Literature Review: Local Post hoc Explainable Artificial Intelligence Techniques

Sumedha Chugh
PhD21123

As AI technology advances, it becomes increasingly challenging for humans to understand how the algorithm generates a particular outcome. The complex models have transformed into a black box that even the engineers who designed the algorithm cannot explain. Explainable AI (XAI) techniques enable humans to understand the outcomes produced by machine learning algorithms. It plays a vital role in characterizing the model's accuracy, fairness, transparency, and results [1]. XAI includes a range of methods that can be used to increase the explainability of machine learning models. Local, global, post-hoc, and anti-hoc are some of the commonly used ones. Local XAI methods focus on explaining the predictions of the model for a single instance or observation. Some popularly used local explainable AI methods areSHAP (Shapley Additive Explanations), LIME (Local Interpretable Model-Agnostic Explanations) etc. . Global XAI methods explain the general behavior of the ML model. It includes decision trees and linear models, which make predictions based on a linear combination of input features. Posthoc XAI methods are applied after ML model has been trained. These methods include feature importance analysis, which identifies the most important features of a model, and partial dependence plots, which shows the relationship between the features and the model predictions.Anti-hoc XAI methods are used during the model's training process for the model to be interpretable. Anti-hoc methods include regularized models, which add constraints to the training process to encourage explainability and prototype-based learning [2]. This work is a literature survey of Local Post-hoc Explainable AI methods.

One of the first and most commonly used Local Posthoc explanation methods is LIME (Local Interpretable Model Agnostic Explanations). This method can be used irrespective of the model type as it is model agnostic. It approximates the machine learning model locally with a simpler surrogate model, which humans can easily understand. The local surrogate model is trained on a subset of the training data close to the explained instance. If the training data is unavailable, it is done by perturbing the given sample for various features and randomly sampling around it. It calculates the surrogate model using

$h'(x') = \arg\min_{f \in F} L(f, g, \pi_x) + \Omega(g)$

where h represents black-box model we want to explain, F is the set of possible models we can use to create a surrogate model, L measures the difference between the predictions of h and f for an instance x according to a proximity measure $\pi(x)$, and $\Omega(x)$ is a complexity penalty term that discourages overfitting of the surrogate model. It is loss function and predicts the same outcome as the original model and generates an explanation for the prediction

by highlighting the most important features used by the surrogate model for prediction. The importance of each explanation feature is calculated using the feature importance score. The feature importance score measures how much the local surrogate model's prediction changes when a feature is perturbed while keeping the other features constant. The perturbation is done by sampling from a distribution defined for each feature. These feature importances is calculated using the following formula:

$weights_i = \sum_{x' \in \mathcal{L}(x)} \frac{\pi_{x'}(x)}{|\mathcal{L}(x)|} \cdot h(x)_i$

where $weights_i$ is the weight assigned to feature i.L(x) is the set of perturbed instances around x used to create the local surrogate model. $h(x)_i$ is the prediction of the surrogate model for the perturbed instance on feature i [3]. LIME can be useful for generating local explanations, however it has limitations like assuming feature independence, sampling bias, sensitivity to hyperparameters, less reliability, and robustness [4].

Random sampling in LIME can lead to issues like lack of consistency, i.e., generating different feature importance every time it's run and more prone to adversarial attacks. Saini et al. use Gaussian Process Sampling to improve the LIME framework and give more robust and reliable explanations. The paper proposes UnRAvEL, a technique based on active learning which utilizes sampling based on the posterior distribution of the probabilistic locality using Gaussian process regression (GPR). It is driven by uncertainty and generates locally faithful explanations. Active learning is used to select the most informative data points for the model to train on, and a probabilistic model based on Gaussian processes generates explanations for the predictions on those data points. The results show that UnRAvEL outperforms existing post-hoc explainability methods in terms of both explanation quality and computational efficiency [5]. ALIME (Autoencoder-based Approach for Local Interpretability) proposed by Shankaranarayanan, P. et al also focuses on changing the sampling technique of LIME to give better results. This method uses an autoencoder neural network and learns the input data's compressed representation to generate synthetic data samples similar to the local neighborhood of a prediction of interest. The synthetic data samples are then used to fit a surrogate model that explains the prediction of interest. ALIME first trains an autoencoder on the input data to learn a compressed data representation. Once the autoencoder is trained, it generates synthetic data samples similar to the local neighborhood of a prediction of interest. These synthetic data samples are then used to fit an interpretable model, that explains the prediction of interest. One of the advantages of ALIME over LIME is its ability to generate more accurate local explanations as it uses synthetic data samples similar to the local neighborhood of a prediction of interest; it can better capture the complex interactions between features that may be missed by other methods that use simpler models to fit the local neighborhood [6]. Another work that aims to make existing works more trustworthy; creates Bayesian versions of LIME and KernelSHAP, which produce credible intervals for the feature importances, capturing the associated uncertainty. These explanations allow user to draw concrete conclusions about their reliability (e.g., there is a 95 probability that the feature importance lies within the given range) but are also highly consistent and dependable efficiently [7]. Another LIME-based model-agnostic method for generating local feature importance explanations is BayLIME. It introduces Bayesian inference to estimate the posterior distribution of feature importance values, which captures the uncertainty associated with the explanation. BayLIME works by sampling data instances from a neighborhood

of the prediction of interest and fitting a simple interpretable model to the sampled data. The feature importance values are then estimated by computing the expected output of the interpretable model for each feature while integrating over the posterior distribution of the model parameters. The resulting feature importance values are accompanied by credible intervals, which measure uncertainty associated with the explanation. One of the advantages of BayLIME over LIME is its ability to handle correlated features. It is also more robust to noise and outliers, as the Bayesian framework allows for regularization and smoothing of the estimated feature importance values [8]. Another method uses SHAP values from game theory to generate Local Post-hoc explanations. It assigns an importance value to each feature for a specific prediction. Suppose a model that takes an input x and outputs a prediction $f(x)$. SHAP starts by defining a reference value $E[X]$, which is typically the average or expected value of the feature values in the dataset. The difference of the input and the reference value is denoted as $\phi = x - E[X]$. The Shapley values ϕ_i for each feature i can be computed using the following formula:

$\phi_i = \sum_{S \subseteq \{1,2,\ldots,p\} \setminus \{i\}} \frac{|S|!(p-|S|-1)!}{p!}(f(S \cup \{i\}) - f(S))$

where p represents total number of features, S is a subset of features excluding the i^{th} feature, and $f(S)$ represents the model's prediction when only considering the features in S. The formula can be interpreted as follows: ϕ_i is the average marginal contribution of feature i across all possible subsets of features, weighted by the number of ways each subset can be formed. It measures how much each feature contributes to the final prediction compared to the expected value of all features. The Shapley values satisfy properties, like missingness, consistency and local accuracy. Local accuracy means that the sum of Shapley values equals the difference of prediction for the input x and the expected value of the output across all possible inputs. Missingness means that if a feature is unimportant, its Shapley value should be close to zero. Consistency means that if two models agree on the output for a certain input, their Shapley values should also be similar [9]. All the methods discussed till now are supervised. Unsupervised Explainable AI (UXAI) methods aim to explain ML models without needing labeled data or prior knowledge. These methods rely on unsupervised learning techniques to extract patterns and structures from data and use them to provide explanations for model predictions. UXAI methods are particularly useful when labeled data is scarce or expensive to obtain. One example of a UXAI method is Cluster Interpretation (CI), which is based on clustering techniques such as k-means or hierarchical clustering. It groups similar instances together and provides explanations based on the characteristics of each cluster. Another example is Principal Component Analysis (PCA), which identifies the most important features in the data and provides explanations based on their contribution to the model predictions. These methods are more scalable and flexible than supervised methods. However, they may suffer from the same limitations such as difficulty in interpreting the meaning of the extracted patterns and structures. Additionally, UXAI methods may not be able to explain some specific instances or predictions, as they rely on general patterns and structures in the data [10]. Selvaraju et al. proposed a gradient-based approach by GradCAM (Gradient-weighted Class Activation Mapping) that highlights the important regions of an image for classification decisions. The basic idea behind this is to compute the target class's gradient concerning the convolutional layer's feature maps. These gradients are then used to weight the feature maps, giving more importance to the ones contributing more to the classification decision. The resulting weighted feature maps are then

averaged to produce a heat map highlighting the image regions that were most important for the classification decision. This method has several advantages over other explainability techniques. First, it can be applied to any convolutional neural network, making it a general-purpose technique. Second, it is easy to implement and computationally efficient. Finally, it produces visually interpretable heat maps that humans can easily understand. GradCAM has been applied to a wide range of applications, including object detection, medical image analysis, and natural language processing. It has also been extended to other types of neural networks, such as recurrent neural networks and graph neural networks [11]. While GradCAM is a powerful and widely used explainability technique, it also has some limitations that must be considered. It has limited spatial resolution i.e. the spatial resolution of the heat map is limited by the size of the feature maps of the last convolutional layer so it may not be able to capture details that are important for the classification. Along with this, it has limited applicability. It may not apply to other types of neural networks, such as recurrent neural networks or graph neural networks, which are used in other domains [12].

Although these methods have proven extremely important, they face several challenges in providing accurate and reliable explanations for already trained models. One of the main challenges is the lack of guarantee that the explanations provided by posthoc methods are a true reprensentation of the model. This is because these methods generate explanations based on the observed behavior of the model on a limited set of input-output pairs rather than the model's internal mechanisms. [13]. Another challenge is the difficulty of providing consistent and stable explanations because explanations are generated based on perturbations of the input data; the explanations can vary significantly depending on the chosen perturbation method and the number of samples used for the perturbation. Furthermore, post-hoc methods can be computationally expensive, especially for complex models with large datasets. This can limit their scalability and practicality in real-world applications. Addressing these challenges is critical for improving the reliability and usefulness of posthoc XAI methods in real-world applications. Overall, addressing these complex challenges is essential for advancing the field of post-hoc XAI and for realizing its full potential in improving human decision-making and enhancing the trustworthiness of AI systems [14].

References

[1] A. Holzinger, A. Saranti, C. Molnar, P. Biecek, and W. Samek, *Explainable AI Methods - A Brief Overview*, pp. 13–38. Cham: Springer International Publishing, 2022.

[2] P. Lopes, E. Silva, C. Braga, T. Oliveira, and L. Rosado, "Xai systems evaluation: A review of human and computer-centred methods," *Applied Sciences*, vol. 12, no. 19, 2022.

[3] M. T. Ribeiro, S. Singh, and C. Guestrin, ""why should i trust you?": Explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, (New York, NY, USA), p. 1135–1144, Association for Computing Machinery, 2016.

[4] A. Datta, S. Sen, and Y. Zick, "Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems," *2016 IEEE Symposium on Security and Privacy (SP)*, pp. 598–617, 2016.

[5] A. Saini and R. Prasad, "Select wisely and explain: Active learning and probabilistic local post-hoc explainability," in *Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society*, AIES '22, (New York, NY, USA), p. 599–608, Association for Computing Machinery, 2022.

[6] S. M S and D. Runje, *ALIME: Autoencoder Based Approach for Local Interpretability*, pp. 454–463. 10 2019.

[7] D. Slack, A. Hilgard, S. Singh, and H. Lakkaraju, "Reliable post hoc explanations: Modeling uncertainty in explainability," in *Advances in Neural Information Processing Systems* (M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, eds.), vol. 34, pp. 9391–9404, Curran Associates, Inc., 2021.

[8] X. Zhao, X. Huang, V. Robu, and D. Flynn, "Baylime: Bayesian local interpretable model-agnostic explanations," *ArXiv*, vol. abs/2012.03058, 2020.

[9] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems* (I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, eds.), vol. 30, Curran Associates, Inc., 2017.

[10] C. A. Ellis, M. S. E. Sendi, S. M. Plis, R. L. Miller, and V. D. Calhoun, "Algorithm-agnostic explainability for unsupervised clustering," *CoRR*, vol. abs/2105.08053, 2021.

[11] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[12] A. Chattopadhyay, A. Sarkar, P. Howlader, and V. Balasubramanian, "Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks," 10 2017.

[13] A. Das and P. Rad, "Opportunities and challenges in explainable artificial intelligence (XAI): A survey," *CoRR*, vol. abs/2006.11371, 2020.

[14] Z. C. Lipton, "The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery.," *Queue*, vol. 16, p. 31–57, jun 2018.