Taxonomy and Survey of Interpretable Machine Learning Method

Saikat Das*, Namita Agarwal*, Deepak Venugopal*, Frederick T. Sheldon† and Sajjan Shiva*
*Department of Computer Science, University of Memphis, Memphis, TN, USA
†Department of Computer Science, University of Idaho, Coeur d'Alene, ID, USA
Emails: {sdas1, nfnu, dvngopal, sshiva}@memphis.edu, sheldon@uidaho.edu

***Abstract*—Since traditional machine learning (ML) techniques use black-box model, the internal operation of the classifier is unknown to human. Due to this black-box nature of the ML classifier, the trustworthiness of their predictions is sometimes questionable. Interpretable machine learning (IML) is a way of dissecting the ML classifiers to overcome this shortcoming and provide a more reasoned explanation of model predictions. In this paper, we explore several IML methods and their applications in various domains. Moreover, a detailed survey of IML methods along with identifying the essential building blocks of a black-box model is presented here. Herein, we have identified and described the requirements of IML methods and for completeness, a taxonomy of IML methods which classifies each into distinct groupings or sub-categories, is proposed. The goal, therefore, is to describe the state-of-the-art for IML methods and explain those in more concrete and understandable ways by providing better basis of knowledge for those building blocks and our associated requirements analysis.**

***Index Terms*—Interpretable machine learning; taxonomy; survey; black box machine learning; machine learning.**

I. Introduction

Machine learning (ML) techniques are being used extensively in various decision-making problems in healthcare, finance, agriculture, criminal justice [1], etc., while there are questions about their reliability due to their lack of transparency. In other words, experts working with classification problems have been using ML classifiers lavishly and blindly without knowing how a classifier functions internally, how a machine-learned response function works, how a model predicts, etc. As a result, the usages of these ML models become untrustworthy to the users/ experts due to the classifiers' black-box characteristics across multifarious domains. For example, relying on ML models blindly has had a significant number of issues in recent days, including some catastrophic failures, such as deadly accidents using self-driving car [2, 3], crime-fighting robot colliding with a child [4], Alexa playing uncensored audio instead of a kid's song [5], Amazon's face recognition that falsely matched 28 members of Congress with mugshots [6], complex artificial intelligence (AI) based stock trading software causing a trillion dollar flash crash [7], and so on.

Human can understand ML-based autonomous decisions and regulatory compliance, if the technique ensures transparency. Explainable Machine Learning or Explainable Artificial Intelligence (XAI) pertains post hoc analyses to understand a pretrained model and/ or its predictions. The analysis behind these techniques enhances understanding in various ways, by providing transparency, insights into the functional aspects of these algorithms, information on how they predict and how these functions operate, etc. In addition, trustworthiness is the primary concern of developing these techniques. Generally, trust in machine learning techniques grows from the tangible accuracy, transparency of the mechanism, and from its security. Explainable ML techniques enhance trust by ensuring fairness with transparency, stability, and dependability of ML algorithms to its users.

Interpretable Machine Learning (IML) explains an ML model in an understandable way and thus achieves trust. Interpretability is the degree of measurement on how much a human can understand the reason behind decisions (i.e.; predictions) made by ML models. The higher the interpretability of a machine learning model, the more understandable it is for human in terms of reasoning of a certain prediction. In other words, interpretability increases the trustworthiness of a model's functionality, whereas a single-metric evaluation, such as model's 'accuracy' is an incomplete description of understandability in most real-world tasks. The primary task of making an interpretable model is to create the model using white-box techniques. Generally, the complex model is often hard to interpret and explain. So, it is necessary to ensure the highest degree of human understanding for the high-stakes decision-making model. In these high-stakes cases, maximum transparency safeguards against fairness and security issues. In addition, interpretability comes at a negligible accuracy penalty for some newer white-box modeling methods, such as explainable neural network (XNN), scalable Bayesian rule lists, monotonic gradient boost machine (GBM), etc. Furthermore, interpretability helps debug an ML model and perform fairness auditing tasks easier [8].

The outline of this research contribution is as follows:

a) We discuss IML in detail with its importance in multiple research domains,
b) As building white-box model is the primary concern of making an ML model interpretable, we identify and discuss the requirements in building IML models,
c) We classify IML methods into several categories by outlining building blocks and then provide a taxonomy

978-1-7281-2547-3/20/$31.00

of IML methods,
d) We discuss recently used IML methods and provide a comparison of these methods using the building blocks.

The objective of this research is to identify the necessary requirements, building blocks of IML methods and to provide the state-of-the-art of interpretable machine learning methods. This survey not only helps understand IML technique but also unveils the future research scope on it.

The remainder of the article is organized as: in Section II, we briefly describe interpretable machine learning (IML). Section III provides the literature review that depicts the current state of the art. Requirements of building white-box models are identified and described in Section IV. In Section V, we present the building blocks and a taxonomy of IML methods. A comparison of several IML methods with respect to their building blocks is shown in Section VI. Finally in Section VII, we present some conclusions and future aspirations.

II. Interpretable Machine Learning (IML)

The interpretation of the ML model is a process in which individuals try to understand its predictions. Interpretability is the ability of understanding the decision-making policies of the machine-learned response function in order to explain the relationship between independent (input) and dependent (target) variables, ideally in a humanly interpretable manner. In many areas, interpretability cannot be sacrificed either because of legal requirements or because it leads to unfair decisions or because it is important for users. Interpretability has multiple benefits, such as:

a) helps users to extract interpretable patterns from trained ML models,
b) users can identify the reasons behind poor predictions,
c) helps users to increase trust in model predictions,
d) users become able to detect bias in ML models,
e) creates additional safety catch that can protect against overfitted models, etc.

Generally, ML algorithms (models) act as black boxes in which the algorithms' internal functionality is not known, and users have no idea of how the models make predictions. As the traditional models prevent users from having deep insight into the algorithms, people became doubtful and started raising most obvious questions about the predictions of the models, such as:

a) Why did the model make this decision and not the other one?
b) Why should I trust the model?
c) How do I correct an error?
d) When does the model succeed?
e) When does the model fail? and so on...

Interpretable machine learning has numerous applications across various domains. Currently, interpretability is the prime focus in commercialized ML solutions and products. As such, some recent commercialized IML solutions like H2O Driverless AI [9], DataRobot [10], etc. provide interpretability as base service. FICO, an analytics software company, which is famous for using AI in credit scoring, published a white paper "XAI Toolkit: Practical, Explainable Machine Learning" [11], and used IML methods in their applications. Several XAI platforms for government, financial services, healthcare, etc. are provided by Kyndi [12].

III. Literature Review

ML systems are increasingly used in complex, high-stakes systems. The need for interpretability of ML models have become undeniable due to some unexpected failure of traditional ML models; the most prominent of these are Google's facial recognition system, which branded some African-American people as gorillas [13], Uber's self-driving vehicle, which crossed a stop sign [3], and so on. In response to the failures of traditional ML, interpretable machine learning has been used with wide varieties of applications. For example in 2019, IBM provided a business AI platform product called IBM Watson AI Open Scale, where trust, transparency, and explainability are the highlighted features [14]. Google has long recognized the need for interpretability, and with their outstanding research, developed Google Vizier, a service for optimizing black box models. Besides, secure VM migration [15] in cloud computing can also be infected. In medical field, Mullenbach et al. [16] proposed a model based on the Convolutional Neural Network (CNN) to annotate automatic code to the Intensive Care Unit (ICU) text discharge summary. Moreover, they employed a per-label attention mechanism named as Convolutional Attention for Multi-Label classification (CAML), which allowed their models to learn distinct representations of documents for each label. Becker et al. [17] presented an interpretability procedure called Layer-wise Relevance Propagation (LRP) to explain the predictions for their Neural Network (NN) model on spoken digits and gender speakers using a novel audio dataset. In the realm of customer-oriented services, personalized recommendations are at the core of many online business, such as eCommerce, social media, content-sharing web portals, etc. The recommendation problem transformed into the matching problem that aims to measure the relevance score between users and items based on their available profiles. Therefore, to address this problem Wang et al. [18] combined the strengths of embedding based models and tree-based models on their proposed Tree-enhanced Embedding Method (TEM) that explains why those recommendations are made to the users by the system.

In the security domain, Marino et al. [19] provided an explanation approach for incorrect classification of data-driven Intrusion Detection Systems (IDS). Smith et al. [20] showed the Detection of Anomalous Activity in Real-Time (DAART) system for detecting anomalous activity without training data, and proposed interactive explanation methods for improved operator trust, and enhanced operator feedback through system transparency. Various feature selection process [21] can be benefited using interpretability of their models. Android malware is one of the major threats to mobile security. Therefore, Melis et al. [22] generalized current explainable android malware detection approaches to any black-box ML model

by leveraging a gradient-based approach to identify the most influential local features. Similarly, Drebin [23], an android malware detector, detects smartphone's malicious application using lightweight method.

IV. IML Requirements

The requirements for building interpretable machine learning models are summarized here [8, 24].

A. Degree of White-Box Modeling

Since IML techniques enhance the capability of transparency of traditional ML models, the best practice would be to increase the degree of white-box modeling (i.e.; maximize transparency). Maintaining a certain degree of white-box modeling helps human understand the model and using it for high-stakes decision-making problems where the maximum transparency safeguards against the fairness and security issues.

B. Data Visualization

Machine learning models, and therefore Interpretable machine learning models represent data. So, understanding data as well as its content is very important as it helps explain reasonable expectations for model behavior and the prediction. In practice, sometimes it is difficult to visualize and understand the datasets due to their complex design, enormous number of variables and instances. A graphical view or plot of higher dimension datasets are often helpful for human comprehensibility.

C. Model Visualization

Model visualization is required in IML techniques which provides graphical insights into the prediction behavior of ML models and helps debug the prediction mistakes (if any). In addition, model visualization helps envision the process flow of a complex model's decision-making process, model's local view and global view, the change of a model's predictions based on input variables, prediction errors while highlighting anomalies and outliers, potential interactions in ML models when multiple models are used combinedly, etc.

D. Variable Importance

Variable importance is the key aspect of explaining machine learning models. To calculate the variable importance, several methods exist where the methods determine the contribution of an input variable that helps a model to predict, either globally or locally. In terms of measuring global variable importance, averaging local measures into global measures are recently more established than the aggregation of local measures.

E. Fairness

Fairness is another required element in ML techniques when the model's outcome affects humans. It often refers to disparate impact analysis that includes assessing model predictions and errors across sensitive demographic segments of ethnicity, gender, etc. In ML arena, fairness helps remove biasness from the training data and from model predictions.

F. Sensitivity

Sensitivity analysis is the most important validation and debugging technique for ML models that maintains the acceptable model behavior and for its predictions when the data is intentionally perturbed or simulated by other means. In practice, numerical instability of regression parameters is being counted seriously on various linear model validation methods due to the correlation, which is measured between two input variables or between input and target variables. Moreover, while transforming linear modeling approach to ML approach, instability of model predictions is counted more seriously than that of parameters.

G. Residuality

Residuals for each data instances in a dataset can be measured by difference between stored dependent value and the predicted dependent value. Unless any random error, these residuals distribute randomly over a well-fitted model. Residuality is necessary in building IML models to make them well-fitted and adopted during any certain abnormal condition apart from errors.

V. Building Blocks of IML

IML methods come in varieties of forms in nature and are widely spread. As a result, sometimes it becomes difficult to classify those methods within a very short range. In this section, we show various IML methods with different classification families. In order to build the taxonomy of the IML methods, several building blocks are depicted in the subsequent sections.

A. Consistency

In the interpretable machine learning, consistency is the degree of measurement that determines the similarity of explanations that are provided by similar data instances for a certain type of model prediction. For an example, flu and COVID-19 have some similar symptoms like fever, cough, runny nose, headache, etc. Assuming we have two COVID-19 patients with similar symptoms, and they are detected as COVID-19 patients through ML models. An explanation is often needed after detecting a COVID-19 patient. In the above case, consistency measurement determines the similarities of two patients' symptoms and dissimilarities with flu patients. Therefore, consistency refers to the stability of a model which denotes during the change of a model's internal functionality, if simplified input's contribution increases or stays the same regardless of the other inputs; but the input's attribution should not decrease. Based on the consistency measurement, IML methods are divided into two classes:

1) Consistent: An IML model is treated as a consistent model if it generates consistent explanations for similar data instances. In other words, the explanation for a certain type of data instance is said consistent when the explanation is similar with same type of data and dissimilar with different type of data.

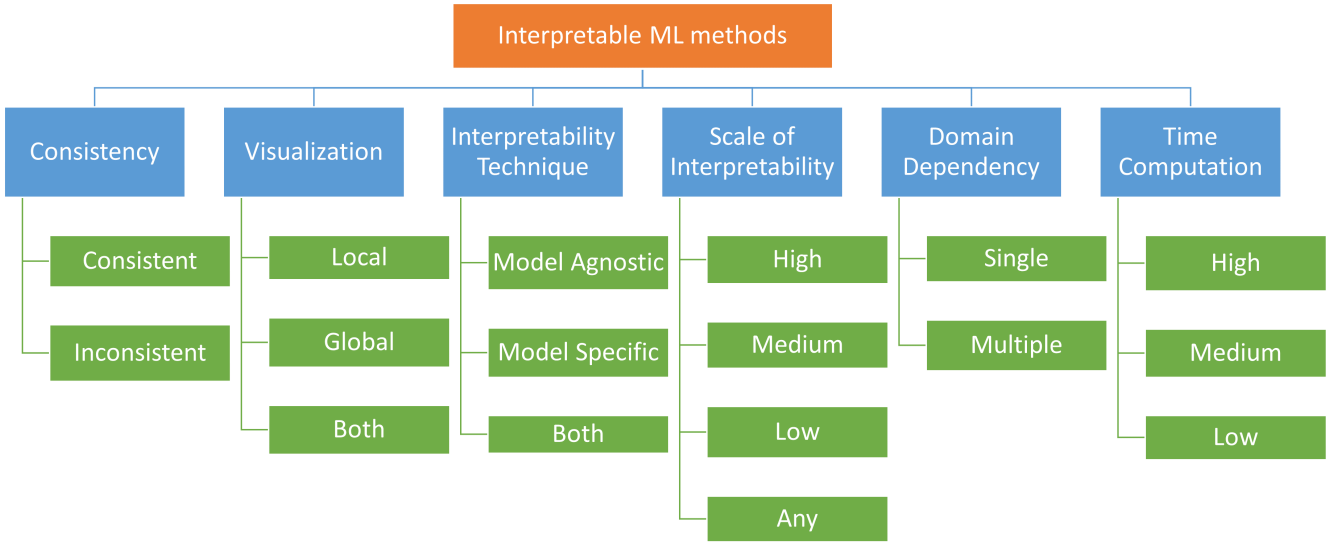

Fig. 1. Taxonomy of IML methods

2) Inconsistent: On the other hand, when the explanations for similar type of data instances are varied, the IML model that is used for these explanations is inconsistent no matter what happens with different types of data.

B. Visualization

As mentioned earlier, data visualization is the key requirement of IML models. Based on visualization, IML methods can be classified into two categories:

1) Global: Global view is the measurement of understanding the relationship between the input variables and the overall model based on the target variable. However, this type of interpretation is often highly presumptive.

2) Local: On the other hand, local visualization provides us with the interpretation of model itself or the prediction made by that model for single data instance or a group of similar instances. Local visualization is more accurate than global visualization, as the machine-learned response functions for a small group are seemed to be linear and monotonic.

Both local and global visualization techniques help interpret a model precisely and accurately with its marginal boundaries. The best interpretability analysis for an IML model can be performed by combining them.

C. Model Interpretability Techniques

Model interpretability technique is an important way to classify IML methods. There are two types of model interpretability techniques.

1) Model Agnostic: This technique applies when the IML methods are used in various types of ML algorithms i.e.; these IML methods can be used irrespectively in any ML model.

2) Model Specific: On the other hand, the IML methods that are designed to apply only for a single type or class of algorithm, are called model specific.

D. Scale of Interpretability

Scale of interpretability is the measurement of the degree of interpretability of certain model. Based on this, IML methods can be classified into four interpretable levels, such as high, medium, low, and any. The order high to low represents the deviation of interpretability i.e.; high interpretability means through these models, human can understand prediction more accurately while from low interpretability, it has less understandability for human. Interpretable level 'any' applies when models are interpretable with any of three levels (high, medium, and low) based on the requirement.

E. Domain Dependency

Domain dependency of IML refers to the models' interactivity with multiple research areas. IML methods can be applied on a single domain or multiple domains. Based on the count of interactions, IML methods can be classified into two:

1) Domain Dependent: Several IML methods are only applicable for a single domain i.e.; the functionality of some IML methods are solely related with certain domain, and their functionalities are not applicable to other domains.

2) Domain Independent: When the functionalities of IML methods are not domain dependent and can be applied over multiple domains, the methods are treated as domain independent methods.

F. Time Computation

Computational time is the major parameter for any algorithms including IML methods. In an IML method, computational time can be measured by the time taken to train the model and/or time taken for explaining a prediction. Based on the computational time, IML methods can be classified into three classes: high computational time, medium computational time, and low computational time methods. High, medium,

and low quantifiers are determined by the comparative measurement of time taken to train the model. For example, a regression model takes longer time to train than a linear model; on the other hand, a decision tree model requires lesser training time than linear and regression model.

VI. Comparison of IML Methods

This section first describes several IML methods with their applications or usages, and then compares them based on the taxonomy provided in Section V.

A. LIME

Local Interpretable Model-agnostic Explanation (LIME) method can generate sparse or simple explanations based on the local important variables. To explain the behavior of a complex machine-learned response function, LIME uses local linear parameters. It is helpful only for local interpretation with a medium complexity for explanation and is often used in pattern-recognition applications [25].

B. Anchors

Another well-known method proposed by the same researchers who introduced LIME method. By finding high-precision sets of rules in terms of input variables, this approach explains individual predictions using reinforcement learning techniques in combination with a graph search algorithm. It has the impact on data mining and pattern recognition domains. Anchor is a model agnostic method that creates local interpretation with medium complexity for explanation [26].

C. PDP

Partial Dependence Plot (PDP) shows the marginal effect of one or two features that contributes to the ML methods' prediction. PDP shows the relationship between the target and a feature which can be linear, monotonic or complex. PDP is a global model agnostic interpreter that explain any type of ML problem with lower computational time [27].

D. ICE

Individual Conditional Expectation (ICE) plots are newer and less adaptive than PDP where the plots show how a model behaves for single instance. ICE is a local model agnostic interpreter that can be used to verify monotonicity constraints with various complexity of explanations [28].

E. ALE

Accumulated Local Effect (ALE) is a plot that describes the influence of features in machine learning predictions. This plot is faster to compute (i.e.; O(n)) and an unbiased alternative to PDP. ALE is a model agnostic local interpretable technique that can be used to explain any type of ML problems [29].

F. SLIMs

Supersparse Linear Integer Models (SLIMs) are used to create predictive models using simple arithmetic operations like addition, subtraction, multiplication, etc. These models are used in high-stakes decision-making problems to explain the predictions. SLIM is a simple global interpretable model that is used for specific ML methods. It is often used to optimize medical scores [30].

G. GAMs

Generalized Additive Models are the alternative regression white-box modeling approaches that use contemporary methods to augment linear models. This type of explanation is easier to build for simple ML problems and can be used globally for specific methods with medium complexity in explanation [24].

H. GBM

Monotonic Gradient Bosting Machine can be used for white box modeling. Due to the monotonic constrains, sometimes it is difficult to interpret nonlinear nonmonotonic models and can be solved by enforcing a uniform splitting approach. GBM is a model specific technique with its global interpretability and has medium complexity in explaining predictions [24].

I. Treeinterpreter

Treeinterpreter decomposes tree type ML models, like decision tree, random forest, etc. into bias for each input variable. Treeinterpreter works locally, sometimes works globally, and has lower time complexity with specific type of model like tree-based model [31].

J. LOFO

LOFO stands for Leave One Feature Out. LOFO first creates local interpretations for each data instance and then for each features in the unlabeled dataset. During scoring, it leaves one variable out of the prediction by setting value to 'zero' or 'missing'. It is model agnostic and often used to build reason coding. LOFO is faster than Shepley in model training and data scoring. It can be used both locally and globally with various types of complexity for explanation [32].

K. CEM

Contrastive Explanations Method (CEM) is an explanation based on missing values, also known as pertinent negatives. To justify an explanation, CEM not only considers the positive or present elements but also finds contrastive perturbations that should be necessarily absent. It is a local model agnostic interpreter that worked with various domains and has a low implementation cost [33].

TABLE I
IML METHODS AND THEIR BUILDING BLOCKS [8, 24]

IML Models	Consistency	Visualization	Interpretability Technique	Scale of Interpretability	Domain Dependency	Time Computation
LIME	Consistent	Local	Model Agnostic	Medium	Independent	Medium
Anchors	Consistent	Local	Model Agnostic	Medium	Independent	Medium
PDP	?	Global	Model Agnostic	Any	Independent	Low
ICE	?	Local	Both	Medium	Independent	Medium
ALE	?	Local	Model Agnostic	Any	Dependent	Low
SLIMS	?	Global	Model Specific	Low	Dependent	High
GAMs	?	Global	Model Specific	Medium	Independent	Low
GBMs	?	Global	Model Specific	Medium	Independent	Medium
Treeinterpreter	?	Both	Model Specific	Any	Dependent	Low
LOFO	?	Both	Model Agnostic	Any	Dependent	Low
CEM	Consistent	Local	Model Agnostic	Medium	Independent	Medium
Residual Plot	?	Both	Model Agnostic	Any	Independent	Medium
Shapley Value	Consistent	Local	Both	Medium	Independent	High
SHAP	Consistent	Local	Both	Medium	Independent	Medium
Tree SHAP	Consistent	Local	Model Specific	Medium	Independent	Low
Kernel SHAP	Consistent	Local	Both	Medium	Independent	High
LRP	?	Local	Model Agnostic	Medium	Independent	Medium
XNN	?	Local	Model Specific	Medium	Independent	High
DeepLIFT	?	Global	Model Specific	Medium	Independent	High

Note: a question mark (?) indicates unknown cases.

L. Residual Plot

Residuals for each data instances in a dataset can be measured by difference between stored dependent value with the predicted dependent value. Unless any random error, these residuals distribute randomly over a well-fitted model. Residual Plots are model agnostic interpreter that use both local and global views and explain different ML methods. These plots are mostly used for debugging purposes [24].

M. Shapely Value

Shapley value interpretable model is constructed based on coalitional game theory where each of features for a certain data instances is considered as a player and the prediction is considered as playout. Explanation of a prediction formulated based on game theory. This interpretable method is used for any agnostic ML model with local visualization but high computation overhead [8].

N. SHAP

Shapley Additive exPlanations (SHAP) are different from shapely values in that the former ones are built with the credible support of both economics and game theory. These explanations unify approaches like LIME, LOFO, treeinterpreter, etc., and creates consistent and accurate global views with global variable importance measurement. SHAP explanations are time consuming to calculate and can be used for both model specific and model agnostic techniques [8, 34].

O. KernelSHAP

Kernel SHAP is the combination of Linear LIME and Shapley Values. The KernelExplainer builds a weighted linear regression by using the input variables, target variable, and machine-learned response function that generates the predicted values. The major limitation of the Kernel SHAP explanation is, it has longer running time for explanation. However, it has local visualization and can be used in both model agnostic and model specific technique [8].

P. TreeSHAP

TreeSHAP explainer is a variant of SHAP that uses tree-based machine learning models, like decision trees, random forests, and gradient boosted trees. TreeSHAP is a model-specific alternative to KernelSHAP which can produce unintuitive feature attributions. It has lower computation time than KernelSHAP and uses local visualization [8, 35].

Q. LRP

Layer-wise Relevance Propagation (LRP) is a Deep Neural Network interpretation technique to explain the predictions using layered network. LRP computes contribution scores and back propagates the scores across the layers from output variables to input variables. LRP has been used in various datatypes like images, text, EKG signals, audio, etc. and with various neural architectures (ConvNets, LSTMs) [36].

R. XNN

eXplanable Neural Network (XNN) is the model-specific explainer for artificial neural network (ANN) to interpret and explain ANN accurately. XNN extracts features from the fully connected neural networks and provides a cost-effective explanation of the relationship between the input variables and the target variable. It is often used for pattern recognition, fraud detection, credit scoring, etc., and can be interpretable with both global and local views [24, 37].

S. DeepLIFT

DeepLIFT (Deep Learning Important FeaTures) explanation method decomposes the prediction of an ANN on a specific input by backpropagating the contributions of all neurons in the ANN to every input variable (feature) [38].

The goal of all the above methods is to interpret ML predictions so that human can understand the process flow, insights of the response function, and the details on how a decision is made based on feature importance and variances. Depending on the problems, like monotonic, nonmonotonic, linear, regression, neural network, etc., IML methods work with certain problem domain from various angles, like visualization, degree of interpretability, time to explain, consistency, interpretable techniques, etc.

VII. Conclusion

Humans cannot trust the Machine learning (ML) based decisions completely in high-stakes applications such as COVID-19 testing, fraud detection, loan sanctions, credit scoring, etc., as the traditional ML models lack in transparency due to their black-box nature. Interpretable machine learning (IML) technique is the way of explaining ML predictions which helps human to understand the internal functionalities of the functioning ML model. To ensure the transparency of these traditional ML models, it is necessary to know the requirements of building a white-box ML model. In this paper, we have identified the key requirements in building IML models. The importance of interpretability on various domains is also mentioned here. Moreover, we conducted a detailed survey on IML methods across multiple domains to produce the state-of-the-art of interpretable machine learning. Apart from these, a taxonomy of IML methods to classify them into several subsections is also proposed. The taxonomy that we have introduced will help the future researchers to dive deeper into IML methods.

We plan to enrich our existing research on network monitoring [39] and ensemble supervised [40] and unsupervised frameworks [41] by including interpretability. In addition, we plan to derive metrics for the consistency of IML methods to add extra confidence of their predictions.

References

[1] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nature Machine Intelligence*, vol. 1, no. 5, pp. 206–215, 2019.

[2] TheGuardian, "Tesla driver killed while using autopilot was watching Harry Potter." https://www.theguardian.com/technology/2016/jul/01/tesla-driver-killed-autopilot-self-driving-car-harry-potter/. [Online; accessed 10-August-2020].

[3] Reuters, "Self-driving Uber car kills Arizona woman crossing street." https://reut.rs/2GJi24F/. [Online; accessed 10-August-2020].

[4] Los Angeles Times, "Crime-fighting robot hits, rolls over child at Silicon Valley mall." https://www.latimes.com/local/lanow/la-me-ln-crimefighting-robot-hurts-child-bay-area-20160713-snap-story.html/. [Online; accessed 10-August-2020].

[5] Entrepreneur, "Whoops, Alexa Plays Porn Instead of a Kids Song!." https://www.entrepreneur.com/video/287281/. [Online; accessed 10-August-2020].

[6] ACLU, "Amazon's Face Recognition Falsely Matched 28 Members of Congress With Mugshots." https://www.aclu.org/blog/privacy-technology/surveillance-technologies/amazons-face-recognition-falsely-matched-28/. [Online; accessed 10-August-2020].

[7] Wikipedia, "2010 flash crash." https://en.wikipedia.org/wiki/2010_flash_crash/. [Online; accessed 10-August-2020].

[8] C. Molnar, "Interpretable machine learning: A guide for making black box models explainable.(2019)," *URL https://christophm. github. io/interpretable-ml-book*, 2019.

[9] H20.ai, "Driverless AI." https://www.h2o.ai/products/h2o-driverless-ai/. [Online; accessed 10-August-2020].

[10] DataRobot, "Model Interpretability." https://www.datarobot.com/wiki/interpretability/. [Online; accessed 10-August-2020].

[11] A. Flint, A. Nourian, and J. Koister, "xai toolkit: Practical, explainable machine learning," *White paper*, 2018.

[12] KYNDI, "Products." https://kyndi.com/products/. [Online; accessed 10-August-2020].

[13] Forbes, "Google Photos Tags Two African-Americans As Gorillas Through Facial Recognition Software." https://www.forbes.com/sites/mzhang/2015/07/01/google-photos-tags-two-african-americans-as-gorillas-through-facial-recognition-softw [Online; accessed 10-August-2020].

[14] IBL, "Watson." https://www.ibm.com/watson/. [Online; accessed 10-August-2020].

[15] S. Das, A. M. Mahfouz, and S. Shiva, "A stealth migration approach to moving target defense in cloud computing," in *Proceedings of the Future Technologies Conference*, pp. 394–410, Springer, 2019.

[16] J. Mullenbach, S. Wiegreffe, J. Duke, J. Sun, and J. Eisenstein, "Explainable prediction of medical codes from clinical text," *arXiv preprint arXiv:1802.05695*, 2018.

[17] S. Becker, M. Ackermann, S. Lapuschkin, K.-R. Müller, and W. Samek, "Interpreting and explaining deep neural networks for classification of audio signals," *arXiv preprint arXiv:1807.03418*, 2018.

[18] X. Wang, X. He, F. Feng, L. Nie, and T.-S. Chua, "Tem: Tree-enhanced embedding model for explainable recommendation," in *Proceedings of the 2018 World Wide Web Conference*, pp. 1543–1552, 2018.

[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*, pp. 3237–3243, IEEE, 2018.

[20] A. Smith-Renner, R. Rua, and M. Colony, "Towards an explainable threat detection tool.," in *IUI Workshops*, 2019.

[21] S. Das, D. Venugopal, S. Shiva, and F. T. Sheldon, "Empirical evaluation of the ensemble framework for feature selection in ddos attack," in *2020 7th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2020 6th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom)*, pp. 56–61, IEEE, 2020.

[22] M. Melis, D. Maiorca, B. Biggio, G. Giacinto, and F. Roli, "Explaining black-box android malware detection," in *2018 26th European Signal Processing Conference (EUSIPCO)*, pp. 524–528, IEEE, 2018.

[23] D. Arp, M. Spreitzenbarth, M. Hubner, H. Gascon, K. Rieck, and C. Siemens, "Drebin: Effective and explainable detection of android malware in your pocket.," in *Ndss*, vol. 14, pp. 23–26, 2014.

[24] P. Hall, *An introduction to machine learning interpretability*. O'Reilly Media, Incorporated, 2019.

[25] M. T. Ribeiro, S. Singh, and C. Guestrin, "" why should i trust you?" explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144, 2016.

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

[27] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *Annals of statistics*, pp. 1189–1232, 2001.

[28] A. Goldstein, A. Kapelner, J. Bleich, and E. Pitkin, "Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation," *Journal of Computational and Graphical Statistics*, vol. 24, no. 1, pp. 44–65, 2015.

[29] D. W. Apley and J. Zhu, "Visualizing the effects of predictor variables in black box supervised learning models," *arXiv preprint arXiv:1612.08468*, 2016.

[30] B. Ustun and C. Rudin, "Supersparse linear integer models for optimized medical scoring systems," *Machine Learning*, vol. 102, no. 3, pp. 349–391, 2016.

[31] Datadrive, "Random forest interpretation with scikit-learn." https://blog.datadive.net/random-forest-interpretation-with-scikit-learn//. [Online; accessed 10-August-2020].

[32] J. Lei, M. G'Sell, A. Rinaldo, R. J. Tibshirani, and L. Wasserman, "Distribution-free predictive inference for regression," *Journal of the American Statistical Association*, vol. 113, no. 523, pp. 1094–1111, 2018.

[33] A. Dhurandhar, P.-Y. Chen, R. Luss, C.-C. Tu, P. Ting, K. Shanmugam, and P. Das, "Explanations based on the missing: Towards contrastive explanations with pertinent negatives," in *Advances in Neural Information Processing Systems*, pp. 592–603, 2018.

[34] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in neural information processing systems*, pp. 4765–4774, 2017.

[35] S. M. Lundberg, G. G. Erion, and S.-I. Lee, "Consistent individualized feature attribution for tree ensembles," *arXiv preprint arXiv:1802.03888*, 2018.

[36] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek, "On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation," *PloS one*, vol. 10, no. 7, p. e0130140, 2015.

[37] J. Vaughan, A. Sudjianto, E. Brahimi, J. Chen, and V. N. Nair, "Explainable neural networks based on additive index models," *arXiv preprint arXiv:1806.01933*, 2018.

[38] A. Shrikumar, P. Greenside, and A. Kundaje, "Learning important features through propagating activation differences," *arXiv preprint arXiv:1704.02685*, 2017.

[39] S. Das and S. Shiva, "Corum: collaborative runtime monitor framework for application security," in *2018 IEEE/ACM International Conference on Utility and Cloud Computing Companion (UCC Companion)*, pp. 201–206, IEEE, 2018.

[40] S. Das, A. M. Mahfouz, D. Venugopal, and S. Shiva, "Ddos intrusion detection through machine learning ensemble," in *2019 IEEE 19th International Conference on Software Quality, Reliability and Security Companion (QRS-C)*, pp. 471–477, IEEE, 2019.

[41] S. Das, D. Venugopal, and S. Shiva, "A holistic approach for detecting ddos attacks by using ensemble unsupervised machine learning," in *Future of Information and Communication Conference*, pp. 721–738, Springer, 2020.