Published in Geek Culture

Gatha Varma Follow

Sep 24, 2021 · 7 min read · Listen

Save

Feature Attribution in Explainable AI

Model explanations in the form of feature importance. What is it & how is it achieved?

Imagine an AI-based model trained to identify whether the food item in the box is a pizza or a burger from provided images. Now the developer had asked their friends and family to share pictures taken by them to build the training dataset. Moreover, two specific makers of pizza and burgers were popular in their town because of their limited geography and extensive franchise system. It is but obvious that the logos of the brands were also present in the training images. Should the presence and shape of the logo on the box be an indicator of whether the food item is a pizza or a burger? Or in other words*, will the trained model output be considered correct if the logo on the box was treated as one of the features to classify the images?* Such unpredictability in data can lead to improper training and possible bias, which Explainable AI (XAI) aims to address.

Coming over to more complex models that need to detect tumors from images, or predict the outcome of certain chemical combinations for drug discovery. Human or algorithmic users of such AI-based models would have expectations like transparency as to how the model reached a particular answer, justification of why the output should be accepted, the information to back decisions, and uncertainty estimation. Feature attribution is an important part of post-modeling (also called post hoc) explanation generation and facilitates such desiderata.

A feature attribution method is a function that will accept model inputs and give a per-feature attribution score based on the feature's contribution to the model's output. The score could range from a positive value that shows its contribution to the model's prediction, to a zero that would mean the feature has no contribution, to a negative value which means that removing that feature would increase the probability of the predicted class. For example, the image background has been used as a feature in certain object recognition tasks. If an attribution algorithm assigned it a negative score for a hand gesture recognition model, it would mean that removing background pixels would improve the recognition of gesture class. In the instance of an improperly trained model on burger and pizza images, the attribution algorithm will present the brand logo as a feature with a substantial

The type of feature attribution that focuses on the importance of a feature and its influence on the results of a trained model for a specific input is called **Local feature attribution**. It is said to be of a local scale since the scope is limited to an observation i.e. the specific input set. A popular method to achieve this is local interpretable model-agnostic explanations (LIME). Instead of starting from local attributions and aggregating up, it is possible to explain the global behavior of a model by inspecting the model's own structure or one of its surrogates through **Global feature attribution**. This approach ensures that the human understands the full picture which would explain model behavior on all inputs.

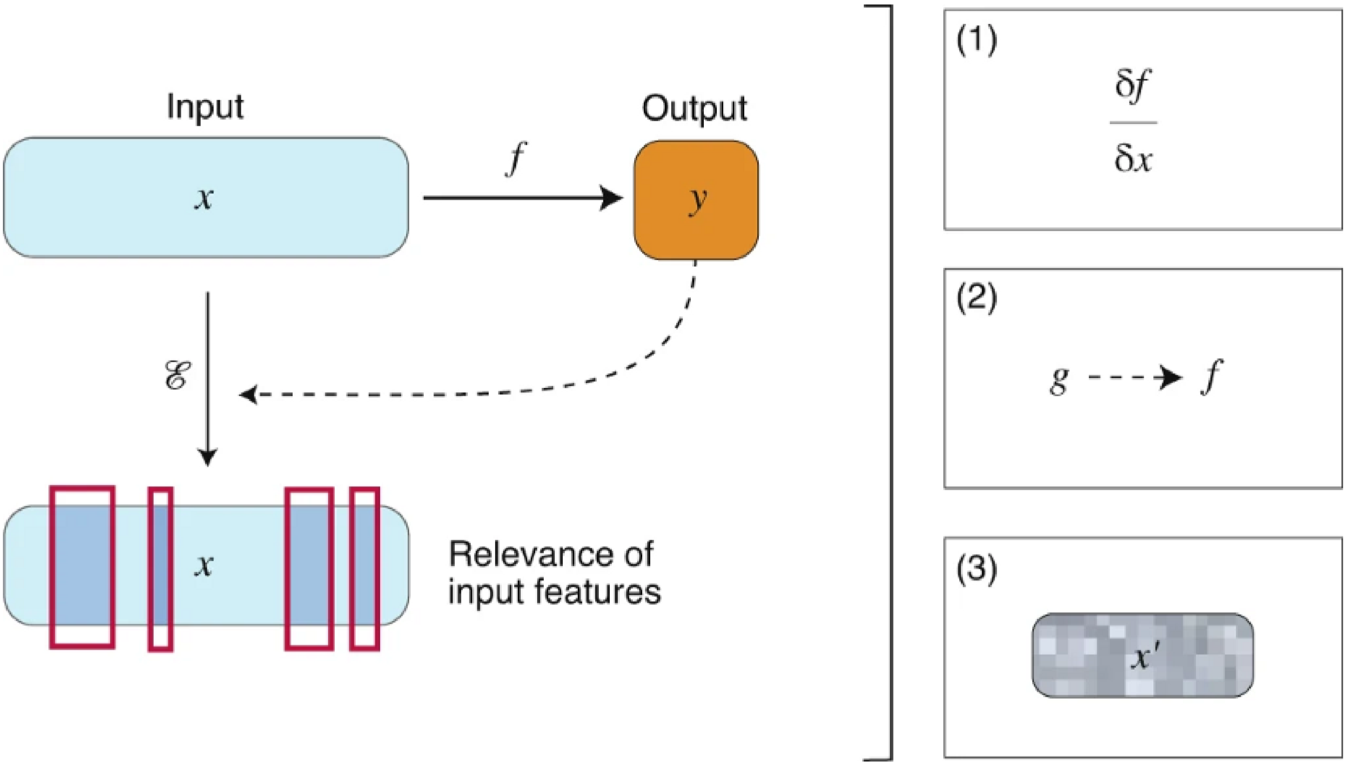
However, it is only possible to communicate the entirety of a model if it is a simple one with a small number of parameters. For a complex model, global attributions can be extracted from its simpler surrogate, but such generated importances may not cover all of the complex scenar[illegible]ider a model working for critical problems in financial services, a su[illegible] might not be acceptable for seeking explanations.

The process of feature attribution that I have explained must be giving you flashbacks of something similar. The features are being analyzed for contribution to the prediction and assigned scores; also there is a provision for weeding out the ones that have dismal scores.

> *Does this sound like feature selection to you?*

Well, the techniques are similar and some researchers have experimented with using attribution techniques for selection, but the major difference between the two lies in their places during the model training lifecycle. Feature selection is generally used before or during the model training while feature attribution is used to explain an already trained model (post hoc explanation).

$f(x)$ for input sample x, a feature attribution method E outputs the relevance of every input feature of x for the prediction. The three basic methods of doing so are:



(1) Gradient-based methods, (2) Surrogate methods, (3) Perturbation-based methods. source

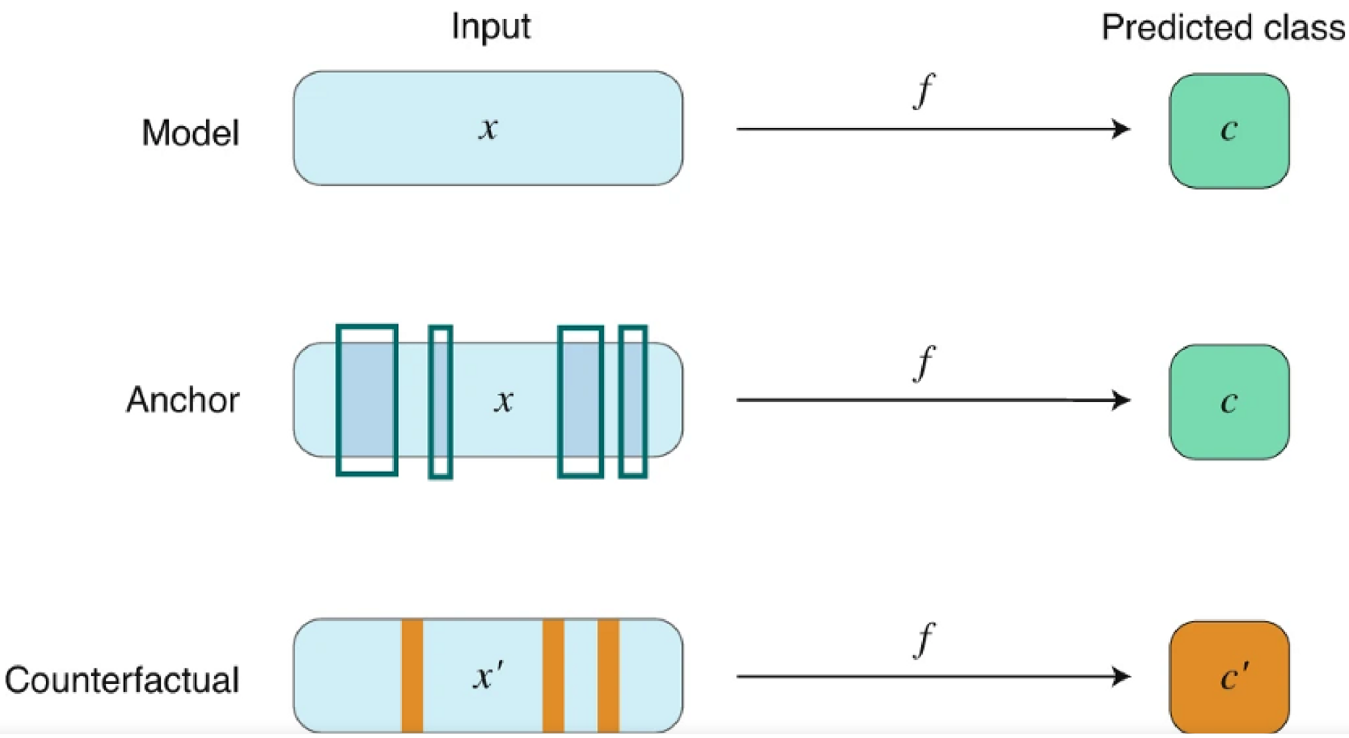
- **Gradient-based methods:** This approach measures how much a change around a local neighborhood of the input x corresponds to a change in the model output $f(x)$. A **common approach** among deep-learning practitioners relies on **the use of the derivative of the output of the neural network with respect to the input i.e., $\delta f/\delta x$ to determine feature importance.** Some of the noteworthy methods include LIME, Deep Learning Important FeaTures (DeepLIFT), Grad-CAM, SmoothGrad, and layer-wise relevance propagation (LRP).

- **Surrogate methods:** As discussed before, this method is usually employed for complex models. **The original function f is approximated by a human-interpretable model g** and used to generate attributions.

The above two methods provide local explanations. While Global surrogate explanation models have been proposed to describe f via a decision tree or decision set model, the approximations may lack precision. Thereby we come to the third information is then used to assess the feature's importance. While perturbation-based methods have the advantage of directly estimating feature importance, they are computationally slow when the number of input features increases, and the final result tends to be strongly influenced by the number of features that are perturbed together.

Recently, Shapley additive explanations (SHAP) was proposed to interpret predictions by complex models such as ensemble methods and deep learning models. This solution combined existing methods such as DeepLift, LIME, Shapley Values, and LRP in various permutations to achieve model-agnostic estimations.

The interpretability of feature attribution methods is limited by the original set of features (model input). Now as the complexity or opaqueness of the input increases, we might need a different approach that does not require decoding of the input type.



instance can be real or generated for the purposes of the method. This approach can provide natural model interpretations for humans because they resemble counterfactual reasoning, i.e., producing alternative sets of action to achieve a similar or different result.

- **Anchor algorithms:** Anchor algorithms offer model-agnostic interpretable explanations of classifier models. They c**ompute a subset of if-then rules based on one or more features that represent conditions to sufficiently guarantee a certain class prediction**. In contrast to many other local explanation methods, anchors therefore explicitly model the coverage of an explanation.

- **Counterfactual instance search:** Given a classifier model f and an original data point x, a counterfactual instance search aims to find examples x' such that (a) x' are as close to x as possible and (b) for which the classifier produces a different class label from the label assigned to x. In other words, **a counterfactual describes small feature changes in sample x such that f classifies it differently**.

- **Contrastive explanation methods:** These **provide instance-based interpretability of classifiers by generating relevant positive and negative sets.** This methodology is related to both anchors and counterfactual search approaches. Relevant positives are defined as the smallest set of features that should be present in an instance for the model to predict a 'positive' result (similar to anchors). Conversely, relevant negatives constitute the smallest set of features that should be absent for the model to be able to sufficiently differentiate from the other classes (similar to a counterfactual instance).

Contrastive explanation methods find such sets by solving two separate optimization problems, namely by (1) perturbing the original instance until it is predicted differently than its original class and (2) searching for critical features in the original input i.e. those features that guarantee a prediction with a high degree of certainty.

entirely different reasoning. An example could be the feature attribution of a cancer prediction model to scientific knowledge, since even a well-performing model may rely on a different set of features.

Another common approach **successively removes features with the highest attribution values and evaluates certain metrics.**

The first metric is model prediction change. These evaluations do not account for nonlinear interactions. For example, if the model learns an OR function of two features and both are active in an instance; the evaluation will incorrectly deem whichever feature was removed first to be useless as its removal does not affect the prediction.

The second metric is model retraining performance. It fails when the retrained model can rely on different features while still achieving the same accuracy. For example, a model might achieve some accuracy by using only feature x1. If a retrained model using only x2 achieves the same accuracy, the evaluation framework would falsely reject the ground truth attribution on x1 due to the same re-training accuracy.

Sources:

- Drug discovery with explainable artificial intelligence
- Comparison of feature importance measures as explanations for classification models
- Do Feature Attribution Methods Correctly Attribute Features?
- Google Cloud AI Explanations Whitepaper

Your tip will go to Gatha Varma through a third-party platform of their choice, letting them know you appreciate their story.

Give a tip

Sign up for Geek Culture Hits

By Geek Culture

Subscribe to receive top 10 most read stories of Geek Culture — delivered straight into your inbox, once a week. Take a look.

Your email

Get this newsletter

By signing up, you will create a Medium account if you don't already have one. Review our Privacy Policy for more information about our privacy practices.