A Comparative Survey of SHAP and LIME: Explaining Machine Learning Models for Transparent AI

Sudipta Dey[1] and Tathagata Roy Chowdhury[2]
[1] *MSc Artificial Intelligence, School of Computing and Engineering, University of Huddersfield*
[2] *Assistant Professor, Department of Computer Science and Engineering, Techno Engineering college Banipur*

Abstract: ***Artificial Intelligence (AI) and Machine Learning (ML) have increasingly become central to decision-making in critical domains such as healthcare, finance, and autonomous systems. However, their complexity has rendered many models opaque, often referred to as "black-box" models, making it difficult for users to understand or trust the decisions made. Explainable AI (XAI) seeks to address this by providing transparency in model decision-making processes. Two prominent XAI techniques, SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-agnostic Explanations), are widely used to interpret complex models. This paper presents a comparative analysis of SHAP and LIME, examining their theoretical foundations, strengths, limitations, and applications. SHAP is rooted in cooperative game theory and offers global interpretability with consistent and reliable explanations, whereas LIME provides efficient, local explanations suited for real-time applications. The paper further discusses the challenges in applying these methods, particularly around scalability and real-time decision-making, and highlights potential future research directions, including hybrid models that combine the strengths of both SHAP and LIME.***

Keywords: ***Explainable AI (XAI), Machine Learning Interpretability, SHAP (Shapley Additive Explanations), LIME (Local Interpretable Model-agnostic Explanations), Black-box Models, Model Transparency, Feature Attribution, Model-agnostic Explanations, Cooperative Game Theory, Local Explanations, Global Interpretability, Model Explainability, Bias Detection, Trust in AI, Ethical AI, Algorithm Transparency, AI Accountability, Model Evaluation, Hybrid Explanatory Models, Computational Complexity in XAI.***

INTRODUCTION

Artificial Intelligence (AI) and Machine Learning (ML) continue to drive innovation and transformation across industries, revolutionizing fields such as healthcare, finance, marketing, autonomous systems, and more. These fields now rely on sophisticated machine learning models for tasks like medical diagnosis, fraud detection, customer recommendations, and risk assessment. However, as these models become more advanced and complex, they also become less interpretable. In many cases, even the data scientists and engineers who design these models may not fully understand how they reach their predictions.

This opacity has led to the rise of the term “black-box models,” referring to models whose internal workings are not accessible to human understanding. While these models might be highly accurate and capable of handling large amounts of data, their lack of transparency poses a significant challenge. When models are deployed in high-stakes environments—such as healthcare, criminal justice, or autonomous driving—stakeholders need to understand the rationale behind their decisions. Without this understanding, the trust and reliability of AI systems are compromised.

Explainable AI (XAI) seeks to address this issue by making machine learning models more transparent and understandable to human users. SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-agnostic Explanations) are two of the most widely adopted XAI methods. SHAP leverages game theory to explain the contribution of individual features to model predictions, providing both global and local interpretability. LIME approximates model behavior locally by perturbing the input data and generating simple, interpretable models for specific instances.

This paper presents a detailed comparison of SHAP and LIME, analyzing their strengths, limitations, computational efficiency, and practical applications. Additionally, it examines the ethical implications of explainability in AI, the challenges of implementing XAI at scale, and potential future research directions, including hybrid approaches that combine the best of both methods.

Background on Explainable AI (XAI)

The Growing Need for Explainable AI

AI models are rapidly being integrated into high-stakes domains such as healthcare, finance, legal systems, and autonomous systems. In these domains, the decisions made by machine learning models can have life-altering consequences. However, when these models act as black boxes—generating predictions without providing clear reasoning—their trustworthiness is severely diminished. For example, in healthcare, doctors may hesitate to follow treatment recommendations made by an AI system if they cannot understand the factors driving those recommendations. Similarly, in finance, opaque models may make credit decisions that disadvantage certain demographic groups without revealing why or how.

The growing need for transparency in AI systems has also been driven by increasing regulatory pressure. For instance, the European Union's General Data Protection Regulation (GDPR) enshrines the "right to explanation," meaning that individuals affected by AI-driven decisions must be provided with an explanation of how those decisions were made. This regulatory requirement is part of a larger push towards ensuring that AI systems are accountable and fair.

In addition to legal and regulatory motivations, ethical concerns are at the forefront of discussions about XAI. Unexplained AI decisions could perpetuate biases, deepen inequalities, or lead to other harmful consequences. Thus, explainability is not just a technical necessity; it is a critical component of ethical AI deployment.

Key Challenges in Developing Explainable AI Models

Achieving true explainability in AI is a complex and multifaceted challenge. Some of the major obstacles include:

1. Balancing Complexity and Interpretability: One of the most persistent challenges in AI is the trade-off between complexity and interpretability. Simple models, such as decision trees or logistic regression, are easy to interpret but may not capture complex patterns in the data. Conversely, more powerful models, such as deep neural networks or ensemble methods, can achieve state-of-the-art performance but are difficult, if not impossible, to interpret intuitively. XAI techniques like SHAP and LIME aim to make these complex models interpretable without sacrificing too much performance.
2. Uncovering and Mitigating Bias: Machine learning models often inherit biases from the datasets on which they are trained. These biases can lead to discriminatory outcomes, particularly in domains like hiring, lending, or criminal justice, where model decisions can disproportionately affect marginalized groups. Without explainability, these biases may go unnoticed, perpetuating unfair and unethical practices. XAI methods can help expose bias by highlighting the features that most influence a model's predictions.
3. Trust and Accountability: Trust in AI systems is essential for their widespread adoption. Users, regulators, and other stakeholders must trust that the AI models are making fair, ethical, and reliable decisions. However, trust cannot be established if users cannot understand or verify the reasoning behind the model's decisions. XAI is crucial for building this trust by offering clear and interpretable explanations for the model's behavior.
4. Ethical Concerns and Fairness: With AI systems increasingly being used in decision-making processes that impact people's lives, there are ethical concerns around fairness and equity. How can we ensure that these systems do not perpetuate existing societal biases or unfairly discriminate against certain groups? XAI methods can play a role in detecting biases in models, but further research is needed to fully address these ethical concerns.
5. Regulatory Compliance and Legal Considerations: As AI models become more integrated into critical decision-making processes, legal frameworks such as the GDPR and the Algorithmic Accountability Act have mandated transparency and explainability. Companies that develop and deploy AI systems must ensure compliance with these regulations, or they risk legal and financial consequences. XAI methods like SHAP and LIME offer potential solutions, but they also raise new questions about how explanations are communicated to users and how comprehensible they are for non-technical audiences.

Overview of XAI Techniques

Several techniques have been developed to make AI models more interpretable. In addition to SHAP and LIME, other XAI methods include:

- Counterfactual Explanations: Counterfactual explanations provide insights into what changes to the input data would lead to a different outcome. For example, a counterfactual explanation for a loan rejection might indicate that if the applicant's income were $5,000 higher, the loan would have been approved. Counterfactuals help users understand how sensitive a model's predictions are to changes in input features.
- Saliency Maps: In image classification models, saliency maps highlight the regions of an image that are most influential in the model's decision. This technique is commonly used in deep learning models, such as convolutional neural networks (CNNs), to provide visual explanations of the model's focus.
- Partial Dependence Plots (PDPs): PDPs show the effect of a single feature on the predicted outcome, holding all other features constant. This technique provides a global view of the relationship between features and predictions, helping users understand how changes in feature values affect the model's output.
- Feature Importance Scores: Feature importance methods assign a score to each feature, indicating its contribution to the model's overall performance. This technique is commonly used in tree-based models like random forests and gradient boosting machines, where feature importance scores can be directly derived from the structure of the trees.

While these techniques are useful, SHAP and LIME are particularly versatile and widely applicable. They have become the go-to methods for practitioners looking to make complex machine learning models more transparent.

SHAP: SHapley Additive exPlanations

Origins and Theoretical Foundations

SHAP is grounded in cooperative game theory and builds on the concept of Shapley values, introduced by Lloyd Shapley in the 1950s. In cooperative game theory, Shapley values represent a way to fairly distribute the payout of a game among the players based on their individual contributions. In the context of machine learning, SHAP applies this concept by treating input features as "players" and the model's prediction as the "payout." The Shapley value for each feature quantifies its contribution to the final prediction.

What sets SHAP apart from other XAI techniques is its strong theoretical foundation, which guarantees that the attributions provided are fair and consistent. Specifically, SHAP satisfies three key properties:

1. Local Accuracy: The explanation provided by SHAP is accurate for each individual prediction. The sum of the SHAP values equals the model's output, ensuring that the explanation is faithful to the model's behavior.
2. Consistency: If a model changes in such a way that the contribution of a feature increases, SHAP guarantees that the importance assigned to that feature will not decrease. This consistency ensures that the explanations generated by SHAP are stable and reliable.
3. Additivity: SHAP ensures that the contribution of each feature is fairly distributed, meaning that the sum of all feature contributions equals the model's prediction. This additivity guarantees that SHAP explanations are globally consistent and interpretable.

How SHAP Works

SHAP decomposes a model's prediction into the contributions of individual features by calculating Shapley values for each feature. These values are computed by evaluating the effect of each feature in combination with all possible subsets of other features. The resulting Shapley value for each feature represents its marginal contribution to the model's output.

For example, in a model predicting house prices, SHAP might reveal that the size of the house, its location, and the number of bedrooms each contribute a specific amount to the final predicted price. By summing these contributions, the model's output can be broken down into interpretable parts, providing clear insight into how the prediction was made.

To efficiently calculate Shapley values, SHAP offers different implementations tailored to specific types of models. These include:

- Tree SHAP: This version is optimized for tree-based models such as decision trees, random forests, and gradient-boosting machines. Tree SHAP takes advantage of the tree structure to efficiently compute Shapley values, making it more computationally feasible than other SHAP variants.
- Kernel SHAP: Kernel SHAP is a model-agnostic implementation that can be applied to any

type of machine learning model. It uses a kernel-based approach to approximate Shapley values, making it suitable for models where exact computation is not feasible.

- Deep SHAP: Deep SHAP is designed for deep learning models, particularly neural networks. It combines the principles of SHAP with gradient-based techniques like DeepLIFT to approximate Shapley values for deep models.

Strengths of SHAP

1. Fairness and Consistency: SHAP guarantees that feature attributions are fair and consistent across all predictions. This property is particularly important in applications where accountability is critical, such as healthcare, finance, and legal decision-making.
2. Global and Local Interpretability: SHAP provides both local and global interpretability, allowing users to understand individual predictions as well as the overall behavior of the model. This flexibility makes SHAP a valuable tool for a wide range of use cases, from explaining specific loan approvals to understanding the general factors driving credit scoring models.
3. Handling Feature Interactions: One of SHAP's key strengths is its ability to model interactions between features. By evaluating all possible feature combinations, SHAP can capture how different features work together to influence a prediction. This capability is particularly useful in domains like genetics or personalized medicine, where feature interactions play a significant role in determining outcomes.
4. Unified Framework for Explaining Models: SHAP provides a unified framework for interpreting different types of machine learning models. Whether the model is a random forest, a neural network, or a support vector machine, SHAP can generate consistent, theoretically sound explanations for its predictions.

Limitations of SHAP

1. Computational Complexity: One of the main drawbacks of SHAP is its computational cost. Calculating exact Shapley values requires evaluating all possible feature subsets, which can be computationally expensive for large datasets or models with many features. Even with optimizations like Tree SHAP and Kernel SHAP, the method can be too slow for real-time applications.
2. Scalability Issues: SHAP's computational complexity makes it challenging to scale to large datasets or real-time systems. While SHAP works well for offline analysis, it may not be practical for applications where quick decisions are required, such as fraud detection or autonomous driving.
3. Applicability to Non-additive Models: While SHAP performs well with additive models, its complexity increases when applied to models that are non-additive or highly non-linear, such as deep neural networks. In these cases, the approximations provided by SHAP may be less reliable or harder to interpret.

LIME: Local Interpretable Model-agnostic Explanations

Overview of LIME

LIME is a model-agnostic explanation technique designed to provide local interpretability for individual predictions. Unlike SHAP, which generates explanations based on the model's global behavior, LIME focuses on explaining the model's behavior in the vicinity of a specific instance. It does this by perturbing the input data around the instance of interest and observing how the model's predictions change. Based on these perturbations, LIME fits a simpler, interpretable surrogate model (often a linear model) to approximate the complex model's decision boundary.

LIME's strength lies in its ability to provide quick, local explanations for individual predictions, making it particularly useful for applications where users need to understand specific outcomes.

How LIME Works

LIME works by perturbing the input features of an instance and observing how the model's predictions change. These perturbed data points are then used to train a simpler, interpretable model (usually a linear model) that approximates the complex model's decision boundary in the local region. The surrogate model provides an explanation for the specific prediction by indicating which features contributed most to the model's decision.

For example, in a text classification model, LIME can generate explanations by perturbing individual words or phrases and observing how the model's prediction changes. By fitting a linear model to the perturbed data, LIME can provide a local explanation that indicates which words were most influential in

determining the model's classification of the document.

LIME can be applied to a variety of data types, including:

- Text: In text classification tasks, LIME can highlight important words or phrases that influenced the model's prediction.
- Tabular Data: For structured datasets, LIME can indicate which features were most important in making a specific prediction.
- Images: In image classification, LIME can highlight the regions of an image that were most important to the model's decision.

Strengths of LIME

1. Model-Agnostic: LIME is a model-agnostic explanation technique, meaning it can be applied to any machine learning model, regardless of its architecture. This flexibility makes LIME suitable for a wide range of applications, from simple linear models to deep neural networks.
2. Local Interpretability: LIME excels at providing local explanations for individual predictions. This localized approach is especially useful in applications where users need to understand specific decisions, such as why a loan application was rejected or why a medical diagnosis was made.
3. Computational Efficiency: LIME is more computationally efficient than SHAP because it focuses on generating local explanations rather than explaining the entire model. By perturbing the input data and fitting a simpler surrogate model, LIME can quickly provide explanations, even for complex models.

Limitations of LIME

1. Locality and Variability: One of LIME's main limitations is its focus on local explanations. While this is useful for understanding individual predictions, the explanations may not generalize well to other instances. As a result, LIME's explanations can vary depending on the instance being explained, making it less consistent than SHAP.
2. Parameter Sensitivity: LIME's results can be sensitive to the choice of parameters, such as the number of features included in the explanation or the kernel width for perturbations. This can lead to variability in the generated explanations, potentially confusing users.
3. Limited Handling of Feature Interactions: LIME assumes that features are independent within its local surrogate models, which can lead to inaccurate explanations when features interact heavily. This limitation makes LIME less effective in domains where feature interactions are important, such as genetics or personalized healthcare.

Comparative Analysis: SHAP vs. LIME

Accuracy and Reliability

SHAP provides more consistent and reliable explanations compared to LIME, largely due to its foundation in cooperative game theory. By calculating Shapley values for all possible feature subsets, SHAP ensures that feature attributions are fair and accurate. This makes SHAP particularly suitable for high-stakes applications where explanation consistency is critical, such as healthcare or financial decision-making.

LIME, in contrast, generates approximate explanations based on local surrogate models. While LIME is computationally more efficient, its reliance on local surrogates can result in variability in the explanations, particularly when applied to highly non-linear models. As a result, LIME's explanations are often less reliable than those generated by SHAP.

Interpretability

Both SHAP and LIME offer interpretable outputs, but the scope of their explanations differs. SHAP provides global interpretability by offering insights into how features influence predictions across the entire dataset. This global perspective is useful for understanding the overall behavior of a model, especially in applications like credit scoring or fraud detection, where stakeholders need to trust the model's general decision-making process.

LIME, on the other hand, focuses on local interpretability, providing explanations for individual predictions. This localized approach is particularly useful in contexts where understanding specific decisions is more important than understanding the overall model. For example, in personalized healthcare, LIME can help doctors understand why a particular treatment was recommended for an individual patient.

Computational Efficiency

SHAP's main drawback is its computational complexity. Calculating Shapley values for all possible feature combinations requires significant computational resources, making SHAP impractical for real-time applications or models with a large number of features.

In contrast, LIME is more computationally efficient because it generates explanations based on local approximations. LIME's ability to quickly produce local explanations makes it well-suited for real-time applications, although this efficiency comes at the cost of accuracy and consistency.

Applicability

LIME's model-agnostic nature makes it highly versatile. It can be applied to any type of machine learning model, including text, image, and tabular data. This versatility makes LIME a valuable tool for interpreting models across various domains and applications.

While SHAP is also versatile, it requires different implementations for different types of models. For example, Tree SHAP is specifically designed for tree-based models, while Deep SHAP is tailored for deep learning models. This complexity can make SHAP more challenging to apply in certain contexts, particularly for non-expert users.

Global vs. Local Explanations

One of the key distinctions between SHAP and LIME is their focus on global versus local explanations. SHAP excels at providing global interpretability, making it ideal for understanding how features influence predictions across the entire dataset. This global perspective is valuable in applications like credit scoring, where understanding the overall behavior of the model is critical for ensuring fairness and accountability.

LIME, by contrast, focuses on local interpretability, explaining why the model made a specific prediction for a given instance. This localized approach is particularly useful in personalized medicine, where doctors need to understand why a particular treatment was recommended for an individual patient.

Handling Feature Interactions

SHAP's ability to handle feature interactions is one of its key strengths. By calculating Shapley values for all possible feature combinations, SHAP can model how features interact to influence predictions. This makes SHAP particularly useful in domains where feature interactions are important, such as genetics or personalized healthcare.

LIME, on the other hand, assumes that features are independent within its local surrogate models. This assumption can lead to inaccurate explanations when features interact heavily, as LIME does not capture these interactions as effectively as SHAP.

Applications of SHAP and LIME

Healthcare

Machine learning is increasingly being used in healthcare to predict patient outcomes, assist in diagnosing diseases, and recommend personalized treatments. However, the complexity of these models often makes them difficult for healthcare professionals to trust and interpret. Both SHAP and LIME have proven valuable tools for explaining healthcare models and improving their transparency.

For example, SHAP has been used to explain complex predictive models that estimate patient mortality rates or the likelihood of hospital readmission. By providing clear, consistent explanations, SHAP helps healthcare providers understand which factors, such as age, medical history, or lab results, contributed most to the prediction. This transparency is crucial for building trust in AI-driven healthcare systems.

LIME has also been applied in healthcare, particularly for explaining individual predictions. In personalized medicine, for example, LIME can explain why a specific treatment was recommended for a patient, helping doctors understand the key factors that influenced the decision. This localized approach is especially useful when doctors need to explain the model's recommendations to patients in clear, understandable terms.

Finance

In the financial sector, transparency and fairness are critical, particularly in areas like credit scoring, fraud detection, and algorithmic trading. Both SHAP and LIME have been used to improve the interpretability of machine learning models in these domains.

SHAP has been used to explain credit-scoring models by showing which features contributed most to a loan approval or rejection. By providing global explanations, SHAP allows lenders to ensure that the model's decisions are fair and unbiased, which is

essential for meeting regulatory requirements. For example, SHAP might reveal that an applicant's income, credit history, and employment status were the most important factors in determining loan eligibility.

LIME has also been applied to fraud detection systems in finance. For instance, LIME can explain why a specific transaction was flagged as fraudulent by highlighting the features that contributed most to the model's decision, such as transaction amount, time, and location. This helps investigators focus on the most important aspects of the transaction and make more informed decisions during their investigations.

Natural Language Processing

Natural language processing (NLP) models, which analyze and generate human language, are another area where explainability is crucial. SHAP and LIME have both been used to interpret text classification models, providing insights into how models make predictions in sentiment analysis, spam detection, and document classification tasks.

In sentiment analysis, for example, SHAP can provide global explanations by showing which words or phrases generally influence the model's predictions. This allows users to understand how the model behaves across different types of text.

LIME, on the other hand, is useful for explaining individual predictions in NLP tasks. For example, LIME can highlight the specific words or phrases that were most important in determining the classification of a document, helping users understand why the model classified a particular email as spam or why it rated a product review as positive or negative.

Image Classification

Interpreting image classification models is challenging due to the high dimensionality of image data. Both SHAP and LIME have been applied to explain image classification models, helping researchers and practitioners understand how these models make decisions.

LIME has been widely used in image classification tasks to generate visual explanations by highlighting the regions of an image that were most important to the model's decision. For example, in a model used for diagnosing medical images, LIME can highlight the areas of an X-ray or MRI scan that were most influential in the model's diagnosis, helping doctors understand the model's decision-making process.

SHAP, although less commonly used in image classification, can still provide valuable insights. For example, SHAP can be used to explain the contributions of different image features, such as color, texture, or shape, to the model's predictions. This transparency is particularly important in medical imaging, where understanding the rationale behind a diagnosis is crucial for building trust in AI-driven diagnostic systems.

Challenges and Future Directions

Scalability Issues

One of the primary challenges with both SHAP and LIME is their scalability to large datasets and complex models. SHAP, in particular, suffers from high computational costs due to the need to evaluate all possible combinations of input features. This makes SHAP difficult to apply in real-time systems or on large-scale datasets, such as those used in image or video classification.

LIME, while more computationally efficient than SHAP, can also struggle with high-dimensional data, particularly when the number of features is large. Future research may focus on optimizing both SHAP and LIME to handle larger datasets more efficiently without sacrificing interpretability.

Integration with Real-time Systems

For XAI methods to be useful in real-world applications, they must be able to generate explanations in real-time. This remains a challenge for both SHAP and LIME, particularly in domains where quick decision-making is critical, such as autonomous vehicles or financial trading. Future work may focus on reducing the computational complexity of these methods to make them suitable for real-time applications.

Standardized Evaluation Metrics

Another challenge in the field of XAI is the lack of standardized metrics for evaluating the quality of explanations. While SHAP and LIME provide intuitive explanations, there is currently no consensus on how to measure the interpretability and usefulness of these explanations. Future research may focus on developing benchmarks and metrics for assessing the quality of XAI methods across different domains.

Ethical AI and Fairness

As AI models become more widely used in decision-making processes, concerns about fairness and bias have come to the forefront. Both SHAP and LIME can play a role in detecting and mitigating biases in models, but there is still much work to be done in this area. Hybrid approaches that combine the strengths of both methods may offer better solutions for detecting bias and ensuring fairness in AI models.

Hybrid Models

One potential future direction in XAI is the development of hybrid models that combine the strengths of SHAP and LIME. For example, SHAP's ability to provide global interpretability could be combined with LIME's local explanations to create a more comprehensive understanding of a model's behavior. This hybrid approach could be particularly useful in applications where both global and local interpretability are required, such as in healthcare or finance.

CONCLUSION

In this paper, we have provided a comprehensive comparison of SHAP and LIME, two of the most widely used techniques for explaining machine learning models. SHAP offers global interpretability and strong theoretical guarantees, making it particularly well-suited for domains where fairness and accountability are critical. However, SHAP's computational complexity can be a limitation, especially for large datasets or real-time applications.

LIME, on the other hand, excels at providing local, model-agnostic explanations that are efficient and intuitive. While LIME may not offer the same level of consistency as SHAP, its flexibility and efficiency make it a valuable tool for interpreting individual predictions.

Both methods have been successfully applied across a wide range of domains, from healthcare and finance to natural language processing and image classification. However, challenges remain, particularly in terms of scalability, real-time integration, and the evaluation of explanations. As the field of XAI continues to evolve, future research will likely focus on addressing these challenges and further improving the interpretability and transparency of machine learning models.

REFERENCES

[1] Lundberg, S. M., & Lee, S.-I. (2017). A Unified Approach to Interpreting Model Predictions. Advances in Neural Information Processing Systems, 30.

[2] Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). 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 (KDD).

[3] Molnar, C. (2020). Interpretable Machine Learning: A Guide for Making Black Box Models Explainable. Lulu.com.

[4] Shapley, L. S. (1953). A Value for N-person Games. In Contributions to the Theory of Games (Vol. 2, pp. 307-317).

[5] Doshi-Velez, F., & Kim, B. (2017). Towards a Rigorous Science of Interpretable Machine Learning. arXiv preprint arXiv:1702.08608.

[6] Caruana, R., Lou, Y., Gehrke, J., Koch, P., Sturm, M., & Elhadad, N. (2015). Intelligible Models for Healthcare: Predicting Pneumonia Risk and Hospital 30-Day Readmission. Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.

[7] Doran, D., Schulz, S., & Besold, T. R. (2017). What Does Explainable AI Really Mean? A New Conceptualization of Perspectives. arXiv preprint arXiv:1710.00794.

[8] Ribeiro, M. T., Singh, S., & Guestrin, C. (2018). Anchors: High-Precision Model-Agnostic Explanations. Proceedings of the AAAI Conference on Artificial Intelligence.

[9] Kim, B., Shah, J. A., & Doshi-Velez, F. (2015). Mind the Gap: A Generative Approach to Interpretable Feature Selection and Explanation for Black Box Models. Advances in Neural Information Processing Systems, 28.

[10] Chen, J., Song, L., Wainwright, M. J., & Jordan, M. I. (2018). Learning to Explain: An Information-Theoretic Perspective on Model Interpretation. Proceedings of the 35th International Conference on Machine Learning.

[11] Montavon, G., Samek, W., & Müller, K.-R. (2018). Methods for Interpreting and Understanding Deep Neural Networks. Digital Signal Processing, 73, 1-15.

[12] Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., & Pedreschi, D. (2018). A Survey of Methods for Explaining Black Box Models. ACM Computing Surveys (CSUR), 51(5), 93.

[13] Wachter, S., Mittelstadt, B., & Russell, C. (2017). Counterfactual Explanations Without Opening the Black Box: Automated Decisions and the GDPR. Harvard Journal of Law & Technology, 31(2), 841-887.

[14] Mittelstadt, B., Russell, C., & Wachter, S. (2019). Explaining Explanations in AI. Proceedings of the Conference on Fairness, Accountability, and Transparency.

[15] Gilpin, L. H., Bau, D., Yuan, B. Z., Bajwa, A., Specter, M., & Kagal, L. (2018). Explaining Explanations: An Overview of Interpretability of Machine Learning. 2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA).

[16] Fryer, D., & Smith, L. (2021). Detecting and Mitigating Bias in Black Box Models: A Survey of Explainable AI Algorithms. IEEE Access, 9, 138377-138396.

[17] Sundararajan, M., Taly, A., & Yan, Q. (2017). Axiomatic Attribution for Deep Networks. Proceedings of the 34th International Conference on Machine Learning.

[18] Gunning, D. (2017). Explainable Artificial Intelligence (XAI). Defense Advanced Research Projects Agency (DARPA).

[19] Samek, W., Montavon, G., Lapuschkin, S., Anders, C. J., & Müller, K.-R. (2021). Explaining Deep Neural Networks and Beyond: A Review of Methods and Applications. Proceedings of the IEEE, 109(3), 247-278.

[20] Lipton, Z. C. (2018). The Mythos of Model Interpretability. Queue, 16(3), 30-57.