involved in different projects with industries related to network and information security. His research interests include vehicular network security, cyber security, intrusion detection, data science, data mining, and artificial intelligence.

Dr. Ghaleb was a recipient of many awards and recognitions, such as the Postdoctoral Fellowship Award from UTM, the Best Postgraduate Student Award from UTM, the Excellence Awards from UTM, and the Best Presenter Award from the School of Computing, Faculty of Engineering, UTM, as well as Best Paper Awards from many international conferences



Abdulmohsen Almalawi received the BS degree in computer science from King Abdul Aziz University, Jeddah, Saudi Arabia, in 2003. He received the MS and PhD degrees in computer science from RMIT University, Melbourne, Australia, in 2009 and 2014, respectively. He is an assistant professor in the School of Computer Science and IT, King Abdul Aziz University, Jeddah, Saudi Arabia. His research interests are intrusion detection and cybersecurity of industrial SCADA systems with emphasis on data mining, machine learning, and fast algorithms.

Abdullah Marish Ali is an assistant professor in King Abdul Aziz University, Jeddah, Saudi Arabia. He received his Ph.D. degree in Computer Science from University Teknologi Malaysia, Malaysia in 2018. His research interests are in the areas of machine learning, machine learning, Data Mining, and IoT

Tawfik Al-Hadhrami received the M.Sc. degree in IT/applied system engineering from Heriot-Watt University, Edinburgh, U.K., the Ph.D. degree in wireless mesh communication from the University of the West of Scotland, Glasgow, U.K., 2015. He was involved in research at the University of the West of Scotland, Networking Group. He is currently a Senior Lecturer with Nottingham Trent University (NTU), U.K, where he is also a member of the Network Infrastructure and Cyber Security (NICS) Group. His research interests include the Internet of Things (IoT) and applications, network infrastructures and emerging technologies, artificial intelligence, computational intelligence, and 5G wireless communications. He is involved in different projects with industries. He is an Associate Editor of IEEE ACCESS and the IEEE SENSORS JOURNALS.