Classification of Malware using Av Labels Technique with Various Approaches

Manish Kumar Rai
Research Scholar, Research and Publication, Rashtriya Raksha University,
Lavad, Dehgam, Gandhinagar, Gujarat, 382305
Manish.it.imps@gmail.com

Prof. Priyanka Sharma
Dean, Research and Publication, Rashtriya Raksha University,
Lavad, Dehgam, Gandhinagar, Gujarat, 382305
Ps.it@rsu.ac.in

Abstract - As world moves towards the digitalization, Thus to satisfy the customer demand, Industry move in the world of cyberspace where everything is connected with a network. Apart from these advance and attractive facilities provided by the different industries, there is a serious concern regarding the security issues shown in the previous years. Several malware attacks happen by the different malicious groups to disrupt the services. Most of the malware identified in these cyber-attacks are from the known malware family. And that malware is easily identified by the available antivirus. Now these days, Antiviruses are able to identify malware by using their signature through signature-based analysis. Signature-based malware detection is an effective method for the identification of known malicious files but it has some limitations. Any malware with the same characteristics with different signature can able to bypass these antivirus securities. In this paper, we use the AV labeling technique with two feature selection strategies in static and dynamic mode to identify new kinds of malicious files. To get the highest accuracy rate, we took different numbers of features in different cases to identify the malware. Our trials accomplish the most elevated order precision of 97.60% for RF classifier malware into two diverse element determination procedures in all sections. Our answer is strong furthermore, adaptable as we have additionally tried our model on pressed and muddled malware tests. The model accomplishes a precision of 96.97%, 97.45% and 95.94% for LMT, NBT, and REPT on stuffed and jumbled malware tests, individually.

Keywords: Malware, AV Labels, Fisher Score, Information Gain, Random Force, LMT, NBT, REPT

1. Introduction

As India is a developing country and is having the second-highest population in the world. Power production, Management, and distribution is an important concern in such geographical conditions.;. In the United Kingdom, their traditional grids are updated into smart grids to become greener in upcoming years [Drayer (2019)]. Smart grids are able to provide more reliable, efficient, and low-carbon energy in the residential as well as industrial areas. Though we can implement the Smart grid concept in India, Security plays an important hurdle after the emergence of the STUNEXT, Aurora project. Modern organizations are significant and powerless, and there are possibly crushing outcomes of a digital occurrence. Instances of genuine digital episodes that happened in the past from CENTCOM to Aurora to Stuxnet—have developed continuously more extreme over the long haul, featuring the advancing idea of dangers against mechanical frameworks [Pieper (2020)]. The new threat [Mirsky (2019)] are evolving into APTs (Advance persistent threat), and the intentions are evolving from information theft to industrial sabotage and the disruption of critical infrastructures. The full degree of what Stuxnet can do isn't known to date. Stuxnet can Infect windows utilizing an assortment of multi-day abuses, endeavors to sidestep conduct hindering, commonly contaminates by infusing the whole DLL into another cycle, checks for windows rendition Anti-infection introduced, spreads horizontally through (tainted organization, USB, stage 7 undertaking record), infuses code blocks into the objective plc that can intrude on cycles, utilizes contaminated PLCs to look for explicit practices by observing Profibus, capacities to (eliminate itself from the inconsistent framework, lie lethargic, reinfect cleaned frameworks[Pieper (2020)] and convey shared to self-update inside contaminated organizations). The normal number of days between when weakness was uncovered freely and when the weakness was found in the charge framework was 331 days: very nearly a whole year. Indeed, even there is a sure case in which weaknesses were found more than 1100 days (very nearly long term) [Yan (2012)]. In spite of the fact that the consequence of discovering a weakness in the framework [Bujorianu (2018)] may lessen from a normal of 331 days to week after week on the off chance that we arranged windows support. Be that as it may, windows upkeep is very troublesome, financially savvy, and tedious. In the below table, We can see a few attacks on different organizations or individuals in the past few years.

Table 1: Attack over different organization in past few years.

No.	Target Group	Malware Class	Malware Family	Properties	Behavior	Loss	Year
1	Fire Eye / Solar Winds	Trojan	(Backdoor Payload or RAT	(.DLL File)	Data Breaches/	$5.2 Million	March 2020
2	MongoDB	Ransomware	(Trojan. Script. Agent)	(.JSON)	Data Delete	15bit	Jul/202 0
3	Mitsubishi Electric	Remote Access Trojan	(Backdoor Payload or RAT)	(.EXE)	Data Breaches	200 MB Files	Jan/2020
4	Twitter	Social Engineering	(Phishing Attack)	NONE	Accounts Compromise	$100,00	July/2020
5	Marriott	Ransomware	Remote Access Trojan (RAT)	(.EXE)	Accounts Credential	5.2 million	2020
6	MGM Resorts	Social Engineering	(Phishing Attack) Misconfigured Cloud Server	NONE Unauthorized	Data Leak/Breaches	$10.6 Million	Feb/2020
7	Mostly desktop users of company	Ransomware	GandCrab	MS Office Macros	Encrypting Data	$13 million	2019
8	Personal Computers	CryptoVirus	Dharma	.gif .AUF, .USA, .xwx,	Encrypting data using	$15 million	2019
10	VPN Users	Worm	BlueKeep	.exe	RDP	$4 million	2019
11	Norsk Hydro, Altran	Wiper	LockerGoga	.exe	Encrypts data,	$8 million	2019

2. Related Work

Findings from the various research will help to spot and predict malicious behavior [Bao, (2015)] of system and maintain those activities which generate longer-term influences on energy culture either on energy saving or on security. The responsibility that computational insight can play inside the savvy framework [Drayer (2019)] environment is grounded in its capacity to empower keen practices under unsure conditions. The developing worries about the natural effect that energy debasement has on the world are bringing about a pristine origination of force systems, inside which sustainable power sources are progressively coordinated into the gathering cycle and furthermore inside which energy productivity and security ought to be augmented. This work has given uncommon consideration to how artificial [Gupta (2020)] and computational insight can add to the accomplishment of more brilliant frameworks to the current end, this work has explored the most keen lattice advancements and furthermore the current and potential effect that shrewd methodologies[Wang (2020)] have upon them. It could be presumed that information designing methodologies can effectively address the matter of dealing with the immense measure of information which will be created in future shrewd grids. Moreover, the new approaches which are evolving in the form of machine intelligence and artificial intelligence of the systems are now in the place of traditional computational system to monitor and manage the whole process involved in the generation and distribution over the plant. Multi-specialist framework [Yang (2020)] approaches have shown their capacity to coordinate and understandable reactions to the appropriate information coming from the different wellsprings of information. At long last, the responsibility of intelligent model or delicate registering strategies can incredibly add to the improvement and control the measure required inside the keen network. Researchers are to quicken progress on key parts of brilliant lattice strategy, innovation, and related principles through intentional investment by governments in explicit undertakings and projects. Informative analysis [Kumar (2019)] that might be wont to develop energy systems in ways in which could meet sustainability goals, especially targeted on electricity systems. Shrewd models are run in new way also help the state to conserve energy which benefits the environment. The resources and toolbox are use to development of practitioners and program designers curious about diagnostics of the critical system and the complex model.

3. Datasets:

To compute this research, we use Microsoft Malware Data set of various families of malware. These datasets are collected from different machine which are using Microsoft operating system. These all malware families are identified based on the Has Detections ground. We also have the information about the machine on which attack has happened or malware resides in that system through Machine Identifier. In order to satisfy such a User policy, both in terms of user privacy and the time span during which the machine was operating, the sampling technique used to build this dataset was developed. Detection of malware is genetically a time series problem, but the emergence of new computers, machines that come online and offline, machines that receive updates, machines that receive new operating systems, etc., makes it complicated. Although the given dataset has been segregated in different time intervals. The previously mentioned complexities and examining models can imply that you may see defective arrangement between your cross-validation, public, and private scores!

4. Proposed Methodology

In this segment, we delineate our technique embraced in this paper. We show the strategy from the highlights that we extricate also, the model design

4.1. Data Preprocessing:

In this phase, we will take input of unlabeled data to make them AV levels [Sebastián (2020)] of different samples. AV labels are used as input for the labeling of unlabeled data present in the malware dataset[Wang (2002)]. After the labeling the each data, the most belonging malware family names relate with that data are returned as a result.

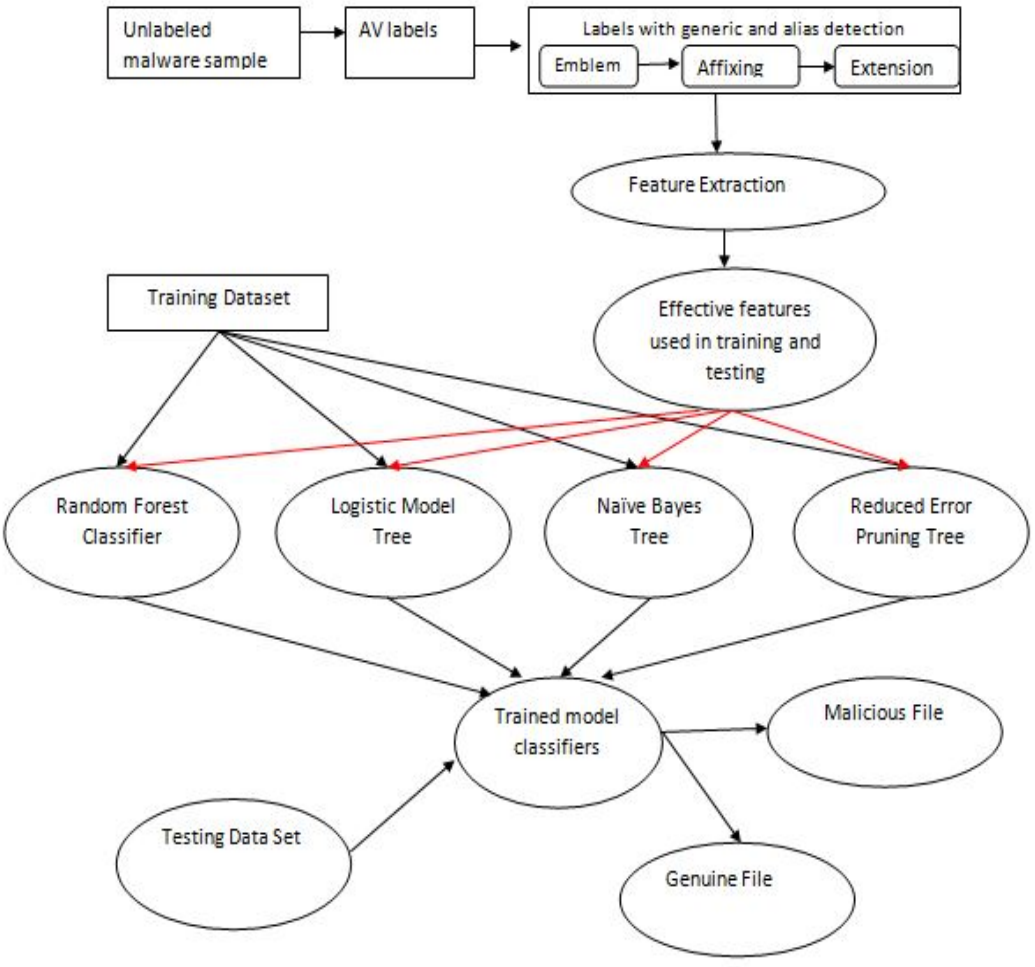

Fig. 1.: Methodology used in this paper.

4.1.1. ***Malware Class***: Individuals will in general mess around with security phrasing. In any case, it's essential to get your malware orders straight since knowing how different kinds of malware spread is crucial to containing. Example (virus, worm, Trojan, Hybrid and Exotic Forms, Ransomware, Fileless Malware, Adware, Malvertising, Spyware).

4.1.2. ***Malware Family:*** Family is the collection of similar malware based on characteristic and code use in creation. Codes are manipulated to create different kind of malware. So we can place this malware in a single group which helps the classifier to detect. In this way there are high possibilities that the examples having a place with a similar family will have code closeness also, will require comparative recognition and evacuation strategies. Example (Conficker, Crypto Wall, Hiddad, etc…). Few other information like file properties (pdf, doc, etc…) and behavior [Kumar (2019)], as a rule, recovered progressively (DOS or DDOS attack type and etc…)

4.2. Labeling Overview:

After preprocessing of unlabeled data with different AV engines [Sebastián (2020)] into AV levels format. We process the AV levels data for feature extraction. It restores an AV labels (or basically a name for short), i.e., a string with data about the malignant example. An AV name can be viewed as a grouping of emblems isolated by delimiters. Those emblems are specific and explicit as they are chosen rather autonomously by the AV merchant. For clearness, we will utilize emblems to affix with different labels in AV names and emblems to allude to passages in the scientific classification. AV labels after extension contains the useful information regarding the malicious file.

4.2.1. Emblems: In this step, each label is divided into different emblems which aim is to preserve the meaning of the each AV levels as most of the long are long enough. Emblems are needed also because the AV vendors are inconsistent in terms of labels.

4.2.2. ***Affixing:*** After replacing the AV labels with small emblems, affixing them with an identification mark for better classification. These affixing steps improve the inconsistent AV labels into a precise format. Small emblems are also get considered which are left till now. The affixing rule replaces emblems that specify the set of the rule. For instance, an affixing rule maps the owner's emblems to the ownership affix. For this situation, we state that that the emblems are a false name for the affix since it catches a similar idea as the affix. Various emblems might be monikers for a similar affix. For example, another affixing rule maps the loader token additionally to the ownership affix. Accordingly, tokens owner and loader are both ownership monikers. If some unknown emblems are detected in these steps are not provided with any affix rule. But we can't ignore these emblems because they can give information about any unknown malicious file. So this information gets passed to the extension phase.

4.2.3. ***Extension:*** Extension of affix emblems works into two different forms of inside family and outside the family. It supports the classifier model in an effective manner to classify by increasing the dimension of findings in that domain. It also maintains a hierarchical level in form of a tree where leaf levels are part of their root (If the classifier determines the file as a specific category and that particular category is a family member of any other malicious file). It helps to find the hybrid or exotic malware family which has multiple functionalities as described in the different classes.

4.3. Features Extraction:

4.3.1. ***Static Feature:*** In steady feature examination, the steady highlights are separated from executable records utilizing python PEFILE [Kumar (2019)] without accomplishing feature in monitoring climate. This analysis separates numerous highlights from header, discretionary header, and various segments. Convenient Executable (CE) designs are utilized by windows executables and Dynamic Link Libraries DLLs. At the time of loading data in windows, DLLs give linkage and execution data of code. It is additionally utilized for examination of imported libraries and the sorts of linkage are utilized in running of the document. CE document contains header and areas. Header formats have numerous other highlights bunches. As DOS, CE, Sections Table is utilized during the step. The record header comprises highlights like Operating System, number_of_segments, arrange_date, number_of_bring_in_images, and qualities. The discretionary header contains highlights like significant_rendition, low_rendition, investigate_amount, checksum, and at_rva. The segment header gives valuable data about a part like its id, memory, and so on. Apart from that other highlights are segment entropy to take out undesirable information in the segment names, size of crude information n, size of uninitialized information, segment virtual location, absolute size of the CE record and to ensure that all areas names are advanced segments like Machine_Identifier, ProductName, EngineVersion, AppVersion, AvSigVersion, DefaultBrowsersIdentifier, AVProductStatesIdentifier, HasTpm.

Table 2: Features selection during Static Analysis

Id No.	Identification Des.	Id No.	Identification Des.
Id1	MachineIdentifier	Id17	Platform
Id2	ProductName	Id18	Processor
Id3	EngineVersion	Id19	OsVer
Id4	AppVersion	Id20	OsSuite
Id5	AvSigVersion	Id21	SkuEdition
Id6	IsBeta	Id22	IsProtected
Id7	DefaultBrowsersIde	Id23	AutoSample0ptiIn
Id8	AVProductStatesIden	Id24	PuaMode
Id9	HasTpm	Id25	Shoed
Id10	CountryIdentifier	Id26	Firewall
Id11	Census_MDC3	Id27	Census_deviceFamily
Id12	Census_ProcssorCorecu	Id28	Census_ProcessorClass
Id13	Census_PrimaryDisk	Id29	Census_SystemVolumeTota
Id14	Census_HasOpticalDisk	Id30	Census_TotalPhysicalRAM
Id15	Census_OSVersion	Id31	Census_IsFlightsDisabled
Id16	Census_FlightRing	Id32	Census_IsSecureBootEnabl

4.3.2. ***Dynamic:*** Most of the identifiable characteristics can be extracted from executables dynamically or at run time. To extract and analyze dynamic features, we use the cuckoo sandbox tool in this paper. Both the testing environment and the actual system are isolated in the cuckoo sandbox tool. Different summarize sections are extracted through this tool and each section has different multiple features present. Some of the features are mentioned like: different actions performed during the execution of executables, Different data files created, modified, or deleted during the execution of malicious executables will also helpful in the identification. Every change in a file can be used as a feature of identification. Different API calls by the executables can be another feature of the identification of malware. Application programming interfaces are used to make connections between different processes through different hardware. The number of different API calls is presently based on their operating system, the hardware used, and the preinstalled library functions. The cuckoo tool uses all the APIs used and also has a record of the reply of that API in that particular execution. This feature tells about the status of execution of a particular code as it was executed successfully or not. Network Analysis can also provide information about the executables like file size sending through the network having an IP address and domain name server. SiLK software was used to extract that information like IP address and the domain name from pcap file. The malicious file can use a number of different IP addresses for the execution of a particular command. So this could another feature of identification. One more important feature selection section is Registry keys. It contains the all information like program installed, hardware accessed and different value before the execution, during the execution and after execution. Also have information about the registry written, registry deleted, opened, and read. So we can analyze that which of the component goes down after the execution or effect of executable to different hardware and software. Malware or any malicious file will try to modify different registry module to cross the security level.

Table 3: Features selection during Dynamic Analysis

Id No.	Identification Des.	Id No.	Identification Des.
Id33	Web Packets	Id47	Network Error
Id34	Web Request Packets	Id48	Ip Entropy
Id35	EXPLORE	Id49	TCP Entropy
Id36	Notification	Id50	HTTP Entropy
Id37	AQUIRE	Id51	ICMP
Id38	Web Reply Packets	Id52	IGMP
Id39	Host Error	Id53	SSL
Id40	TCP	Id54	Nbns
Id41	Data	Id55	Udp
Id42	Host	Id56	Server
Id43	Netapi	Id57	Openweathermap
Id44	GET	Id58	POST
Id45	PUT	Id59	DELETE
Id46	TOKEN	Id60	ERROR Message

4.4. Target Feature Selection:

Highlight choice procedures can dodge the scourge of dimensionality and consequently empower the disentanglement of models, compiling deciphering exploratory outcomes simpler for analysts. Special feature of the directed element determination techniques, the Fisher score calculation chooses each component freely as per their marking. Feature determination utilizing the Fisher score calculation brings about elite of qualities that are positioned by their significance.

In this experiment, The vital thought of the Fisher score is to discover a subset of highlights, with the end goal that in the information space traversed by the chose highlights, the distances between information focuses in various classes are as extensive as could reasonably be expected, while the distances between information focuses in a similar class are as little as could reasonably be expected. We have a data set

$$\sum_{l=1}^{n}(A_p, B_q) \quad (1)$$

Where $A_p \in$ data set present in the form of matrix and $B_q \in \{$ number of different feature upto C$\}$. Total C feature were selected based on their enlightening features. Each feature score calculation through Fisher Score:

$$F(A^j) = \frac{\sum_{i=1}^{c} n_l (\alpha_l^j - \alpha^j)^2}{(\pi^j)^2} \quad (2)$$

Where α_l^j is mean of each class corresponding to the j-th feature and π^j & α^j is mean, standard deviation of the whole class. After generation of each score.

A different feature selection strategy is used to extract the feature from same dataset. It will help to select top feature which help in identification. Next feature extraction method is relies on entropy based detection. This extraction methodology is mainly used in training machine learning models. IG is likewise be utilized for include choice, by assessing the increase of every factor with regards to the objective variable. In this somewhat unique utilization, the count is alluded to as common data between the two irregular factors. Information gain calculation:

$$IG(A, i) = E(A) - E(A \mid i) \quad (3)$$

Where IG(A, i) represent the full information of dataset A for an arbitrary variable, E(A) is the entropy of original dataset or without modified dataset (portrayed above) and E(A | i) is the contingent entropy of dataset for particular information i.

4.5. Model Training:

Different type of classifier is trained with training data set. In this experiment we use two feature selection strategy and four classifier. The dataset in the proportion of 75%-25% for preparing and testing of our four models using ten-fold cross validation. Each model trained with two feature selection method of same data set which extracted before.

5. Result:

Now we have the result of all four classifier through the analysis of the dataset using two feature extraction methods (Information gain & Fisher Score). Result and performance charts are shown in the figure and the tables are given below. For the selection major features in the dataset for identification, Two feature selection strategy was used in this research with different number of features are in consider which effect in the result outcome like with more number of features the accuracy of the model is slightly increased as can be seen in the figure. At first, we just take the fisher score method of feature selection into consideration for model training and testing. Different number of features is used to train and tests four models. Result shown in Table 4 that if we increase the number of features for the identification then the model shows high accuracy like with 15 features the RF classifier shows 95.38% accuracy rate in the detection of the malicious file but if we increase the number of identification feature the accuracy rate slightly gets increased as accuracy rate goes up to 98.67% which is very effective in terms of identification.

Table 4: Accuracy of Four Classifier with different number of features extracted through Fisher Score

Number of Features	Random Forest	Logistic Model Tree	Naïve Bayes Tree	Reduced Error Pruning
15 Features	95.38%	94.6%	95.73%	94.66%
30 Features	97.86%	97.67%	97.8%	96.43%
45 Features	98.5%	97.89%	98.31%	96.77%
60 Features	98.67%	97.74%	97.99%	95.89%

Similarly we can see the same pattern for the rest of three classifier where all models shows above then 94% accuracy rate and it gets increase for the more number of features. With the fisher score features selection methodology, RF shows highest rate of accuracy in the segments.

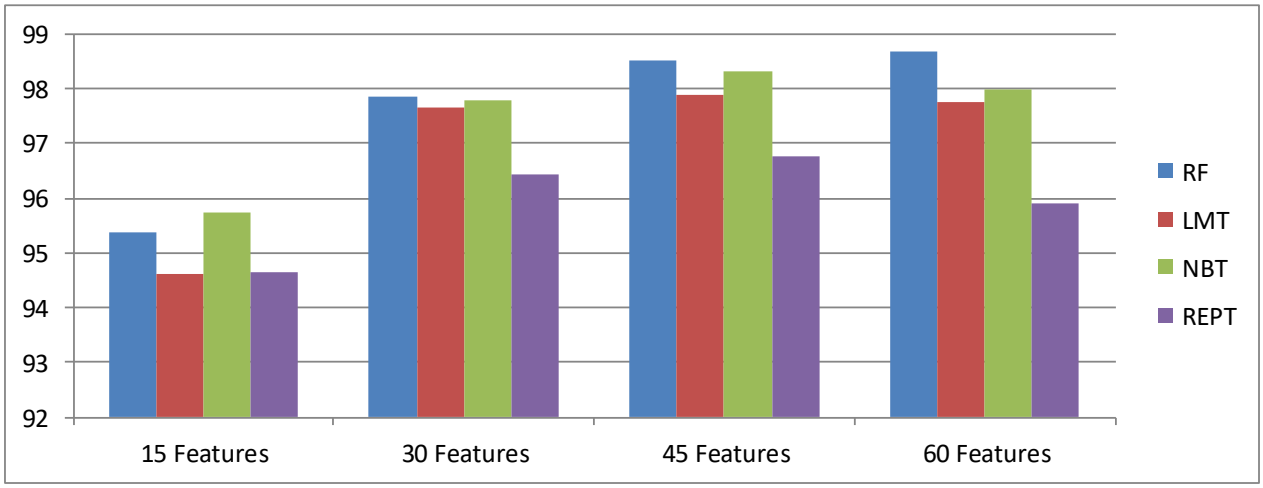

Figure 2: Accuracy of Four Classifier with different number of features extracted through Fisher Score

Despite the fact that there is a minor improvement in the precision, the distinctive number of highlight feature used is significant. Table 5 shows the accuracy rate of the all four classifier with information gain feature selection strategy. From this table we came to know that the performance of all four classifier gets decreased in comparison of the fisher score. For the 15 features RF model shows 93.67% accuracy rate for information gain where it is 95.38% in fisher score.

Table 5: Accuracy of Four Classifier with different number of features extracted through Information Gain

Number of Features	Random Forest	Logistic Model Tree	Naïve Bayes Tree	Reduced Error Pruning
15 Features	93.67%	93.21%	90.73%	89.53%
30 Features	93.89%	92.67%	93.8%	91.43%
45 Features	95.19%	94.89%	95.31%	91.77%
60 Features	96.76%	95.74%	95.99%	92.89%

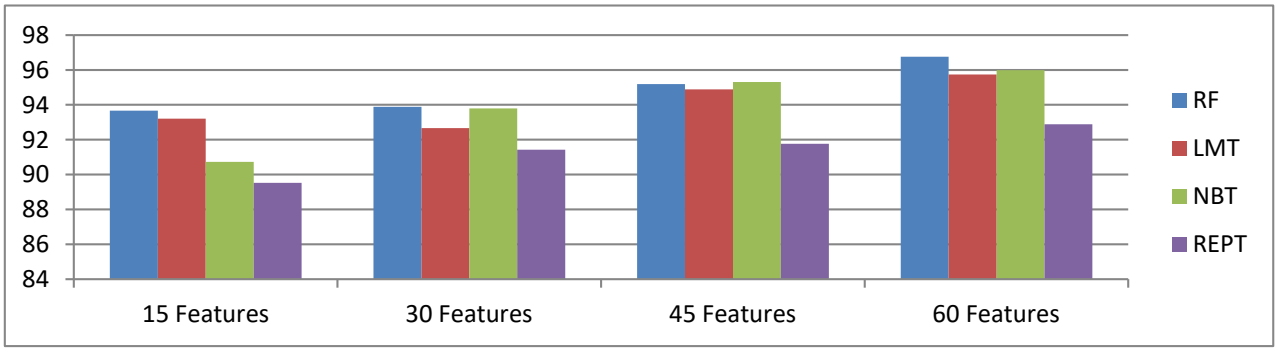

Figure 3: Accuracy of Four Classifier with different number of features extracted through Information Gain

Overall comparison of the classifier with these two features selection strategy shown in the Table 6. RF shows efficient result over the all other model in both strategy and all different segments. Fisher Score with Random forest classifier achieve 97.60% of identification accuracy with this dataset.

Table 6: Classifier Average Performance over different Feature

Name of the Classifier	Information Gain	Fisher Score
Random Forest	94.88%	97.60%
Logistic Model Tree	94.12%	96.97%
Naïve Bayes Tree	93.96%	97.45%
Reduced Error Pruning	91.40%	95.94%

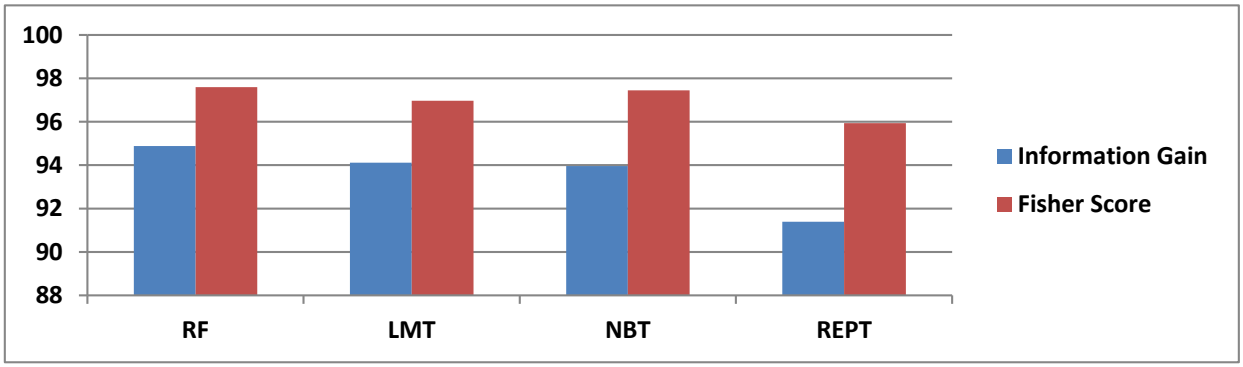

Figure 4: Classifier Performance over different Feature

6. Conclusion:

Threats over cyber space to utilities are probably going to fill in numbers and it will become more complex. In this paper we used an AV class which labels the data into AV labels for all malware dataset. AV labels are well organized to classification family group and indexed which shows high identification rate in all four classifier discussed in this paper. Two feature extraction methods (Fisher Score and Information Gain) is used to extracting the feature from AV labels data set. We have present RF, LMT, NBT and REPT tools to detected malware samples. The extracted features like behavior, file properties, class, and family are used by these

classifiers for identification. These samples show the ground truth of the models for comparisons. RF classifier tool shows the highest classification rate in both cases with an average classification rate 96.24%.

References

[1] Alabadi, Montdher, and Yuksel Celik. "Anomaly Detection for Cyber-Security Based on Convolution Neural Network: A survey." In 2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), pp. 1-14. IEEE, 2020.

[2] Bao, Haiyong, et al. "BLITHE: Behavior rule-based insider threat detection for smart grid." IEEE Internet of Things Journal 3.2 (2015): 190-205.

[3] Bensaoud, Ahmed, Nawaf Abudawaood, and Jugal Kalita. "Classifying Malware Images with Convolutional Neural Network Models." arXiv preprint arXiv:2010.16108 (2020).

[4] Bujorianu, Manuela L. "CPS Dependability Framework Based on Inhomogeneous Stochastic Hybrid Systems." In Cyber Physical Systems. Model-Based Design, pp. 134-153. Springer, Cham, 2018.

[5] Drayer, Elisabeth, and Tirza Routtenberg. "Cyber Attack Localization in Smart Grids by Graph Modulation (Brief Announcement)." In International Symposium on Cyber Security Cryptography and Machine Learning, pp. 97-100. Springer, Cham, 2019.

[6] Drayer, Elisabeth, and Tirza Routtenberg. "Intrusion detection in smart grid measurement infrastructures based on principal component analysis." In 2019 IEEE Milan PowerTech, pp. 1-6. IEEE, 2019.

[7] Gupta, Maanak, Sudip Mittal, and Mahmoud Abdelsalam. "AI assisted Malware Analysis: A Course for Next Generation Cybersecurity Workforce." arXiv preprint arXiv:2009.11101 (2020).

[8] Jha, Sudan, Deepak Prashar, Hoang Viet Long, and David Taniar. "Recurrent neural network for detecting malware." Computers & Security 99 (2020): 102037.

[9] Kumar, Nitesh, Subhasis Mukhopadhyay, Mugdha Gupta, Anand Handa, and Sandeep K. Shukla. "Malware Classification using Early Stage Behavioral Analysis." In 2019 14th Asia Joint Conference on Information Security (AsiaJCIS), pp. 16-23. IEEE, 2019.

[10] Kumar, Amit, Nitesh Kumar, Anand Handa, and Sandeep Kumar Shukla. "PeerClear: Peer-to-Peer Bot-net Detection." In International Symposium on Cyber Security Cryptography and Machine Learning, pp. 279-295. Springer, Cham, 2019.

[11] Li, Xiang, Kefan Qiu, Cheng Qian, and Gang Zhao. "An Adversarial Machine Learning Method Based on OpCode N-grams Feature in Malware Detection." In 2020 IEEE Fifth International Conference on Data Science in Cyberspace (DSC), pp. 380-387. IEEE, 2020.

[12] Mills, Alan, and Phil Legg. "Investigating anti-evasion malware triggers using automated sandbox reconfiguration techniques." Journal of Cybersecurity and Privacy 1, no. 1 (2020): 19-39.

[13] Mirsky, Reuth, Ahmad Majadly, Kobi Gal, Rami Puzis, and Ariel Felner. "New Goal Recognition Algorithms Using Attack Graphs." In International Symposium on Cyber Security Cryptography and Machine Learning, pp. 260-278. Springer, Cham, 2019.

[14] Pieper, Tobias. "Distributed co-simulation framework for hardware-and software-in-the-loop testing of networked embedded real-time systems." (2020).

[15] Roy, Sohini, and Arunabha Sen. "Identification of the K-most Vulnerable Entities in a Smart Grid System." In 2020 3rd International Conference on Advanced Communication Technologies and Networking (CommNet), pp. 1-6. IEEE, 2020.

[16] Singh, Ajay, Anand Handa, Nitesh Kumar, and Sandeep Kumar Shukla. "Malware classification using image representation." In International Symposium on Cyber Security Cryptography and Machine Learning, pp. 75-92. Springer, Cham, 2019.

[17] Sebastián, Silvia, and Juan Caballero. "AVclass2: Massive Malware Tag Extraction from AV Labels." Annual Computer Security Applications Conference. 2020.

[18] Schauer, Stefan, Stefan Rass, Sandra König, Klaus Steinnocher, Thomas Schaberreiter, and Gerald Quirchmayr. "Cross-Domain Risk Analysis to Strengthen City Resilience: the ODYSSEUS Approach.

[19] Singh, Ajay, Anand Handa, Nitesh Kumar, and Sandeep Kumar Shukla. "Malware classification using image representation." In International Symposium on Cyber Security Cryptography and Machine Learning, pp. 75-92. Springer, Cham, 2019.

[20] Wang, Wenbo, Yurong Song, Yinwei Li, and Yaqiong Jia. "Research on Cascading Failures Model of Power Grid Based on Complex Network." In 2020 Chinese Control And Decision Conference (CCDC), pp. 1367-1372. IEEE, 2020.

[21] Wang, X., & Miikkulainen, R. (2020). MDEA: Malware Detection with Evolutionary Adversarial Learning. arXiv preprint arXiv:2002.03331.

[22] Yang, Li, and Junlin Liu. "TuningMalconv: Malware Detection With Not Just ." IEEE Access 8 (2020): 140915-140922.

[23] Yan, Y. Qian, H. Sharif, and D. Tipper, "A survey on cyber security for smart grid communications," IEEE Communications Surveys and tutorials, vol. 14, no. 4, pp. 998–1010, 2012.

Authors Profile

I "Manish Kumar Rai" am pursuing doctorate degree from Rashtriya Raksha University (A Institute of National Importance) under the department of Research and Publication in Cyber Security. I have completed my Post Graduation from Central University of South Bihar in Computer Science. I am currently holding the position of Junior Research fellow. My key research area is industrial security and SCADA security. I have qualified GATE, NET and JRF in computer Science of Engineering.