EXPLAINING ARTIFICIAL NEURAL NETWORKS WITH DECISION TREE ENSEMBLES

YAPAY SİNİR AĞLARININ KARAR AĞACI TOPLULUKLARI İLE AÇIKLANMASI

SAYİT KILIÇ

ASSOC. PROF. DR. BURKAY GENÇ
Supervisor

Submitted to
Graduate School of Science and Engineering of Hacettepe University
as a Partial Fulfillment to the Requirements
for the Award of the Degree of Master of Science
in Computer Engineering

April 2023

ABSTRACT

EXPLAINING ARTIFICIAL NEURAL NETWORKS WITH DECISION TREE ENSEMBLES

Sayit Kılıç
Master of Science, Computer Engineering
Supervisor: Assoc. Prof. Dr. Burkay GENÇ
April 2023, 73 pages

With the development of efficient algorithms, artificial intelligence (AI) applications have become ubiquitous in almost every aspect of our lives. They have even started to be used in critical areas such as defense industry, economy, and healthcare. However, the use of AI models in these important areas raises concerns about their reliability. Therefore, explaining how these black box models work has become an important goal. This thesis, we propose a simple and fast model to explain the decisions of any black box model. To achieve this, we attempt to explain the basic behavior of the model through a set of semi-random decision trees. Our approach only requires the data used to train the black box model and the model itself to work. Current state-of-the-art explainable AI (XAI) models typically produce local explanations for a black box model's decision regarding a single observation. On the other hand, models that produce global explanations use complex computations to understand the effect of each feature on the model's decisions. However, our proposed approach defines separate regions in the model's general decision space to explain the decision-making process of the model, and requires significantly less computational power than other advanced XAI techniques while producing both local and global explanations.

Keywords: Explainable Artificial Intelligence, Interpretability, Ensemble Model, TEXAI

ÖZET

YAPAY SİNİR AĞLARININ KARAR AĞACI TOPLULUKLARI İLE AÇIKLANMASI

Sayit Kılıç
Yüksek Lisans, Bilgisayar Mühendisliği
Danışman: Assoc. Prof. Dr. Burkay GENÇ
March 2023, 73 sayfa

Verimli algoritmaların gelişmesiyle yapay zeka uygulamaları hayatımızın neredeyse her alanında kullanılır hale geldi. Savunma sanayi, ekonomi ve sağlık gibi insan hayatı için çok önemli konularda bile kullanılmaya başlandı. Bu önemli konularda yapay zeka modellerinin kullanımı, bu modellerin güvenilirliği hakkında soru işaretlerine neden olmaktadır. Bu nedenle, bu siyah kutu modellerinin nasıl çalıştığını açıklayabilme önemli bir hedef haline gelmiştir. Bu tezde, herhangi bir kara kutu modelin kararlarını açıklamak için basit ve hızlı bir model öneriyoruz. Bunu yapmak için, modelin temel davranışını yarı rastgele karar ağaçlarının bir kümesi aracılığıyla açıklamaya çalışıyoruz. Yaklaşımımızın çalışması için sadece kara kutu modeli eğitmek için kullanılan verilerine ve modelin kendisine ihtiyaç duymaktayız. Mevcut son teknoloji açıklanabilir yapay zeka modelleri genellikle siyah kutu modellerin tek bir gözlem hakkındaki kararı için yerel açıklamalar üretmektedir. Öte yandan, genel açıklamalar üreten modeller, her özniteliğin modelin kararları üzerindeki etkisini anlamak için karmaşık hesaplamalar kullanır. Ancak, önerdiğimiz yaklaşım, modelin karar verme sürecini açıklamak için modelin genel karar uzayında ayrı ayrı bölgeler tanımlar ve hem yerel hem de genel açıklamalar üretirken diğer ileri XAI tekniklerine göre önemli ölçüde daha az hesaplama gücü gerektirir.

Keywords: Açıklanabilir Yapay Zeka, Açıklanabilirlik, Topluluk Modeli, TEXAI

ACKNOWLEDGEMENTS

To our noble martyrs who have filled every page of the glorious history of Türkiye with epic heroism and ensured us to live in peace and security on this sacred land under the crescent and star flag,
To my valuable advisor **Assoc. Prof. Dr. Burkay GENÇ**, who has enlightened my path with his knowledge and experience and taught me how to learn throughout this process,
To my mother **Hacer KILIÇ**, my father **Adem KILIÇ** and my brother **İsmail KILIÇ**, who have made all kinds of sacrifices for me to come this far,
To the **MORALABS** family, starting with my valuable employers **Ms. Kamer KEMERKAYA** and **Mr. Kemal ŞEN**, who provide every kind of support for me to continue my education alongside my professional life,
To **Ms. Gözde ORHAN**, who has supported me with her knowledge and experience during my master's degree,
To **Mr. Ertunç EFE**, **Mr. Mehmed Derviş BARUTÇU**, **Ms. Zümrüt UYSAL**, and **Ms. Elif ÜSKUP**, who have guided me and never withheld their assistance during my working life,
To my dear friend **Mr. Muhammed GÜL**, who has shown me that blood ties are not necessary for brotherhood,
To my dear friend and the greatest advisor on my life, **Atty. Cemre Gözde ATİK**, who has always been by my side in difficult times,
And to my precious wife **Gözde KILIÇ**, who has taught me how to love and has been my biggest supporter in all my decisions and my companion on this life journey.

I offer my endless respect and gratitude...

CONTENTS

Page

TABLES

FIGURES

ABBREVIATIONS

TEXAI	:	**T**ree **E**nsemble for e**X**plainable **A**rtificial **I**ntelligence
ANN	:	**A**rtificial **N**eural **N**etworks
ML	:	**M**achine **L**earning
EM	:	**E**nsemble **M**odel
AI	:	**A**rtificial **I**ntelligence
XAI	:	e**X**plainable **A**rtificial **I**ntelligence
LIME	:	**L**ocal **I**nterpretable **M**odel-Agnostic **E**xplainations
SHAP	:	**SH**apley **A**dditive ex**P**lainations

1. INTRODUCTION

Artificial intelligence (AI) has become an increasingly essential part of our daily routines, from voice assistants to fraud detection systems in our banks. However, AI is often viewed as a black box, making it difficult for humans to understand the decision-making process behind AI-generated outcomes [2, 3]. The absence of transparency regarding the implementation of AI in crucial domains like healthcare and finance can result in doubt and suspicion towards its use. To address this issue, the domain of explainable artificial intelligence (XAI) has emerged, which aims to develop models and methods that can be understood and interpreted by humans [4].

XAI is a multidisciplinary research field that seeks to develop algorithms and models capable of providing clear and understandable explanations for decision-making processes. With the rapid expansion of AI usage across a range of domains, including healthcare, finance, and security, the demand for accountable and trustworthy AI systems has grown considerably.

The significance of XAI stems from the fact that many modern AI systems function as "black boxes" meaning their decision-making processes are opaque to human understanding. This poses a critical challenge in situations where these systems make decisions with real-world consequences, such as medical diagnosis, financial trading, or autonomous vehicles. In these cases, it is essential to have a clear grasp of the reasoning behind an AI system's outputs, particularly when they have an impact on human welfare.

XAI draws on expertise from various fields, including computer science, cognitive science, philosophy, and others, to develop techniques and approaches that enable AI systems to provide transparent and understandable explanations of their process of making a decision. These approaches include methods from machine learning, natural language processing, and visualization, among others, with the aim of creating more transparent and interpretable AI systems.

In summary, the ultimate goal of XAI is to improve the transparency, interpretability, and accountability of AI systems, especially in contexts where the consequences of their decisions are significant. By leveraging interdisciplinary expertise and developing techniques and approaches that enable AI systems to provide transparent and interpretable explanations, XAI promotes trust and confidence in AI systems, mitigates unintended consequences or errors, and paves the way for the responsible and ethical deployment of AI in society.

1.1. Motivation

Explainable Artificial Intelligence (XAI) not only enhances the credibility of artificial intelligence models, but also offers several advantages that drive our motivation in this thesis. XAI can potentially increase the accuracy of Artificial Intelligence (AI) models in several ways. One way is by allowing humans to better understand and correct errors in the AI model's decision-making process. XAI models provide human-interpretable explanations of the AI model's decisions, allowing users to identify and correct any errors or biases in the model's logic. In addition, XAI models can enable users to fine-tune the AI model's parameters and features to better fit the specific problem or domain, which can lead to improved accuracy. For example, a study on medical diagnosis found that an XAI model that allowed clinicians to adjust the importance of different features achieved higher accuracy than a traditional AI model [5]. Moreover, XAI can help detect and prevent errors or biases in the training data used to develop the AI model. By providing explanations of the AI model's decisions, XAI can reveal any underlying biases or inaccuracies in the data and allow users to address them before they affect the model's performance.

The comprehension of an AI model's decision-making process can provide individuals with valuable insights into the problem domain being addressed. XAI can aid medical professionals and researchers in the medical field by enabling them to comprehend how AI models diagnose diseases or prescribe treatments. Through the provision of natural language explanations of the features or variables that the model employs to make decisions, XAI can assist medical professionals in identifying patterns or relationships that may not be immediately evident from the data. Similarly, in the financial domain, XAI can facilitate analysts in understanding how AI models predict stock prices or recognize fraudulent transactions. By visualizing the connections between various variables and emphasizing the factors that significantly influence the model's decision-making process, XAI can enable analysts to acquire insights into the market trends or patterns detected by the model.

As AI technologies continue to be integrated into important domains such as defence industry, finance, and criminal justice, ethical values such as fairness, transparency, accountability, and privacy are becoming increasingly paramount in AI development. One of the major ethical worries regarding AI systems is that they may entrench or maintain existing biases and discrimination. This can occur if the training data of AI model is biased or if the algorithms used to make decisions are designed with bias. Another important ethical consideration in AI is the potential for these systems to cause harm to individuals or society as a whole. This can occur if AI systems are used inappropriately or if they make decisions that have unintended consequences.

The assurance of ethical alignment in the training of AI models is crucial in order to avoid unintentional negative consequences and promote responsible AI development. XAI plays a vital role in achieving this objective by providing developers and users with the ability to comprehend AI model decisions. This transparency assists in identifying potential biases or unfairness in the data or algorithms employed to train the models and implementing corrective actions to align with ethical values.

AI systems must be designed to ensure that they do not perpetuate or amplify existing biases or discrimination and must uphold basic human rights and values. XAI can play a critical role in ensuring ethical alignment in AI systems. By providing transparent and interpretable explanations for their decision-making processes, AI systems can be made more accountable and can be designed in a way that is consistent with ethical values.

One important example of unethical behavior by an AI model is the case of a hiring algorithm developed by Amazon in 2014. The algorithm was designed to evaluate job candidates' resumes and assign them a score based on their qualifications. However, it turned out that the system was biased towards female candidates. The system's bias against female candidates was that they were trained over a ten-year period with resumes sent to Amazon, with the majority of them male candidates. As a result, the algorithm gave lower scores to resumes that contained words associated with women, such as “women’s,” “female,” and “she.” Amazon eventually scrapped the algorithm after the bias was discovered, and this case highlights the potential of AI models to maintain and prolong exist discrimination if not trained with care and ethical considerations [6, 7]. Therefore, the development of AI models featuring explainability and transparency represents a positive stride in building trust and confidence in these technologies and promoting responsible and ethical development.

1.2. Problem Definition

Despite significant advances in the research on eXplainable Artificial Intelligence (XAI), contemporary models have certain limitations and drawbacks. LIME (Local Interpretable Model-Agnostic Explanations) [8] and SHAP (SHapley Additive exPlanations) [9] are the most widely well-known models in this area.

LIME is an approach for explaining black-box machine learning model predictions at a local level by generating a simplified “local surrogate model” that imitates the behavior of the black box model. To create the surrogate model, LIME perturbs the feature values of the instance and uses the black-box model to predict the labels of these instances, then trains the surrogate model based on these predictions. The surrogate model’s feature weights serve as feature importances, indicating the impact of each feature to the original model’s decision. However, the drawback of LIME is that it may not always provide a faithful explanation of the model’s behavior. LIME approximates the global behavior of the original model by building a local surrogate model, which may not capture all the nuances of the original model’s behavior. The choice of kernel function used in LIME may also impact the resulting explanations, and determining the appropriate kernel function for a given problem may be challenging.

SHAP is an algorithmic approach that explains the contribution or importance of each feature for a specific prediction made by an AI system. It uses shapley values to distribute credit or blame fairly among the features. The method assigns an relevance point to each feature based on its impact to the output. However, SHAP can be computationally expensive and time-consuming to calculate for large datasets or complex models because it requires calculating the marginal contribution of each feature to the prediction for every possible combination of features, which is computationally intensive.

Another potential issue with both LIME and SHAP is that they may not provide a comprehensive picture of the model's behavior. Both methods focus on explaining the impact of individual features to the AI system predictions but may not capture complex interactions between features or the overall structure of the model. Furthermore, LIME and SHAP explanations may be specific to a particular input example and may not generalize well to other examples.

In the "Tree Ensemble for Explainable Artificial Intelligence(TEXAI)" model we presented, all the structural behaviors of the black box model are uncovered without requiring any additional data beyond the model's training data. This is achieved by analyzing the model's behavior on observations instead of computing complex feature importance scores. In this way, we introduce a novel XAI approach that is simple, rapid, and resource-efficient, with minimal computational requirements.

1.3. Organization

The organization of the thesis is as follows:

- Chapter 1 presents problem definition, our motivation, contributions and the scope of the thesis.
- Chapter 2 provides an overview of the overall scope and methodology of the literature.
- Chapter 3 gives a comprehensive information about state of the art XAI models.
- Chapter 4 introduces our TEXAI model.
- Chapter 5 demonstrates the metric results of our TEXAI Model.

- Chapter 6 provides a summary of the thesis and outlines potential future directions.

2. BACKGROUND OVERVIEW

2.1. Artificial Intelligence

The term Artificial Intelligence (AI) pertains to the creation of computer systems that have the ability to carry out functions which would usually necessitate human intelligence, like recognizing visual inputs, comprehending speech, making decisions, and translating languages. According to Russell and Norvig, AI can be defined as “the study of agents that receive percepts from the environment and take actions that affect that environment” [10] as shown in figure 2.1. AI has been around for over sixty years and has made remarkable progress recently due to greater accessibility to data and the advent of more advanced computing systems.

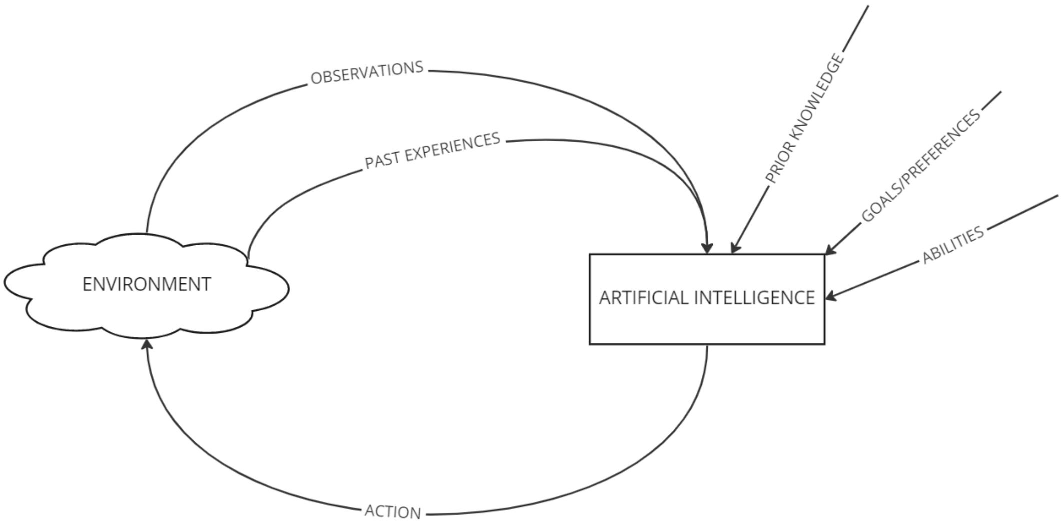


Figure 2.1 Artificial Intelligence

There are many different AI methods, and they can be classified in different ways based on model type, problem-solving approach, learning type, and other criteria. For the purpose of our discourse, we shall limit our discussion to the types of models that are pertinent to our thesis or those that are prevalent in the field of AI.

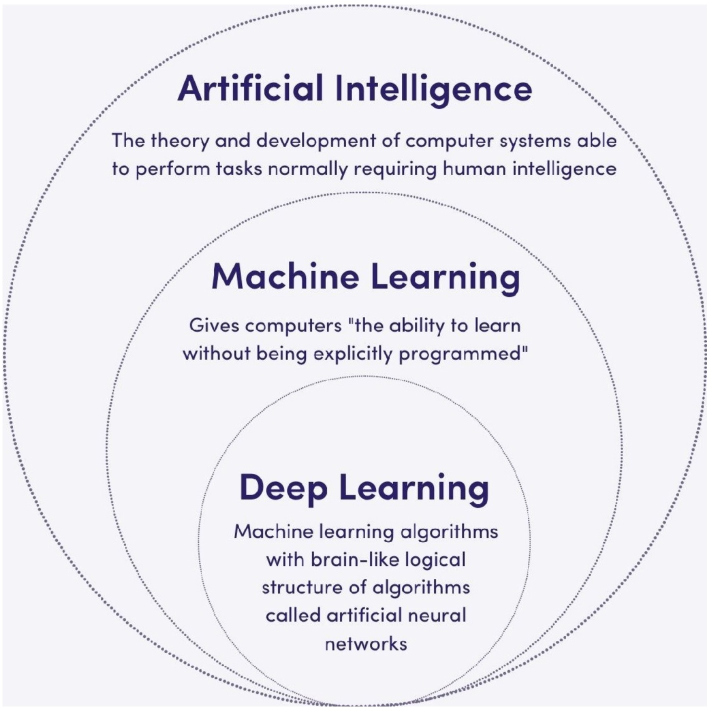

Artificial Intelligence
The theory and development of computer systems able to perform tasks normally requiring human intelligence
Machine Learning
Gives computers "the ability to learn without being explicitly programmed"
Deep Learning
Machine learning algorithms with brain-like logical structure of algorithms called artificial neural networks

Figure 2.2 Relations of AI, machine learning, and deep learning. [1]

2.1.1. Machine Learning

Machine learning (ML) is a branch of artificial intelligence (AI) that enables computer systems to learn and enhance their performance based on experience, without requiring explicit programming [11]. According to Tom Mitchell, a renowned computer scientist, machine learning is the study of algorithms that enable computer programs to automatically improve through experience [11].

Supervised learning, unsupervised learning, and reinforcement learning are the three main types of machine learning. In supervised learning, a dataset with known input and output pairs is used to train the computer, which then learns to predict the output for new inputs. In contrast, unsupervised learning involves the computer being given an unlabeled dataset and tasked with discovering patterns and structures within the data. Reinforcement learning is another approach in which an agent learns how to act within an environment by taking actions and receiving penalties or rewards according to those actions.

[1] Adapted from “AI vs Machine Learning vs Deep Learning” by Emerj. Available at: `https://emerj.com/ai-vs-machine-learning-vs-deep-learning-whats-the-difference/`

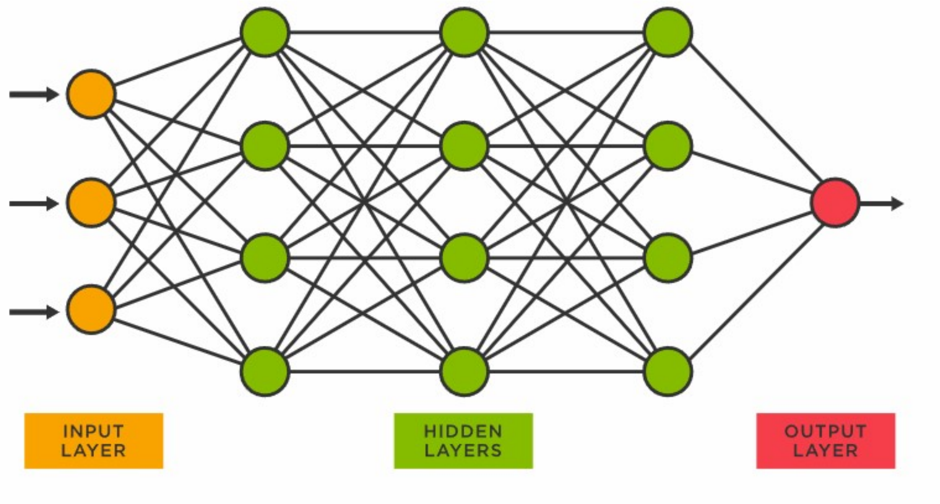


Figure 2.3 An artificial neural network diagram. [2]

Machine learning is used in various fields such as natural language processing, speech recognition, autonomous vehicles etc. Different types of algorithms are used in machine learning, including decision trees, k-nearest neighbors, support vector machines, neural networks, and deep learning models [12, 13].

2.1.2. Artificial Neural Networks

Artificial Neural Networks (ANNs) are a branch of Artificial Intelligence (AI) that has gained a lot of interest in recent times. ANNs can learn from input data and make decisions according to that learning, by taking inspiration from the organization of the biological neural networks found in the human brain.

According to Haykin, an ANN is a “massively parallel distributed processor that has a natural propensity for storing experiential knowledge and making it available for use” [14]. Artificial Neural Networks (ANNs) consist of numerous interconnected processing nodes, called neurons, that receive inputs and generate outputs. as shown in figure 2.3. The connections between the neurons are weighted, which allows the network to learn from data by adjusting the weights to improve its performance on a given task.

[2] Adapted from TIBCO Software Inc., 2021). Available at : `https://www.tibco.com/sites/tibco/files/media_entity/2021-05/neutral-network-diagram.svg`

Artificial Neural Networks (ANNs) have been effectively implemented in various fields such as image recognition, natural language processing, and prediction tasks [15]. ANNs are particularly useful in learning intricate patterns and relationships in data, which makes them suitable for tasks that cannot be modeled using conventional methods. This attribute of ANNs is one of their significant advantages.

Artificial Neural Networks (ANNs) can be categorized based on various factors such as their architecture, learning algorithm, and activation function. Several common types of ANNs exist, including feedforward propogated networks, convolutional neural networks, among others [16].

Despite their success, ANNs are not without their challenges and limitations, including the need for large amounts of training data, overfitting risk, and the difficulty of interpreting the internal workings of the network [17]. Nonetheless, despite the challenges and limitations, ANNs are still considered a potent approach for addressing various AI problems. As such, their ongoing improvement and advancement are expected to have a significant impact on the development of the field in the future.

2.1.3. Decision Trees

Decision Tree is a versatile algorithm that can be used for both classification and regression tasks. It has been applied in a wide range of fields and industries, including healthcare, finance, marketing, engineering, and many more. [18]. The Decision Tree algorithm is a robust and uncomplicated method that builds a tree-like structure by dividing the data recursively according to the feature values that offer the highest information about the target variable or reduce the variance in it.

Both classification and regression tasks can be performed by decision trees. In classification task, the goal is to assign a label to a given instance according to its features. In regression, the goal is to predict a numerical value for a given instance based on its features. The construction of the decision tree algorithm involves recursively dividing the data using feature values that enhance the distinction between classes or reduce the variance in the target variable.

Decision Trees have several advantages over other machine learning algorithms. First, they are easy to understand and interpret, as they represent a set of decision rules that can be visualized as a tree. This makes them useful for explaining the reasoning behind the model's predictions and gaining insights into the data. Second, they are capable of handling both categorical and continuous data, as well as NA values and noisy data, without the need for data normalization or transformation. Lastly, they can capture complex interactions between features, such as nonlinear and hierarchical relationships, by constructing decision rules based on multiple features.

In conclusion, Decision Tree is a powerful and versatile algorithm in artificial intelligence that can be helpful for various fields like classification, regression, and feature selection. Decision Trees have several advantages such as interpretability, flexibility, and robustness which make them a popular choice for individuals working in the field of artificial intelligence.

2.1.4. Decision Tree Ensembles

Ensemble methods refer to the use of multiple machine learning models to improve the overall performance of a predictive task, as compared to using a single model. Decision tree ensembles, are a popular type of ensemble method that use multiple decision trees to create a stronger model. Two primary types of tree ensembles are "bagging" and "boosting".

Bagging, also known as bootstrap aggregating, entails building multiple decision trees using randomly selected subsets of the training data, and then combining their predictions, typically through averaging, to obtain the final prediction. This technique was introduced by Breiman in 1996 [19], and studies have shown that utilizing bagging can improve the accuracy and robustness of decision tree models. [20].

Boosting, on the other hand, involves iteratively constructing decision trees that focus on the data points that were misclassified by previous trees. The most well-known algorithm for boosting decision trees is AdaBoost, which was proposed by Freund and Schapire in 1997 [21]. AdaBoost has been shown to increase the performance of decision tree models on a variety of datasets [22].

Random Forest, Gradient Boosting, and XGBoost are some of the most common tree ensembles. Random Forest is a machine learning technique that involves creating numerous decision trees on different subsets of the training data and consolidating their predictions to obtain a final prediction [23]. The Gradient Boosting algorithm constructs decision trees in a sequential manner, where each subsequent tree aims to rectify the mistakes made by the previous tree [24]. XGBoost is known for its speed and scalability, and uses a regularized gradient boosting algorithm to build a strong model [25].

Ensemble methods using decision trees have emerged as a popular and effective machine learning technique and using in diverse applications, such as speech recognition, bioinformatics, finance, and many others thanks to improve predictive accuracy and reduce overfitting [26].

2.2. Explainable Artificial Intelligence

Explainable artificial intelligence (XAI) is a specialized domain within AI that emphasizes the creation of AI systems that are transparent, easily comprehensible, and capable of being explained to people. XAI aims to provide users with a better understanding of how AI systems make decisions by revealing the internal workings of the AI models. XAI has been defined by the Defense Advanced Research Projects Agency(DARPA) as “the ability to understand, appropriately trust, and effectively use AI systems.” [27, 28]. XAI research employs various techniques, including model interpretation, visualization, and explanation, to make the AI models decision process more transparent and interpretable.

The origins of the requirement for XAI can be traced back to the initial stages of AI research. In the 1960s and 1970s, AI researchers were primarily focused on developing rule-based systems, which were designed to mimic human reasoning and decision-making processes. These systems were inherently interpretable, as the rules used to make decisions were explicitly defined and transparent to the user. As the AI field advanced, researchers started to concentrate on more intricate and potent machine learning models, like neural networks. Even though these models were proficient in accomplishing remarkable performance in various tasks, they were frequently opaque and hard to interpret. This lack of transparency led to concerns about the reliability, fairness, and accountability of AI systems.

The importance of XAI became even more apparent in the 1990s and 2000s, as AI systems became increasingly embedded in everyday life. For example, AI models were used in credit scoring, healthcare decision-making, and criminal justice systems, raising concerns about bias, discrimination, and lack of accountability. In response to these concerns, researchers began to develop a range of XAI methods, with the aim of making AI systems more transparent and understandable to human users.

Moreover, XAI has become increasingly important due to legal obligations that require AI systems to be transparent and interpretable. For example, the General Data Protection Regulation (GDPR) of the European Union enforces that individuals possess the entitlement to receive an explanation of the verdicts taken by AI systems that impact them [29]. Similarly, in the United States, as per the Fair Credit Reporting Act (FCRA), automated systems are mandated to provide customers with explanations regarding credit determinations [30]. These legal obligations highlight the significance of XAI in ensuring that AI systems are accountable, fair, and unbiased.

XAI has a significant impact on the development and deployment of AI systems, particularly in sensitive domains. By providing users with insight into the decision-making process of AI systems, XAI can enhance the accountability, transparency, and fairness of AI models.

2.2.1. Categorization of explanations

Various classification schemes have been proposed for explanations. One commonly used approach involves categorizing explanations according to two factors: the basis of the explanation and the timing of the explanation. The basis of explanation pertains to the classification of explanation that either clarifies a single observation or clarifies the entire architecture of an black box model. The timing of disclosure refers to when the explanation is provided, either during the decision-making process or after the fact. As a combination of these distinctions, we can classify explanations in 4 main categories.

Local Post-Hoc explanations refer to a set of methods used to provide an understanding of the rationale behind a model's prediction for a given input instance. These methods are employed after the model has generated its output, and they help to shed light on the reasoning behind the decision made by the model.

The purpose of local post-hoc methods is to pinpoint the input features that held the most sway in the decision-making mechanism of the model. One popular technique for doing this is known as "feature importance analysis," which involves evaluating the impact of each input feature to the model's decision. Other methods include perturbation analysis, which involves altering the input features and observing the effect on the model's output, and LIME (Local Interpretable Model-Agnostic Explanations), which creates simple, interpretable method to approximate the original model's behavior for the given input[8].

Local post-hoc methods can be helpful in diverse applications where understanding the reasoning behind a model's prediction is important. For example, in medical diagnosis, local post-hoc explanations can help doctors understand which features of a patient's medical history were most influential in the diagnosis. Similarly, in credit risk assessment, local post-hoc explanations can help lenders understand which factors led to a given loan application being approved or denied.

Global Post-Hoc explanations are a set of methodologies utilized to comprehend the overall operation of a model, regardless of the input instance. In contrast to local post-hoc explanations, which focus on expounding the prediction for a single input instance, global post-hoc techniques examine the model holistically to detect input-independent factors that contribute to the model's decision-making process.

Global post-hoc techniques endeavor to determine the most crucial input features for the entire model instead of individual input cases. This can be achieved by utilizing methods such as permutation feature importance, where each input feature's values are randomly shuffled, and the impact on the model's overall performance is assessed. As an example SHAP (Shapley Additive Explanations) method can be presented as an example.

Global post-hoc methods are advantageous in applications where comprehending the overall behavior of a model is vital, such as in regulatory compliance or auditability. By examining the model as a whole, global post-hoc methods provide insights into the decision-making process of the model and facilitate identification of potential biases or inaccuracies in the model's output. Moreover, these techniques are applicable for optimizing the model by identifying the most significant features that should be prioritized in future iterations.

Local Self-Explaining explanations refer to the set of methods that aim to explain how the model works by analyzing the intermediate results that occur during the model's training or prediction phase. These methods focus on providing transparency into the inner workings of the model and can be useful in applications where model interpretability is important.

An instance of a local self-explanatory model is the Layer-wise Relevance Propagation (LRP) approach, wherein importance scores are attributed to each input feature based on their contribution to the model's output [31]. LRP has been shown to provide accurate and reliable explanations for diverse models, including neural networks and deep learning models.

Global Self-Explaining explanations pertain to the process of clarifying the intermediate outcomes that arise during the model's training and operation without necessitating an extra process or a separate model. This type of explanation is commonly utilized in deep learning models, where the model's intricacy makes it challenging to construe the output based solely on the input features.

An instance of a Global Self-Explaining Explanation technique is Integrated Gradients, which computes the average gradient of the model output concerning the input feature, incorporated over a course from a baseline input to the current input[32]. This method has been shown to provide intuitive and reliable explanations of deep neural network models.

Integrated Gradients have been extensively studied and have been used in diverse domains, such as vision systems and natural language processing.

2.2.2. Explainability Techniques

Explainability techniques in Explainable Artificial Intelligence (XAI) allude to a collection of methodologies employed to offer an understanding of the decisions of AI models. These techniques aim to increase transparency, accountability, and trust in the models by explaining how they arrive at their predictions or decisions.

Feature Importance is a method used in machine learning to identify which features or variables in a dataset have the most impact on a model's output. The knowledge derived from such techniques can be significant in multiple scenarios, such as recognizing the essential factors that govern business performance, comprehending the elements that lead to a medical diagnosis, or refining the design of a product. Various techniques can be employed to compute feature importance, each with its own strengths and limitations.

A frequently used technique for evaluating feature importance is based on the notion of information gain. Information gain is an indicator of the reduction in entropy (or uncertainty) that is achieved by segregating a dataset based on a specific feature. Attributes that lead to the most significant decline in entropy are deemed to be the most informative and, thus, the most vital. This method is commonly used in decision tree algorithms such as C4.5 and CART. [33, 34].

Another commonly used approach for determining feature importance is permutation. It entails randomly shuffling the values of an individual attribute and observing the influence on the model output. Attributes that have the most substantial effect on the model's performance when shuffled are deemed to be the most crucial [23].

Over the years, there has been an increasing interest in utilizing feature importance methodologies to enhance the interpretability and transparency of black box models. By understanding which features are most important to a model's predictions, stakeholders can gain insight into the underlying drivers of a decision and assess the model's fairness and bias [9].

While feature importance techniques can be useful, it is important to note that they have limitations and should be used in conjunction with other explainability techniques such as model visualization and local interpretation. Additionally, the choice of feature importance method may depend on the specific characteristics of the dataset and the model being used.

Surrogate Model refers to a simplified model that is constructed to imitiate the behavior of a more intricate model. The goal of a surrogate model is to provide a more interpretable representation of the underlying model, allowing stakeholders to gain insight into the factors driving the model's output.

Surrogate models can be created using a diverse techniques, such as decision trees, linear regression, and neural networks. The key is to create a model that is simpler and easier to understand than the original model, while still capturing the essential features of the original model's behavior.

Surrogate models are often utilized when the internal workings of a model are not easily accessible, particularly in opaque models. By creating a surrogate model that imitiates the black box model, stakeholders can gain insight into the factors driving the model's predictions without needing to understand the intricacies of the original model [35].

Surrogate models can also be used to perform sensitivity analysis, which involves varying the inputs to the model and observing the impact on the model's predictions. By creating a surrogate model that imitiates the original model, sensitivity analysis can be performed more quickly and efficiently than by directly varying the inputs to the original model [36].

Example Driven explainability technique involves providing interpretable explanations for a black box model's predictions based on specific examples or instances. The idea is to present an explanation of how the model arrived at a particular prediction by highlighting the most influential features or attributes that contributed to the prediction for that particular instance.

Example Driven explanations can be created using a variety of methods, such as LIME [8], counterfactual explanations [37], and prototype explanations [38]. LIME generates an explanation by fitting a local linear model to the neighborhood of the instance of interest. Counterfactual explanations provide a set of alternative instances that would have led to a different prediction. Prototype explanations identify a small set of representative examples that can explain the model's behavior across a range of instances.

Example Driven explanations can be particularly useful in situations where the model's behavior is non-intuitive or unexpected. By providing explanations that are specific to particular instances, Example Driven techniques can help build trust in the model's predictions and can provide insight into how the model is processing information.

Provenance-Based approach to explainability focuses on providing information about the origin and history of the data and model used in making a prediction. This can include information about the source of the data, the transformation steps of data, and the specific features or variables that were used in the model. By providing this information, users can better understand how the model arrived at its prediction and make more informed decisions about how to use the model.

Provenance-Based techniques have been applied in various domains, such as healthcare, finance, and machine learning. In healthcare, the provenance of a model's output can be helpful for clinicians to understand the reason behind a diagnosis or treatment recommendation. In finance, it can assist regulators and auditors in tracing the data lineage for compliance purposes. In machine learning, it can aid in identifying the data and features that are most influential in a model's decision-making process.

One of the popular methods in Provenance-Based techniques is Provenance Graph, which is a directed acyclic graph (DAG) that represents the lineage of data and computations in a workflow [39]. Provenance Graphs have been used in various domains, such as scientific workflows and database systems, to capture the lineage of data and provide explanations for the results.

In recent times, Provenance-Based techniques have garnered significant attention owing to their capacity to offer transparency and interpretability for intricate models. However, there are still challenges in applying these techniques, such as the scalability of capturing and analyzing the provenance of large datasets and models[40].

Declarative Induction approach to explainability focuses on generating human-readable rules or decision trees that summarize the behavior of a black box model [41]. This approach involves training an interpretable model that is as accurate as possible while still being simple and understandable [42]. The resultant model can be utilized to furnish justifications for particular predictions, in addition to obtaining discernments regarding the performance of the opaque model in a broader sense.

One example of the Declarative Induction approach is the use of decision trees to summarize the behavior of a black box model. Decision trees can be trained using techniques such as CART or C4.5 to generate a tree-like structure that represents a set of rules for making predictions [43]. These rules can be easily understood by humans and can provide insights into how the black box model is making its predictions.

2.2.3. Operations To Enable Explainability

As the need for explanation of artificial intelligence models increases, several techniques have been developed to enable explainability operations.

First-Derivative Saliency refers to a method for interpreting machine learning models that involves calculating the first derivative of a model's output with respect to its input.

This approach has gained popularity in recent years as a way to provide insight into the decision-making processes of complex machine learning models.

The concept underlying the First Derivative Saliency method is to determine the most relevant input features that affect a model's output by computing the differentiation of the output base on input. This technique is especially valuable in image and text classification problems, where comprehending the rationale behind a model's decisions can be challenging.

There are several different techniques for calculating First Derivative Saliency, including the popular "gradient-based" methods which includes Gradient-weighted Class Activation Mapping (Grad-CAM) and Guided Backpropagation (GBP). These methods have been shown to be effective in identifying salient features in image data [44, 45].

In addition to image classification, First Derivative Saliency has also been applied to natural language processing assignments like sentiment analysis and named entity recognition [46, 47]. Applications have demonstrated the usefulness of First Derivative Saliency for interpreting complex models in a wide range of domains.

Overall, "First Derivative Saliency" is an important tool for interpreting machine learning models and gaining insight into their decision-making processes. As machine learning gains prominence and becomes integral to diverse applications, the ability to understand and explain these models will become even more crucial.

Layer-wise Relevance Propagation(LRP) is a method used to interpret the predictions of neural networks by assigning correlation scores to input features. The idea behind LRP is to propagate the correlation scores of the output of a model back to its inputs, in order to identify which input features were important for making the prediction. LRP can be used to understand how a neural network arrives at its decisions, and to identify potential biases or errors in its predictions [31, 48].

The basic idea of LRP is to assign relevance scores to each neuron in the network, which represent the contribution of that neuron to the final prediction. These relevance scores are then propagated backwards through the network using a set of propagation rules, which depend on the architecture and activation functions of the network.

One of the advantages of LRP is that it provides a principled approach for interpreting the output of a neural network, without requiring any external knowledge or preconceptions about the problem being solved. LRP has demonstrated its efficacy in explaining the predictions of diverse neural network architectures, spanning convolutional neural networks, deep belief networks, and recurrent neural networks. Its applications span across image classification, natural language processing, and drug discovery, where it has proven to be a useful tool for detecting possible biases and identifying significant features in neural network predictions.

Input Perturbations is employed to interpret the predictions made by machine learning models by altering the input data and evaluating its impact on the output. The underlying concept of this technique is to identify the features of input data that are critical for the model's predictions. This method is beneficial in comprehending the model's decision-making process, discovering any possible predispositions or inaccuracies, and enhancing the model's effectiveness.

Input perturbations can take many forms, including adding noise to the input data, removing or replacing certain features, or manipulating the input in other ways. The effect of these perturbations on the model's output can be analyzed to determine which attributes are most critical for model's predictions.

Input perturbations are a powerful and versatile technique as they can be applied to any type of machine learning model, regardless of its structure or training approach. This model-agnostic technique has been successfully utilized to explain the predictions of a broad range of machine learning models, such as decision trees, support vector machines, and neural networks.

One common method for analyzing the effect of input perturbations is to use sensitivity analysis, which involves calculating the change in the model's output for a given change in the input. Another method is to use visualization techniques, such as heatmaps or scatterplots, to identify patterns in the model's predictions and their relationship to the input data.

Input perturbations have found applications across diverse fields, including but not limited to image classification, natural language processing, and healthcare. It has been shown to be effective at identifying relevant features, detecting potential biases in machine learning models, and improving the interpretability and transparency of these models.

Attention is a deep learning technique that enables the model to concentrate on specific parts of the input data during processing [49]. This mechanism is inspired by human cognition, where the brain selects information to process based on its importance to the given task. The implementation of attention in deep learning has been proven to enhance the model's performance in various applications such as image classification, speech recognition, and machine translation [50].

The attention mechanism functions by assigning a significance score to each element in the input data, depending on its significance to the task. These scores are then utilized to compute a weighted summation of the input, which is subsequently propagated through the remainder of the network. The significance scores are obtained through training and can be viewed as an indication of the significance of each input element.

Attention offers the advantage of enhancing the interpretability of deep learning models, as it enables us to comprehend the specific regions of the input data that the model is emphasizing during processing. This ability can be valuable in gaining insight into the decision-making process of the model, detecting possible biases or inaccuracies, and refining the model's performance.

LSTM (Long Short-Term Memory) is a variant of recurrent neural networks that excels in handling sequential data. What distinguishes LSTM is its capacity to selectively retain and retrieve information over a period, aided by gating signals that regulate the information flow through the network. These gating signals are acquired through training, and offer an indication of the significance of distinct segments of the input sequence.

In an LSTM, the gating signal consists of three gates: the input gate, the forget gate, and the output gate. The input gate regulates the amount of new input to be added to the memory cell, the forget gate determines how much old information should be retained in the memory cell, and the output gate determines the extent to which the current memory cell value should be output to the next layer in the network.

The gating signal is computed by taking a linear transformation of the input and the previous hidden state, and then applying a sigmoid function. By doing so, the model is capable of learning a function that is flexible enough to adapt to various input sequences, and can manage the flow of information in the network efficiently.

In an LSTM, the gating signal has been shown to be a useful tool for understanding the inner structure of the model, along with for improving the performance of the network on certain tasks [51–53]. By analyzing the gating signals, researchers can gain insights into which parts of the input sequence are most crucial to the task at hand, and can use this information to refine the network architecture and training process.

Explainability-aware architecture design is a growing tendency in deep learning involves integrating explanation techniques within the model architecture to improve interpretability. This approach is gaining popularity as it can lead to more reliable and better-performing models in diverse domains, including finance, healthcare, and autonomous driving.

The attention mechanism is one such technique that enables the model to selectively focus on relevant parts of the input and produce a more interpretable output. This approach has been effectively utilized in natural language processing (NLP) tasks, such as machine translation [54] and text classification [55].

Another example of an explainability-aware architecture design is the use of graph neural networks (GNNs), which can model the structure and relationships between entities in a graph and provide a more interpretable output. GNNs have been applied in various domains such as social networks [56], drug discovery [57], and recommendation systems [58].

In addition to attention and GNNs, there are other techniques that can be used to design explainability-aware architectures, includes decision trees, and causal models [59].

By incorporating explainability techniques directly into the model architecture, explainability-aware architecture design can provide a more interpretable and transparent model, which can lead to better performance and more reliable predictions.

2.2.4. XAI Visualizing Techniques

Another important aspect of XAI is visualizing the decision-making process of these models, which can help humans better understand and trust their outputs. Some popular methods for visualizing the internal workings of machine learning models has revealed.

A common technique is to visualize the importance of different features, which can reveal the features that have the greatest influence on a model's predictions. For instance, permutation-based techniques such as “Permutation Feature Importance (PFI)” and “Partial Dependence Plot (PDP)” can be utilized to analyze the impact of individual attributes on the model's output [60]. Alternatively, “Integrated Gradients” and “Layer-wise Relevance Propagation (LRP)” can be used to visualize the contribution of each feature to the model's output by computing the output's gradient with respect to the input [31].

Another approach to visualizing machine learning models is to use saliency maps, which highlight the parts of an image that the model is focusing on when making its predictions. For example, Grad-CAM (Gradient-weighted Class Activation Mapping) calculates the gradient of the output concerning the feature maps of the last convolutional layer, a heatmap is generated that pinpoints the image regions that are most significant for the model's prediction. [44].

Raw declarative and natural language processes are another XAI visualization methods that convert model behavior into natural language for human comprehension. Raw declarative models map complex model decisions to simple, understandable rules or logic that a non-expert can easily interpret. Natural language processes, on the other hand, convert the model's behavior into a more human-readable text, allowing users to understand the reasoning behind a particular prediction.

One example of raw declarative methods is the "Decision Trees", which are commonly used for classification tasks and can be visualized as a hierarchical structure of if-else statements that explain the reasoning behind the model's predictions [61].

Natural language processes have also been applied in XAI to enhance model interpretability. One example is the work of Lei[62], who proposed a natural language generation method to explain model behavior. The method generates textual explanations for model predictions, which can be used to understand the reasoning behind the model's decisions.

Both raw declarative and natural language methods provide a way to translate complex model decisions into human-readable formats. These methods can help to improve model transparency and foster greater trust in machine learning systems.

2.2.5. Evaluation XAI Methods

Evaluation of XAI methods is an essential step in determining the effectiveness and usefulness of these methods. The evaluation can be conducted using various metrics such as, accuracy, precision, recall, and F1 score. In addition to these traditional metrics, there are also several other metrics specific to XAI, such as interpretability and transparency [63].

A common method for evaluating explainable artificial intelligence (XAI) techniques involves employing human participants to assess the quality of the explanations generated by these methods. This approach is particularly relevant when evaluating methods designed to explain complex models, includes ANNs. For instance, Li et al. [64] utilized human participants to evaluate the effectiveness of explanation methods in elucidating the predictions made by neural networks.

Another approach to evaluating XAI methods is to use simulation experiments to measure the effectiveness of these methods in identifying errors or biases in the models they are designed to explain. For example, Kaur et al. used simulation experiments to make assesment the effectiveness of different explainable artificial intelligence methods in identifying biases in a predictive model for predicting breast cancer recurrence [65]. Simulation experiments involve creating artificial datasets to evaluate the performance of XAI methods. These datasets are created to mimic real-world scenarios and are designed to test the ability of XAI methods to accurately explain the decisions made by machine learning models. By simulating different scenarios, researchers can evaluate the robustness of XAI methods and determine their effectiveness in different contexts.

3. RELATED WORK

3.1. Local Interpretable Model-Agnostic Explanations

Local Interpretable Model-Agnostic Explanations(LIME) is a method to explain the predictions of opaque machine learning models by generating understandable and localized approximations of these models. In 2016, Ribeiro et al. introduced LIME, which has since become a commonly used technique for enhancing interpretability in machine learning. [61].

The basic idea behind LIME is to create interpretable model that which imitates the actions of a complex model in a limited area surrounding a specific instance. This approximation can then be used to understand why the black box model made a certain prediction for that instance. To create this local approximation, LIME first selects a set of important features for the instance in question, then generates a dataset of perturbed instances by randomly sampling from the feature distributions. The black box model is then used to predict the output of these perturbed instances, and a linear regression model is trained on the perturbed instances and their corresponding black box model predictions. The coefficients of this linear regression model represent the importance of each feature in the local approximation.

LIME has several advantages over other explainability methods. First, it can be used with any black box model, including deep learning models. It makes LIME “Model-Agnostic”. Second, it produces interpretable explanations that are easy to understand and can be used to build trust in machine learning systems. Finally, it can be used to explain both individual predictions and the overall behavior of a model.

There have been many applications of LIME in various fields, including healthcare [66], finance [67], and NLP [68]. However, there are also some constraints to LIME, such as the fact that it only approximates the black box model in a small region around the instance, and that the explanations may not be globally consistent.

In conclusion, LIME is a powerful and widely used technique for explaining black box machine learning models. Its ability to create simple, interpretable approximations of complex models has made it a popular choice for building trust in machine learning systems. However, as with any explainability method, it is important to understand its limitations and use it appropriately.

3.2. SHAP (SHapley Additive exPlanations)

The SHAP technique is an approach for interpreting the results of intricate black box models. It offers a method to evaluate the impact of each input feature on the model's prediction for a specific instance. The SHAP values demonstrate the alteration in the anticipated model output when a specific feature value is observed, compared to when the feature value is substituted by a baseline value. [9].

The SHAP technique utilizes the concept of Shapley values from cooperative game theory, which provides a way to fairly allocate the total value of a coalition of players to each player based on their individual contributions [69]. The Shapley value is a method for assigning a payoff to each player in a cooperative game. It is based on the idea that the contribution of each player to the total payoff should be proportional to their marginal contribution to any coalition that achieves the payoff. In other words, the Shapley value is a way of fairly dividing the payoff among the players based on their contributions. In machine learning, the coalition of players in the context of SHAP is the set of input features, and the value is the output of the model. The Shapley values represent the marginal contribution of each feature to the difference between the actual prediction and the expected prediction for a given input instance.

$$\phi_j^S(x) = \sum_{T \subseteq 1,\ldots,p \setminus j} \frac{|T|!(p-|T|-1)!}{p!} \left[f_{T \cup j}(x) - f_T(x) \right] \tag{1}$$

In equation 1, $\phi_j^S(x)$ represents the SHAP value for feature j in the context of input x. p is the total number of attributes in the dataset, T is a subset of the feature indices excluding feature j. $f_T(x)$ represents the predicted output of the machine learning model when only the features in subset T are present in the input x. $f_{T \cup j}(x)$ represents the predicted output of the model when both feature j and the features in subset T are present in the input x. The equation calculates the difference in predicted output between the input x with feature j and the features in subset T, and the input x with only the features in subset T. These differences are then weighted and averaged over all possible subsets of features, resulting in a SHAP value for feature j that provides a measure of feature importance and can be used to interpret the predictions of the any black box model.

The SHAP method is model-agnostic, meaning it can be used with various machine learning models, such as ANNs, decision trees, and linear models. This approach makes the SHAP method versatile and applicable to a wide range of problems. The method approximates the model locally to calculate the SHAP values, enabling it to handle complex models and large datasets without significant computational overhead [9].

The SHAP method has found applications in various domains, including healthcare [70], finance [71], and natural language processing [72]. It has also been integrated into popular machine learning libraries, such as XGBoost [25] and scikit-learn [73], making it accessible to a wide range of users.

To summarize, the SHAP method is a potent approach to interpret the results of complicated machine learning models. It utilizes Shapley values from cooperative game theory to estimate the input feature's contribution to the model's prediction for a given instance, regardless of the model's type. The SHAP method uses a model-agnostic approach to compute the SHAP values through a local approximation of the model, which makes it scalable to large datasets and complex models. Due to its widespread applications and its integration into popular machine learning libraries, the SHAP method is an invaluable tool for both researchers and practitioners.

4. PROPOSED METHOD

In the context of our thesis, we introduce a novel technique denominated as the “Tree Ensemble for Explainable Artificial Intelligence(TEXAI)” which related to the concept of explainable artificial intelligence. This approach is aligned with the theoretical underpinnings and empirical evidence we have expounded in the preceding sections. Broadly speaking, the TEXAI model we have devised takes as input an artificial neural network (ANN) and it’s training dataset, as depicted in figure 4.1. Subsequently, it generates a comprehensive explication for the decisions made by the ANN.

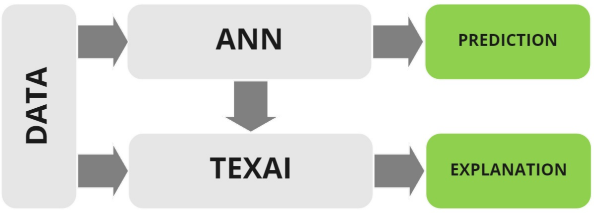


Figure 4.1 Basic workflow of TEXAI model

The structure of our TEXAI possesses the capability to elucidate the decision-making process of any black-box model. As such, our TEXAI can be categorized as **model-agnostic**. It can be classified as a **post-hoc** approach, given its ability to generate explanations regarding the decision-making of the black-box model subsequent to its training phase. It does not require any supplementary processing during the training of the black box model. And our TEXAI has the capacity to generate two types of explanations: one pertaining to an individual observation and the other concerning the overall structure of the opaque model. This duality of explanations allows us to assert that our model generates both **local** and **global** explanations.

The research carried out by the Defense Advanced Research Projects Agency(DARPA) has shown that there is a trade-off between the predictive accuracy and the explainability accuracy of a model[1]. The model prediction accuracy is inversely proportional to its explainability. The models that achieve high predictive performance tend to have lower explainability accuracy. The comparative scheme of DARPA, which is illustrated in figure 4.2, highlights this trade-off between predictive performance and explainability. Thus, there is a need to find a balance between accuracy and explainability in order to obtain a model that is both accurate and interpretable.

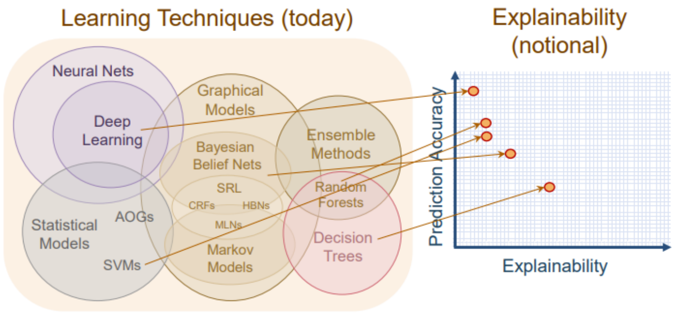


Figure 4.2 Learning techniques and their explainability.[1]

The central question we aim to address in this study is whether models that exhibit high predictive accuracy can be transformed into highly explainable models. This question serves as the primary inspiration for our proposed method. In light of this, we have chosen to utilize the decision tree method as it is considered to be one of the most suitable techniques for generating explainable models. However, it is not realistic to expect that a single decision tree can achieve the same level of performance as a neural network. Therefore, we have opted to construct an ensemble that comprises multiple decision trees to simulate the behavior of a neural network. Our methodology involves the utilization of the **surrogate model** technique, whereby an additional model is developed to emulate the black box model in order to provide an explanation of its decision process. Simultaneously, it can be asserted that the our TEXAI was conceived utilizing the **Declarative Induction** explanation technique, as it was built through the utilization of self-explanatory decision trees.

4.1. TEXAI Model Development

The subsequent sections of our thesis shall provide an elaborate elucidation of the phases involved in the development of our TEXAI. For the purpose to gain preliminary insight, we present the pseudo code of our model in Algorithm and the visual flowchart in figure 4.3.

Algorithm TEXAI model development

```
procedure MAIN($Dataset$,$ANN$)
    $D$ =DATAAUGMENTATION($Dataset$)
    $predictions = ANN.predict(D)$
    $D[Target] = predictions$
    $TreeForest$ =CREATEDECISIONTREEFOREST($Dataset$)
    $Paths$ =TREEFOREST.CREATEPATH
    $Paths$ =PRUNE($Paths$)
    $Paths$ =REMOVEOVERLAPPEDPATHS(Paths)
    Return $Paths$
end procedure
procedure DATAAUGMENTATION($Dataset$)
    $length = Dataset.length$
    for each integer $i$ in $length$ do
        $Pickrandom2Observations$
        $Dataset.append(mean(twoobservations))$
    end for
    Return $Dataset$
end procedure
procedure CREATEDECISIONTREEFOREST($Dataset$)
    $ColumnCount = Dataset.Columns.Count$
    for each integer $i$ in $MaxTreeCount$ do
        $RandomlySelectedColumns = Dataset.Columns.randomSample(TempColCount)$
        $TempDataset = Dataset[RandomlySelectedColumns]$
        $Tree = sklearn.DecisionTreeClassifier().Fit(TempDataset[RandomlySelectedColumns], TempDataset[Target])$
        $Forest.Add(Tree)$
    end for
    Return $Forest$
end procedure
procedure PRUNE($Paths$)
    for each path $p$ in $paths$ do
        if CHECK($p$) $isFalse$ then
            $Paths.remove(p)$
        end if
    end for
    Return $Paths$
end procedure
procedure CHECK($Path$)
    if $Path.Coverage > MinCoverage$ & $Path.Purity > MinPurity$ & $Path.Proba > MinProba$ then
        Return $True$
    else
        Return $False$
    end if
end procedure
procedure REMOVEOVERLAPPEDPATHS($Paths$)
    $groups = Paths.GroupBy(Breakdownfeatures, direction)$
    for each group $g$ in $groups$ do
        $tempPaths.add(group.pickMoreInclusive())$
    end for
    Return $tempPaths$
end procedure
```

In the initial phase of this study, a rudimentary artificial neural network (ANN) was trained and assessed using the dataset, attaining an accuracy rate of roughly 39%. Though this accuracy rate may appear low or unsatisfactory, it is noteworthy that our objective in this thesis does not entail the presentation of a highly efficacious ANN model. Rather, our aim is to explicate the decision-making mechanisms of the ANN in a manner that can be

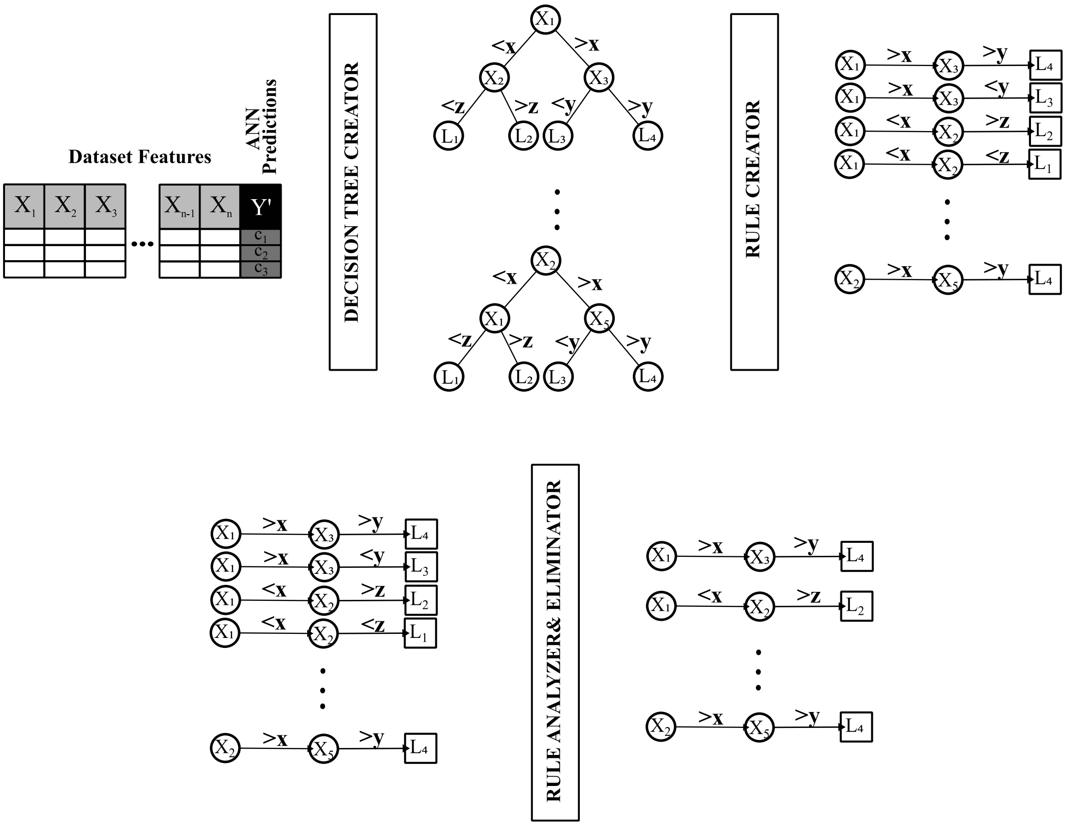


Figure 4.3 TEXAI model work flow schema.

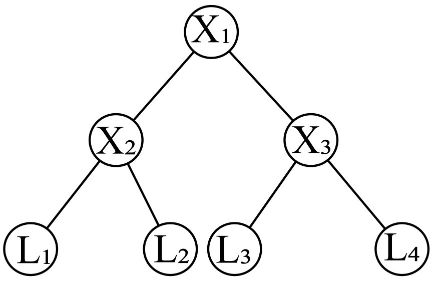
comprehended by human beings, even when the accuracy rate is diminished. Consequently, no attempts were made to enhance the performance of the model.

It is commonly understood that there exists a positive correlation between the amount of data and the accuracy of models. Furthermore, as the amount of the data increases, a more comprehensive understanding of the model's behavior can be attained. Based on these suppositions, we have taken the initiative to augment our input dataset. By doing so, we aim to ensure that the threshold values obtained in the node breakdown of a decision tree are more consistent. For instance, suppose a dataset comprises of two observations, where the x_1 value of the first observation is denoted as t_1 and its target class is c_1, and the x_1 value of the second observation is t_2 and its target class is c_2. In a model trained with this data, the threshold value for attribute x_1 would be the average of t_1 and t_2. However, it is evident that a more precise threshold value could be derived if additional observations between t_1 and t_2 for the x_1 attribute were present in the dataset. As a result, we have augmented our input dataset to incorporate more data using a data augmentation method that generated new observations by averaging two randomly chosen observations, thereby doubling the data volume.

$$n = \sum_{i=2}^{n_f/2} \binom{n_f}{i} \tag{2}$$

The proposed TEXAI takes an augmented and preprocessed dataset, and a neural network model as input, and generates a set of rules as output. The algorithm initiates by constructing a set of n decision trees, each independently built on a subset of the original dataset. The subset is created by randomly selecting a subset of features from the original dataset. It is important to note that the value of n indicates the maximum number of sample datasets that can be generated through random feature selection from the original dataset. The equtation for n represented in equation 2 where n is the maximum number of trees to be created and n_f is the number of features in dataset. An example of these trees is shown in figure 4.4.Each path of each tree represents a rule. In the information presented about the leaf in the figure 4.4, the array named "classes" shows the target class distribution in that leaf. The class with the maximum value in this array is the prediction of the path that reaches that leaf. And the other informations such as gini impurity, probability, sample count and coverage in the figure 4.4 will be explained subsequent part of the our thesis. Among the n trees, many different rules may make the same decisions as the ANN that requires explanation, thus locally simulating its behavior. The core idea of our approach is to identify the paths resulting in these rules, which provide a local explanation of the ANN model, and then create an ensemble of them to explain the entire ANN model.

Note that, each tree was trained using different and random sub-samples of features and observations. Due to the use of a randomized subset of features, different decision trees were created and the consistency of the node thresholds in these decision trees was observed. Not all paths were immediately accepted though. Only decisive paths which satisfy the threshold



X_1,X_2,X_3 some features in dataset
L_1,L_2,L_3,L_4 leaf nodes of decision tree
For each leaf node we consider;
Classes = [C_1,C_2,C_3...,C_n]
GI = Gini Impurity
P(L) = Probability at leaf
C(L) = Coverage at leaf
N(L) = Sample count at leaf

Figure 4.4 Decision tree example

we established based on the following three metrics were used, namely, **Gini impurity**, **probability** and **coverage** metrics.

Gini impurity is a measure of the impurity or heterogeneity of a set of data points. It is commonly used in decision tree algorithms for classification tasks. A smaller value of Gini impurity indicates a more pure or homogeneous set of data points, where all data points belong to the same class. Conversely, a larger value of Gini impurity indicates a more impure or heterogeneous set of data points, where the data points belong to multiple classes with roughly equal probabilities. Formally, the Gini impurity of a set S is defined in equation 3 where p_j is the proportion of data points in S that belong to class j. The summation is taken over all classes j.

$$gini(S) = 1 - \sum p_j^2 \tag{3}$$

The probability assigned to a leaf node is a computed estimate of the likelihood that a novel data point would be classified as a certain class label, which is determined based on the ratio of observations of that class label present in the leaf node to the overall number of samples in that node. In a formal context, the probability associated with a class label c for a given leaf node L may be mathematically expressed as illustrated in equation 4. Herein, c signifies the class label that is most commonly observed in the leaf node, $n(c, L)$ represents the number of samples in the leaf node L belonging to the class c, and $n(L)$ refers to the total number of samples contained in the leaf node L.

$$p(L) = \frac{n(c, L)}{n(L)} \tag{4}$$

In our study, we utilized the concept of coverage as the third criterion for evaluating decision tree paths. This addition was motivated by the issue of overfitting. The aim was to ensure that the rule set generated by the model would exhibit similar levels of accuracy on both training and test datasets. The conventional definition of coverage for a leaf in decision trees is the ratio of the number of observations in the leaf to the total number of observations. However, in our classification approach, the target attribute was based on predictions generated by ANN. Occasionally, the ANN may predict a specific class for a limited number of observations, which may be contained within a single leaf. Using the standard coverage equation to evaluate this path to leaf may result in it being categorized as nondecisive. However, upon further examination, it may be discovered that this leaf covers the entire area associated with the relevant class in hyperspace. To address this issue, we formulated an alternative coverage equation and presented it as equation 5. In this equation, c represents the class with the maximum number of observations in the leaf, $n(c, L)$ denotes the number of observations in leaf node L that belong to class c, and $n(c)$ indicates the total number of observations belonging to class c. This modified equation enables us to obtain a more accurate assessment of the coverage for each leaf in the decision tree, even when dealing with non-uniform class distributions.

$$C(L) = \frac{n(c, L)}{n(c)} \tag{5}$$

Each individual path of the decision trees was evaluated based on the three metrics previously mentioned. Those paths that were determined to be non-decisive were subsequently eliminated, resulting in a reduced set of paths. As examples, the following two rules were generated from the abalone dataset. In this context, the symbol *P* is used to denote the probability associated with a leaf node, while *N* represents the number of observations classified to that specific leaf node. Additionally, *GI* pertains to the measure of impurity known as “Gini Impurity”, while the “Classes” array indicates the distribution of observations based on their target class.

Rule 1: (Viscera weight>0.44475) then (Height>0.155) then (Length>0.6825) ⇒ “Class11” P:97.0 N:128 GI:0.02 Classes:[0,0,4,124]

Rule 2: (Diameter>0.335) then (I≤0.5) then (Shell weight≤0.21975) then (Whole weight>0.761) then (Whole weight≤1.088) ⇒ “Class9” P:97.3 N:150 GI:0.01 Classes:[0,146,4,0]

Despite the elimination of non-decisive paths, it is possible that some remaining paths may overlap on the same plane. In such scenarios, the path that covers a larger area on the plane is preserved while the other path is discarded. During this process, rules that are on the same hyperplane and provide the same target class as a prediction are consolidated into groups. Among these groups, the most comprehensive rule is selected, and the rest are eliminated. To provide an illustration, two rules generated by our model on the same hyperspace are presented below. Rule 3 is more comprehensive, as evident from the N value, which denotes the total number of observations covered by that rule. Hence, rule 3 is retained while rule 4 is eliminated.

Rule 3: (I≤0.5) then (Length>0.4475) then (Length<0.59) ⇒ “Class9” P:100.0 N:158 GI:0.0 Classes:[0,158,0,0]

Rule 4: (I≤0.5) then (Length>0.4925) then (Length≤0.59) ⇒ “Class9” P:100.0 N:79 GI:0.0 Classes:[0,79,0,0]

After applying the filtering criteria as described previously, a set of rules that define each target class of the neural network are obtained. These rules are derived from the decision trees’ leaves and are linear in nature, dividing the space into two halves, one of which corresponds to the neural network’s class decision and the other half suggests a different class. The set of these rules describes a region that potentially contains open half-spaces, which closely matches the neural network’s decisions locally. It should be noted that the neural network is a non-linear model and, as a result, the regions described by our set of rules are only piece-wise approximations of the original regions of the neural network.

By manipulating the parameters of algorithm , specifically the minimum gini impurity, purity and probability values, we can generate tighter or looser regions. When the regions are tighter, the resulting explanations of the neural network is more accurate, but may fail to account for some observations. Conversely, when the regions are looser, the explanations of the neural network is less accurate, but the number of labeled observations increases. In the next section, we present experimental results obtained from the "Wine Quality" and "Abalone" datasets.

5. EXPERIMENTAL RESULTS

5.1. Datasets

In the process of constructing our TEXAI model, we employed two distinct datasets, designated as “Abalone” and “Wine Quality”. A substantial portion of each dataset, that is 80%, was allocated for utilization in the development of the TEXAI model. The remaining 20% was reserved as a test phase for evaluation purposes. In order to ensure the accuracy and effectiveness of the TEXAI model development process, it is crucial to utilize pre-processed datasets that were originally used to train the black-box model.

5.1.1. Abalone Dataset

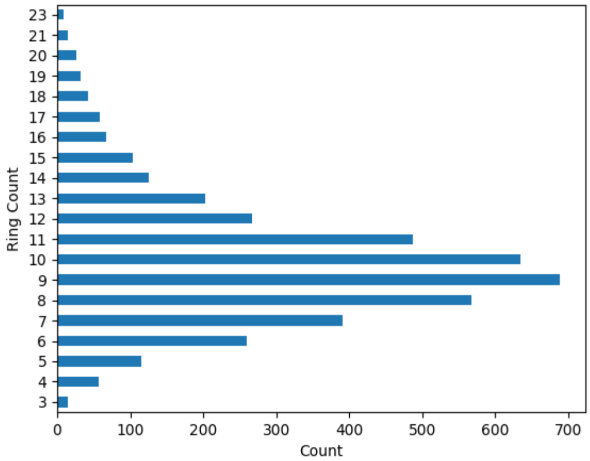


Figure 5.1 Abalone dataset target class distribution

Primarily, in the process of developing our model TEXAI, we employed the *abalone* dataset, which can be accessed from [74]. This dataset contains a total of 4,177 observations, comprising of 9 attributes and a target column that indicates the number of rings. These attributes include *sex(F,I,M)*, *length*, *diameter*, *height*, *whole weight*, *shucked weight*, *viscera weight*, and *shell weight*.

The target attribute of the dataset utilized in this study comprises numerical values, rendering it suitable for regression tasks. However, the objective of the thesis is to investigate the decision-making process of classification models. Unfortunately, as illustrated in figure 5.1, there are a large number of classes and the number of observations in each class is quite unbalanced. Due to the scarcity of observations in certain target classes, black box models have a tendency to predict target classes with a high number of observations for all instances. Our TEXAI model, developed with the predictions and inputs of the ANN model, will have no observations for target classes characterized by a low number of instances. Consequently, we eliminated instances belonging to target classes with low observation count from our dataset. Thus, by subsampling the four most populated classes (8, 9, 10, and 11 rings) we obtained a more balanced classification problem. Upon completion of the elimination process, a final count of 2378 observations remained. During the development process of TEXAI, 1902 observations were employed for training purposes while the remainder of the dataset was allocated for testing the model's performance. The training dataset was augmented to twice its original size using the augmentation technique described in the previous section.

5.1.2. Wine Quality Dataset

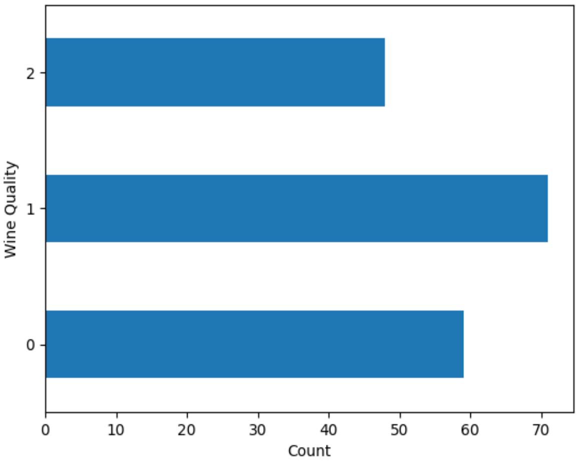


Figure 5.2 Wine quality dataset target class distribution

Additionally to *Abalone* dataset, during the testing phase, the *wine quality* dataset, which is also available on [75] was utilized. This dataset contains a total of 178 observations, comprising of 13 attributes and a target column that indicates the quality of wine. These attributes include *alcohol*, *malic_acid*, *ash*, *alcalinity_of_ash*, *magnesium*, *total_phenols*, *flavanoids*, *nonflavanoid_phenols*, *proanthocyanins*, *color_intensity*, *hue*, *od280/od315_of_diluted_wines*, *proline*. The target attribute of the dataset comprises three unique classes, and their distribution is depicted in figure 5.2. During the development process of TEXAI, 142 observations were employed for training purposes while the remainder of the dataset was allocated for testing the model's performance. The training dataset was augmented to twice its original size using the augmentation technique described in the previous section.

5.2. Threshold Tuning

In the preceding section, where rule elimination was to be performed, the utilization of probability, coverage, and gini impurity thresholds was discussed. In this section, we present an experimental approach for determining an optimal threshold value.

To determine minimum probability threshold, we conducted a systematic investigation in which we gradually increased the probability value by 1 percent increments while maintaining all other parameters constant. We then evaluated the resulting TEXAI model's performance. The findings of our study are presented in figure 5.3 where the y-axis shows the normalized values about results and the x-axis shows the minimum probability value. Our analysis uncovered a direct relationship between the minimum probability threshold and the TEXAI model's decision accuracy. Increasing the minimum probability threshold decreases the area covered by the TEXAI in the ANN's decision space and increases the number of observations for which the TEXAI cannot make a decision. However, this decrease in the decision space also results in a reduction in the number of incorrect predictions made by the TEXAI. Conversely, decreasing the minimum probability threshold increases the number of incorrect predictions made by the TEXAI, but reduces the number of observations for which the TEXAI is uncertain. Therefore, a trade-off exists between the TEXAI being indecisive and making incorrect decisions. Consequently, it is crucial to select an appropriate minimum threshold to achieve an optimal balance. Based on our empirical findings, for *Abalone* dataset, we have determined that a probability value higher than the 44 percent is required for a path to be decisive.

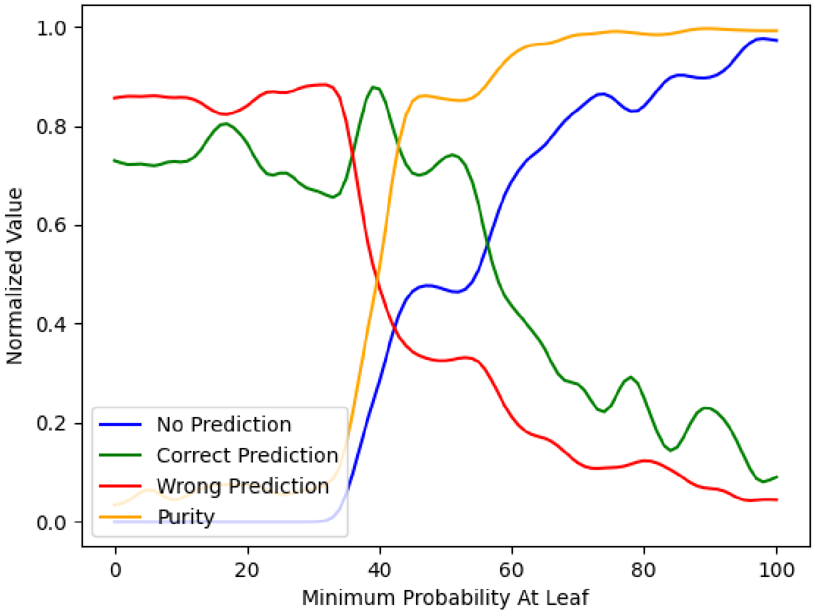


Figure 5.3 Minimum probability evaluation(Abalone)

As observed in our minimum probability analysis, a similar trade-off exists between the TEXAI model providing an incorrect estimate and being undecisive. Similar to the assessment of the minimum probability threshold, we employed a systematic methodology. While maintaining constancy of all other variables, we systematically incremented the coverage parameter by 1% increments and assessed the outcomes. Based on our experimental analysis, which is presented in figure 5.4 where the y-axis shows the normalized values about results and the x-axis shows the minimum coverage value and considering the generic nature of our coverage equation 5, we found that a minimum coverage value of 13% represents a reasonable compromise between model error and model indecisiveness.

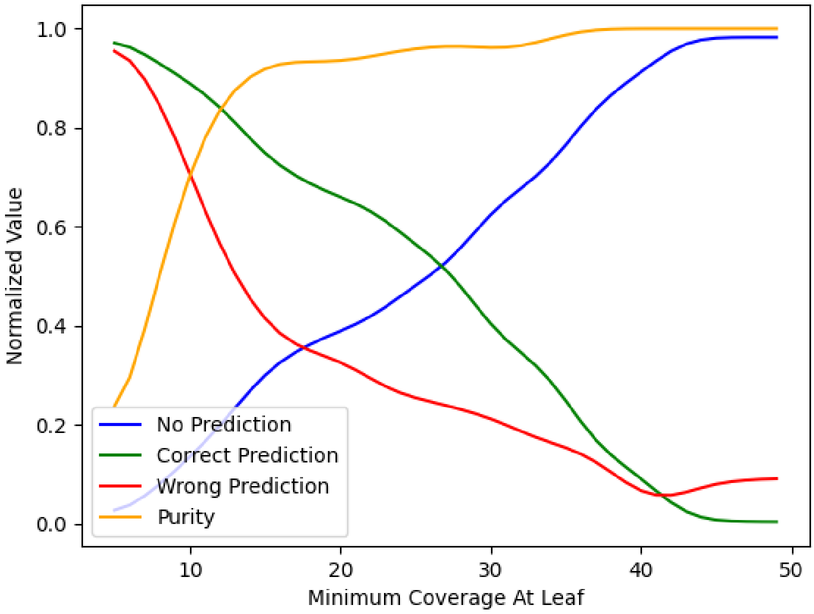


Figure 5.4 Minimum coverage evaluation(Abalone)

Both figure 5.3 and figure 5.4, The “No Prediction” displays the normalized quantity of observations that were left unclassified by TEXAI, whereas the “Wrong Prediction” exhibits the normalized quantity of observations that were incorrect classified by TEXAI. Conversely, the “Correct Prediction” line demonstrates the normalized quantity of observations that were accurately classified by TEXAI. The term “purity” pertains to the purity of TEXAI. The purity of TEXAI was computed according to test dataset using equation 6 where the symbol $\sigma(r)$ represents the purity of the TEXAI, the symbol ε represents the number of rules that predict the same target class as the ANN for that observation, while the symbol lambda denotes the total number of rules that fit the observation. Additionally, the symbol N represents the total number of observations.

$$\sigma(T) = \frac{\sum \frac{\varepsilon}{\lambda}}{N} \tag{6}$$

Upon analyzing both figures 5.3 and figure 5.4, it becomes apparent that the optimal values for probability and coverage are 44% and 13%, respectively. This conclusion is based on the observation that up to these threshold values, there was a significant and rapid increase in the number of correct predictions and purity, accompanied by a sharp decline in the number of incorrect predictions. However, beyond these threshold values, while the purity and number of correct predictions continued to increase gradually, the number of observations in which the model was indecisive increased rapidly.

5.3. Results

As a consequence of the procedures outlined in the preceding segment, a total of 203 rules were derived for the abalone dataset and 150 rules were obtained for the wine quality dataset. Due to the random sampling of datasets from the main dataset for the purpose of constructing decision trees, these figures may vary, even for the identical dataset and model. Nevertheless, given the creation of a substantial number of decision trees, any disparity in these figures is expected to be negligible. Each rule indicates a target class. The set of rules indicating a class defines the area covered by that class in hyperspace. Table 5.1 illustrates the count of rules that pertain to target classes in the “Abalone” dataset, while table 5.2 presents the corresponding rule count for the target classes in the Wine dataset.

Target Class	**Rules Count**
Class 8	85
Class 9	45
Class 10	23
Class 11	75
Total	228

Table 5.1 Abalone dataset rules count

Target Class	**Rules Count**
Class 0	111
Class 1	59
Class 2	49
Total	219

Table 5.2 Wine quality dataset rules count

As we previously discussed, our starting point was to mimic high-performing models with low interpretability, such as ANN, by transforming them into decision tree ensembles that have high interpretability. Therefore, our evaluation should focus on the extent to which our TEXAI mimics ANN. Broadly speaking, all AI models take an input and produce an output.

Consequently, the similarity between two AI methods or models is determined by comparing the outputs they generate for the same inputs.

The decision-making process for the TEXAI model that we developed can be described as follows: For a given observation, all rules that match the observation are considered, and each of these rules has one vote which is the target class of them. The target class that receives the highest number of votes is then selected as TEXAI's estimate for that particular observation.

During the evaluation of our results, we employed two distinct approaches. The first approach involved a comparison between the decisions made by the Artificial Neural Network (ANN) and those made by the TEXAI model, using the test datasets.

The visual representations provided below illustrate prediction plots made over the "Abalone" dataset, with figure 5.5 representing the predictions of ANN for observations, figure 5.6 representing the predictions of TEXAI for observations, and figure 5.7 displaying the differences between the predictions of ANN and TEXAI.

The prediction plots for the wine quality dataset that we have used as secondary dataset are presented in figure 5.8 for ANN predictions, figure 5.9 for TEXAI predictions, and figure 5.10 for the differences between ANN and TEXAI predictions.

According to both datasets we used, the results are presented in table 5.3. As can be seen from the results, there is an agreement of almost 94% between the ANN's and TEXAI's decisions on the wine dataset, and an agreement of 81% on the abalone dataset. Note that, in the abalone dataset there is also a large number of observations where the TEXAI model couldn't make a prediction. That is because of the tightness of the TEXAI model constructed. We could easily decrease the number of "no predictions" in favor of the true and false predictions. For each dataset the desired ratio can be obtained by the model builder using the parameters described above.

As a secondary approach for evaluation, we conducted individualized assessments for each rule that was created. It was explicitly stated that each of our rules is linked to a target class.

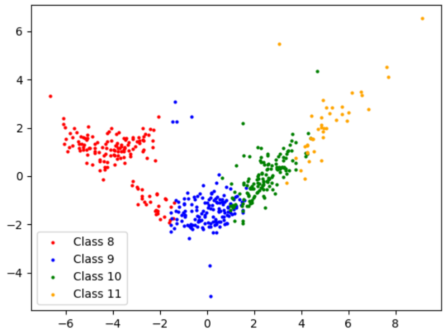


Figure 5.5 ANN prediction plots(Abalone)

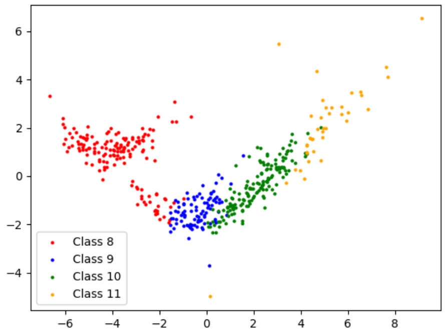


Figure 5.6 TEXAI prediction plots(Abalone)

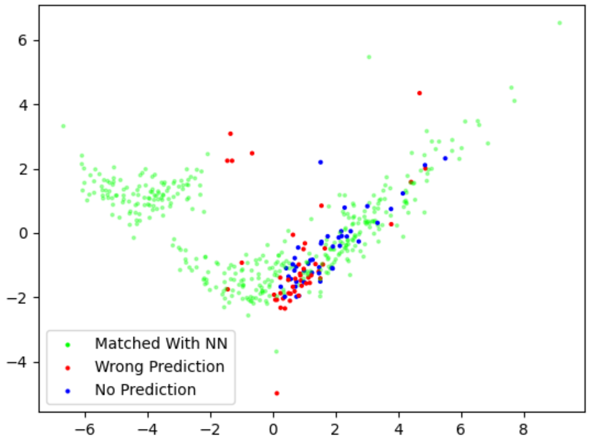
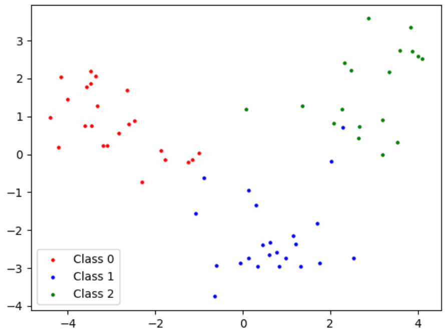


Figure 5.7 ANN vs TEXAI predictions(Abalone)

Dataset	Test Observation Count	Matched with NN	Wrong Prediction	No Prediction
Abalone Dataset	476	386	46	44
Wine Quality Dataset	64	60	4	0

Table 5.3 Metric results of TEXAI



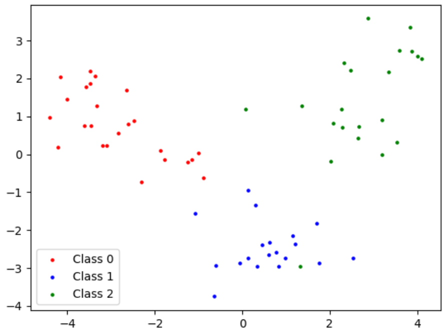


Figure 5.8 ANN prediction plots(Wine Quality) Figure 5.9 TEXAI prediction plots(Wine Quality)

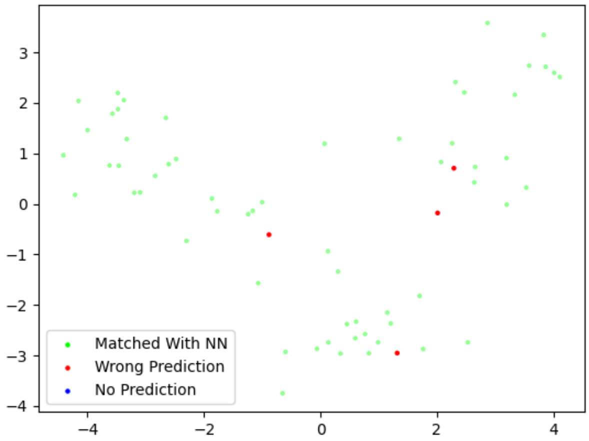


Figure 5.10 ANN vs TEXAI predictions(Wine Quality)

In order to carry out this evaluation, we extracted all the observations that satisfied rule from the test data and subsequently predicted their corresponding target classes using an Artificial Neural Network (ANN) model. The degree of correspondence between the target class of the rule and the predictions made by the ANN were then determined as a percentage. As an example, the observations and ANN predictions that followed a specific rule which obtained from abalone dataset are depicted in figure 5.11. Notably, out of 58 observations that adhered to the specified rule, 55 of them exhibited agreement between the ANN and the rule. This corresponded to a compliance rate of approximately 95%. It is important to consider the possibility that the differences in predictions may also arise due to the "overfitting" issue of ANN. Out of the 203 rules obtained from the Abalone dataset, a total of 153 and, similarly, out of the 219 rules obtained from the wine quality dataset, 162 of them demonstrate a perfect match with the predictions generated by the Artificial Neural Network (ANN) at a 100% level of accuracy. The average compliance rate for all rules generated within the Abalone dataset was estimated to be 97%, while that of the wine quality dataset was 98%. Therefore, we can conclude that each rule is fairly successful in recognizing a specific class.

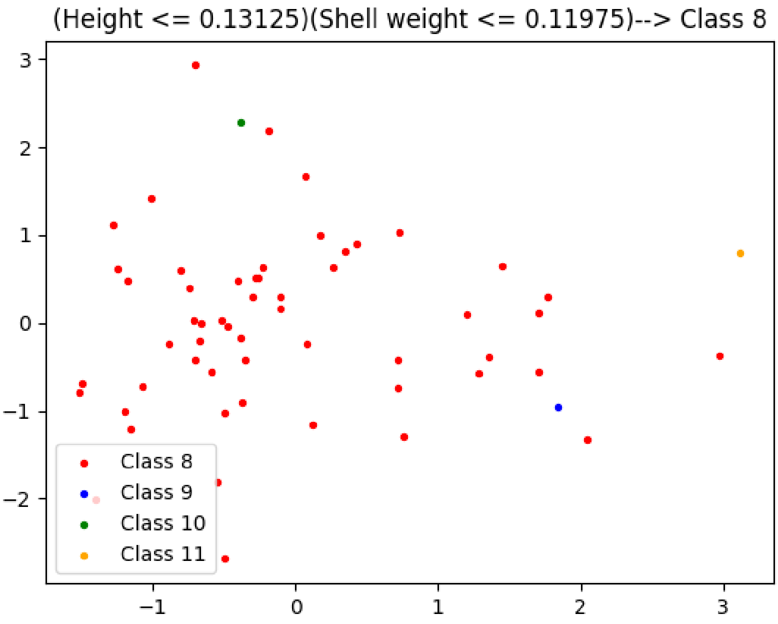


Figure 5.11 ANN predictions for observations which fits one rule (Abalone)

Moreover, the TEXAI framework we have established enables us to identify which features hold greater significance in the model's decision-making process. Hence, TEXAI can also function as a feature importance extraction method. Figure 5.12 shows the number of rules containing each feature for wine quality dataset. For example, in the ”Wine Quality” dataset, where we obtained 219 rules, more than half of the rules contain the color_intensity property. Based on this observation, we can infer that the feature of color_intensity holds greater significance in the model's decision-making process compared to other features.

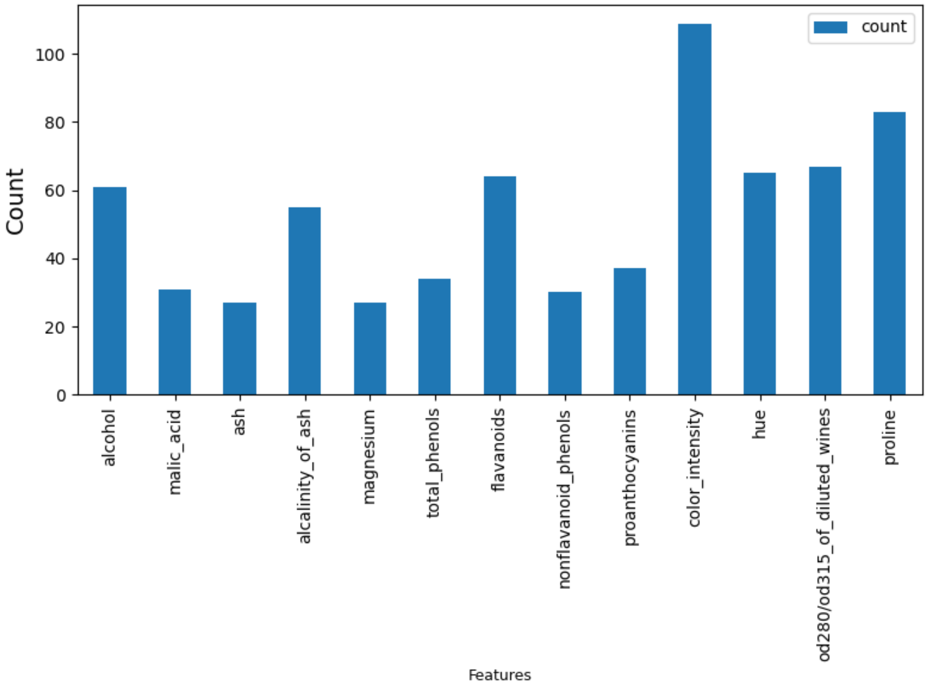


Figure 5.12 Feature importance (Wine Quality)

6. CONCLUSION

The concept of Explainable Artificial Intelligence (XAI) has become a noteworthy area of focus in recent times, mainly because of the surge in the utilization of intricate black box models, such as deep neural networks, in different fields. XAI aims to provide interpretability and transparency to these models, enabling humans to understand the decision-making processes and reasoning behind the models' predictions. In addition, the development of XAI techniques is essential in ensuring ethical and trustworthy AI systems. XAI can enable the detection and mitigation of biased or discriminatory decisions made by black box models, which is critical in domains such as healthcare and finance.

While the development of XAI techniques is still in its infancy, there have been notable advancements in the development of XAI techniques in recent times, including the proposal of various XAI methods, such as LIME, SHAP, and our proposed Tree Ensemble for Explainable Artificial Intelligence (TEXAI). These methods offer varying degrees of interpretability and transparency, with trade-offs in terms of computational complexity and accuracy.

This thesis proposes a novel ensemble model of decision tree-based rules, coined as "Tree Ensemble for Explainable Artificial Intelligence (TEXAI)" with the primary objective of providing insights into the decision-making processes of black box models, such as neural networks. The model takes a dataset and a black box model as inputs, generating a rule set that endeavors to describe the decision boundaries of the black box model. The model demonstrates a remarkable degree of efficacy, as evidenced by the metric results presented in the preceding section. Additionally, we enabled the model to be adjusted through the use of parameters in the filtering process of the rules, thereby allowing for its tunability. By implementing more stringent rule filters, one can augment the accuracy of the estimated regions; however, this also has the potential to introduce areas where no predictions are made. Moreover, it is worth noting that the effectiveness of the rule set may be constrained in scenarios where there exists a prominent non-linearity within the decision-making process. Notwithstanding, we contend that our approach offers several benefits over other state-of-the-art XAI models, such as SHAP and LIME, in terms of performance and simplicity. We firmly believe that our approach will make a positive contribution to the field of explainable artificial intelligence and we hope that it will facilitate the responsible and ethical deployment of AI systems in various domains.

In the future, we want to look into different ways of increasing both prediction power and accuracy of the TEXAI. And each rule that is created in the process of modeling defines separate regions within the hyperplane. Through further refinement, these separate regions can be consolidated into a single contiguous region, allowing for the identification of clear regions that effectively delineate each target class.

In conclusion, this thesis proposes a new and innovative ensemble model, TEXAI, that provides interpretability and transparency to black box models, such as neural networks, thereby enabling a better understanding of the decision-making process behind the model's predictions. By demonstrating the presented metric results, we have shown its efficacy. However, it is worth noting that the effectiveness of the rule set may be limited in scenarios where there exists a prominent non-linearity within the decision-making process. Despite this limitation, we believe that our approach offers several advantages over other XAI models in terms of performance and simplicity. We hope that this model will contribute positively to the field of explainable artificial intelligence, enabling the responsible and ethical deployment of AI systems in various domains.

REFERENCES

[1] M. Turek. DARPA-Explainable Artificial Intelligence(XAI) Program. `https://www.darpa.mil/program/explainable-artificial-intelligence`, **2017**.

[2] David Gunning. Explainable artificial intelligence (xai). Technical report, Defense Advanced Research Projects Agency (DARPA), Arlington, VA (United States), **2017**.

[3] Christoph Molnar. Interpretable machine learning: A guide for making black box models explainable. *The 2019 Springer*, **2021**.

[4] Alejandro B Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Andres Bennetot, Siham Tabik, Alberto Barbado, and Ana Garcia-Serrano. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information Fusion*, 58:82–115, **2020**.

[5] Kyosuke Hirasawa, Noriyuki Takahashi, Tomoya Nishino, Lucila Ohno-Machado, and Hiroshi Fujita. Explainable artificial intelligence for medical diagnosis: Improving the performance of deep learning models. *PloS one*, 15(11):e0241875, **2020**.

[6] Jeffrey Dastin. Amazon scraps secret AI recruiting tool that showed bias against women. *Reuters*, **2018**.

[7] Xiaowei Liu, Zhongxuan Liu, and Sujuan Zhou. How to prevent AI from discriminating against females: A study based on Amazon recruiting algorithm. *Journal of Big Data*, 6(1):73, **2019**.

[8] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. “why should i trust you?”: Explaining the predictions of any classifier. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, **2016**.

[9] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Journal of Machine Learning Research*, 18(1):1–54, **2017**.

[10] S. J. Russell and P. Norvig. *Artificial intelligence: A modern approach*. Pearson Education, **2010**.

[11] Tom M Mitchell. *Machine Learning*. McGraw Hill, **1997**.

[12] Ethem Alpaydin. *Introduction to Machine Learning*. MIT press, **2010**.

[13] Christopher M Bishop. *Pattern Recognition and Machine Learning*. Springer, **2006**.

[14] Simon Haykin. *Neural networks: A comprehensive foundation*. Macmillan Publishing, **1994**.

[15] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, **2015**.

[16] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, **2016**.

[17] Christopher M Bishop. *Neural networks for pattern recognition*. Oxford university press, **1995**.

[18] I. H. Witten, E. Frank, and M. A. Hall. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, **2016**.

[19] Leo Breiman. Bagging predictors. *Machine learning*, 24(2):123–140, **1996**.

[20] Thomas G Dietterich. Ensemble methods in machine learning. In *Multiple classifier systems*, pages 1–15. Springer, **2000**.

[21] Yoav Freund and Robert E Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1):119–139, **1997**.

[22] Harris Drucker, Donghui Wu, and Vladimir Vapnik. Support vector machines for spam categorization. *IEEE Transactions on Neural Networks*, 10(5):1048–1054, **1999**.

[23] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, **2001**.

[24] Jerome H Friedman. Greedy function approximation: A gradient boosting machine. *Annals of statistics*, pages 1189–1232, **2001**.

[25] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794, **2016**.

[26] Robi Polikar. Ensemble based systems in decision making. *IEEE Circuits and Systems Magazine*, 6(3):21–45, **2006**.

[27] Defense Advanced Research Projects Agency. Explainable artificial intelligence (xai). `https://www.darpa.mil/program/explainable-artificial-intelligence`, **2019**.

[28] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Andres Bennetot, Siham Tabik, Alberto Barbado, Ana Garcia-Serrano, Sergio Gil-Lopez, Delia Molina, Rami Benjamins, et al. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information Fusion*, 68:28–62, **2021**.

[29] Bryce Goodman and Seth Flaxman. European union regulations on algorithmic decision-making and a “right to explanation”. *AI Magazine*, 38(3):50–57, **2016**.

[30] Julian Angwin, Jeff Larson, Chris Levy, and Jennifer Singer. Machine learning and data mining: Introduction to principles and algorithms. In Luis Torgo and Paulo Ribeiro, editors, *Data Mining with R: Learning with Case Studies*, pages 75–97. CRC Press, Boca Raton, FL, **2016**.

[31] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. Pixel-wise explanations of non-linear classifier decisions via gradient-based localization. In *International Conference on Computer Vision (ICCV)*, pages 1630–1638. **2015**.

[32] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3319–3328. JMLR. org, **2017**.

[33] Leo Breiman, Jerome H Friedman, Richard A Olshen, and Charles J Stone. *Classification and Regression Trees*. CRC press, **1984**.

[34] J Ross Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, **1993**.

[35] Alex Goldstein, Adam Kapelner, Justin Bleich, and Emily Pitkin. Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation. *Journal of Computational and Graphical Statistics*, 24(1):44–65, **2015**.

[36] Dongdong Wang, Zongming Li, Yanpeng Cai, and Hao Wu. A surrogate modeling approach for global sensitivity analysis of computer models. *Environmental Modelling & Software*, 79:216–228, **2016**.

[37] Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *Harvard Journal of Law & Technology*, 31(2):841–887, **2017**.

[38] Been Kim, Cynthia Rudin, Julie A Shah, and Alexander J Smola. The bayesian case model: A generative approach for case-based reasoning and prototype classification. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1952–1960. **2014**.

[39] Xinyun Chen, Xiapu Zhang, Bo Xie, and Yang Liu. Provenance-based explanation of black box models. In *2019 IEEE 35th International Conference on Data Engineering (ICDE)*, pages 1010–1021. IEEE, **2019**.

[40] Mohammad Rezwanul Huq and Joy Bose. Provenance in machine learning: a systematic review of recent research. *Knowledge and Information Systems*, 56(2):245–295, **2018**.

[41] Benjamin Letham, Cynthia Rudin, Tyler H McCormick, and David Madigan. Interpretable classifiers using rules and bayesian analysis: Building a better stroke prediction model. *Annals of Applied Statistics*, 9(3):1350–1371, **2015**.

[42] Himabindu Lakkaraju, Sebastian H Bach, and Jure Leskovec. Interpretable decision sets: A joint framework for description and prediction. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1675–1684. ACM, **2016**.

[43] J Ross Quinlan. Induction of decision trees. *Machine learning*, 1(1):81–106, **1986**.

[44] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 618–626. IEEE, **2017**.

[45] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, **2014**.

[46] Leila Arras, Franziska Horn, Gr'egoire Montavon, Klaus-Robert M"uller, and Wojciech Samek. Explaining predictions of non-linear classifiers in nlp. *arXiv preprint arXiv:1705.01054*, **2017**.

[47] Jiwei Li, Will Monroe, Alan Ritter, Michel Galley, Jianfeng Gao, and Dan Jurafsky. Understanding neural networks through representation erasure. *arXiv preprint arXiv:1612.08220*, **2016**.

[48] Sebastian Lapuschkin, Alexander Binder, Gr'egoire Montavon, Klaus-Robert M"uller, and Wojciech Samek. Analyzing classifiers: Fisher vectors and deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2912–2920. **2016**.

[49] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st Conference on Neural Information Processing Systems*, pages 6000–6010. **2017**.

[50] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, **2014**.

[51] Felix A Gers, Jürgen Schmidhuber, and Fred Cummins. Learning to forget: Continual prediction with lstm. *Neural Computation*, 12(10):2451–2471, **2000**.

[52] Haşim Sak, Andrew Senior, and François Beaufays. Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition. *arXiv preprint arXiv:1402.1128*, **2014**.

[53] Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, **2014**.

[54] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems*, pages 3104–3112. **2014**.

[55] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489. **2016**.

[56] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *Proceedings of the International Conference on Learning Representations*. **2016**.

[57] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *Proceedings of the 34th International Conference on Machine Learning*, pages 1263–1272. **2017**.

[58] Rex Ying, Ruining He, Kaifeng Chen, Phitchaya Eksombatchai, William L Hamilton, and Jure Leskovec. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 974–983. **2018**.

[59] Zachary C Lipton. The mythos of model interpretability. In *Proceedings of the 2018 ICML Workshop on Human Interpretability in Machine Learning*, pages 1–8. **2018**.

[60] Maximilian Alber, Sebastian Lapuschkin, Philipp Seegerer, Philipp Häusser, Wojciech Samek, and Klaus-Robert Müller. innvestigate neural networks. *Journal of Machine Learning Research*, 20(93):1–8, **2019**.

[61] Marco Tulio Ribeiro, Sameer Singh, and Carlos 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*, pages 1135–1144. ACM, **2016**.

[62] Tao Lei, Regina Barzilay, and Tommi Jaakkola. Rationalizing neural predictions. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 107–117. **2016**.

[63] Ying Zhang, Yijun Song, Jinghui Ren, and Xin Chen. Quantifying the effect of explanation on human trust in ai systems. *ACM Transactions on Interactive Intelligent Systems*, 12(1):1–24, **2022**.

[64] Yong Li, Jing Liu, Furu Wei, Sujian Zhang, and Houfeng Wang. A benchmark study of different explanation methods for text classification. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 179–190. Association for Computational Linguistics, **2018**.

[65] Jasleen Kaur, Parneet Bhatia, Inderjit Kaur, and Dinesh Kumar Singh. Xai-based breast cancer recurrence prediction model: An explainability study. *Journal of Biomedical Informatics*, 109:103520, **2020**.

[66] Brianne Kompa, David Kompa, and Animesh Acharjee. Explaining the predictions of a machine learning model in critical care: a case study on mortality prediction. *Journal of Biomedical Informatics*, 103:103389, **2020**.

[67] Yasir Qureshi and Volodymyr Kuleshov. Explainable ai in finance: An application of lime for credit risk assessment. *arXiv preprint arXiv:1911.07958*, **2019**.

[68] José M Álvarez, Mathieu Salzmann, and Svetlana Lazebnik. Towards robust interpretability with self-explaining neural networks. In *Advances in Neural Information Processing Systems*, pages 7775–7784. **2018**.

[69] Lloyd S Shapley. A value for n-person games. *Contributions to the Theory of Games*, 2(28):307–317, **1953**.

[70] Alvin Rajkomar, Eyal Oren, Kai Chen, Andrew M Dai, Nemit Hajaj, Moritz Hardt, Peter J Liu, Xiaobing Liu, Jake Marcus, Mimi Sun, et al. Scalable and

accurate deep learning with electronic health records. *npj Digital Medicine*, 1(1):1–10, **2018**.

[71] James Bailey, Jonathan Borwein, Harsh Kapoor, and Bryan Wood. Evaluating the interpretability of deep learning models: A human-centered approach. *arXiv preprint arXiv:1806.00069*, **2018**.

[72] Di Jin, Kun Xu, Yong Zhang, Yunqi Li, Long Wang, and Ping Li. Analyzing and improving interpretability of neural machine translation by shap-based methods. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:1521–1535, **2021**.

[73] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830, **2011**.

[74] Abalone data set. `https://archive.ics.uci.edu/ml/datasets/abalone`.

[75] Wine quality data set. `https://archive.ics.uci.edu/ml/datasets/wine+quality`.