The Analysis and Development of an XAI Process on Feature Contribution Explanation

SUBMISSION DATE / POSTED DATE

12-11-2022 / 12-12-2022

CITATION

Huang, Jun; Wang, Zerui; LI, DING; Liu, Yan (2022): The Analysis and Development of an XAI Process on Feature Contribution Explanation. TechRxiv. Preprint. https://doi.org/10.36227/techrxiv.21545478.v2

DOI

10.36227/techrxiv.21545478.v2

The Analysis and Development of an XAI Process on Feature Contribution Explanation

Jun Huang*, Zerui Wang*, Ding Li, Yan Liu
Department of Electrical and Computer Engineering, Concordia University
Montréal, Québec, Canada
{*jun.huang, zerui.wang, ding.li*}*@mail.concordia.ca, yan.liu@concordia.ca*

Abstract—**Explainable Artificial Intelligence (XAI) research focuses on effective explanation techniques to understand and build AI models with trust, reliability, safety, and fairness. Feature importance explanation summarizes feature contributions for end-users to make model decisions. However, XAI methods may produce varied summaries that lead to further analysis to evaluate the consistency across multiple XAI methods on the same model and data set. This paper defines metrics to measure the consistency of feature contribution explanation summaries under feature importance order and saliency map. Driven by these consistency metrics, we develop an XAI process oriented on the XAI criterion of feature importance, which performs a systematical selection of XAI techniques and evaluation of explanation consistency. We demonstrate the process development involving twelve XAI methods on three topics, including a search ranking system, code vulnerability detection and image classification. Our contribution is a practical and systematic process with defined consistency metrics to produce rigorous feature contribution explanations.**

Index Terms—**Explainable AI, Feature Importance, XAI Process, Machine Learning, Deep Learning**

1. Introduction

The topic of "*eXplainable Artificial Intelligence* (XAI)" has drawn increasing attention from AI experts and scientists. XAI focuses on various methods that provide a human understanding of AI models [1], [2], [3]. With the explanations, humans can decide if the AI model is trustworthy.

Academic search engines are always essential as the priority tools researchers must use. Scientists obtain the latest research progress and inspiration through searching and reading literature. The Semantic Scholar [4] is a real-world case of the scientific literature search engine combined with the Artificial Intelligence (AI) model. A typical question is: *Why does one paper rank higher than the other?* Another question is: *Which feature of a paper contributes to the ranking prediction more?*

Natural Language Processing (NLP) has wide applications. A novel usage for NLP is to detect programming code vulnerabilities or bugs. Unlike static code analytics tools, the AI-powered code vulnerability detector commonly does not explain the decision-making process. Hence there is a need to use the XAI techniques to address problems.

Computer Vision (CV) models are commonly studied in AI. As the complexity of the model increases, the vision model performs better while losing its interpretability. An issue often occurs is that the deep learning models could learn biases from the vast data set. Without explanation analysis, people cannot find the model's bias, which cannot make the model trustworthy.

XAI-related research is still in the stage of early development. Many XAI methods focus on solving particular questions or attributes of the explainability of the specific type of model. Afterwards, multiple XAI methods may provide diverse explanations for the same cases. It is hard to judge which explanation is reliable without evaluation. With the diversities of XAI methods, the decision to select an XAI method and the following up development with the XAI method with the model evaluation is beyond a single task but involves a complete process.

In this work, We develop an XAI process to provide a general guideline for XAI practitioners. The current development of our XAI process and evaluation metrics mainly focus on analyzing the XAI methods that generate feature contribution explanation.

To develop the XAI process for feature contribution explanation, we derive three research questions:

RQ1: ***Given the types of XAI methods and diversities of AI models, what is the basis for the process development to cover the combination?***
The answer to this question analyzes to what extent the process is subject to the types and attributes of XAI methods. In Section 3.1 and 3.2, we summarize the existing XAI goals regarding feature contribution explanation into a criterion and construct a taxonomy for XAI methods as the base of the process.

RQ2: ***What are the core activities and major entities produced by those activities in a process definition for feature contribution explanation?***
The process generally has multiple activities, each of which handles different functionality. The execution of these activities achieves the XAI goals. In Section 3.3, we define a micro process with associated

*. These authors contributed equally to this work

978-1-6654-8045-1/22/$31.00 ©2022 IEEE

activities. We further validate the outputs using three case studies in the following sections.

RQ3: ***What metrics for feature contribution explanation are used to evaluate different XAI methods based on the measurement derived for explainability?*** The metrics are necessary to quantify the comparison among XAI methods. Thus, XAI practitioners can select a specific XAI method based on the results of the systematic evaluation. Section 4 presents clear definitions of feature contribution explanation consistency in two aspects.

The contribution of this paper is three-fold as follows:

1. We summarize the criterion for feature contribution explanation. We construct a taxonomy for the state-of-the-art XAI methods and guide XAI practitioners in selecting suitable candidate methods for the AI model explanations.
2. We present an XAI micro process which provides general guidance to practitioners to achieve AI model explanation, particularly by feature contribution. The process enables practitioners to evaluate and compare their explanation results between multiple approaches. We define two metrics to assess the stability and consistency of the feature contribution explanation methods.
3. We present three case studies to demonstrate the usage of the process and the assessment of multiple types of feature contribution explanations across different models. The case studies are a scholarly literature search ranking model, a natural language processing classification model, and an image classification model. We apply the process to these case studies, get the feature contribution explanations, and follow the evaluation of stability and consistency.

The following is the organization of this study. Section 2 discusses related works, including the goal of XAI, existing XAI methods, and evaluation methods. Section 3 presents the XAI goals definition under the input/output relation study criterion of XAI. It then summarizes the XAI method taxonomy based on their properties. Finally, it presents the XAI process development for feature contribution explanation. Section 4 gives the evaluation metrics definition for feature contribution explanation consistency. Section 5, 6, and 7 provide three case studies examining twelve XAI methods regarding the feature contribution explanation. Finally, Section 8 concludes our experiment and development and the further usage and limitations of the XAI process.

2. Related Works

Research on XAI arises regarding the growth of AI applications in expanding domains. This section discusses the existing techniques and works regarding the evaluation metrics of XAI methods to understand SOTA research and its relevance to XAI practices.

2.1. Methods for Achieving XAI

Generally, there are two main branches of methods for XAI techniques, namely, building an inherent interpretable AI model or explaining the model with post-hoc methods.

Algorithms such as linear regression, logistic regression, and decision tree models are adopted in building an inherent interpretable AI model for a specific domain [5]. While deep learning models are hard to be interpreted inherently, this motivates the development of post-hoc methods.

Post-hoc methods can be classified into model-specific and model-agnostic methods. The model-specific methods focus on explaining the specific types of models. A study [6] initially proposed a technique named Class Activation Mapping (CAM) from the global average pooling layer of the Convolutional Neural Networks (CNN) model. CAM visualizes and highlights the discriminative object parts on any given image to the CNN model. Afterwards, Grad-CAM [7], EigenCAM [8], Grad-CAM++ [9], XGrad-CAM [10], and HiResCAM [11] are a series of methods that solve the drawbacks of the CAM method, optimize the map and provide faithfulness.

The model-agnostic methods explain the model by the model's input and output. They can be applied to any model. Some model-agnostic methods are based on rules or simplified interpretable models for prediction. The decision set [12] is a framework for independent rules. A model based on rules and bayesian analysis [13] aims to build interpretable predictive models.

Alternatively, researchers are focusing on explaining which feature affects the prediction. Anchor [14] computes the individual explanations with high-precision rules. The partial dependence plot (PDP) [15] provides observations on how the outcome prediction changes with the variation of a single input feature in the scope of the entire data set. The computation complexities of PDP are significantly high because all the data set samples to involve in the calculation for each example. In addition, the mutual offset between related features makes PDP unreliable [16]. An optimized approach of PDP named accumulated local effects (ALE) plot [16] was proposed. It eliminates the unreliable bias cases in the PDP method and reduces the computation complexity, making it possible to apply to large data sets. These methods explain the model by varying the input feature and observing the output prediction results. We call them *mutation-based* methods.

Shapley Value [17] is a reliable feature attribution method with a solid theoretical background [18], which explains a single data sample by calculating the contribution value for each feature. Another novel and well-known method, SHAP [19] gives an individual explanation by predicting the contribution values instead of calculating them. These methods calculate the feature contribution values by removing or masking the features. The study [19] defines these methods as *removal-based*. This paper summarizes them as *masking-based* methods.

2.2. Assessment of XAI Method

The metrics to assess XAI methods are diverse. A study [20] introduces the definition of *soundness* and *completeness* as XAI metrics. Soundness indicates if the explanation is correct. The measurement of soundness requires access to the ground truth. However, the ground truth of the model may not always be available. Then, completeness describes the explanation of the entire task model. Completeness is an assessment metric for global explanations but is not necessary for local explanations. A study [21] about an explainable book search system evaluates the *trustworthiness* in terms of the retrieval performance. The system utilizes three approaches: the ranking by user clicks, questionnaires, and the user eye tracker system. The ranking results are compared and evaluated for the trustworthiness of explanations.

To assess those metrics, XAI practitioners should conduct user studies through interviews or questionnaires. In our study, we present computational metrics to measure the performance of the XAI method.

3. The Development of XAI Process

Understanding commonalities and discrepancies of XAI methods are essential to driving the XAI process. This section presents a summarized XAI criterion from the XAI works. Then, Figure 1 shows a taxonomy of XAI methods, which guides the definition of a general-purpose XAI process. Finally, an XAI process for the feature contribution explanation is presented in Figure 2.

3.1. XAI Criterion: Input/Output Relation (*RQ1*)

To develop an XAI process, we need to establish the criteria to evaluate the process outcome. The criterion is discovering the logic between a model's inputs and outputs. It matches the following XAI goals, including:

Feature Influence. This goal quantifies the correlation between predictive features and class variables in learning. The influence of the input features does not directly reveal the underlying mechanism of an AI model. Instead, it helps quantify the importance of features and further improves the quality of feature selection.

Feature Causality. This goal focuses on the causal relations between predictive features and class variables. For example, the objective in the image classification case presents causality between feature and prediction. The causal effects help to discover the underlying mechanism of an AI model.

3.2. XAI Taxonomy (*RQ1*)

We organize the XAI methods into a taxonomy, as shown in Figure 1. The taxonomy has a tree-based topology to structure the levels of categories. The main categories are building interpretable models and using post-hoc XAI methods. Under the post-hoc methods, practitioners generate the explanation with model-specific or model-agnostic methods. The model-specific methods are developed based on the specific structures of machine learning models. The model-agnostic methods apply the techniques, including explaining the feature importance, data visualization, and model simplification. The developed XAI taxonomy answer the first research question. The category allows adding new methods developed in the future. We describe the branches of the taxonomy structure as follows.

3.2.1. Build Interpretable Models. The interpretable model allows extracting decision rules from the model structure. The machine learning algorithms such as Linear regression, Logistic regression, Decision trees and Decision rules are commonly used to develop interpretable models.

3.2.2. Post-hoc Explanation Methods. Post-hoc XAI methods aim to extract relationships between feature values and predictions. Models such as deep neural networks are not usually interpretable since the decision rules can not be directly extracted from the model structure. In terms of the ways to approximate the model's behaviour to understand decisions, the taxonomy has branches as model-specific and model-agnostic methods.

3.2.3. Model-specific Methods. Model-specific methods mean the XAI techniques apply to specific contexts and conditions. Those techniques use the properties of the underlying algorithm or specific structure of an AI model. Reverse engineering approaches are applied to probe the internals of algorithms. Grad-CAM [7] methods provide visual explanations for convolutional neural network models. The CAM-based methods produce a localization map from a convolutional layer showing the essential regions in the image for predicting the concept. A Layer-wise Relevance Propagation [22] is used for recurrent neural network explanation. Methods such as EfficientNET [23], and Axiomatic Attribution [24] are also used for explaining CNN models.

3.2.4. Model-agnostic Methods. Model-agnostic methods focus on relations between feature values and prediction results. Under this classification, we further distinguish the XAI methods by how they present the explanations.

Explaining by Visualization. This kind of method summarizes visual tools to help humans understand model behaviours. An obvious example is that explanations illustrate the image sample or data set by highlighting and visualizing the relevant feature area.

Explaining by Feature Importance. This class indicates methods that explain the influence of the features for prediction based on the feature masking method or feature mutation method.

Feature Masking. Feature masking methods remove the input feature or set the element to the default value. The idea is to observe how the model predicts with the masked information. A framework [19] for the XAI method associated with removing features presents the choices made by the removal-based explanations.

Feature Mutation. Feature mutation methods assign other values from the data set to replace one or a few input feature values and show how the individual prediction changes according to the data variation. For example, the partial dependence plot [15] shows the mutation impact of one feature on the model's outcomes.

Explaining by Simplification. Explaining by simplification means training another interpretable model to describe the current complex black-box model. It contains two groups of methods, the Rule-based Learner and the Additional Interpretable Model.

Rule-based Learner. Decision sets [12], and their variants are sets of classification rules. Rule-based models are initially developed as decision rules that explain how they reach a particular prediction. By understanding the branches of these decisions, humans can understand the path for predictions.

Additional Interpretable Model. The methods explain the black-box classification model by training another interpretable model. For example, the Local Interpretable Model-Agnostic Explanations (LIME) [25] build simple linear models around the predictions to provide explanations. Anchor [14] gives individual explanations with new high-precision rules. These methods can only supply local explanations.

In the experiments, we prepare three case studies for the XAI process. Altogether, we have twelve XAI methods deployed. These are six Model-agnostic methods that contain both feature masking methods and feature mutation methods. The six Model-specific methods are CAM-based for the CNN model. We indicate our selection path with grey background in the boxes and the bold lines in Figure 1.

3.3. XAI Process Development for Feature Contribution Explanation (*RQ2*)

The XAI process development focuses on the explanation activities and the essential data flow. We present the core activity associated with the second research question for feature contribution explanation. We identify the significant activities and structure them as an XAI process shown in Figure 2. In the following part of this section, We present the main functionality of each task and the data flow between tasks.

3.3.1. Select XAI Method. XAI practitioners explore multiple XAI methods through the taxonomy and identify those to fulfill the XAI criterion. For instance, Accumulated Local Effects (ALE) [16] and Partial Dependence Plot (PDP) [15] both are feature mutation methods by calculate the varied feature values affect the prediction. ALE accumulates the effects of feature value variation within a well-separated interval, while PDP captures the effects of feature value variation within the entire feature space. ALE is a faster and unbiased alternative to the PDP method, but this does not indicate that it is the default better choice. If the practitioners can not find the proper order of the categorical feature of the target model, they can not explain this feature with ALE since ALE can only be executed on the ordered feature space. Another example is the method of "Model Class Reliance for Feature Importance Estimation [26]". Such a method requires actual outcomes of the data samples to calculate the expected $loss$ values to generate the explanation. With the selection of multiple XAI methods, XAI practitioners should follow the process and evaluate the explanation consistency for these methods.

3.3.2. Develop XAI Method. Adopting a selected XAI method may further customize the implementation in alignment with the AI model and the data set. Customization is a form of general software configuration. Therefore the practices and tools in software configuration management can be referred to and applied. We omit the details in the discussion. During the execution of XAI methods, hyper-parameter searching and tuning are necessary to gain optimal performance. In addition, cross-validation is essential if the data set may incur unbalanced results for XAI evaluation. One way is to execute the XAI methods with segments of the input data and observe the variation of the XAI results.

3.3.3. Define Explanation Consistency. The same input data set explained by various XAI techniques may yield inconsistent results. Hence we derive explanation consistency as an evaluation metric. The XAI results consistency brings trust to the XAI practitioners. In this work, we present the definition of consistent evaluation in section 4 and set it as a unit in the process.

3.3.4. Process Data and Execute XAI method. The data processing activity is to perform data cleaning, extraction, segmentation and classification in preparing the data format and quality suitable to an XAI method. In addition to those tasks akin to data preprocessing for model training, the data preparation is also under requirements imposed by the selected criterion. For example, the original data may lead to unbalanced result distribution, which is unsuitable for inputting directly into an XAI method for evaluation. Hence the data preparation needs to perform extra tasks such as classifying the data sets whose results of XAI methods are more balanced. Then, what follows is the execution of XAI methods. The XAI methods produce explanation results.

3.3.5. Present Explanation Summary and Evaluate Consistency. The process presents the output after XAI runtime. Besides analysis utilizing statistics, tables, or graph plots, the result should be further unified and summarized to match the evaluation metrics. Then, the process validates the results with the defined explanation consistency. Suppose the consistency level does not satisfy the explanation. XAI practitioners may select another XAI method and repeat another round of the process. If the consistency level meets the evaluation, the method can be present for this model and data set. Then the explanation results can conclude.

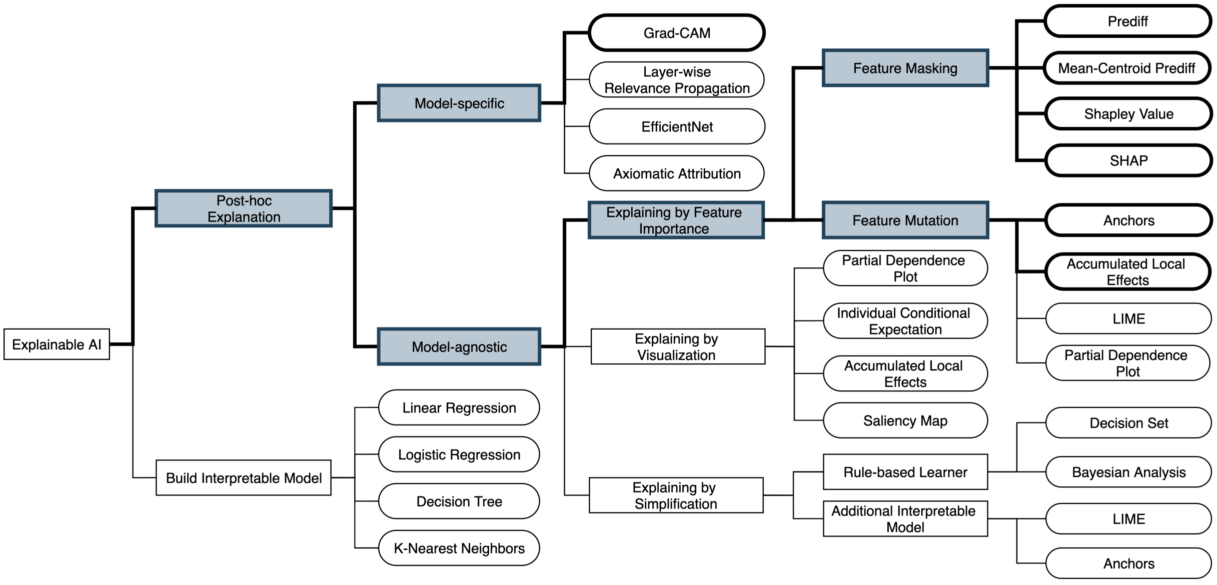


Figure 1: Taxonomy of explainable AI methods.

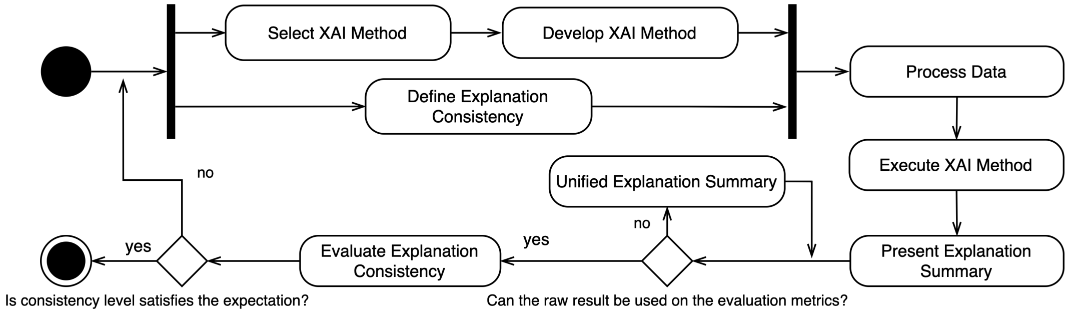

Select XAI Method
Develop XAI Method
Define Explanation Consistency
Process Data
Execute XAI Method
Present Explanation Summary
Unified Explanation Summary
no
yes
Evaluate Explanation Consistency
no
yes
Is consistency level satisfies the expectation?
Can the raw result be used on the evaluation metrics?

Figure 2: A feature contribution explanation micro process: main activities

4. Metric Definition for XAI Method (*RQ3*)

We focus on the metrics to measure, observe and evaluate the XAI explanation on the same data set across multiple XAI methods and the explanation on multiple data sets generated by the same XAI method. We aim to define the unified metrics that cover the above three scenarios. According to the process definition, the XAI criterion leads to consistency metrics.

4.1. Derive Feature Contribution Value and Explanation Summary

For the explanation that can not be used to build a new data instance, such as ALE explanation or Anchor explanation, we derive the *feature contribution value* for each feature by aggregating the raw explanation result, and we further derive the explanation summary from those values. Suppose $< e_j^{x^{[1]}}, e_j^{x^{[2]}}, \dots, e_j^{x^{[n]}} >$ is the raw explanations for feature j on the given data set X. The *feature contribution value* $\phi_j(e_j^X)$ could be the absolute mean value of the e_j^X. The first and second case studies in Section 5 and Section 6 derive the feature contribution value $\phi_j(e_j^X)$ by such aggregation. By ranking features' contribution values in descending order, we derive the feature importance order on each data set as the explanation summary.

For the explanation that can be used to build a new data instance, such as saliency map explanation from CAM-based methods, we derive the explanation summary directly by calculating the prediction difference between the original data instance and the new-built data instance. Consider $\hat{f}(x^{[i]})$ is the model prediction on instance $x^{[i]} < x_1^{[i]}, x_2^{[i]}, \dots, x_p^{[i]} >$, where p is the number of features. $x^{[i]+}$ represents the instance $x^{[i]}$ under a transformation such as an image with masks. Suppose S is the subset of all the features that we are interested in their feature importance. j is a feature that belongs to S. R contains the rest of the features, and $P = S \cup R$ makes the whole features P. By marginalizing over features R, the prediction depends on features in S, including the interactions with other features in R. For each instance x of the data set, we define the *prediction difference* caused by the feature $j \in S$ of interest as the absolute difference in Equation 1 and the relative difference in Equation 2 respectively.

$$\delta_j^{x^{[i]}}|_{abs} = |\hat{f}_S(x^{[i]}) - \hat{f}_P(x^{[i]})| \tag{1}$$

and

$$\delta_j^{x^{[i]}}|_{rel} = \left| \frac{\hat{f}_S(x^{[i]+}) - \hat{f}_P(x^{[i]})}{\hat{f}_P(x^{[i]})} \right| \tag{2}$$

Such prediction difference of the given data instance can be used as the explanation summary. For the third case study in Section 7, multiple XAI methods are applied to the same model prediction on the same data set. The XAI methods produce saliency maps as an explanation, which are used to transform an original image into a masked image for prediction. The explanation summary is computed by Equation 2.

4.2. Explanation Summary Distance

So far, we have set up the one-on-one mapping between an explanation summary and the feature contribution values regardless of the form of an explanation summary. Hence we define the metric *distance* to measure the difference between any pair of explanation summaries. Suppose there are m explanation summaries. The set E contains all the explanation summaries, $E = \{\varepsilon^1, \varepsilon^2, \ldots, \varepsilon^m\}$. Then the distance between any two pairs of summaries is defined as $f_d(\varepsilon^i, \varepsilon^j)$. The choice of selecting any pair of summaries in $E = \{\varepsilon^1, \varepsilon^2, \ldots, \varepsilon^m\}$ has the combination of $K = \binom{m}{2}$. Each choice k produces a distance value as $f_d^{[k]}(\varepsilon^i, \varepsilon^j), (i \neq j, i \leq m, j \leq m)$. The shorter the distance value, the more consistent the XAI explanation summaries are on two data sets. We can now observe and evaluate XAI explanation summaries at two levels, namely *explanation stability* and *explanation consistency* as follows.

4.2.1. Explanation Stability. Explanation stability represents the intra-XAI method explanation consistency. When an XAI method is applied to the model prediction with multiple data sets, each data set has an explanation summary produced by the same XAI method.

We use a violin plot to observe the explanation stability to visualize the distance distribution and probability density of $K = \binom{m}{2}$ number of distance values. We further aggregate the mean distance value for one XAI method across multiple data sets as follows.:

$$f_d^K = \frac{1}{K} \sum_{k=1}^{K} f_d^{[k]}(\varepsilon^i, \varepsilon^j), (i \neq j, i \leq m, j \leq m) \tag{3}$$

4.2.2. Explanation Consistency. Explanation consistency presents the inter-XAI explanation consistency across multiple XAI methods on the same data set. It compares different XAI methods' explanation summaries. Here the number m of explanation summaries are generated from m numbers of XAI methods on the same data set, $E = \{\varepsilon^1, \varepsilon^2, \ldots, \varepsilon^m\}$. For each XAI method I, we compute the distance of its explanation ε^I to other $m-1$ XAI methods, which produces $m-1$ numbers of distance values $f_d^{[k]}(\varepsilon^I, \varepsilon^j), (I \neq j, I \leq m, j \leq m, k = 1, 2, ..., m-1)$.

Likewise, a violin plot can be applied to visualize the distance distribution and probability density of $m-1$ number of distance values for each XAI method to observe the explanation consistency across XAI methods. We further aggregate the mean distance value for one XAI method I, compared with other $m-1$ XAI methods on the same data set as follows:

$$f_d^I = \frac{1}{m-1} \sum_{k=1}^{m-1} f_d^{[k]}(\varepsilon^I, \varepsilon^j), (I \neq j, I \leq m, j \leq m) \tag{4}$$

Considering the complex comparison scenario under L numbers of data sets, we compute the inter-XAI method explanation distance using Equation 4 for each data set. We finally collect $(m-1) \times L$ distance data plots for each XAI method to visualize using the violin plot and compute the mean distance values of $(m-1) \times L$ data instances.

5. Case Study I: Academic Paper Ranker

The first case study explains the academic papers ranking model, a black-box model with tabular inputs. We select four XAI methods: ALE, Shapley Value, Anchor, and SHAP to derive the overall feature importance order. We evaluate those methods by explanation stability and explanation consistency.

5.1. Model and Data

The target model is an open-source machine learning ranking model called "s2search" [27] from Semantic Scholar [4], a scientific literature search engine. *s2search* model ranks papers according to the collected user behaviour data, such as search logs and user clicks. For the data set, we choose the arXiv data set [28] meta topics as classification labels under the field of *Computer Science*. We use the subset with secondary categories of *Computer Science* provided by arXiv. Each paper is categorized with one or more meta topics. Finally, we have thirty-eight data sets categorized by arXiv with 542,877 papers' metadata for the case study experiment. In this case study, our objective is to determine the order of the importance of the six features :*title*, *abstract*, *venue*, *authors*, *year*, *n_citations*.

5.2. Explanation Consistency Analysis

According to the taxonomy in Figure 1, we select ALE, Shapley Value, Anchor, and SHAP to derive feature importance order for the black-box model. We develop the software solution for the four XAI methods. The source code is available from this GitHub repository[1].

[1] https://github.com/youyinnn/s2search

5.2.1. Summary from Feature Importance Order. For all methods that we selected, we define the feature contribution value for feature j over data set X as:

$$\phi_j(e_j^X) = \frac{1}{n}\sum_{i=1}^{n}|e_j^{x^{[i]}}|$$

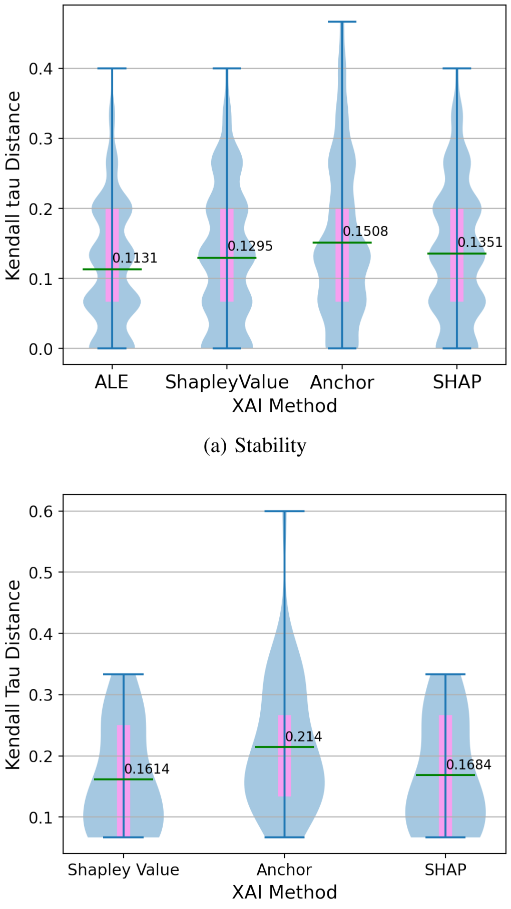
where the $e_j^{x^{[i]}}$ is the raw explanation for feature j on data sample $x^{[i]}$. For the ALE explanation, the local effect for each data instance will be the $e_j^{x^{[i]}}$. For the Shapley Value or SHAP explanation on data sample $x^{[i]}$, we set each contribution value from it for feature j as $e_j^{x^{[i]}}$. For the Anchor explanation on each data sample $x^{[i]}$, we set the partial precision value for feature j as $e_j^{x^{[i]}}$. Repeating this process to every feature and get feature contribution values over the data set X. For thirty-eight data sets, we get thirty-eight groups of feature contribution values for each method. We take the mean value of those thirty-eight groups of values as the overall feature contribution value, and further derive the overall feature importance order explained by each method. The results are listed in Table 1.

5.2.2. Stability Observation. To observe the explanation stability, we run each XAI method on thirty-eight data sets. This experiment produces thirty-eight feature importance orders as the explanation summaries. We apply *Kendall tau Distance* algorithm as the distance function to compute $f_d^{[k]}(\varepsilon^i, \varepsilon^j), (i \neq j, i \leq 38, j \leq 38)$. By Equation 3, we compute 703 distances for each XAI method and produce the violin plot with mean value bar depicted in Figure 3. The result indicates that the ALE method has the lowest mean value on the stability metric.

5.2.3. Consistency Observation. We select ALE as the baseline more for inter-XAI method consistency observation since ALE performs better in term of method stability across multiple data sets on the same prediction model. To calculate the consistency between the ALE method and other methods, we use the Equation 4 with the *Kendall tau Distance* algorithm as the distance function $f_d^{[k]}(\varepsilon^{ALE}, \varepsilon^j), (I \neq j, j \leq 4, k = 1, 2, 3)$. Figure 3b reads as the explanation difference between Shapley Value and ALE, Anchor and ALE, and SHAP and ALE in term of distances measured on thirty-eight explanation summaries. Shapley Value and SHAP has approximate mean distance values compared to ALE. The plot indicates Shapley Value and SHAP produce more consistent explanation with ALE than Anchor does.

6. Case Study II: Code Vulnerability Classifier

Popular code vulnerability data sets like the Open Web Application Security Project (OWASP) [29] Benchmark and Juliet test suite [30] provide a reliable training corpus. These data sets contain mined method-level software code files with the Common Weakness Enumeration (CWE) labels. The code files usually contain the code comments, code body, and import packages. A class of technique is that machine learning models treat the code file as the text content and classify it into different CWE labels. It is a natural language processing classification problem. In this case study, we observe whether machine learning models capture semantics from code bodies, code comments or import packages and make the classification decision.



(a) Stability

(b) Consistency

Figure 3: (a) Explanation stability for each method where the mean value indicates the stability. (b) Explanation consistency between ALE and other methods. The green line indicates the mean value.

6.1. Model and Data

The data sets are from The Open Web Application Security Project (OWASP) Benchmark [29] and Juliet test suite [30] for Java. OWASP Benchmark contains 2,740 test cases with 52% files of vulnerable code and 48% non-vulnerable files. The 52% vulnerable files contain 11 CWE (Common Weakness Enumeration) labels. The Juliet test suite contains 217 vulnerable and 297 non-vulnerable files with 112 different CWE labels. The CWE labels are the ground truth for multiclass classification.

We practice this case study with the state-of-the-art Natural Language Processing (NLP) model XLNet [31]. It

TABLE 1: Feature contribution explanation summary derived from feature importance order for each method

Method	Overall Feature Importance Order					
ALE	*abstract(5.4923)*	*title(2.6864)*	*venue(2.5473)*	*year(0.506)*	*n_citations(0.266)*	*authors(0.2504)*
Shapley Value	*abstract(5.4701)*	*title(1.6359)*	*year(1.0298)*	*venue(0.8989)*	*n_citations(0.171)*	*authors(0.0987)*
Anchor	*abstract(0.4659)*	*year(0.1724)*	*title(0.1592)*	*venue(0.0853)*	*n_citations(0.0461)*	*authors(0.0074)*
SHAP	*abstract(4.5293)*	*title(1.2729)*	*year(0.8781)*	*venue(0.7461)*	*n_citations(0.1473)*	*authors(0.0899)*

is a model that could capture bi-directional text information and outperform other state-of-the-art NLP models. Each text content in a code file containing a method with a CWE label is considered the data instance and the label pair. To observe which part of these contexts highly impacts the machine learning classifier, we identify three features in the data: comments, code body and import packages. Comments contain the file description and the code comments. Further, the description comment of the Juliet test case includes the CWE types directly. We remove this content to avoid data leakage during the model's training. Data leakage is defined as unintentionally leaking the signal. In our case, the CWE type content to the model potentially increases the accuracy [25]. We directly remove one of the three features and remain the other two features to get the prediction from the model as masked the feature.

The label distributions in both data sets are also unbalanced, and the non-vulnerable code accounts for over 50%. In practice, we combine labels whose numbers count less than ten percent of the total size and generate four labels for both data sets.

The Juliet test case for Java and the OWASP Benchmark test suite are treated as two data sets to perform the experiments. We shuffle each data set into a training set and testing set by the percentage of eighty and twenty. The training set is used for fine-tuning the XLNet model. The testing set is used for the masking feature and collects the prediction for XAI processing. The features are removed before input to XLNet model [31] to output the log-odd probability of the ground truth CWE label.

6.2. Explanation Consistency Analysis

Under the objective of the case study, observing which feature among the three features affects the most model deciding, we select methods that could reflect the feature influence. Meanwhile, features in text content can be evaluated by masking. We pick up Shapley Value [17], SHAP [19], Preddiff [32] and Mean-Centroid Preddiff [33] to perform the feature masking-based explanation.

6.2.1. Summary from Feature Importance Order. In this case, the way of deriving the explanation summary for the Shapley Value, SHAP, and Prediff method is the same as we did in the first case study. For Mean-Centroid Prediff method, the raw explanation for each data instance is the prediction difference in paper [33] as $\delta_j^{x^{[i]}}$. This method computes feature contribution value $\phi_j(\delta_j^X)$ as the tangent value of the centroid point of clusters formed by the data instances of $\delta_j^{x^{[i]}}$. Then, we derive the feature importance order for the two data sets in Table 2.

6.2.2. Stability Observation. In this case, we only have two data sets. Hence the explanation stability metric value of Shapley Value and SHAP method is 0 since their feature importance order in the two data sets are the same. For Prediff and Mean-Centroid Prediff, it is 0.67. The result indicates that the Shapley Value and SHAP method are considered more stable than the Prediff and the Mean-Centroid Prediff method.

6.2.3. Consistency Observation. The feature importance order results of four XAI methods are consistent for Juliet test cases. *comment* has the most importance than *code* and *import*. Each method has a zero *Kendall tau distance* with other methods. For the OWASP Benchmark data set, Preddiff and Mean-Centriod Preddiff have different insights on features. Measuring by *Kendall tau distance*, we calculate the average feature importance order distance for Preddiff and Mean-Centroid Preddiff are 0.33 from the other two methods. We observe Shapley Value and SHAP achieve a higher consistency.

7. Case Study III: Image Classifier

The third case study explains the image classifier model with tabular inputs. The selected XAI methods for computer vision are all model-specific. We select six pixel-attribution methods, all CAM-based methods and further evaluate them with our defined metrics over 1,000 image data.

7.1. Model and Data

We select the public-released pre-trained CNN model called ResNet50 [34] from Pytorch[2] on the data set Imagenet-1000. The pre-trained model is used for image classification on 1,000 classes. We sampled 1,000 image data from the validation set of ImageNet[3].

7.2. Explanation Consistency Analysis

We consider the model-specific XAI method for CNN model and also the pixel-attribution methods that can produce explanations reflecting the active region of the image. According to the taxonomy in Figure 1, we select the Grad-CAM method first and we also identify other CAM-based

[2] https://pytorch.org/hub/nvidia_deeplearningexamples_resnet50/
[3] https://www.image-net.org/

TABLE 2: Feature importance order summary of code vulnerability detection case study

XAI Methods	Juliet test cases for Java	OWASP Benchmark
Preddiff	*comment* > *code* > *import*	*code* > *import* > *comment*
Mean-Centroid Preddiff	*comment* > *code* > *import*	*code* > *import* > *comment*
Shapley Value	*comment* > *code* > *import*	*comment* > *code* > *import*
SHAP	*comment* > *code* > *import*	*comment* > *code* > *import*

method, namely, EigenCAM [8], Grad-CAM++ [9], Grad-CAMEW, XGrad-CAM [10], and HiResCAM [11]. The implemetation of those methods are public accessible at this GitHub repository[4].

7.2.1. Summary from Prediction Change Aggregation. Suppose the $\hat{f_P}(x^{[i]})$ is the model prediction on the original image $x^{[i]}$ and the $\hat{f_S}(x^{[i]+})$ is the model prediction on the masked image where $x^{[i]+}$ is the masked image transformed from saliency map $\rho(x^{[i]})$ of the original image. Figure 4 shows the example of deriving the prediction change value. For all the selected CAM-based methods, we define the prediction change on data sample $x^{[i]}$ with Equation 2 as:

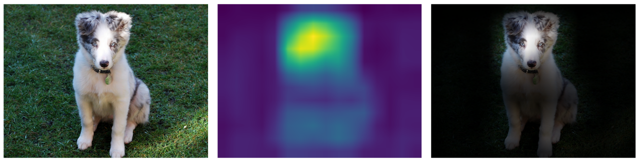

(a) original image (b) saliency map (c) masked image

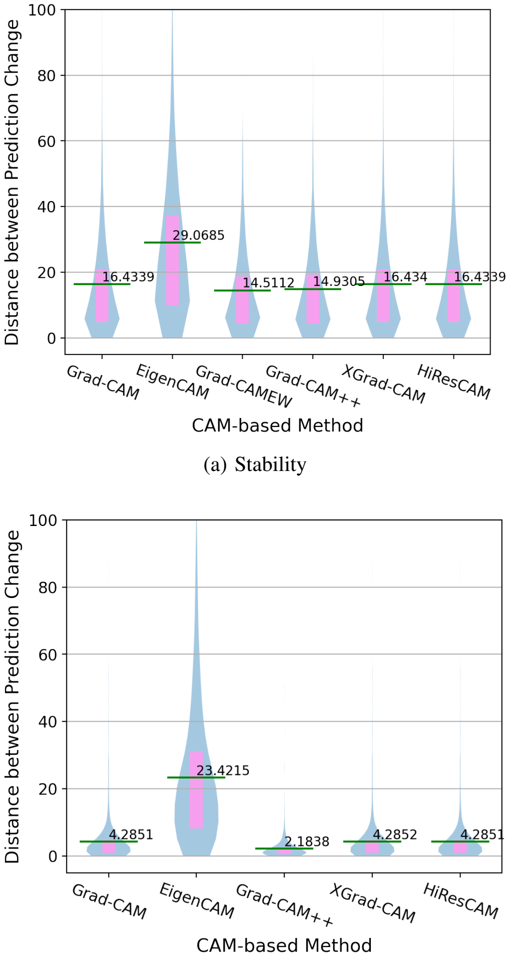
Figure 4: Images of the original image, saliency map generated by Grad-CAM method, and masked image. The ground truth label for the original image is “collie”. The prediction score is 7.5674 on the original and is 5.4336 on the masked. Hence the prediction change is 28.17%.

$$\left|\frac{\hat{f_S}(x^{[i]+})-\hat{f_P}(x^{[i]})}{\hat{f_P}(x^{[i]})}\right| \times 100, (x^{[i]+} \leftarrow \rho(x^{[i]}))$$

We then get all the prediction changes on data set X for each method.

7.2.2. Stability Observation. To derive the stability metric in this case, we set each prediction change on the data sample as one summary and get the distances of any two pairs of the prediction change with the distance function $\hat{f_d}(\varepsilon_i, \varepsilon_j) = |\varepsilon_j - \varepsilon_j|$. Figure 5a shows the distribution of the distances between the prediction changes. By using Equation 3, we evaluate the stability for a method. As can be observed, the Grad-CAMEW method has the lowest mean value on the stability metric.

7.2.3. Consistency Observation. To calculate the consistency between Grad-CAMEW method and the other methods, we use the Equation 4 with the distance function $\hat{f_d}(\varepsilon_i, \varepsilon_j) = |\varepsilon_i - \varepsilon_j|$. Figure 5b shows the distribution of consistency over 1,000 images.



(a) Stability

(b) Consistency

Figure 5: (a) Explanation stability for each method where the mean value indicates the stability. (b) Explanation consistency between Grad-CAMEW and other methods. The green line indicates the mean value.

8. Conclusion

This paper proposes a micro process for structuring the activities, entities and artifacts essential to an XAI project for feature contribution analysis. We analyze the state-of-the-art XAI goals and derive criteria. We define a taxonomy as the basis of developing the XAI process. The micro process has the benefit of producing instances of XAI according to the data set classes and methods. We

[4] https://github.com/jacobgil/pytorch-grad-cam

define two explanation consistency metrics by analyzing the explanation generation consistency within and across the methods. Thus, it becomes practical for large-scale evaluation of such XAI projects. We provide three use cases for explaining an AI-powered scholar literature ranking model, an NLP code vulnerability detection model, and an image classification model. The first case study performs a total of 152 experiments with four instances of the XAI process. Each instance maps to one XAI method and runs thirty-eight classes of data sets which include 542,877 papers. The second case study performs feature contribution explanation analysis on the NLP classification model with also four instances of the XAI process. The third case study reaches the problem of image classification. Hence, the detailed implementation of the metric evaluation differs from the previous two cases while we conduct six instances of the XAI process. The feature contribution analysis micro-process becomes the composition unit for large-scale XAI projects with large-scale data sets and many target models. Future work could develop the comprehensive XAI process for more kinds of XAI goals and different kinds of XAI methods. More explanation evaluation metrics could be identified to discover different characteristics of the XAI method.

References

[1] T. Miller, "Explanation in artificial intelligence: Insights from the social sciences," pp. 1–38, 2019.

[2] S. Mohseni, N. Zarei, and E. D. Ragan, "A multidisciplinary survey and framework for design and evaluation of explainable ai systems," pp. 1–45, 2021.

[3] C. Molnar, *Interpretable machine learning*. Lulu. com, 2020.

[4] S. Fricke, "Semantic scholar," *Journal of the Medical Library Association: JMLA*, vol. 106, no. 1, p. 145, 2018.

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

[6] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2921–2929.

[7] 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*, 2017, pp. 618–626.

[8] M. B. Muhammad and M. Yeasin, "Eigen-cam: Class activation map using principal components," in *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2020, pp. 1–7.

[9] A. Chattopadhay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, "Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks," in *2018 IEEE winter conference on applications of computer vision (WACV)*. IEEE, 2018, pp. 839–847.

[10] R. Fu, Q. Hu, X. Dong, Y. Guo, Y. Gao, and B. Li, "Axiom-based grad-cam: Towards accurate visualization and explanation of cnns," *arXiv preprint arXiv:2008.02312*, 2020.

[11] R. L. Draelos and L. Carin, "Hirescam: Faithful location representation in visual attention for explainable 3d medical image classification," *arXiv preprint arXiv:2011.08891*, 2020.

[12] H. Lakkaraju, S. H. Bach, and J. 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*, 2016, pp. 1675–1684.

[13] B. Letham, C. Rudin, T. H. McCormick, and D. Madigan, "Interpretable classifiers using rules and bayesian analysis: Building a better stroke prediction model," *The Annals of Applied Statistics*, vol. 9, no. 3, pp. 1350–1371, 2015.

[14] M. T. Ribeiro, S. Singh, and C. Guestrin, "Anchors: High-precision model-agnostic explanations," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.

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

[16] D. W. Apley and J. Zhu, "Visualizing the effects of predictor variables in black box supervised learning models," pp. 1059–1086, 2020.

[17] E. Štrumbelj and I. Kononenko, "Explaining prediction models and individual predictions with feature contributions," *Knowledge and information systems*, vol. 41, no. 3, pp. 647–665, 2014.

[18] L. S. Shapley, "A value for n-person games, contributions to the theory of games, 2, 307–317," 1953.

[19] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," *Advances in neural information processing systems*, vol. 30, 2017.

[20] T. Kulesza, S. Stumpf, M. Burnett, S. Yang, I. Kwan, and W.-K. Wong, "Too much, too little, or just right? ways explanations impact end users' mental models," in *2013 IEEE Symposium on visual languages and human centric computing*. IEEE, 2013, pp. 3–10.

[21] S. Polley, R. R. Koparde, A. B. Gowri, M. Perera, and A. Nuernberger, "Towards trustworthiness in the context of explainable search," in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021, pp. 2580–2584.

[22] L. Arras, G. Montavon, K.-R. Müller, and W. Samek, "Explaining recurrent neural network predictions in sentiment analysis," 2017.

[23] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," pp. 6105–6114, 2019.

[24] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," PMLR, pp. 3319–3328, 2017.

[25] M. T. Ribeiro, S. Singh, and C. Guestrin, "Why should i trust you? explaining the predictions of any classifier," pp. 1135–1144, 2016.

[26] A. Fisher, C. Rudin, and F. Dominici, "All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously," 2018.

[27] A. I. for AI. (2020) Github - allenai/s2search: The semantic scholar search reranker.

[28] arXiv.org submitters, "arxiv dataset," 2022. [Online]. Available: https://www.kaggle.com/dsv/4498393

[29] P. R. Phil, "Owasp top 10: The top 10 most critical web application security threats enhanced with text analytics and content by pagekicker robot phil 73," 2014.

[30] P. E. Black and P. E. Black, *Juliet 1.3 test suite: Changes from 1.2*. US Department of Commerce, National Institute of Standards and Technology, 2018.

[31] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le, "Xlnet: Generalized autoregressive pretraining for language understanding," *Advances in neural information processing systems*, vol. 32, 2019.

[32] L. M. Zintgraf, T. S. Cohen, T. Adel, and M. Welling, "Visualizing deep neural network decisions: Prediction difference analysis," *arXiv preprint arXiv:1702.04595*, 2017.

[33] D. LI, Y. Liu, J. Huang, and Z. Wang, "A trustworthy view on xai method evaluation," *TechRxiv. Preprint. techrxiv.21067438.v2*, 2022.

[34] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.