[Red Hat logo]

● ✕ Search
☰ ✕

Search this site…
Search

Enable Architect

- Articles
  Architect career Architectural design strategy Architecture patterns Architecture tools Cloud computing Enterprise architecture design Kubernetes Security
- Portfolio Architecture

Enable Architect

- Articles
  Architect career Architectural design strategy Architecture patterns Architecture tools Cloud computing Enterprise architecture design Kubernetes Security
- Portfolio Architecture

🖎 Subscribe to our RSS feed or Email newsletter. 🔍

Search this site…
GO

3 AI algorithms for understanding business process automation results

Understanding, debugging, and benchmarking AI and ML models is a fundamental issue, especially when their predictions affect business decisions and people's lives.

Posted: March 10, 2022 | | by Rui Vieira (Red Hat)

Image
[Building with a grid pattern]

Photo by Vlado Paunovic from Pexels

Business process modeling and process automation are powerful techniques for businesses to represent, abstract, and automate tasks. These techniques are commonly built on open standards such as the Object Management Group's Business Process Model and Notation (BPMN) and Decision Model and Notation (DMN). These standards allow expressing complex workflows in a well-documented and standardized format, formally verifying them, and applying best practices such as testing and simulation.

Edge computing

- What is edge architecture?
- How to explain edge computing in plain English
- How edge solves problems
- What is edge machine learning?

-

Artificial intelligence (AI) and machine learning (ML) are becoming prevalent in modern life, including in business decisions and process automation. It is increasingly common to find ML predictive models embedded in automation workflows to facilitate automated decision-making, for instance.

Although learning from historical data or classifying and predicting scenarios is beneficial, ML techniques have not always been subject to the same level of transparency, audibility, and interpretability as their process-automation counterparts.

Being able to assess, understand, debug, and benchmark AI and ML models is a fundamental issue when used in processes that could directly impact business decisions and people's lives. Not only is this [an ethical concern](#) but also a legal-compliance issue as AI and ML become more regulated.

This article is an overview of my presentation [Explainable AI for business processing models](#) at DevConf.CZ 2022, which you can watch in its entirety below.

AI and business decisions

In AI/ML, black-box predictive models make inputs and outputs observable, but not their internal functions.

One example is a simple black-box model that recommends whether a loan application should be approved. What if the outputs do not correspond to your intuition? For example, what if people with higher incomes have loan applications rejected at higher rates? Or what if a traditionally underserved group has a higher than expected rate of approvals? Using a model with hundreds of parameters, this might go undetected. How can you understand the model's predictions?

In practice, some model types are so highly complex that, even with access to the training data, algorithm, and predictions, an expert wouldn't be able to explain why the model returned a certain prediction.

It all boils down to understanding the model *in simple terms*. And quite often, you realize that the less you know about it, the less you trust it.

[IT infrastructure plays in important role in making your organization faster, more agile, and more flexible. Learn more in [An architect's guide to multicloud infrastructure](#).]

Explainability algorithms

Explainability or Explainable AI (XAI) is the set of methodologies and algorithms that allow the production of human-understandable explanations for black-box or opaque models.

XAI is an active area of research. It includes **intrinsic methods**, where the complexity of the model is restricted to produce more interpretable ones, and **post-hoc methods**, which deal with an already-trained model with any degree of complexity.

Post-hoc methods can be further divided into **global** or **local explanations**. While global methods aim to provide explanations for a model's general behavior, local explanations restrict information to a small neighborhood of the prediction space. **Post-hoc local** methods can be further divided into **model-specific** explainability (which employs methods targeted to the specific structure and algorithms of certain machine learning models) and **model-agnostic** methods (which provide tools to obtain explanations for any type of model regardless of the structure).

This article provides an overview of three post-hoc, local, model-agnostic explainability methods: LIME, counterfactuals, and SHAP.

What is LIME?

The first explainability method I'll discuss is called **Local Interpretable Model-Agnostic Explanations** (LIME). LIME tries to answer the questions: *Which features are more important?* and *How do they affect the result?*

LIME performs a perturbation of the original data. For example, you could implement permutation by adding noise to the original input. The permutation method clearly depends on the type of features you are dealing with, for instance, image data, text, or tabular data. These new data points are then weighted according to their distance to the original input.

The original model predicts the outcome for each of your simulated data points. Using this new dataset, you train a surrogate model with higher interpretability, and locally it will approximate your black-box model. For instance, you could train a weighted linear regression model and consider the regression weights as the *feature importances*. These importances are contrastive, and in tabular data, they give a quantification of how important each feature is to the final result.

This is useful for interpreting the black-box model prediction because you can, at a glance, quantify which features are more important in relation to your outcome, diagnose your model accordingly, and better understand the decision.

[Check out Red Hat's [Portfolio Architecture Center](#) for a wide variety of reference architectures you can use.]

What are counterfactuals?

Counterfactual explanations answer questions in the form of: *To get this specific outcome, what should my inputs be?*

If you go back to your example loan approval model and a specific input, imagine the model predicts a loan should not be approved. Counterfactuals provide an alternative set of inputs that will lead to the desired outcome (that is, the loan approval) in the form of: *If your income was X and your number of installments was Y, the loan would have been approved*.

The actual desired properties of a counterfactual explanation are part of an active research area, but they mainly focus on the most common ones: validity, sparsity, and actionability.

The **validity** property states that not just any solution that satisfies the desired outcome is a valid counterfactual. A counterfactual should be as close as possible to the original inputs.

Sparsity states that counterfactuals that change the minimum amount of features are preferred. This can be understood from an explainability point of view, as it is easier to interpret the counterfactual if it only changes a few inputs instead of a large number (for a model with a large number of inputs).

The final property is **actionability**, which refers to the ability of the counterfactual to distinguish between mutable and immutable properties.

Cloud architecture

What is SHAP?

The final method I'll present is Shapley Additive explanations (SHAP). Where LIME answered the question: *What is the importance of each feature to my final result?*, SHAP answers the question: *How much did each individual feature contribute to the result?* For example, if you were using a model that would give an estimated value for a used car using a set of vehicle characteristics, LIME would tell *which* characteristics are more important for the result. But SHAP would give a breakdown of how much each characteristic adds to (or removes from) the car's final value.

SHAP relies on a concept from game theory called Shapley values. Shapley values, simply put, try to establish how much each player in a coalition contributed to the final result. To apply the concept of Shapley values to explainability, you consider each input to be a player and the game's result to be the model's prediction.

If you had a model where you could remove inputs and get an outcome based on a partial input, by calculating the difference between the outcome from all inputs and the outcome of a coalition without a certain input, you would get a marginal contribution to the coalition. The outcome of all coalitions that differ by having an input or not is the mean marginal contribution, or Shapley, for that input.

There are two obvious problems here. The first is that the number of coalitions that need to be calculated suffers from a combinatorial explosion with the number of inputs. For all but a few features, you quickly hit prohibitive computational costs. The second problem is that most models do not allow for missing inputs to calculate marginal contributions.

The solution presented by the SHAP authors is to use something called a SHAP kernel. This works by replacing the missing inputs with values taken for a synthetic background dataset, allowing you to calculate the average output for the coalition with the values from the background data taking the place of the missing inputs. This set of coalitions and mean contributions can be viewed as a linear system that, when solved, will provide coefficients that will be equivalent to the Shapley values.

AI's place in process automation

[Skip to the bottom of list](#)

Kubernetes and OpenShift

- [Introduction to Kubernetes architecture](#)
- [Hybrid cloud and Kubernetes: A guide to successful architecture](#)
- [Red Hat OpenShift and Kubernetes ... what's the difference?](#)
- [Get started with Red Hat OpenShift Service on AWS (ROSA)](#)
- [Try for 60 days: Red Hat OpenShift Dedicated](#)

AI and ML are a perfect fit for process automation and decision management. Since, by definition, streamlining and automating tasks is the goal, the ability to enrich decisions and anticipate scenarios is highly desirable. However, accountability, transparency, and auditability are key for trusting AI/ML as a core component of business workflows.

An example of AI/ML explainability in a process automation scenario is TrustyAI, a component of Kogito. Kogito is an open source end-to-end business process automation (BPA) project for building intelligent cloud-

native applications. It is a cloud-first runtime environment. On top of leveraging business process models and decision technologies such as jBPM, Drools, and OptaPlanner, it also provides an explainability service featuring some of the methods mentioned before. The explainability features are also available as a library. Kogito provides a bridge between process automation and machine learning explainability.

Trust the process

I see explainability and interpretability as critical concerns, whether from a legal compliance point of view or a valuable service provided to users and customers. It is also a useful tool for data scientists to debug, profile, and better understand ML models.

As AI/ML becomes more prevalent in all aspects of life, we must have trust in these methods, which due to their inherent complexity, might be difficult to understand without proper tools. Although this is an active area of research, several effective methods are available. When applied to a wide variety of scenarios and model types, they offer good results for explaining and adding interpretability to ML.

As usual, open source communities are taking a leading role here. Many of these tools are available for immediate use and contribution by anyone that wants to be involved and help improve them.

For more on this topic, please view our presentation, [Explainable AI for business processing models](#), from DevConf.CZ 2022.

Image
[Clouds seen through tunnel](#)
[Using Operate First to host cloud-native AI](#)
Learn why Open Data Hub selected Operate First, which applies open source concepts to operations, to host its project.
Posted: January 12, 2022
Author: [Tom Coufal (Red Hat)](#)
Image
[Pen, paper, smartphone, and laptop](#)
[7 AI features changing speech recognition](#)
Artificial intelligence is bringing near-human-level accuracy to automatic speech recognition applications.
Posted: March 9, 2022
Author: [Dylan Fox](#)
Image

Robots overwhelmed by parts (graffiti art)
[DevOps for machines: thinking outside the MLOps spin cycle](#)
AI projects can struggle when they hit production. Machine learning, paired with DevOps, does offer a way around this problem—just beware of the hype around MLOps.
Posted: December 8, 2020
Author: [Gavin Clarke](#)
Topics: [Artificial intelligence](#) [Automation](#)
Author's photo

Rui Vieira

Rui is a Senior Software Engineer at Red Hat working on data science, machine learning, and business automation. He has a PhD in Bayesian statistics, specifically Sequential Monte Carlo methods in long-running streaming data. [More about me](#)

Navigate the shifting technology landscape. Read An architect's guide to multicloud infrastructure.

Related Content

Image
The number 10 painted on a wall
[Planning for 6G wireless, understanding Podman features, using ChatGPT effectively, and more insights for IT architects](#)
Check out Enable Architect's top 10 articles from March 2023.
Posted: April 4, 2023
Author: [Vicki Walker (Editorial Team, Red Hat)](#)
Image
Robots overwhelmed by parts (graffiti art)
[ChatGPT: 3 productivity tips for IT professionals](#)
ChatGPT can help you write better code, faster and expedite problem solving, but you need to be aware of its limitations.
Posted: March 21, 2023
Author: [Oleg Silkin (Red Hat)](#)
Image
Neon lights circling a padlock on top of a computer laptop keyboard
[Security automation planning: Tips from three experts](#)
Looking to implement security automation to help protect your organization from threats? Consider this advice from people who have done it.
Posted: March 28, 2023
Author: [Marjorie Freeman (Editorial Team, Red Hat)](#)

OUR BEST CONTENT, DELIVERED TO YOUR INBOX
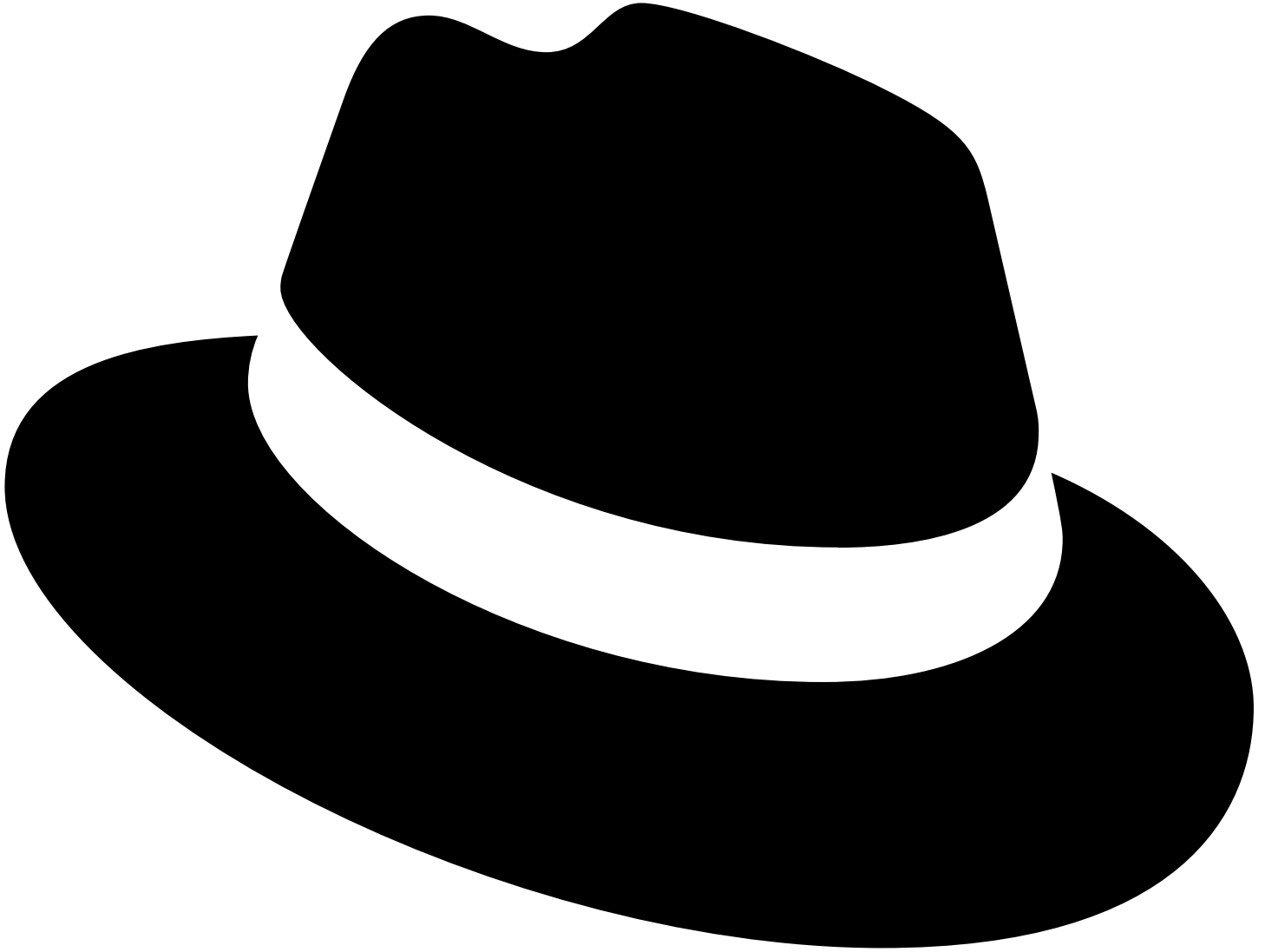
Work email *

Country/Region *
-- Please Select --

Submit

NOT, AND ARE NOT INTENDED TO BE, RED HAT DOCUMENTATION, SUPPORT, OR ADVICE.

- About Red Hat
- Jobs
- Events
- Locations
- Contact Red Hat
- Red Hat Blog
- Diversity, equity, and inclusion
- Cool Stuff Store
- Red Hat Summit

- Privacy statement
- Terms of use
- All policies and guidelines
- Digital accessibility
- Préférences de cookies