Attribution Methods Assessment for Interpretable Machine Learning

Alfredo Cuzzocrea [1,2,*], Qudrat E. Alahy Ratul [3], Islam Belmerabet [1] and Edoardo Serra [3]

[1] *iDEA Lab, University of Calabria, Rende, Italy*
[2] *Department of Computer Science, University of Paris City, Paris, France*
[3] *Boise State University, Boise, Idaho, United States*

Abstract

In this study, we introduce a generic experimental framework for measuring the degree of *attribution methodologies generality and precision in terms of machine learning interpretability*. In addition, we detail a way for gauging the *consistency* of two attribution approaches. In our experimental work, we concentrate on two well-known model-independent attribution techniques, namely *SHAP* and *LIME*, and evaluate them using two applications in the *attack detection* sector. Our introduced methodology demonstrates the lack of precision, generality, and consistency in both LIME and SHAP. As a result, attribution research needs to be examined more carefully.

Keywords

Artificial Intelligence, Feature Attribution Methods, Machine Learning Interpretability

1. Introduction

Machine learning models are commonly used for solving different types of problems. From *"simple" movie recommendation systems* and *personal voice assistants*, to more *"sensitive"* domains that involve taking *"high" impact* decisions, such as *mortgage approval models* and *healthcare decision support systems*. Therefore, it became inevitable to democratize *Artificial Intelligence (A.I.)* in our society [1]. Regardless of the adoption expansion of ML models, the logic and mechanisms behind these models is still unknown to end users and experts, i.e. making these models be considered *black boxes* [2]. Therefore, depending on these ML algorithms for *decision-making* tasks that are sophisticated such as in *aircraft collision detection systems* without well understanding these models can lead to serious consequences [3]. Hence, in order to solve this problem, many *interpretable* models and explanation methods [2] have been proposed.

Being social impact of ML algorithms significantly increasing, the need for understanding the mechanism behind the decision-making process is also increasing along with it [4]-[7].

A big number of studies targeting this problem has been done. Specifically, *Explainable Artificial Intelligence (X.A.I.)* i.e. a field of study aiming at interpretable ML models development in order to make a transition to *transparent* A.I. [2], i.e. producing more explainable models and methods that explain existing black box models without compromising their predictive performance. *Defense Advanced Research Projects Agency (DARPA)* is a remarkable initiative in this research field, funded by the *U.S. Department of Defense*, which created the X.A.I. program for funding academic and military research [8]. Likely, *"Preparing for the Future of Artificial Intelligence"*- is a report published by the *White House Office of Science and Technology Policy (OSTP)* which represents another example of governmental initiatives emphasizing that A.I. systems should be open, transparent, and understandable for interrogating the assumptions behind the models decisions [9]. Many countries

* This research has been made in the context of the Excellence Chair in Big Data Management and Analytics at University of Paris City, Paris, France
SEDB 2023: 31st Symposium on Advanced Database System, July 02–05, 2023, Galzignano Terme, Padua, Italy
EMAIL: alfredo.cuzzocrea@unical.it (A. Cuzzocrea); qudratealahyratu@u.boisestate.edu (Q. E. A. Ratul); ibelmerabet.idealab.unical@gmail.com (I. Belmerabet); edoardoserra@boisestate.edu (E. Serra)
ORCID: 0000-0002-7104-6415 (A. Cuzzocrea); 0009-0003-7878-0991 (I. Belmerabet); 0000-0003-0689-5063 (E. Serra)

outside of the U.S.A. have already taken the initiative towards transparent A.I. . i.e., the French Strategy for Artificial Intelligence, The *United Kingdom's Academy of Sciences*, and Portugal government, have published their roadmap towards interpretable A.I. [10]-[12]. The *European Union* announced that "A.I. systems should be developed in a manner which allows humans to understand (the basis of) their actions" in order to increase transparency and minimize the risk of bias error [13].

Interpretable A.I. practices in various A.I.-related fields have already became common in the *Tech* industry. Additionally, numerous companies are interested in interpretability in order to commercialize interpretable A.I. products as investments in this field. *Google* is promoting interpretability by including as a core part of their user experience the interpretability planning and treating, designing interpretable models, understanding and communicating their trained models explanations to the users [14]. *FICO*, is a credit score company that also addressed interpretability in credit scoring systems [15] as it has published a white paper with the title of *"Developing Transparent Credit Risk Scorecards More Effectively: An Explainable Artificial Intelligence Approach"*.

In this paper, the main focus is outcome (e.g., classification result) explanations of machine learning models for a specific instance and the explanation methods, by using and experimental A.I. framework that is straightforwardly composed by some well-known tools for ML-based A.I. explainability. This category of methods include rule-based methods (explain instances with simple logic rules) and attribution methods (which give an importance score for each input feature of the complex machine learning model). Rule-based outcome explanation methods, [16] are the first to introduce precision and generality as a requirement. i.e., Precision enforces that, if a rule explains an instance with a specific classification outcome (e.g., class 1), it should never explain other instances with a different classification outcome (e.g., class 0). And generality suggesting that, if a rule explains an instance with a particular classification outcome (e.g., class 1), it should also likely explain other instances having the same classification outcome. This makes precision and generality two important requirements that raise human trust in ML models explanations. However, these requirements are not tested or even implied in attribution methods, but they are defined and measured only on rules.

Machine Learning models are like black boxes that can only classify instances, where it is unknown how the classification procedure is performed. Hence, this paper provides an overview of model agnostic attribution methods along with a methodology on how to evaluate these method based on the precision and the generality of the attribution, and how to measure consistency between different generic attribution methods.

2. Model Agnostic Explanation Methods: An Introduction

Model agnostic methods separate the explanation from the ML model, which gives flexibility in using any interpretable ML method regardless of how the ML model is defined i.e., a model agnostic method uses the ML model as an oracle model. This category covers some basic approaches such as *Dependency Plot (PDP)* [17], *Individual Conditional Expectation (ICE)* [18], feature interaction based on *H-statistic* [19], and *local surrogate models*. Differently from other models, local surrogate ones focus on explaining a specific outcome of a single instance. Several related works in the field of machine learning are reported in [20]-[34].

We report in the following two relevant local surrogate models that are also attribution methods, i.e. LIME and SHAP, which provide for each feature of the classification model inputs an importance score.

2.1. The First Explainability Tool: LIME

In order to explain an outcome of a model, LIME learns an interpretable model locally around the instance and modifies the data sample of a single instance by *tweaking* its feature values and observing how the changes affect the resulting output.

The main idea behind LIME is that: in order to understand the prediction sensitivity w.r.t. each feature of a particular instance, a *"local sensitivity analysis"* is performed on every outcome made by each individual instance that is passed on to the model. Figure 1 shows how LIME works theoretically.

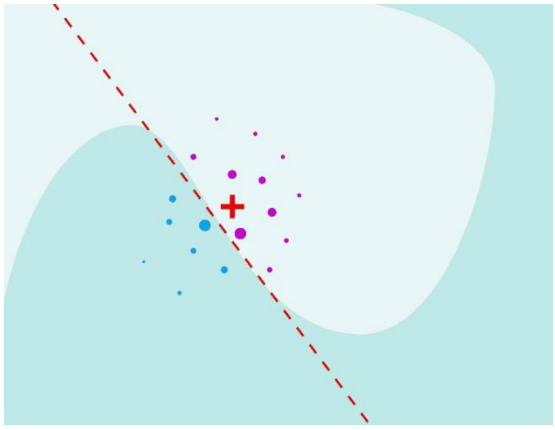


Figure 1: LIME [35]

The blue/orange background represents the original *decision function* i.e. clear to see that it is not *linear*, and the instance to explain is the largest red X mark I. LIME simply generates new instances in the *neighborhood* of I (i.e. *perturbations*) and assigns them weights based on their *proximity* to I. The weights are represented in Figure 1 by the sizes of the symbols i.e. blue circles and red X marks. Based on the model's outcome confidence on these perturbations, LIME approximates the complex model well by learning a linear model i.e. presented with a red line in Figure 1 locally around I. It should be noted that, in this case, the explanation that LIME produces at a local point p is true only locally around the instance I and not globally. The generic formula that returns this explanation is the following:

$$\theta(p) = argmin_{s \in S} F(c, s, \delta_p) + \varepsilon(s) \quad (1)$$

where c is the real function (known as the black box model i.e. the complex machine learning model to explain), s is a surrogate model i.e. the simple model used to approximate c locally around I where δ_p defines this *locality*. This formulation can be applied with different surrogate explanation families S, fidelity functions F, and complexity measures ε. LIME assumes that complexity is opposed to explainability as s usually belongs to the family of linear functions i.e. low in complexity. The loss function F (i.e. *fidelity function*) minimizes the *local mismatch* between the complex machine learning model c and the approximating function s (i.e. the simple model), which is the well-known root mean square error loss function RMSE. Where, typically, s is a linear combination of the input features of the predictive model.

LIME is considered as an attribution method by making use of a linear model, where the coefficients of this linear model are determining the *importance scores* for each feature.

2.2. The Second Explainability Tool: SHAP

SHAP (***SH****aply* ***A****dditive ex****P****lanations*) [36] is a unified approach introduced for interpreting model prediction from different interpretable techniques i.e. LIME [35], *Shapley Sampling Values* [37], *DeepLIFT* [38], *QII* [39], *Layer-wise Relevance Propagation* [40], *Shapley Regression Values* [41] by defining the class of *additive feature attribution* methods. SHAP assigns an importance value for each input feature by employing *game theory* in order to compute the attribution, particularly, by using the *Shapley values*. Which indicate the *reward* received by each player in a cooperative game for his participation in the *coalition*. Let R be the set of all the input features of the ML model, given an instance p and a complex ML model c and by adopting Shapely values we obtain for each feature $i \in T$ an attribution score $\psi_i^c(p)$ as follows:

$$\psi_i(c, p) = \sum_{T \subseteq R \setminus \{i\}} \frac{|T|!\,(|T| - |R| - 1)!}{R!} [c_f(p_{T \cup \{i\}}) - c_f(p_T)] \quad (2)$$

where c_f represents the *confidence* value of the complex ML model c for a particular outcome (e.g., a specific class) and p_T is the instance p having each value $p[r]$ of the feature $r \in R \setminus T$ substituted with the mean value of all r values had among all the possible instances. This score indicates the relevance

of the particular value assigned to each feature r for the classification compared to the mean value of the feature in the instance p.

Clearly, we see that the complexity of computing the feature score is exponential in the number of features. Therefore, approximations are provided in order to overcome such complexity.

Kernel SHAP [36], being one of these approximations, founds on fitting a linear model i.e. defined as follows:

$$s(T) = \psi_0 + \sum_{j \in T} \psi_j \tag{3}$$

This fitting procedure aims to minimize the loss function i.e. defined as follows:

$$\sum_{T \subseteq R} \xi(T)(s(T) - c_f(p_T))^2 \tag{4}$$

where $\xi(T)$ is the kernel, and it is defined as$\xi(T) = \frac{|R|-1}{\binom{|R|}{|T|}|T|(|R|-|T|)}$.

The speed up is obtained by optimizing the loss function $\sum_{T \in A} \xi(T)(s(T) - c_f(p_T))^2$ considering only a random sub-samples $A \subset \{T | T \subseteq R\}$.

All which allows this kernel to align SHAP with LIME.

[42] provides a fast algorithm computation for SHAP in the context of explaining *decision tree*-based machine learning models, e.g., *Random Forest.*

3. How To Measure Precision, Generality, and Consistency of Attribution Models

In this Section, we describe how Precision, Generality, and Consistency are measured in attribution models using our methodology. In [16], precision and generality measurement methods for rules are already defined, but still not for attributions. In this Section, additionally, we propose a method for consistency measurement between two generic attribution techniques.

3.1. Measuring Precision

It is enforced by Precision that, if a rule explains an instance with a specific classification outcome (e.g., class 1), it should never explain other instances with a different classification outcome (e.g., class 0). Thus, it is intuitive that, it is inconsistent and not trustable by a human to provide a rule that explains two different outcomes of a machine learning model. [16], measures inversely the precision of a rule l that explains an outcome t by the percentage of instances that obtain from the machine learning model a different outcome from t and l applies to those instances. In order to define precision to the attribution results, we introduce first two functions. (i) $sel(T, p)$ which returns in $\mathbb{R}^{|T|}$ a vector i.e. the selection of the values of the feature in T. Given $T = i_1, \dots, i_k$ the subset of features for each $j \in \{1, \dots, k\}, sel(T, p)[j] = p[i_j]$. (ii) top_k: $\mathbb{R}^n \to 2^{|k|}$ which returns the top-k feature according to attribution vector att. Let P_t be the set of instances receiving the outcome t and $P_{\neg t}$ be the set of instances not receiving it, given an instance $p \in P_t$ and its attribution att_p, we can inversely measure the attribution precision using the *Reverse Precision (RP)* as follows:

$$RP^k(p, att_p) = \frac{|\{\hat{p} | \hat{p} \in P_{\neg t}, sel(T_{tx}^p, p) = sel(T_{tx}^p, \hat{p})\}|}{|P_{\neg t}|} \tag{5}$$

where $T_{tx}^p = top_k(att_p)$. It is intuitive that, using the reverse precision, we measure the percentage of instances that receive the outcome t and have the same values of top-k features as the specifically explained instance. Given a particular outcome t we can compute the average of the reverse precision scores at k for each instance in P_t as follows:

$$avgRP^k(P_t) = \frac{\Sigma_{p \in P_t} RP^k(p, att_p)}{|P_t|} \tag{6}$$

3.2. Measuring Generality

It is suggested by Generality that, if a rule explains an instance with a particular classification outcome (e.g., class 1), it should also likely explain other instances having the same classification outcome. Given two instances p_1 and p_2 along with their attribution vectors att_{p_1} and att_{p_2}, we define the function that measures how many top-k features does att_{p_1} and att_{p_2} have in common as follows:

$$common_k(att_{p_1}, att_{p_2}) = |top_k(att_1) \cap top_k(att_2)| \quad (7)$$

where the function top_k is already defined in the precision Section. Given an instance $p \in P_t$ the generality of its attribution att_p in the context of the top h neighbour instances in P_t of p is measured using the function $generality(x, k, h, agg)$ defined as follows:

$$agg(\{common_k(att_p, att_{\hat{p}}) | \hat{p} \in topNeighbour_h(p, P_t)\}) \quad (8)$$

where $agg \in \{sum, min, max\}$ and $topNeighbour_h(x, I_a)$ return the top h neighbour instances in P_t of p. These different aggregation functions can provide us with more information about the commonalities of the attributions and their distribution among the top h neighbor instances. It should be noted that the function $common_k$ does not consider the values of the top_k features from the attributions, i.e., justified by the fact that, in the $generality$ function, we use the $common$ function only among the nearest neighbors, then we assume the similarity requirement of the values of these instances. Given a particular outcome t the average of the generality scores at k for the top-h neighbour instances for each instance in P_t is computed as follows:

$$avgGen(P_t, k, h, agg) = \frac{\Sigma_{p \in P_t} generality(p, k, h, agg)}{|P_t|} \quad (9)$$

3.3. Measuring Consistency

As the attribution method SHAP unifies a number of other attribution methods, one of them is LIME, we propose a simple technique to compare the attribution between two different attribution methods. Given an instance p and an attribution method m, we denote the attribution scores for the instance p provided by the attribution method m by $attr_m(p)$.

Given the set of instances P_t receiving an outcome t and the two attribution methods m_1 and m_2, we define the consistency score for the top-k features between m_1 and m_2 as follows:

$$cons_k(P_t, m_1, m_2) = \frac{\Sigma_{p \in P_t} common_k(att_{m_1}(p), att_{m_2}(p))}{|P_t|} \quad (10)$$

where the function $common_k$ is already defined in the *generality* Section.

4. Experimental Evaluation and Analysis

In this Section, in order to evaluate attribution methods in two attack detection contexts, i.e., network traffic and power system. We apply our methodology focusing on the explanation (i.e. the attribution) of *attack instances* particularly. We start by describing the two datasets. Then, we display the classification results of different classification models in detecting such attacks. Finally, and after selecting the best classification model, and using our methodology, we evaluate LIME and SHAP attribution approaches in correctly interpreting attack instances that have been classified by this model.

4.1. The Datasets

The datasets used in the two attack detection contexts are described as follows. *UNSW-NB 15*: Network Traffic: Represents a comprehensive network-based dataset [43] that can reflect the network traffic's modern scenarios, numerous types of low footprint intrusions, and depth structured information about this traffic. The IXIA PerfectStorm tool in the Cyber Range Lab of the Australian Centre for Cyber Security (ACCS) has created this raw network packets of this dataset that contains real normal behavior and synthesized attack activities of network traffic on Jan 22, 2015, and 15 hours on Feb 17,

2015 with a simulation period of 16 hours. This dataset consists of 2, 540, 044 records and contains 49 distinct features. *ICS: Power System*: captures different scenario of power system disturbance. This dataset [44] is a derivation from an original dataset containing 15 sets of 37 power system event scenario for each set (28 attack events and 9 normal events). It consists of a total of 128 features and 78, 377 records.

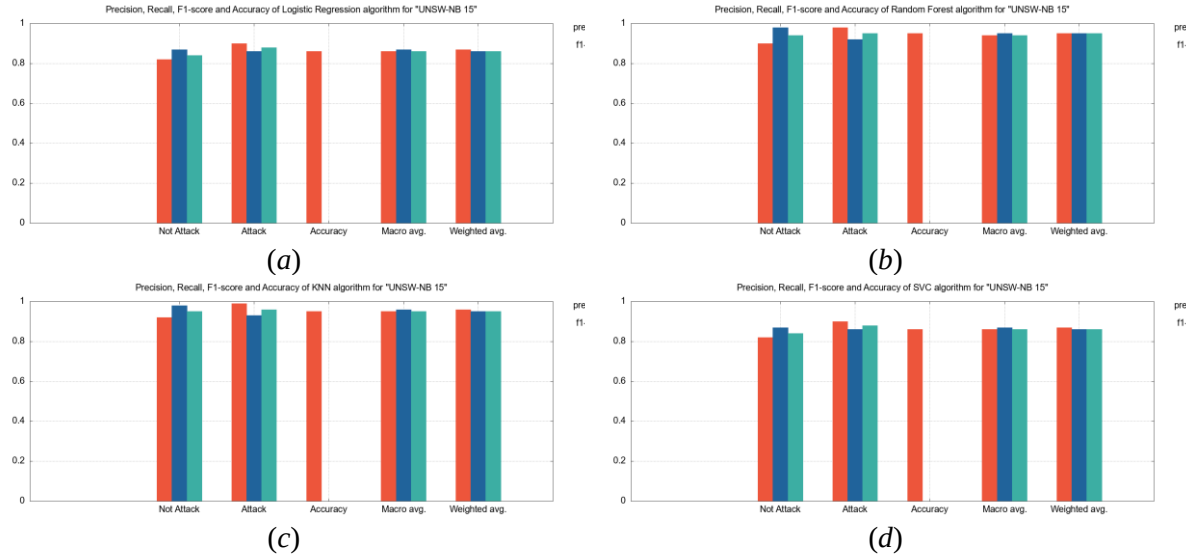


Figure 2: "UNSW-NB 15" experimental patterns of: Linear Regression (*a*), Random Forest (*b*), KNN (*c*), and SVC (*d*) algorithms

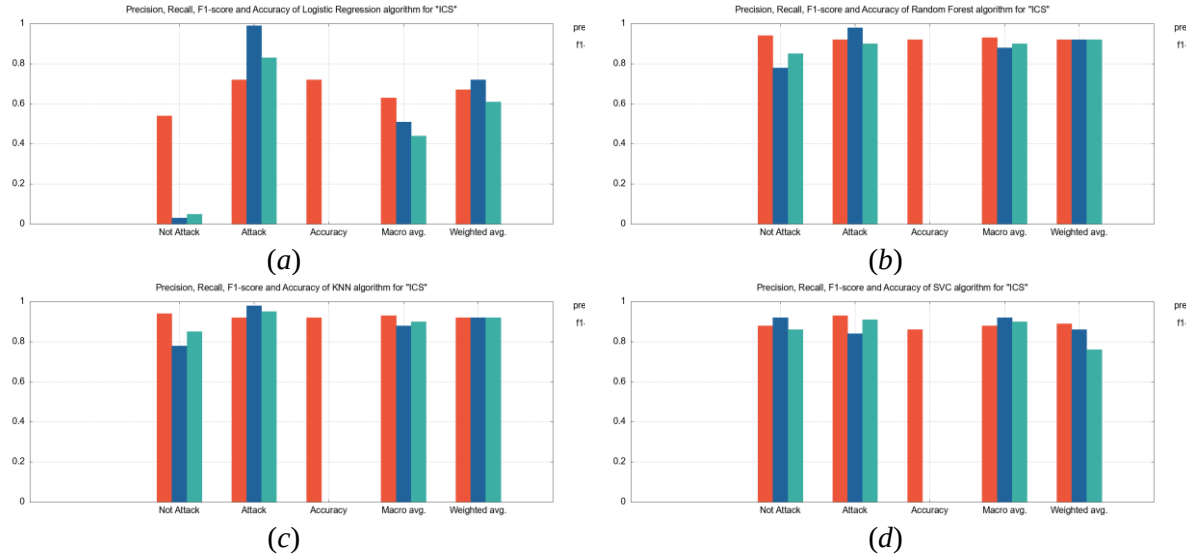


Figure 3: "ICS: Power System" experimental patterns of: Linear Regression (*a*), Random Forest (*b*), KNN (*c*), and SVC (*d*) algorithms

4.2. Experimental Results

Before the start of the classification, we perform a *one-hot encoding transformation* on all the non-numeric features. First, we split the dataset into a 70% training set and a 30% test set. The split was initially provided for "UNSW-NB 15" but with the same percentages. Then, we train and test the: *Logistic Regression*, Random Forest, *KNN*, *Support Vector Classification* (with *RBF kernel*) classification models. Figure 2 displays the classification results (*Precision*, *Recall*, *F1-score* and *Accuracy*) of all the classification models for "UNSW-NB 15", and Figure 3 for "ICS: Power System", respectively. We state that KNN and Random Forest provide the best and most comparable results. Finally, while applying our methodology in order to evaluate the attribution methods, we only concentrated on the Random Forest classifier as SHAP offers a more efficient computation.

4.3. Analysis of Attribution Precision

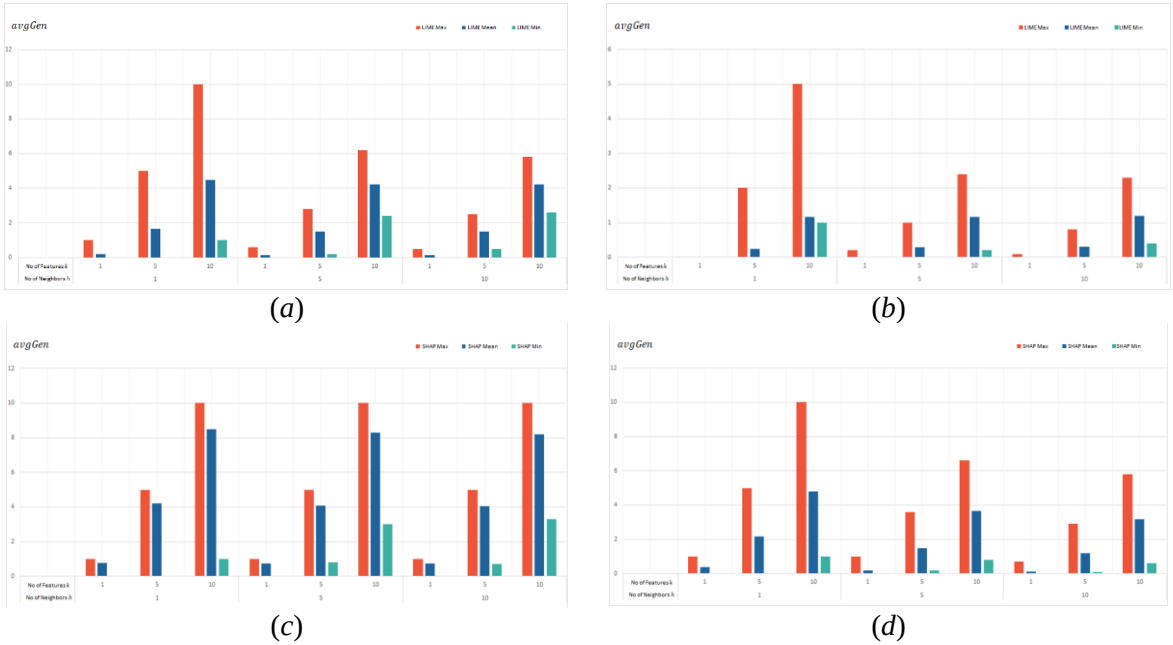
In Table 1 and Table 2 we show $avgRP^k(P_t)$ for LIME and SHAP when varying the number k in "UNSW-NB 15" and "ICS: Power System", respectively. We can see that, even for $k = 6$, the values of the top-6 features, according to the attributions of the attack instances for both the methods (LIME and SHAP), are the same for the normal behavior in both the datasets. In fact, $avgRP^6(P_t) > 0$. Which highlights the fact that as the explanation provided by LIME and SHAP for the attack instances also apply to normal behavior instances, then both LIME and SHAP are not really precise. Additionally, no attribution technique is better than the other. It is also interesting to see that according to the specific attribution procedure, the value of the top-1 most important feature is the same in numerous normal behavior instances, i.e. more than 70% in "UNSW-NB 15" and 50% in "ICS: Power System". Intuitively, this shows how precision still remains an open problem for attribution methods.

Table 1
$\boldsymbol{avgGen(I_a, k, h, agg)}$ by varying $\boldsymbol{k}$ and $\boldsymbol{h}$ for "UNSW-NB 15"

$\boldsymbol{k}$	1	2	3	4	5	6	7	8	9	10
LIME $\boldsymbol{avgRP^k}$	78	64	31.5	31	28	27	19	17	16	14
SHAP $\boldsymbol{avgRP^k}$	81	63	23	11	3	0.5	0	0	0	0

Table 2
$\boldsymbol{avgGen(I_a, k, h, agg)}$ by varying $\boldsymbol{k}$ and $\boldsymbol{h}$ for "ICS: Power System"

$\boldsymbol{k}$	1	2	3	4	5	6	7	8	9	10
LIME $\boldsymbol{avgRP^k}$	59	32	18	5	3	2	1	0	0	0
SHAP $\boldsymbol{avgRP^k}$	52	17	9	6	4	2	0	0	0	0



(*a*) (*b*) (*c*) (*d*)

Figure 4: $avgGen(I_a, k, h, agg)$ by varying k and h for: LIME in "ICS: Power System" (*a*) and in "ICS: Power System" (*b*), SHAP in "ICS: Power System" (*c*) and in "ICS: Power System" (*d*)

4.4. Analysis of Attribution Generality

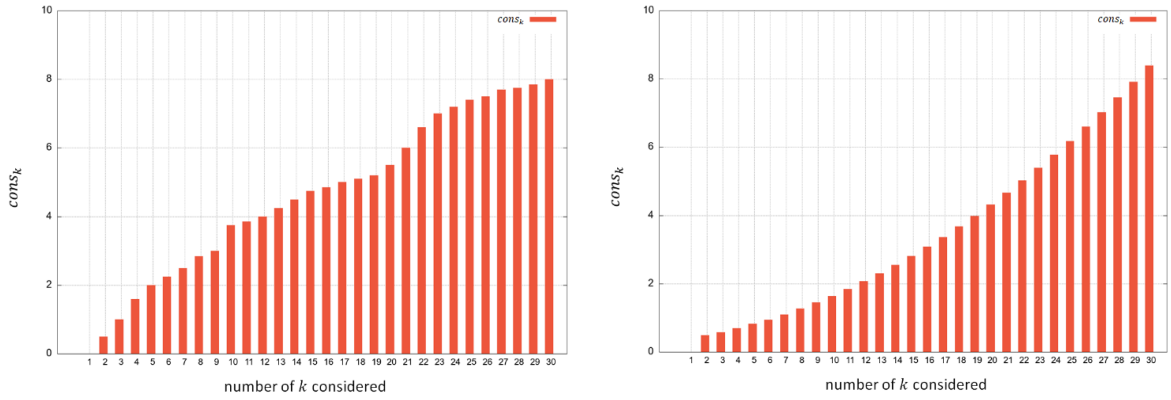
Figure 4 (*a*) shows the values of $avgGen(P_t, k, h, agg)$ by varying the number of top features k and the number of close neighbors h for LIME and Figure 4 (*c*) for SHAP in "UNSW-NB 15", and Figure 4 (*b*) shows the average generality values for LIME, and Figure 4 (*d*) for SHAP in "ICS: Power

System", respectively. Based on the results, even the closest instances produce attributions that are drastically different. Therefore, the attributions of both methods is not so general and attribution seems unique for the specific instance rather than being generic.

4.5. Analysis of Attribution Consistency

In this experiment, we show how LIME and SHAP are consistent, especially when considering SHAP as a method that unifies a number of other attribution methods including LIME. Figure 5 (*a*) shows $cons_k(P_t, LIME, SHAP)$ when varying the number k in "UNSW-NB 15" and Figure 5 (*b*) in "ICS: Power System" respectively. We can see that LIME and SHAP agree over less than third of the top-k features for both the datasets. Particularly, in terms of top-1, top-2, and top-3 features, where the methods have a strong disagreement.

This evaluation highlights the fact that the attributions provided by the two methods are different, and it is difficult to determine which attribution method is the best, as they have low performances in terms of precision and generality.



(*a*) (*b*)

Figure 5: $cons_k(I_a, LIME, SHAP)$ by varying k in: "UNSW-NB 15" (*a*), and "ICS: Power System" (*b*)

5. Conclusions and Future Work

Attribution methods are crucial for machine learning models interpretability evaluation. In this paper, we provided a new methodology to evaluate the precision, generality, and consistency of attribution methods. And, we applied it in order to evaluate the two common model agnostic attribution models, LIME and SHAP, on two attack classification tasks related to network traffic and power systems in the *industrial control system* field. Our methodology highlighted the lack of precision and generality in these two methods and the fact that no method is really better than the other. Regardless of SHAP being proposed as the unification model and that it should generalize LIME, we inspected the attribution results of the two attribution methods in numerous cases, and they were very different. Based on this evaluation, we came with a conclusion that there is no best model for attribution and that still more research is needed to overcome the limitations of precision and generality in this topic.

Future work has many aspects. One concerns with integrating the methodologies with emerging big data analytics tools, in different settings (e.g., [45-48]).

6. References

[1] A. Adadi, M. Berrada, "Peeking Inside the Black-box: A Survey on Explainable Artificial Intelligence (XAI)", *IEEE Access 6* (2018) pp. 52138–52160

[2] D. V. Carvalho, E. M. Pereira, J. S. Cardoso, "Machine Learning Interpretability: A Survey on Methods and Metrics", *Electronics 8.8* (2019) art. 832

[3] S. Temizer, M. Kochenderfer, L. Kaelbling, T. Lozano-Pérez, and J. Kuchar, “Collision Avoidance for Unmanned Aircraft Using Markov Decision Processes”, in: *AIAA Guidance, Navigation, and Control Conference*, 2010, p. 8040
[4] G. Montavon, S. Lapuschkin, A. Binder, W. Samek, and K.-R. Müller, “Explaining Non-Linear Classification Decisions with Deep Taylor Decomposition”, in: *Pattern Recognition 65*, 2017, pp. 211–222
[5] M. Du, N. Liu, and X. Hu, “Techniques for Interpretable Machine Learning”, *Communications of the ACM 63.1* (2019) pp. 68–77
[6] D. Golovin, B. Solnik, S. Moitra, G. Kochanski, J. Karro, and D. Sculley, “Google Vizier: A Service for Black-Box Optimization”, in*: 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2017, pp. 1487–1495
[7] C. Rudin, “Stop Explaining Black Box Machine Learning Models for High Stakes Decisions and Use Interpretable Models Instead”, *Nature Machine Intelligence 1.5* (2019) pp. 206–215
[8] D. Gunning, “Explainable Artificial Intelligence (XAI)”, *Defense Advanced Research Projects Agency (DARPA), nd Web 2.2* (2017) art. 1
[9] P. H. Press, *“Preparing for the Future of Artificial Intelligence”*, 2016, pp. 2–12
[10] C. Villani, “French National Strategy for Artificial Intelligence”, 2019. URL: https://www.aiforhumanity.fr/en/
[11] “Portuguese National Initiative on Digital Skills. AI Portugal 2030. 2019”, 2019. URL: https://www.incode2030.gov.pt/sites/default/files/draft_ai_portugal_2030v_18mar2019.pdf
[12] “Machine Learning: The Power and Promise of Computers that Learn by Example”, 2019. URL: https://royalsociety.org/topics-policy/projects/machine-learning/
[13] European Commission, “Algorithmic Awareness-Building. 2018”, 2019. URL: https://ec.europa.eu/digital-single-market/en/algorithmic-awareness-building
[14] A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-Lopez, D. Molina, R. Benjamins et al., “Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges Toward Responsible AI”, *Information Fusion 58* (2020) pp. 82–115
[15] G. Fahner, “Developing Transparent Credit Risk Scorecards More Effectively: An Explainable Artificial Intelligence Approach”, *Data Anal. 2018* (2018) art. 17
[16] M. T. Ribeiro, S. Singh, C. Guestrin, “Anchors: High-Precision Model-Agnostic Explanations”, in: *AAAI Conference on Artificial Intelligence*, 2018, pp. 1527–1535
[17] J. H. Friedman, “Greedy Function Approximation: A Gradient Boosting Machine”, *Annals of Statistics 29.5* (2001) pp. 1189–1232
[18] A. Goldstein, A. Kapelner, J. Bleich, E. Pitkin, “Peeking Inside the Black Box: Visualizing Statistical Learning with Plots of Individual Conditional Expectation”, *Journal of Computational and Graphical Statistics 24.1* (2015) pp. 44–65
[19] J. H. Friedman, B. E. Popescu, “Predictive Learning via Rule Ensembles”, *The Annals of Applied Statistics* (2008) pp. 916–954
[20] M. Ceci, A. Cuzzocrea, D. Malerba, “Supporting Roll-Up and Drill-Down Operations Over OLAP Data Cubes with Continuous Dimensions via Density-Based Hierarchical Clustering”, in: *SEBD*, 2011, pp. 57–65
[21] E. Serra, M. Joaristi, A. Cuzzocrea, “Large-Scale Sparse Structural Node Representation”, in: *2020 IEEE International Conference on Big Data (Big Data)*, IEEE, 2020, pp. 5247–5253
[22] P. Braun, A. Cuzzocrea, T. D. Keding, C. K. Leung, A. G. Padzor, and D. Sayson, “Game Data Mining: Clustering and Visualization of Online Game Data in Cyber-Physical Worlds”, *Procedia Computer Science 112* (2017) pp. 2259–2268
[23] A. Guzzo, D. Sacca, E. Serra, “An Effective Approach to Inverse Frequent Set Mining”, in: *2009 9th IEEE International Conference on Data Mining*, IEEE, 2009, pp. 806–811
[24] K. J. Morris, S. D. Egan, J. L. Linsangan, C. K. Leung, A. Cuzzocrea, C. S. Hoi, “Token-Based Adaptive Time-Series Prediction by Ensembling Linear and Non-linear Estimators: A Machine Learning Approach for Predictive Analytics on Big Stock Data”, in: *2018 17th IEEE International Conference on Machine Learning and Applications*, IEEE, 2018, pp. 1486–1491

[25] E. Serra, V. Subrahmanian, “A Survey of Quantitative Models of Terror Group Behavior and an Analysis of Strategic Disclosure of Behavioral Models”, *IEEE Transactions on Computational Social Systems 1.1* (2014) pp. 66–88
[26] L. Bellatreche, A. Cuzzocrea, S. Benkrid, “F &A : A Methodology for Effectively and Efficiently Designing Parallel Relational Data Warehouses on Heterogenous Database Clusters”, in: *International Conference on Data Warehousing and Knowledge Discovery*, Springer, 2010, pp. 89–104
[27] O. Korzh, M. Joaristi, E. Serra, “Convolutional Neural Network Ensemble Fine-Tuning for Extended Transfer Learning”, in: *International Conference on Big Data*, Springer, 2018, pp. 110–123
[28] S. Ahn, S. V. Couture, A. Cuzzocrea, K. Dam, G. M. Grasso, C. K. Leung, K. L. McCormick, B. H. Wodi, “A Fuzzy Logic Based Machine Learning Tool for Supporting Big Data Business Analytics in Complex Artificial Intelligence Environments”, in: *2019 IEEE International Conference on Fuzzy Systems*, IEEE, 2019, pp. 1–6
[29] E. Serra, A. Sharma, M. Joaristi, O. Korzh, “Unknown Landscape Identification with CNN Transfer Learning”, in: *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, IEEE, 2018, pp. 813–820
[30] E. Serra, A. Shrestha, F. Spezzano, A. Squicciarini, “Deeptrust: An Automatic Framework to Detect Trustworthy Users in Opinion-Based Systems”, in: *10th ACM Conference on Data and Application Security and Privacy*, 2020, pp. 29–38
[31] M. Joaristi, E. Serra, F. Spezzano, “Inferring Bad Entities through the Panama Papers Network”, in: *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, IEEE, 2018, pp. 767–773
[32] M. Joaristi, E. Serra, F. Spezzano, “Detecting Suspicious Entities in Offshore Leaks Networks”, *Social Network Analysis and Mining 9.1* (2019) pp. 1–15
[33] M. Joaristi, E. Serra, “SIR-GN: A Fast Structural Iterative Representation Learning Approach for Graph Nodes”, *ACM Transactions on Knowledge Discovery from Data 15.6* (2021) pp. 1–39
[34] M. Joaristi, A. Putnam, A. Cuzzocrea, E. Serra, “Ribs: Risky Blindspots for Attack Classification Models”, in: *2019 IEEE International Conference on Big Data (Big Data)*, IEEE, 2019, pp. 5773–5779
[35] M. T. Ribeiro, S. Singh, C. Guestrin, “Why Should I Trust You?: Explaining the Predictions of Any Classifier”, in: *22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 1135–1144
[36] S. Lundberg, S.I. Lee, “A Unified Approach to Interpreting Model Predictions”, *Advances in Neural Information Processing Systems 30* (2017) pp. 4765–4774
[37] S. Kaufman, S. Rosset, C. Perlich, “Leakage in Data Mining: Formulation, Detection, and Avoidance”, *ACM Transactions on Knowledge Discovery from Data 6.4* (2012) pp. 1–21
[38] A. Shrikumar, P. Greenside, A. Kundaje, “Learning Important Features through Propagating Activation Differences”, in: *International Conference on Machine Learning*, PMLR, 2017, pp. 3145–3153
[39] A. Datta, S. Sen, Y. Zick, “Algorithmic Transparency via Quantitative Input Influence: Theory and Experiments with Learning Systems”, in: *2016 IEEE Symposium on Security and Privacy*. IEEE, 2016, pp. 598–617
[40] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, W. Samek, “On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation”, *PloS one 10.7* (2015) art. e0130140
[41] S. Lipovetsky, M. Conklin, “Analysis of Regression in Game Theory Approach”, *Applied Stochastic Models in Business and Industry 17.4* (2001), pp. 319–330
[42] S. M. Lundberg, S.I. Lee, “Consistent Feature Attribution for Tree Ensembles”, *arXiv preprint arXiv:1706.06060*, 2017
[43] N. Moustafa, J. Slay, “UNSW-NB15: A Comprehensive Data Set for Network Intrusion Detection Systems (UNSW-NB15 Network Data Set)”, in: *2015 Military Communications and Information Systems Conference*, 2015, pp. 1–6

[44] R. C. Borges Hink, J. M. Beaver, M. A. Buckner, T. Morris, U. Adhikari, S. Pan, “Machine Learning for Power System Disturbance and Cyber-Attack Discrimination”, in: *2014 7th International Symposium on Resilient Control Systems*, 2014, pp. 1–8
[45] A. Cuzzocrea, F. Martinelli, F. Mercaldo, G. V. Vercelli, “TOR traffic analysis and detection via machine learning techniques”, in: *2017 IEEE International Conference on Big Data*, 2017, pp. 4474–4480
[46] P. P. F. Balbin, J. C. R. Barker, C. K. Leung, M. Tran, R. P. Wall, A. Cuzzocrea, “Predictive analytics on open big data for supporting smart transportation services”, in: *2020 KES International Conference*, 2020, pp. 3009–3018
[47] C. K. Leung, A. Cuzzocrea, J. J. Mai, D. Deng, F. Jiang, “Personalized DeepInf: Enhanced Social Influence Prediction with Deep Learning and Transfer Learning”, in: *2019 IEEE International Conference on Big Data*, 2019, pp. 2871–2880
[48] A. Coronato, A. Cuzzocrea, “An Innovative Risk Assessment Methodology for Medical Information Systems”, *IEEE Trans. Knowl. Data Eng. 34.7* (2022) pp. 3095–3110