Towards the Integration of a Post-hoc Interpretation Step into the Machine Learning Workflow for IoT Botnet Detection

Alejandro Guerra-Manzanares, Sven Nõmm and Hayretdin Bahşi
Department of Software Science, Faculty of Information Technology
Tallinn University of Technology, Akadeemia tee 15 a, 12618,
Tallinn, Estonia
E-mail: {alejandro.guerra, sven.nomm, hayretdin.bahsi}@taltech.ee

***Abstract*—The analysis of the interplay between the feature selection and the post-hoc local interpretation steps in a machine learning workflow followed for IoT botnet detection constitutes the research scope of the present paper. While the application of machine learning-based techniques has become a trend in cyber security, the main focus has been almost on detection accuracy. However, providing the relevant explanation for a detection decision is a vital requirement in a tiered incident handling processes of the contemporary security operations centers. Moreover, the design of intrusion detection systems in IoT networks has to take the limitations of the computational resources into consideration. Therefore, resource limitations in addition to human element of incident handling necessitate considering feature selection and interpretability at the same time in machine learning workflows. In this paper, first, we analyzed the selection of features and its implication on the data accuracy. Second, we investigated the impact of feature selection on the explanations generated at the post-hoc interpretation phase. We utilized a filter method, Fisher's Score and Local Interpretable Model-Agnostic Explanation (LIME) at feature selection and post-hoc interpretation phases, respectively. To evaluate the quality of explanations, we proposed a metric that reflects the need of the security analysts. It is demonstrated that the application of both steps for the particular case of IoT botnet detection may result in highly accurate and interpretable learning models induced by fewer features. Our metric enables us to evaluate the detection accuracy and interpretability in an integrated way.**

***Index Terms*—Botnet detection, machine learning, interpretation.**

I. Introduction

This paper aims to provide interpretable results for machine learning-based IoT botnet detection system while minimizing the feature set. IoT technology has been an important enabler in many sectors such as energy, manufacturing, transportation or health. However, physical-, network-, and application-layer attacks may cause important consequences ranging from privacy violations or business interruptions to physical damage in these systems [1]. Besides these problems, the botnets composed of compromised IoT devices constitute a significant threat to all Internet-faced systems. For instance, in 2016, some companies such as a hosting company, OVH [2] and an internet performance management company, Dyn DNS [3], suffered from massive denial of service attacks originated from IoT bots. Considering the high adaptation rate of IoT technology into real-world applications, these attacks could be assumed as initial warnings for the more detrimental attacks in the future. The research community has addressed the intrusion detection in these environments [4]. Identifying the new attack types remains as a significant difficulty, but machine learning methods have provided solutions for this problem [5].

Although there exists a huge amount of research with convincing results for the adaptation of machine learning to the intrusion detection field, the optimization of the detection accuracy has been the only focus in those studies. However, the detection result is just only a starting phase of an incident handling process that mostly takes place in a layered tier structure of the security operations center [6]. In an ideal case, a full-automatized protection system itself should identify the intrusion, decide and take the necessary actions. However, human-in-the-loop character of these systems has not reduced, inversely, increased in time due to the complexity of the cyber threats. The responsibility of experts is so vital in terms of reducing the false positives, assigning priority levels to the findings and conducting in-depth forensic analysis during the incident handling period. Therefore, machine learning method is highly required to provide the details of explaining the detection decision to the experts, which means interpretability arises as a significant performance metric besides the accuracy rate of the detection.

A model in which the expert can easily understand the reason behind the decision is considered as an interpretable model (a.k.a. comprehensible, understandable or explainable [7]). Reducing the model size via dimensionality reduction is one of the approaches for making models more interpretable [8] despite the fact that it does not guarantee the acceptability in all cases (i.e., the experts may not trust over-simplistic models in some situations) [9]. Interpretability is an ill-defined concept as some scholars relate it to a more general notion such as trust of users to learning models, some of them find it as an instrument for identifying the casual structure or others may simply assume that it helps to gather more useful data from the model [10].

Interpretability methods can be divided into two, global or local. Global interpretability aims to provide understanding of the whole logic of a model and all of its possible outcomes while local interpretability deduces the reasoning behind an individual prediction. In practice, the global interpretability of a model is hardly achievable while local interpretability being more feasible.

Application of the classical machine learning techniques

978-1-7281-4550-1/19/$31.00 ©2019 IEEE
DOI 10.1109/ICMLA.2019.00193

usually presumes that the feature selection process was conducted[1]. Model-agnostic and post-hoc local interpretation methods are applied to the outputs of learning models. In this sense, feature selection is a prior step and local interpretation is the following step for the creation and validation of the learning models in classical algorithms. Usually, local interpretation for each particular instance is presented by the set of inequalities (i.e., assuming that the features are numeric), whereas the number of inequalities is greater or equal to the dimensionality of the feature set. Therefore, although they take place in different steps, feature selection and local interpretation may have an interplay, in the whole machine learning work-flow, that may have an impact on the quality of the interpretation. Besides this, it is obvious that the feature selection has also an impact on the accuracy of the model. Intrusion detection designers would prefer to understand the implication of the feature selection not just only to the accuracy but to the quality of the interpretation as well.

In this study, we analyzed the impact of feature selection on the detection accuracy and the interpretation quality. In the first part, we analyzed the impact of hyper-parameters and feature selection on data accuracy. In the second part, we analyzed the impact of feature selection on the interpretation results. Here, we introduced a quality metric for the interpretation results from the cyber security analyst perspective. This metric, which is based on entropy notion, assumes that an ideal explanation should bring explanation for just only one category as others could create confusion for the analyst. As our focus is not the optimization of the local interpretation step but just to understand the interplay between feature selection and post-hoc local interpretation, we used well-known Local Interpretable Model-agnostic Explanation (LIME) as a local interpretation method [11] although there are other methods in the literature. The decision quality of the interpretation method and comparing it with other similar methods are also out of our scope.

Recently deep-learning-based techniques have gained a lot of popularity. Unlike the classical machine learning algorithms, the deep-learning methods do not require separate feature selection procedure[2]. However, as IoT environments are limited with various computational resource restrictions, classical learning models induced by fewer features can be considered more suitable than deep-learning methods which are computationally intensive. The very high accuracy values that we obtained by classical methods in this problem domain reinforce our method selection decision.

The studies dealing with the application of machine learning to the cyber security problems, in general, and to the intrusion detection, in particular, give the whole focus to the detection accuracy, and very little attention is paid to the interpretability. This paper addresses such an important research gap. Our work is unique as it provides an interplay analysis of feature selection and interpretability steps within the context of the IoT botnet detection problem.

The content of our paper is presented as follows: Section II gives background information about the addressed topic and summarizes the literature. In Section III, we described the data utilized in this study. Section IV explains the method of our research and the obtained results. Our work is concluded in Section V.

II. Background Information and Literature Review

Artificial Intelligence has become a widely adopted solution to deal with some complex tasks such as prediction in a great variety of fields ranging from biology to cybersecurity. The inherent complexity of most machine learning models makes them powerful but lacking transparency, posing as black-boxes, where the explanation each decision remains hidden and unknown to the end-user. Thus, becoming one of the main obstacles to the spreading of AI to fields where the explanation behind each decision is important and needed to trust and justify the AI-system predictions, for instance in disease diagnosis or compliance with General Data Protection Regulation (GDPR).

In recent years, explainable AI (XAI) has emerged as a new research field aiming to create more human-interpretable models, that will empower transparency and trust in machine learning outcomes, whilst preserving their high-performance capabilities [12]. The main issue for XAI models is dealing properly with the interpretability and accuracy tradeoff. One of the successful approaches that allows keeping high-performance metrics while providing interpretability to complex machine learning models is post-hoc methods, which provide explanations without disturbing or having any knowledge about the inner works of the model they are explaining [12].

Local interpretation methods have focused the attention especially to explain the complex and opaque Deep Neural Networks, methods which usually claim to be model-agnostic [12]. In this regard, [11] proposed the Local Interpretable Model-Agnostic Explanation (LIME) method which explains individual instances of non-linear models by sampling perturbed instances around the individual decision, weighting them according to their proximity, getting original model's output for the perturbed instances and learning a linear model in the neighbourhood of the explained instance. From the same authors and more recently, in [13] they proposed Anchors, which extend LIME's model-agnostic explanations on the basis of if-then rules, claiming to provide more coverage, interpretability, and enhanced generalization. Local Rule-based Explanations (LORE), proposed in [14], builds a decision tree model based on a set of neighbor instances of a concrete decision, using a genetic algorithm and given an original black-box model. Explanation of individual decision is extracted from the learned decision tree. In [15], local explanations are based on the local gradients which identify what directions an instance has to be moved in order to change its predicted label, thus

[1] While some techniques do not suffer from the curse of dimensionality, feature selection is still necessary to provide stable classifiers.

[2] It may be seen that feature selection becomes an implicit integral part of the deep learning process

indicating the most influential directions to the prediction. Lastly, [16] applied influence functions that measure the effect of local changes in instances to understand individual predictions.

Cybersecurity is a potential field of application where XAI models are needed [12]. Regarding it, network security is one of the emerging issues where interpretability may help to explain and improve detection mechanisms such as intrusion detection systems. In this regard, the framework for explainable Deep Neural Networks (DNN) based anomaly detection implemented in [17] relied on input feature relevance scores on network-based anomaly detection to explain individual decisions and enhance human trust on machine learning models in the context of critical and industrial systems monitoring. Input relevance scores measured quantitatively the influence of certain input features on the detected anomaly by the DNN model. In the same context, but as a different approach, Situ [18] proposed the use of p-value anomaly scoring to explain anomalies detected on network data streams. P-value anomaly scoring was calculated on a set of statistics of interest and used to discriminate and explain network suspicious behavior. Adversarial learning samples, mainly used to deceive ML classifier models, are adopted in [19] to explain Deep Learning models used as classifiers in Intrusion Detection Systems (DNN-IDS). In this regard, a model-agnostic adversarial machine learning method is implemented to explain misclassified samples by feeding the classifier with misclassified samples, making minimum amount of modifications on them until they are correctly classified. Comparison of modified samples correctly classified and original samples is used to find out the most relevant features that produced the misclassification, thus explaining the classifier's output.

III. Data set description

The dataset is composed of various statistics obtained from the network traffic of 9 IoT devices such as a security camera, webcam, baby monitor, thermostat, and door-bell [20]. Each record contains 115 numeric features and is labeled as benign (normal) or malicious (attack). The malicious traffic covers different attacks (i.e., denial of service, spam or scan) which are conducted by the devices compromised with Mirai or Bashlite (Gafgit) malware. In a typical botnet life-cycle, there exist four phases, formation, command and control (C&C), attack and post-attack [21]. This dataset does not include malicious activities that are related to the first two phases which correspond to exploitation of the device and creation of a remote control channel but covers the post-exploitation activities (attack and post-attack phases).

We analyzed the features within five categories, host-IP, host-MAC&IP, channel, network jitter and socket as shown in Table I. Host-IP category tracks the network traffic of each host regardless of the other communication entity and provides the statistics such as packet counts, mean and variance of packet sizes. Host-MAC&IP category is very similar to the previous one, the only difference is that it dissects each host by its MAC and IP addresses in order to eliminate the artifacts of possible IP spoofing attempts. Channel category captures the sames statistics produced by the source and destination host pairs whereas socket category also includes the source and destination ports in addition to host information. The statistics of the last two categories are extended by magnitude, radius, covariance and correlation coefficient of packet sizes. We classified the network-jitter of the channel type communication (i.e., the time intervals between the packet arrivals) into a separate category. All these statistics are obtained from the most recent five different time windows (100ms, 500ms, 1.5 sec, 10 sec and 1 min). In order to improve the readability, we represent a feature as "Feature Category Type-Time Window-Statistic Type". For instance, "Host_IP-100ms- Pkt Count" means the packet count of host-IP category obtained at the most recent 100ms interval.

The source dataset has 502,605 normal, 2,835,317 Bashlite and 2,935,131 Mirai records, meaning that the label distributions are 8%, 45%, and 47%, respectively. In this study, we covered the three-class classification problem. We utilized the accuracy as a detection metric for the simplicity as we focus on the interaction between feature selection and interpretation steps, and such an analysis can be conducted by other metrics deemed to be useful.

TABLE I
Feature Categories

Feature Categories	Features
Host-IP	Packet count, mean and variance (outbound)
Host-MAC&IP	Packet count, mean and variance (outbound)
Channel	Packet count, mean and variance (outbound) Magnitude, Radius, Covariance, Correlation Coef. (inbound and outbound)
Network Jitter	Count, mean and variance of packet jitter in channel
Socket	Packet count, mean and variance (outbound) Magnitude, Radius, Covariance Correlation Coefficient (inbound and outbound)

IV. Method & Results

The feature selection step is an integral part of any ML workflow where a classification algorithm is used. According to [22] it may be either a stand-alone step or integrated as a part of a wrapper or an embedded technique. Within the frameworks of the present research, only the case when feature selection is a stand-alone step is considered. Namely Fisher's score (given in Equation 1) based filter model is applied to provide initial grading of the features with respect to their discriminating power. Such simplification is possible due to the numeric nature of the features.

A typical work-flow for the application of supervised learning method is depicted in Figure 1. According to [22] three different approaches may be used for feature selection: filter models, wrapper models and embedded models.

Fisher's score is closely related to the information gain and usually defined as given in Equation 1.

$$F_i = \frac{\sum_{k=1}^{K} p_k(\mu_k - \mu)^2}{\sum_{k=1}^{K} p_k \sigma_k^2} \tag{1}$$

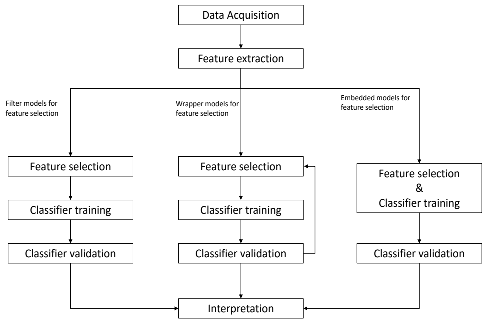

Fig. 1. Usual machine learning work-flow for supervised learning.

where K is the number of classes, p_k proportion of the observation points belonging to the class k, μ_k is the mean value of the class k along the feature i, μ is the overall mean and σ_k is the standard deviation of the class k along the feature i [22]. The higher value of (6) signals greater discriminating power of the feature and vice verse. Fisher's score is computed for each of the 115 features. The features are ordered according to their Fisher's scores as shown in Figure 2. Then feature selection may be performed either with respect to certain threshold or simply by choosing desired number of features based on their Fisher's score values. In Figure 2 one may clearly see that Fisher's score values for some features are negligibly low which is a clear indicator that these features may be omitted.

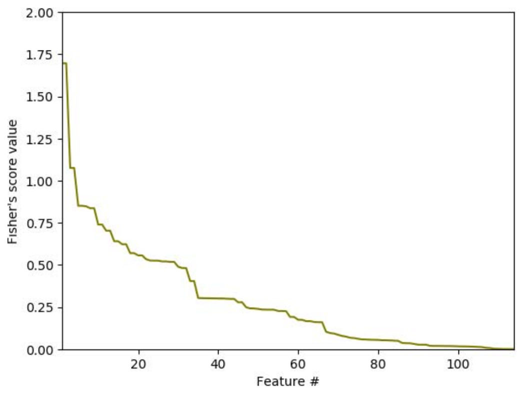

Fig. 2. Values of Fisher's score ordered in descending order.

A. Optimization of Model Hyper-Parameters

Detection of botnet traffic is considered in this subsection as the classification problem where three classes are considered. The first class corresponds to the benign traffic (normal operation of the devices), the second class is Mirai attack and the third class corresponds to the Bashlite attack. Decision trees (DT), k - nearest neighbors (kNN), and logistic regression (LR), are frequently considered as classical classification techniques. This list may be complemented by random forest classifiers (RF) and support vector machines (SVM). Usually these techniques require lesser levels of computational power than deep learning techniques. The application of these methods leads different trade-offs between the hyper-parameters of the algorithms their accuracy. Let us first consider how number of features affects overall accuracy.

For the case of kNN, number of nearest neighbors k and number of features are the hyper-parameters. In Figure 3 accuracy of this method computed during 5-fold cross-validation for different values of hyper-parameters is depicted. The most optimal value for the number of nearest neighbours is $k = 1$. In the framework of the present research, the depth of the decision trees was not limited in any manner, therefore, the number of the features is the sole hyper-parameter. Dependence between the number of the features and accuracy of the DT classifier is depicted in Figure 5.

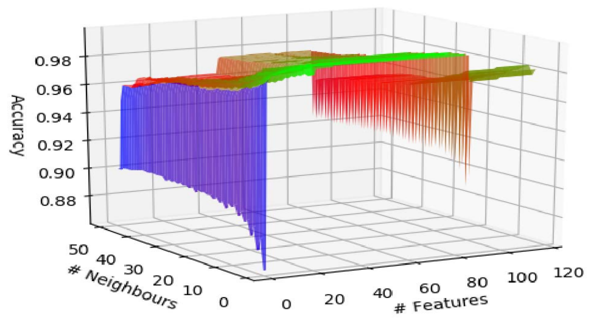

Fig. 3. Trade-offs between accuracy and hyper-parameters in kNN classifiers

For the RF classifier, there are two hyper-parameters number of the features and number of the trees. Like in the case of decision trees depth of the individual trees was not limited in any way. Figure 4 depicts the accuracy as the function of the number of features and number of trees. The accuracy of the classifiers is stable high when number of the features is greater or equal 11 and number of trees is greater or equal than 11.

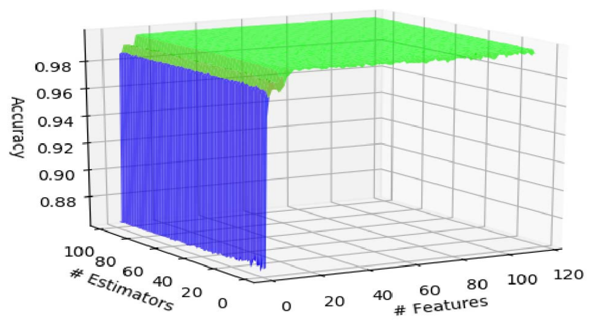

Fig. 4. Trade-offs between accuracy and hyper-parameters in RF classifiers

Figure 5 demonstrates the change in accuracy with the number of the ordered features in learning models induced by DT, RF and kNN (i.e., note that such values are derived from the learning models with the hyper-parameters providing highest accuracy). Logistic regression and support vector machine classifiers have demonstrated lower accuracy rates in comparison to the other methods and therefore omitted from further studies.

Observing the trade-offs between the accuracy and number of features depicted in Figure 5, one may easily see that it is not necessary to base the classification process on all the available features. A feature set having 10-15 features provide optimal accuracy performance. Results of the present section clearly demonstrate that, in our problem, smaller number of features is enough to reach very high accuracy rates, which eliminates the need to use deep-learning algorithms requiring high computational resources. It is important to note that, in most of the cases, intrusion detection in IoT network should be done with limited resources due to the hardware constraints in IoT devices, justifying the minimization of required resources for inducing learning models.

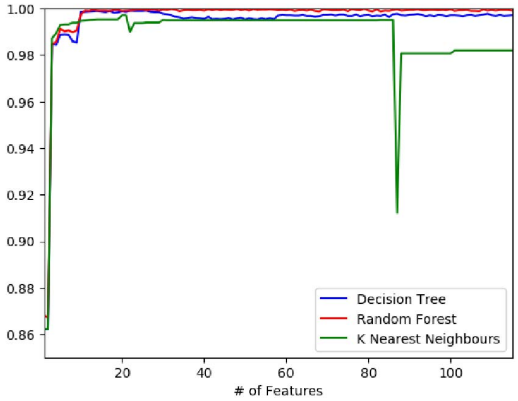

Fig. 5. Comparison of the trade-off between the number of features and accuracy

B. LIME based interpretation

We applied the LIME method [11] to the instances labeled by learning models. In order to compare the local explanations generated for the outputs of these models, we conducted several experiments by using supervised learning methods such as Random Forest, k-NN and Decision Tree with varying sizes of selected features. If not otherwise stated, we chose the hyper-parameters that utilize the best accuracy according to the results of experiments given in Section IV-A. Table II gives the explanations of three test instances belonging to the categories, Benign, Bashlite and Mirai which are correctly classified by the corresponding model (the data points are [4.53 4.53 2.19], [480,00 480.00 219.69] [798.87 798.87 267.78] respectively). In Table II letter 'w' in the feature name column refers to 'weight'. The value of k was chosen as 3 for kNN and the number of estimators was utilized as 10 in Random Forest. The learning models were induced with the best three features selected by the Fisher's Score. In the data-set, the features that belong to Host-IP and Host-MAC&IP categories are very similar. Therefore, local explanations for the same corresponding features (for instance, MI_dir_L1_weight and H_L1_weight) have the same value. It can be observed that all learning methods provide similar intervals for the selected test instances. In case of k-NN, for example, a security analyst can easily deduce from the explanations that if the packet count of a host captured in 1.5 sec and 500 ms intervals are lower than 277.96 and 112.94 respectively, then that host is not compromised by any malware type. Although the acquired explanations are not global, meaning that they do not represent the whole characteristics of the corresponding category, and they are just local explanations of the selected test instances, the results can give intuition to the analyst about the difference between Bashlite and Mirai categories (i.e., Bashlite infection produced more packets then Mirai infection).

TABLE II
EXPLANATIONS FOR SELECTED TEST INSTANCES

Label	**Feature**	**kNN**	**Rand. For.**	**Decision**
		$(k = 3)$	$(10Est.)$	tree
	MI_dir_L1_w	≤ 277.96	≤ 268.37	≤ 270.17
Benign	H_L1_w	≤ 277.96	≤ 268.37	≤ 270.17
	MI_dir_L3_w	≤ 112.94	≤ 110.04	≤ 110.21
	MI_dir_L1_w	≥ 679.91	≥ 680.42	≥ 677.42
Bashlite	H_L1_w	≥ 679.91	≥ 680.42	≥ 677.42
	MI_dir_L3_w	≥ 246.09	≥ 247.44	≥ 244.75
	MI_dir_L1_w	> 277.96	> 268.37	> 270.17
		≤ 595.68	≤ 594.29	≤ 593.59
Mirai	H_L1_w	> 277.96	> 268.37	> 270.17
		≤ 595.68	≤ 594.29	≤ 593.59
	MI_dir_L3_w	> 193.25	> 191.15	> 194.09
		≤ 246.09	≤ 247.44	≤ 244.75

We applied LIME to the randomly selected 50 test instances from each category. The results of correctly categorised instances are given in Table III (note that f1 refers to the feature, MI_dir_L1_weight and f2 refers to MI_dir_L3_weight). As the Host-IP and Host-MAC&IP categories have the same values for the same statistical feature, we did not include the Host-IP category in the table. Each row gives the inequality set for the explanation and the number of instances explained by such an exactly the same inequality set. The results show that all instances belonging to the benign category are represented by one common inequality set whereas five and eight inequality sets are created for Bashlite and Mirai, respectively. As the identified explanations are local and the models are trained with only three features, we observed the same exact explanations for the instances of different categories as one may expect (namely, explanation overlap). The rows shaded by the same gray tone show the same inequality set that is recognized in different categories. For instance, all benign, seven Bashlite and seven Mirai instances are explained by the same inequality set, $MI\_L1\_weight \leq 277.96$,MI_L3_weight≤ 112.94. Another similar result is obtained for the set, $MI\_L1\_weight > 679.91$, MI_L3_weight> 246.09, which explains 26 Bashlite and 7 Mirai instances.

It is obvious that providing the same explanation to different classes would create big confusion for the security analysts

TABLE III
SAMPLE EXPLANATIONS FOR THE OUTPUT OF K-NN

Class	Inst.	Explanation Rules
Benign	50	f1 ≤ 277.96, f2 ≤ 112.94
Bashlite	7	f1 ≤ 277.96, f2 ≤ 112.94
	4	f1 > 679.91, 193.25 < f2 ≤ 246.09
	26	f1 > 679.91, f2 > 246.09
	1	f1 ≤ 277.96, f2 ≤ 193.25
	1	277.96 < f1 ≤ 595.68, 112.94≤ f2 ≤ 193.25
Mirai	4	277.96 < f1 ≤ 595.68, 193.25 < f2 ≤ 246.09
	12	277.96 < f1 ≤ 595.68, 112.94 < f2≤ 193.25
	7	f1 > 679.91, f2 > 246.09
	12	595.68 < f1 ≤ 679.91,193.25 < f2≤ 246.09
	7	f1 ≤ 277.96, f2 ≤ 112.94
	3	595.68 < f1 ≤ 679.91,112.94 < f2 ≤ 193.25
	1	f1 ≤ 277.96,112.94 < f2 ≤ 193.25
	1	277.96 < f1 ≤ 595.68, f2 ≤ 112.94

even if the accuracy of the model is so high. The hyper-parameter choices, including the number of selected features, applied at each step before the post-hoc interpretation would definitely have an impact on the overlaps of the local explanations. In this work, we focus on the implications of these choices rather than the decision quality of the interpretation algorithm itself. In order to show the explanation overlap generated by the LIME interpretation in a better way, we restructured the sample data given in Table III as shown in Table IV so that each row gives a distinct explanation with the distribution of the instance categories described by that explanation. For instance, the first row, $f1 \leq 277.96, f2 \leq 112.94$, explains 50 benign, 7 Bashlite and 7 Mirai instances, which is not the ideal case for an analyst as the same inequality explains three categories. On the other side, the second row, $f1 > 679.91, 193.25 < f2 \leq 246.09$ just explains 4 Bashlite instances but no benign and Mirai ones, not creating a confusion.

TABLE IV
CLASS DISTRIBUTION FOR EACH EXPLANATION (LEARNING MODEL IS K-NN)

Explanation Rule	Benign	Bashlite	Mirai
f1 ≤ 277.96, f2 ≤ 112.94	50	7	7
f1 > 679.91, 193.25 < f2 ≤ 246.09	0	4	0
f1 > 679.91, f2 > 246.09	0	26	7
f1≤ 277.96, f2 ≤ 193.25	0	1	0
277.96 < f1 ≤ 595.68, 112.94 ≤ f2 ≤ 193.25	0	1	12
277.96 < f1 ≤ 595.68, 193.25 < f2 ≤ 246.09	0	0	4
595.68 < f1 ≤ 679.91,193.25 < f2 ≤ 246.09	0	0	12
595.68 < f1 ≤ 679.91,112.94 < f2 ≤ 193.25	0	0	3
f1 ≤ 277.96,112.94 < f2≤ 193.25	0	0	1
277.96 < f1 ≤ 595.68, f2≤ 112.94	0	0	1

We introduced an interpretability quality metric in Equation 2 which computes the degree of explanation overlap by using the entropy notion. Assume that e_i is the ith explanation in an explanation set E, K is the number of categories and p_k is the ratio of instances labeled by category k to the all instances described by e_i.

$$e_i = \sum_{k=1}^{K} -p_k \cdot \log_2 p_k \tag{2}$$

The value of $p_k \cdot \log_2 p_k$ is considered as zero when p_k is zero. Equation 2 gets the lowest value, zero, when all instances explained with one inequality belongs to the same category (namely, explanation overlap is zero) and provides the highest value in case of instances are equally distributed among the categories. Therefore, the first, third and fifth rows in Table IV have entropy values greater than zero whereas all others are exactly zero.

Let's assume that we have N instances and apply LIME to get an explanation for each instance. The explanation overlap, E, of an entire explanation set having N elements is computed as follows:

$$E = \sum_{i=1}^{N} e_i \tag{3}$$

where e_i is computed by Equation 2.

Figure 6 shows the explanation overlap (EO) of a randomly selected instance set (recall that we selected 50 from each category) and explained by LIME for the learning models created by decision trees, kNN and random forest with varying selected features. The x-axis of the graph gives the number of features ordered according to the Fisher's Score and the y-axis demonstrates the value of explanation overlap obtained by the chosen features (i.e., using Equation 3). The results show that all machine learning methods reach the zero value for EO between 13 and 17 features (i.e., although all of them provide non-zero values for some greater feature numbers), meaning that, at the post-hoc interpretation step, the LIME requires at least such number of features to assign one explanation to just only one category. If the machine learning model utilizes fewer features, the explanations may, in turn, confuse the analysts so that one inequality set may explain more than one category. On the other side, once reaching the zero value, LIME provides more zero-valued results for the interpretation of the decision tree models for the greater number of features as there exist fewer fluctuations in the remaining of the graph for this learning model. However, RF and kNN still give much non-zero EO values after 13-17 features.

Recall, in Figure 5, it is shown that the learning models have already reached the optimal accuracy values around 10-15 features. Therefore, we can deduce that it is possible to have a clear explanation figure and optimal accuracy with 13-17 features in our problem. However, such a number of features can not be comprehensible by the experts as the feature set has so many inequalities. Miller's psychological theory states that humans can handle $7(+/-)2$ abstract entities at the same time [23]. Although there is no clear definition of interpretability (i.e., whether it includes comprehensibility or acceptability), it is obvious that 13-17 features may not be preferable by the experts in spite of the high detection accuracy rates. Such high

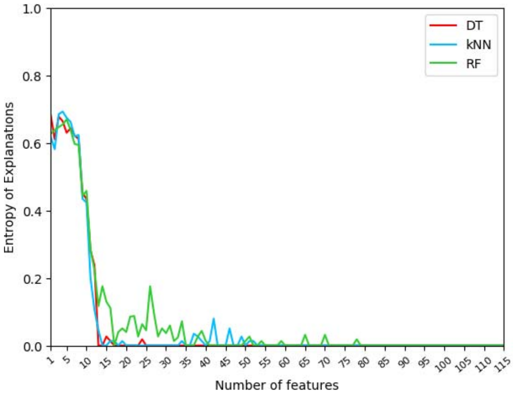

Fig. 6. Equation overlap results by using the features ranked by Fisher's Score

numbered features can not easily explain the decision to the human analyst even in the decision trees which are considered as one of the most inherently interpretable models.

Additionally, the dataset has various features belonging to different time-variants but seem semantically similar, which may be very hard for analysts to dissect the categories. Figure V shows the best 20 features selected by Fisher's score. It can be observed that most of the packet counts in different time intervals (for instance, "MI_dir_L1_weight", "MI_dir_L3_weight" or "MI_dir_L0.1_weight") are determined as the best features which are, in turn, included in the interpretation sets. For an analyst, analyzing all of them may not be much sense if there are no big behavioral deviations in such different intervals. The similarity of data in feature categories, Host-IP and Host-MAC&IP, makes the comprehensibility issue more problematic.

TABLE V
SELECTED FEATURES BY FISHER'S SCORE

Features (in a descending order)
MI_dir_L1_weight
H_L1_weight
MI_dir_L3_weight
H_L3_weight
MI_dir_L0.1_weight
H_L0.1_weight
MI_dir_L5_weight
H_L5_weight
MI_dir_L0.01_weight
H_L0.01_weight
HH_L1_weight
HH_jit_L1_weight
MI_dir_L1_variance
H_L1_variance
HH_L3_weight
HH_jit_L3_weight
MI_dir_L0.01_mean
H_L0.01_mean
MI_dir_L0.1_variance
H_L0.1_variance

Here, it is important to note that the rank of Fisher's score can not be treated as ground truth for the determination of discriminatory power. Wrapper or embedded feature selection methods may yield better accuracy rates with less number of features. One alternative could be also to eliminate the dependent features. Nevertheless, the comprehensive analysis of feature selection methods is beyond the scope of the paper. However, we conducted additional experiment to see the results of another feature set which we selected in the following way: We traversed the list ranked by Fisher Score, but included the 20 features that belong to different feature category (i.e., refer to Table I for these categories) in our final list given in Table VI . We also eliminated the "Host-MAC&IP" category due to its similarity to the "Host-IP" category. This selection means that the final list also includes features with the lower Fisher's Score.

TABLE VI
CUSTOMARY SELECTION OF THE FEATURES

Features (in a descending order)
H_L1_weight
HH_L1_weight
HH_jit_L1_weight
H_L1_variance
H_L0.01_mean
HH_L0.01_std
HpHp_L0.01_mean
HH_L1_mean
HH_jit_L5_mean
HH_jit_L0.01_variance
HpHp_L0.01_std
HpHp_L0.01_magnitude
HH_L0.01_magnitude
HH_L0.01_radius
HH_L0.01_pcc
HpHp_L0.01_radius
HpHp_L5_covariance
HpHp_L5_pcc
HH_L0.1_covariance
HpHp_L0.1_weight

Explanation overlap and accuracy results for the custom feature set are given in Figures 7 and 8. It is interesting to note that the LIME interpretation of all models reached zero with 6 features for the value of equation overlap, and there does not exist any fluctuation in the remaining part of the graph for greater numbers of features. When the results in Figures 8 and 5 are compared, it can be deduced that the optimal detection accuracy is already reached with 5 features, and the overall accuracy figure is similar to the previous case in which the strictly ranked feature set was used. The number of selected features is still within the range of limits stated in Miller's theory. As the features belong to different categories, it can be argued that the security analysts could better perceive the interpretation rules and understand the distinctions between the benign and malware types.

Although we have not thoroughly investigated all feature selection methods, the reduction in the size of the optimal feature set from the accuracy point of view could be attributed to the possible dependencies among features or it can be argued that a filter method which is computationally cheap is not enough. However, the quality metric, explanation overlap, that we proposed in this study, supported the interpretability analysis of the selected features.

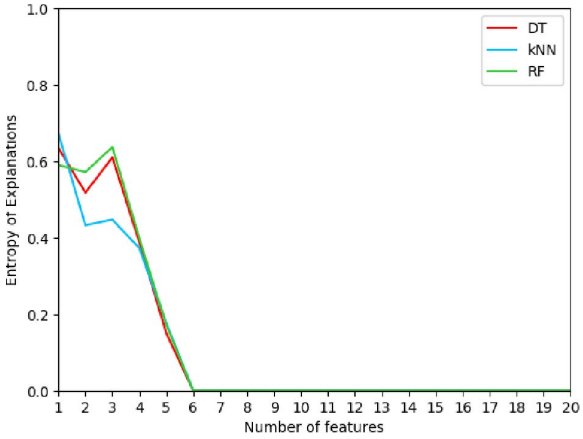

Fig. 7. Using custom features

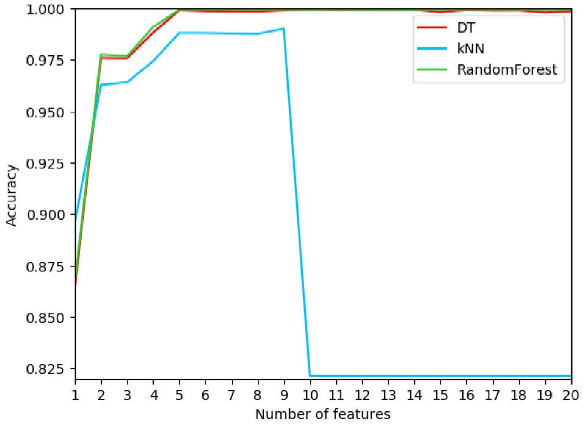

Fig. 8. 3-label accuracy using custom features

V. Conclusion

In this study, we analyzed the feature selection and its impact on the post-hoc local interpretation of the learning model outputs within the context of the botnet detection in IoT networks. A quality metric for the explanations of model decisions reflecting the security analyst perspective is introduced to facilitate this analysis. This metric basically evaluates whether the given explanation describes just only one category or not. If the explanation is prone to explain one category, this is preferable. In our experiments, we utilized a well-known method, LIME, in the post-hoc interpretation phase.

This paper demonstrated that, by using the selected dataset chosen from the problem domain, it is possible to have very high accurate classical machine learning models which can produce explanations that do not create confusion for the security analysts. Our quality metric enabled us to provide such an analysis of interpretability and accuracy in a common picture. Our work is distinguished as it investigates the feature selection and interpretability within the IoT botnet detection domain.

References

[1] I. Andrea, C. Chrysostomou, and G. Hadjichristofi, “Internet of things: Security vulnerabilities and challenges,” in *Computers and Communication (ISCC), 2015 IEEE Symposium on*. IEEE, 2015, pp. 180–187.

[2] P. Paganini, “Ovh hosting hit by 1tbps ddos attack, the largest one ever seen,” *https://securityaffairs.co/wordpress/51640/cyber-crime/tbps-ddos-attack.html*, 2016.

[3] S. Hilton, “Dyn analysis summary of friday october 21 attack (2016),” *URL https://dyn. com/blog/dyn-analysis-summary-of-fridayoctober-21-attack*, 2016.

[4] B. B. Zarpelão, R. S. Miani, C. T. Kawakani, and S. C. de Alvarenga, “A survey of intrusion detection in internet of things,” *Journal of Network and Computer Applications*, vol. 84, pp. 25–37, 2017.

[5] A. L. Buczak and E. Guven, “A survey of data mining and machine learning methods for cyber security intrusion detection,” *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1153–1176, 2016.

[6] C. Zimmerman, “Ten strategies of a world-class cybersecurity operations center,” *MITRE corporate communications and public affairs. Appendices*, 2014.

[7] A. Bibal and B. Frénay, “Interpretability of machine learning models and representations: an introduction,” in *Proceedings on ESANN*, 2016, pp. 77–82.

[8] A. Vellido, J. D. Martín-Guerrero, and P. J. Lisboa, “Making machine learning models interpretable.” in *ESANN*, vol. 12. Citeseer, 2012, pp. 163–172.

[9] A. A. Freitas, “Comprehensible classification models: a position paper,” *ACM SIGKDD explorations newsletter*, vol. 15, no. 1, pp. 1–10, 2014.

[10] Z. C. Lipton, “The mythos of model interpretability,” *Commun. ACM*, vol. 61, no. 10, pp. 36–43, Sep. 2018. [Online]. Available: http://doi.acm.org/10.1145/3233231

[11] M. T. Ribeiro, S. Singh, and C. Guestrin, “Model-agnostic interpretability of machine learning,” *arXiv preprint arXiv:1606.05386*, 2016.

[12] A. Adadi and M. Berrada, “Peeking inside the black-box: A survey on explainable artificial intelligence (xai),” *IEEE Access*, vol. 6, pp. 52 138–52 160, 2018.

[13] M. T. Ribeiro, S. Singh, and C. Guestrin, “Anchors: High-precision model-agnostic explanations,” in *AAAI Conference on Artificial Intelligence*, 2018.

[14] R. Guidotti, A. Monreale, S. Ruggieri, D. Pedreschi, F. Turini, and F. Giannotti, “Local rule-based explanations of black box decision systems,” *arXiv preprint arXiv:1805.10820*, 2018.

[15] D. Baehrens, T. Schroeter, S. Harmeling, M. Kawanabe, K. Hansen, and K.-R. MÃžller, “How to explain individual classification decisions,” *Journal of Machine Learning Research*, vol. 11, no. Jun, pp. 1803–1831, 2010.

[16] P. W. Koh and P. Liang, “Understanding black-box predictions via influence functions,” *arXiv preprint arXiv:1703.04730*, 2017.

[17] K. Amarasinghe, K. Kenney, and M. Manic, “Toward explainable deep neural network based anomaly detection,” in *2018 11th International Conference on Human System Interaction (HSI)*. IEEE, 2018, pp. 311–317.

[18] J. R. Goodall, E. D. Ragan, C. A. Steed, J. W. Reed, G. D. Richardson, K. M. Huffer, R. A. Bridges, and J. A. Laska, “Situ: Identifying and explaining suspicious behavior in networks,” *IEEE transactions on visualization and computer graphics*, vol. 25, no. 1, pp. 204–214, 2019.

[19] D. L. Marino, C. S. Wickramasinghe, and M. Manic, “An adversarial approach for explainable ai in intrusion detection systems,” in *IECON 2018-44th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 2018, pp. 3237–3243.

[20] Y. Meidan, M. Bohadana, Y. Mathov, Y. Mirsky, D. Breitenbacher, A. Shabtai, and Y. Elovici, “N-baiot: Network-based detection of iot botnet attacks using deep autoencoders,” *IEEE Pervasive Computing*, vol. 13, no. 9, 2018.

[21] J. Leonard, S. Xu, and R. Sandhu, “A framework for understanding botnets,” in *Availability, Reliability and Security, 2009. ARES’09. International Conference on*. IEEE, 2009, pp. 917–922.

[22] C. C. Aggarwal, *Data Mining: The Textbook*. Springer Publishing Company, Incorporated, 2015.

[23] G. A. Miller, “The magical number seven, plus or minus two: Some limits on our capacity for processing information.” *Psychological review*, vol. 63, no. 2, p. 81, 1956.