NOVA IMS Information Management School

# MDSAA

Master Degree Program in

**Data Science and Advanced Analytics**

# Explainable AI: Enhancing Machine Learning Model Interpretability with Generative AI

Inês Dinis Castelhano

Master Thesis

presented as partial requirement for obtaining a Master's Degree in Data Science and Advanced Analytics

**NOVA Information Management School**

**Instituto Superior de Estatística e Gestão de Informação**

Universidade Nova de Lisboa

**NOVA Information Management School**
**Instituto Superior de Estatística e Gestão de Informação**
Universidade Nova de Lisboa

**Explainable AI: Enhancing Machine Learning Model Interpretability with Generative AI**

by

Inês Dinis Castelhano

Master Thesis presented as partial requirement for obtaining the Master's degree in Data Science and Advanced Analytics, with a specialization in Business Analytics

## Supervised by

Nuno Rosa, PhD, NOVA Information Management School
Márcia Baptista, PhD, NOVA Information Management School

September, 2025

# STATEMENT OF INTEGRITY

I hereby declare having conducted this academic work with integrity. I confirm that I have not used plagiarism, any form of undue use of information or falsification of results along the process leading to its elaboration. I further declare that I have fully acknowledged the Rules of Conduct and Code of Honor from the NOVA Information Management School.

*Sesimbra, 07-09-2025*

*Inês Castelhano*

*To my family and friends, for always being by my side.*

# ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my parents, for always being the first to support me in everything I need and for constantly encouraging me to give my best every single day. You are my greatest inspiration, and I am deeply grateful for your unwavering support and encouragement.

To my family, and especially my grandparents, thank you for your constant support and for reminding me always to fight for what I truly want in life. Your unconditional love and strength have been a guiding force along the way.

To my friends, who are always the first to believe in my potential and to be there whenever I need someone to listen. A special thank you to Catarina, Mariana, and Rute, for standing by my side and supporting me in every moment.

Finally, to Professor Nuno Rosa, thank you for the attention and dedication you gave to this project. The brainstorming sessions and guidance were invaluable, and I am truly grateful for your encouragement to always aim higher and explore new ideas. Your mentorship has been an inspiration.

# ABSTRACT

In today's data-driven world, machine learning (ML) powers critical decisions across sectors such as healthcare and finance, but the opacity of complex "black-box" models undermines trust, accountability, and adoption. This thesis tackles the urgent challenge of explainability by exploring how Large Language Models (LLMs) can transform post-hoc explanations into clear, actionable insights for diverse stakeholders. Moving beyond traditional techniques like SHAP and Counterfactuals, which often overwhelm with complexity, this research introduces a dynamic framework that integrates LLMs as narrative explainers. The methodology combines robust ML pipelines with post-hoc interpreters, enhanced through prompt engineering, to generate explanations in both technical and business-friendly formats. Experiments on real-world datasets, including emergency healthcare and bank fraud detection, benchmarked leading LLMs such as GPT-4o, Claude 3, LLaMA 3, and DeepSeek. Results show that GPT-4o consistently delivers the most accurate, fluent, and stakeholder-aligned explanations, while local open-weight models offer competitive, privacy-preserving alternatives. The evaluation, comprising linguistic heuristics, semantic similarity metrics, and human judgment, demonstrated significant gains in clarity, completeness, and trustworthiness over conventional explainers. Crucially, counterfactual-based narratives proved highly intuitive for decision-making, while SHAP-based explanations achieved greater technical depth. By reframing LLMs as interpretable mediators rather than mere translators, this study provides empirical evidence that generative AI can close the gap between ML performance and human understanding. The contributions extend beyond academic insight, offering practical guidelines for deploying Explainable AI in high-stakes domains where transparency is not optional but essential for fairness, accountability, and trust.



# KEYWORDS

Explainable AI (XAI); Machine Learning; Large Language Models; Post Hoc Explainers; Interpretability


## Sustainable Development Goals (SDG):

# TABLE OF CONTENTS


1. Introduction ..... 1
   1.1. XAI Applications ..... 1
   1.2. XAI Regulamentation ..... 2
   1.3. Problem Definition ..... 3
2. Literature review ..... 5
   2.1. Model Explainability ..... 5
   2.2. Interpretability Methods ..... 6
      2.2.1. Intrinsic Interpretability vs. Agnostic Interpretability ..... 6
      2.2.2. Evaluation of Interpretability ..... 8
   2.3. Model-Agnostic Methods ..... 9
      2.3.1. Partial Dependence Plot ..... 9
      2.3.2. Permutation Feature Importance ..... 10
      2.3.3. Counterfactual Explanations ..... 10
      2.3.4. Local Interpretable Model-Agnostic Explanations ..... 10
      2.3.5. SHapley Additive exPlanations ..... 11
      2.3.6. Comparison of the methods ..... 12
   2.4. Large Language Models ..... 13
      2.4.1. Families of LLMs ..... 13
         2.4.1.1. GPT models ..... 14
         2.4.1.2. Anthropic models ..... 15
         2.4.1.3. Meta models ..... 15
         2.4.1.4. DeepSeek models ..... 16
      2.4.2. Opportunities and Challenges in LLM Interpretation ..... 16
   2.5. Prompt Design ..... 17
      2.5.1. In-Context Learning ..... 18
      2.5.2. Chain-of-Thought Prompting ..... 18
   2.6. Related Work ..... 19
3. Methodology ..... 22
   3.1. Research Design ..... 22
      3.1.1. Design Science Research (DSR) ..... 22
      3.1.2. Cross-Industry Standard Process for Data Mining (CRISP-DM) ..... 23
      3.1.3. Pipeline Architecture ..... 24
   3.2. Data Understanding and Treatment ..... 25
      3.2.1. Medical Information Mart for Intensive Care Emergency Department Data 25

3.2.2. Bank Account Fraud Data ..... 26
3.3. Machine Learning Framework ..... 27
3.4. Post Hoc Explainer Framework ..... 27
3.4.1. SHAP ..... 27
3.4.2. Counterfactuals Explanations ..... 28
3.5. Prompt Design ..... 29
3.6. Textual Explanation ..... 30
3.7. Evaluation ..... 31
4. Results and discussion ..... 35
4.1. Quantitative Analysis ..... 35
4.1.1. SHAP ..... 35
4.1.1.1. Evaluation of Metrics ..... 35
4.1.1.2. Comparative Analysis of the models ..... 38
4.1.1.3. Comparative Analysis in different datasets ..... 39
4.1.2. Counterfactuals Explanation ..... 41
4.1.2.1. Evaluation of Metrics ..... 41
4.1.2.2. Comparative Analysis of the models ..... 43
4.1.2.3. Comparative Analysis in different datasets ..... 44
4.2. Qualitative analysis ..... 45
4.2.1. SHAP ..... 46
4.2.1.1. Human Case Evaluation ..... 46
4.2.1.2. Explanations Examples ..... 48
4.2.2. Counterfactuals ..... 50
Conclusions and future works ..... 53
4.3. Work Contribution ..... 55
4.4. Constraints and Limitations ..... 55
4.5. Future Work ..... 55
Bibliographical References ..... 57
Appendix A ..... 61

# LIST OF FIGURES

Figure 1.1 - Explanations with an overview over different entities and stakeholders ..... 3
Figure 2.1 - Differences between white and black box models ..... 6
Figure 2.2 - Trade-Off between performance and interpretability on ML Models ..... 7
Figure 2.3 - Importance of explainability and performance ..... 8
Figure 3.1 - Design Science Research Framework ..... 23
Figure 3.2 - Cross-Industry Standard Process for Data Mining Framework ..... 24
Figure 3.3 - Final Pipeline Architecture ..... 25
Figure 3.4 - Phases of the Prompting Design (SHAP example) ..... 30
Figure 4.1 - Boxplot of score distributions across all evaluation metrics (SHAP explanations) ..... 38
Figure 4.2 - Mean total score across LLMs of SHAP explanations ..... 39
Figure 4.3 - Boxplot of score distributions across all evaluation metrics (Counterfactuals explanation) ..... 42
Figure 4.4 - Mean total score across LLMs of Counterfactuals Explanations ..... 43

# LIST OF TABLES

Table 2.1 - Metrics considered in a good explanation for a human ..... 9
Table 2.2 - Comparison of post-hoc explainers' methods ..... 12
Table 3.1 - Metrics for evaluating LLMs ..... 32
Table 4.1 - Results of total score of SHAP generated explanations ..... 36
Table 4.2 - Mean total scores by each model and dataset (SHAP explanations) ..... 40
Table 4.3 - Results of total score of Counterfactuals generated explanations ..... 41
Table 4.4 - Mean total scores by each model and dataset (Counterfactuals explanations) ... 44
Table 4.5 - Human Evaluation Results ..... 46

# LIST OF ABBREVIATIONS AND ACRONYMS

| | |
|---|---|
| **AI** | Artificial Intelligence |
| **ALE** | Accumulated Local Effects |
| **BAF** | Bank Account Fraud |
| **CCI** | Charlson Comorbidity Index |
| **CoT** | Chain-of-Thought |
| **CRISP-DM** | Cross-Industry Standard Process for Data Mining |
| **DK** | Design Knowledge |
| **DSR** | Design Science Research |
| **ECI** | Elixhauser Comorbidity Index |
| **ED** | Emergency Department |
| **EHR** | Electronic Health Record |
| **EU AI Act** | European Union's Artificial Intelligence Act |
| **GPT** | Generative Pre-trained Transformer |
| **ICD** | International Classification of Diseases |
| **ICL** | In-Context Learning |
| **ICE** | Individual Conditional Expectation |
| **LCS** | Longest Common Subsequence |
| **LIME** | Local Interpretable Model-Agnostic Explanations |
| **LLMs** | Large Language Models |
| **LLaMA** | Large Language Model Meta AI |
| **ML** | Machine Learning |
| **MIMIC-IV-ED** | Medical Information Mart for Intensive Care in Emergency Department |
| **MoE** | Mixture of Experts |
| **PDP** | Partial Dependence Plot |
| **PG-ICL** | Perturb+Guide ICL |

| | |
|---|---|
| **P-ICL** | Perturb ICL |
| **PLMs** | Pre-Trained Language Models |
| **RAG** | Retrieval-Augmented Generation |
| **RFE** | Recursive Feature Elimination |
| **RL** | Reinforcement Learning |
| **RLHF** | Reinforcement Learning from Human Feedback |
| **SFT** | Supervised Fine-Tuning |
| **SHAP** | Shapley Additive exPlanations |
| **XAI** | Explainable AI |

# 1. INTRODUCTION

Artificial Intelligence (AI) has become a cornerstone of digital transformation, shaping how organizations use data to drive decision-making and innovation. Machine Learning (ML), once confined to academic research, now supports business processes, products, and real-world applications, driven by society's demand for automation and efficiency. At the same time, Large Language Models (LLMs) are emerging as tools to rethink traditional approaches, simplifying complex concepts, clarifying opaque methods, and enabling more interactive ways of interrogating data-driven systems.

However, this progress brings a critical challenge: the ability to interpret and trust AI-driven predictions. In high-stakes fields such as healthcare and finance, understanding why a model reaches a conclusion is as important as the outcome itself, whether explaining why a patient is at risk of readmission or why a loan application is rejected. While simpler models are often chosen for their clarity, the rise of big data has amplified the advantages of more complex approaches capable of capturing intricate patterns in large datasets (Lundberg & Lee, 2017). These powerful models, such as ensemble methods and deep learning architectures, are known as black box models because the reasoning that underlies their predictions remains opaque. In practice, improvements in predictive accuracy are frequently achieved at the expense of interpretability, reinforcing a trade-off between performance and explainability. The resulting lack of transparency introduces significant challenges in understanding model behavior, identifying potential biases, and ensuring accountability (Hassija et al., 2023).

Traditional performance metrics like accuracy or precision, fail to bridge the gap between technical outcomes and human trust. Stakeholders, from data scientists to regulators and decision-makers, now demand not only high-performing models but also clear, comprehensible explanations. This expectation has boosted the rise of Explainable AI (XAI), a field dedicated to making opaque models more transparent through post-hoc interpretability techniques. Increasingly, these efforts intersect with LLMs, which function as versatile reasoning tools. Beyond enhancing interpretability and trust, LLMs also have the capacity to question, reframe, and critically examine model outputs, encouraging deeper reflection on how AI-driven decisions are produced and understood.

## 1.1. XAI APPLICATIONS

Businesses use ML models for impactful decisions, such as loan approvals, where explanations build trust by clarifying unexpected outcomes and addressing concerns about fairness and non-discrimination (Slack et al., 2019). Examples of areas where XAI is relevant include:

- Healthcare: XAI assists in enhancing patient care by supporting critical tasks such as understanding hospital readmissions, optimizing emergency department utilization, detecting diseases, tracking disease progression, and providing accurate diagnoses and treatment

recommendations. Its contributions are vital in healthcare, where decisions directly impact human lives.

- Finance: In finance, XAI supports fraud detection, credit scoring, compliance, and customer service by explaining decisions, reducing risks, ensuring adherence to regulations, and improving customer satisfaction, addressing the sector's high demand for transparency and trust (Reddy et al., 2023).

In addition, XAI is also valuable in sectors such as law, marketing, cybersecurity, and transportation, where it helps with tasks such as legal document analysis and contract review, personalized marketing, threat identification, and optimizing transportation systems (Reddy et al., 2023).

## 1.2. XAI REGULAMENTATION

XAI involves several ethical challenges, particularly related to privacy, which stem from issues like non-representative training data and improper use of personal information. The European Union's Artificial Intelligence Act (EU AI Act) underscores the importance of XAI, particularly for high-risk systems, which prioritize transparency, interpretability, and human oversight, by supporting responsible AI development.

Under Article 13 of EU AI Act, it mandates high-risk systems to ensure output transparency and provide clear instructions on capabilities, limitations, and context for interpretation, while Article 14 requires human oversight to interpret, override, and ensure accountability in decision-making.

From a broader perspective, three categories of concerns regarding AI systems and human oversight were identified by Ali et al. (2023):

- User Concerns: Real-world validation with users is crucial to ensure fairness, transparency, and trust in AI systems that impact individual rights.

- Application Perspective: Clear explanations are critical in high-stakes domains (e.g., healthcare, finance) where AI decisions can impact lives.

- Government Oversight: Regulatory frameworks demand that AI systems meet standards for robustness, data governance, and accountability, scaling oversight according to risk levels.

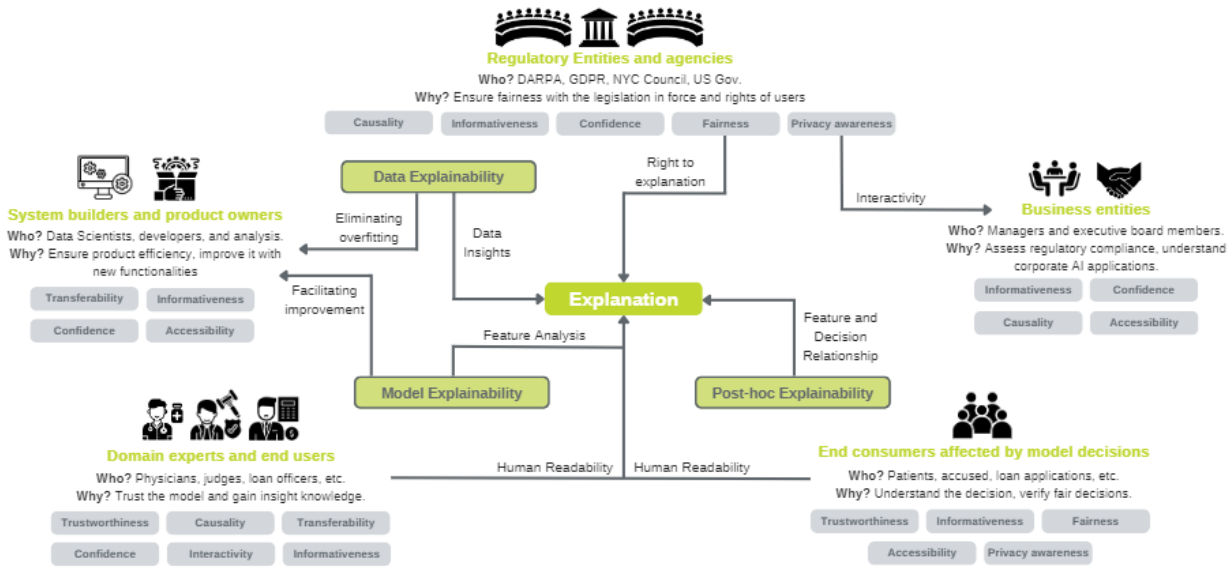



Figure 1.1 - Explanations with an overview over different entities and stakeholders

## 1.3. Problem Definition

Current post-hoc explanation methods, such as SHAP and LIME, have gained popularity to deliver trustworthy interpretations in high-stakes applications. Their strength lies in their ability to explain model outcomes despite opaque internal reasoning (Kroeger et al., 2024). These methods are applied after model training, offering flexibility since they work with any model and can generate various visual explanations (Ribeiro, Singh, and Guestrin, 2016).

Despite their benefits, these explanation methods face limitations that call for the development of new approaches to better understand ML predictions. A key drawback is they often generate explanations or visualizations that appear counterintuitive or inconsistent, which increases the burden of interpretation, risks miscommunication in complex scenarios, and raises the cognitive load on end users (Molnar, 2025). In high-stakes fields, however, stakeholders require models that align with business needs and provide fair, transparent, and easily understandable explanations for both data scientists and decision-makers. In domains such as healthcare and finance, this lack of clarity becomes more than a limitation: it undermines trust, reduces adoption, and creates risks in decision-making. Addressing this gap is the core focus of this work.

LLMs offer a promising opportunity to address these challenges. While originally developed for complex language understanding and reasoning tasks (Radford et al., 2017; Brown et al., 2020), their capacity to generate accessible, text-based explanations remains underexplored (Kroeger et al., 2024). Integrating LLMs with existing methods, such as SHAP or counterfactuals, could enhance interpretability by fitting explanations to different stakeholder needs and fostering trust and actionable insights in high-stakes settings where decisions directly affect lives, financial stability, or legal outcomes.

To address this gap, the research aims to:

- Develop a framework that investigates and compares different LLMs (e.g., GPT, Claude, DeepSeek, LLaMA) for generating post-hoc explanations of ML predictions.

- Incorporate established explainability methods, such as SHAP, and extend the framework with counterfactual explanations to provide complementary perspectives on model interpretability.

- Design two explanation styles, technical and business-oriented, to address the needs of different stakeholder groups, to simplify and enhance trust, transparency, and decision-making.

- Test the effectiveness of the proposed LLM-based framework across diverse datasets (e.g., tabular) and in critical domains such as healthcare and finance.

- Evaluate the quality and impact of LLM-based explanations through heuristic assessment, LLM-based metrics, and a domain-specific use case in the finance sector.

# 2. LITERATURE REVIEW

XAI is a crucial area of study, as understanding model predictions is essential for building trust and ensuring responsible decision-making. The literature on this topic covers a wide range of studies, but this review will focus on three major themes: model explainability (understanding how a model arrives at its decisions), interpretability (gaining insight into its internal mechanisms), model-agnostic methods, and the potential of LLMs as a new tool for generating explanations of model predictions. While these themes are discussed in various contexts, this review will focus on the application of a generative tool that simplifies the use of XAI methods, such as SHAP and Counterfactuals, and enhances ML model explanations for data scientists and stakeholders.

## 2.1. MODEL EXPLAINABILITY

The creation of standards for trustworthy and safe AI systems has led to the growing concern over opaque models. XAI strives to address this challenge by offering alternatives to model interpretation. It offers two primary approaches:

1. Developing transparent white/gray-box models that maintain interpretability but can sacrifice accuracy which restricts its use in everyday applications.

2. Enhancing the interpretability of black-box models, particularly when simpler models cannot achieve the required level of accuracy by using model-agnostic methods.

Adapting explainability strategies to real-world applications is crucial, as AI models often rely on complex, non-linear data that is difficult to interpret. For example, deep neural networks excel at handling complex data by using multiple layers of convolutional filters and millions of parameters to learn intricate patterns. However, their increasing complexity makes internal representations and data flow difficult to interpret, rendering their decision-making untraceable. The same mechanisms that enhance their performance, deep layering, complex connectivity, and optimization functions, also make them opaque, exacerbating the black-box problem. This trade-off between accuracy and interpretability highlights the need for XAI to bridge the gap between performance and trust in AI systems (Ali et al., 2023).

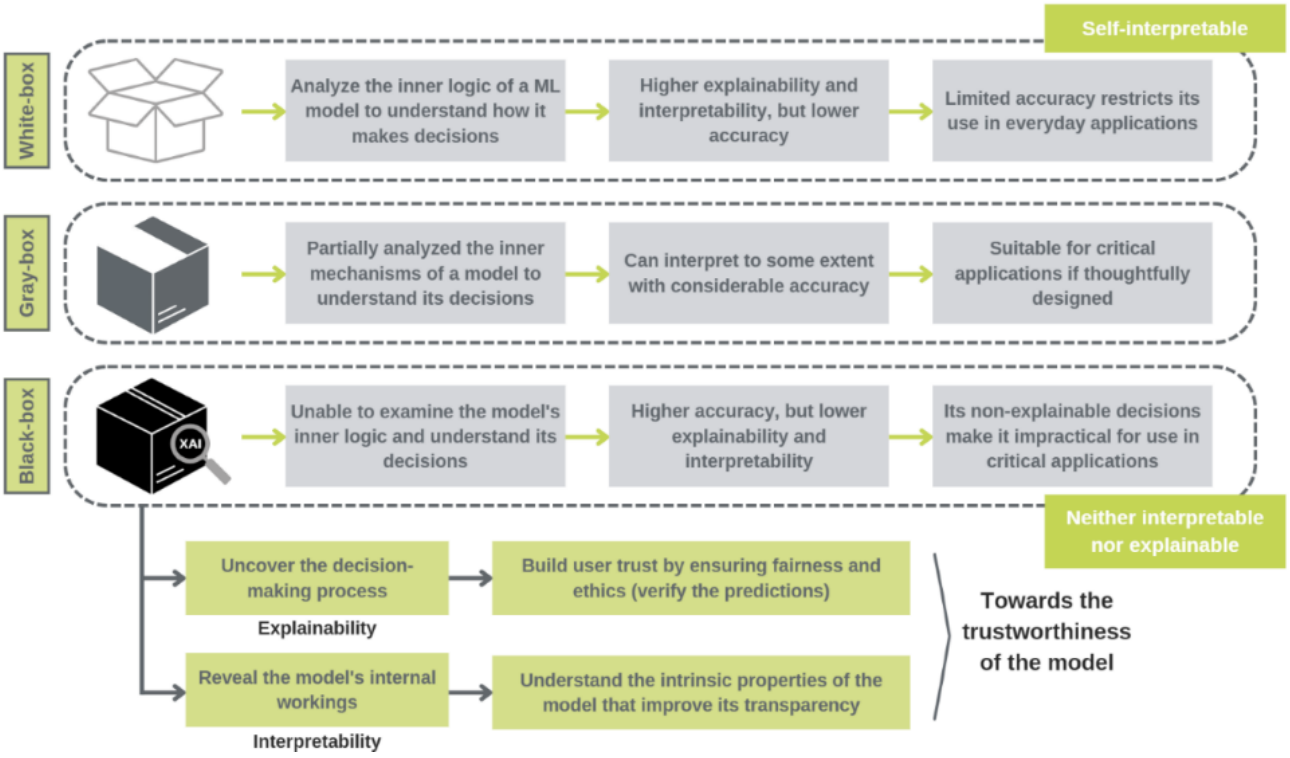



Figure 2.1 - Differences between white and black box models

## 2.2. Interpretability Methods

### 2.2.1. Intrinsic Interpretability vs. Agnostic Interpretability

An important distinction in model interpretability is whether it is built into the model itself (model-intrinsic interpretability) or applied afterward without accessing its internal mechanisms (model-agnostic interpretability) (Ai et al., 2021).

With intrinsic interpretability, the model is designed to be simple and transparent from the start. Data scientists must carefully evaluate model complexity with interpretability, as shown in the Figure 1.3. Simpler models like linear regression and decision trees are often chosen when transparency is prioritized. However, these models may sacrifice performance compared to more complex ones, as they typically assume linear or smooth relationships and rely on straightforward decision processes. In contrast, models like neural networks and ensemble methods (e.g., random forests) achieve higher accuracy by capturing complex, non-linear relationships, but they are harder to interpret and require longer computation times.

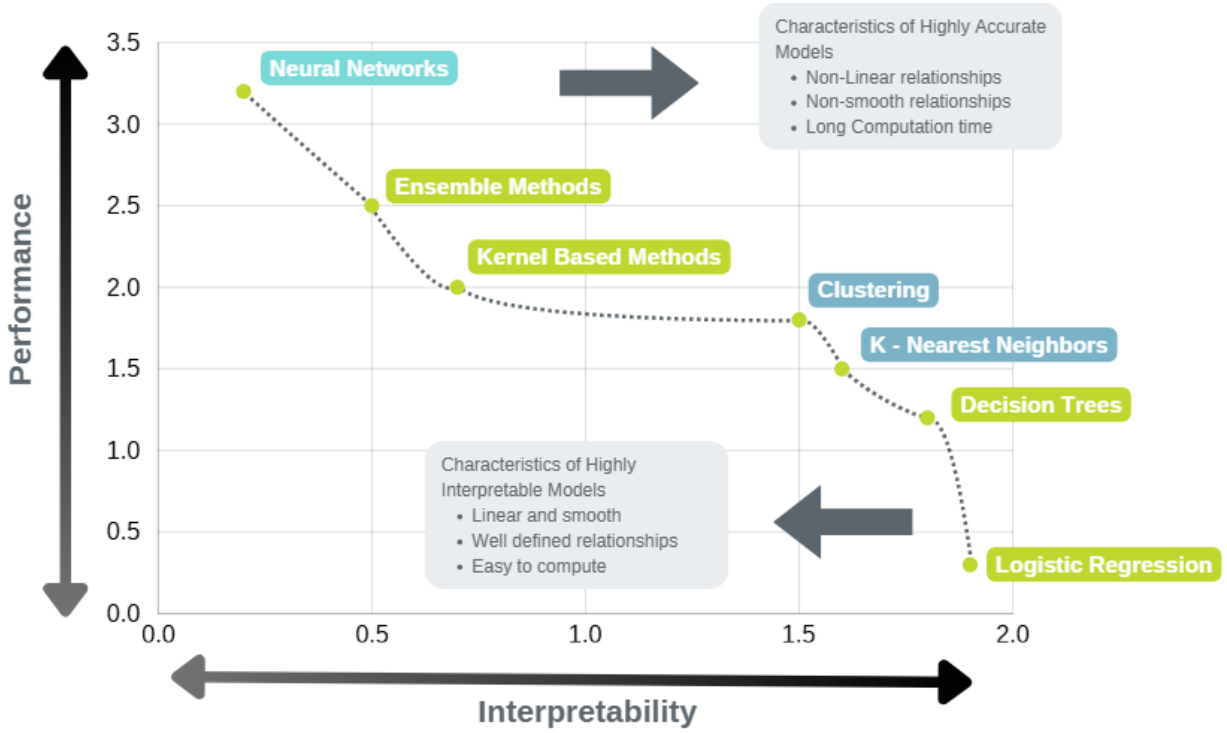



Figure 2.2 - Trade-Off between performance and interpretability on ML Models

Model-agnostic interpretability (or post-hoc interpretability), on the other hand, involves analyzing the model after it has been trained. This approach allows the use of complex models maintaining high performance while still providing some level of interpretability. Post-hoc methods can demonstrate interpretability through visualizations or summary statistics that show how features influence predictions across the dataset.

Post-hoc interpretability methods can further be divided into local and global approaches. Local interpretability explains individual predictions, providing insight into why a specific instance received a particular outcome. On the other hand, global interpretability explains a model's overall behavior by analyzing its average patterns and decision mechanisms, often using expected values based on the data distribution.

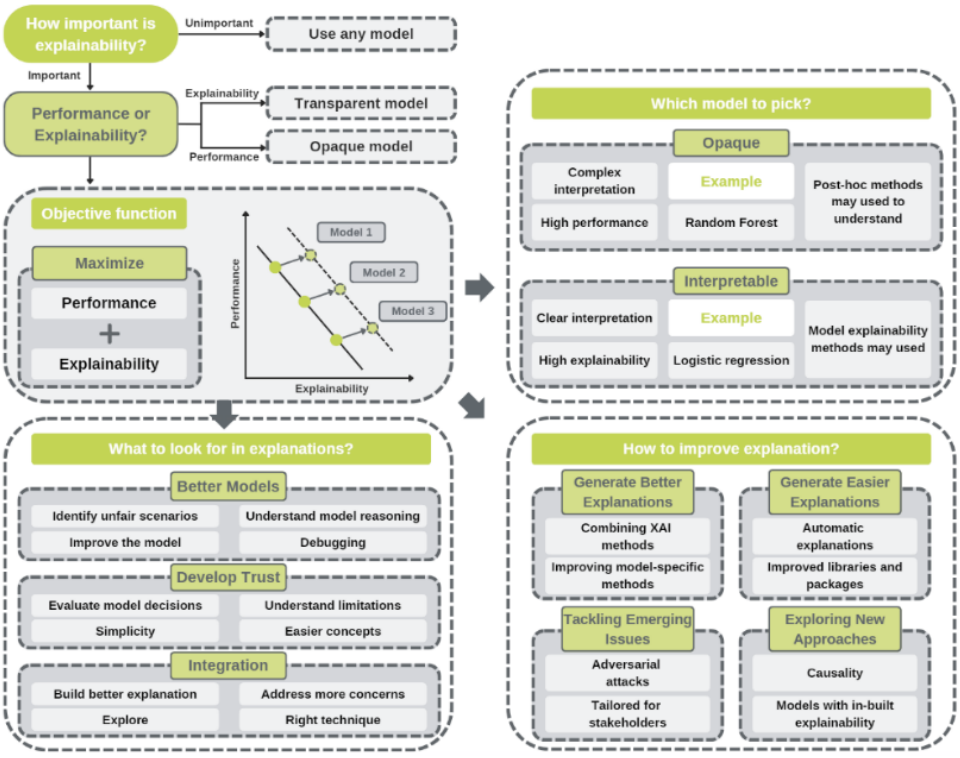



Figure 2.3 - Importance of explainability and performance

### 2.2.2. Evaluation of Interpretability

The evaluation of interpretability does not rely on a precise metric, as the quality of an interpretation depends on various external factors. However, Doshi-Velez and Kim (2017) proposed three main levels for evaluating interpretability:

1. Application-level evaluation: This involves testing the explanation directly in a real-world application by the intended end-users. For example, radiologists could directly evaluate fracture detection software by interacting with it to assess the quality of its explanations.

2. Human-level evaluation: Simplified version of the application-level evaluation. Instead of involving domain experts, the evaluation is carried out with laypersons or non-experts. This approach assesses the interpretability of a model from a more general user's perspective.

3. Function-level evaluation: This assumes that if a class of models has already been assessed and validated through human-level review in prior studies, new studies can focus on the functional performance of the model class rather than involving new human assessments.

The critical question is: what makes a good explanation for a human? A good explanation effectively balances multiple factors, considering the specific needs and context of the user. The following table outlines essential metrics that contribute to the quality of an explanation.

Table 2.1 - Metrics considered in a good explanation for a human

| Metric | Description |
|---|---|
| **Contrastive** | Explains why one outcome occurred over another. |
| **Selective** | Highlights one to three key reasons for clarity. |
| **Socially aware** | Considers the context and target audience. |
| **Truthful** | Provides accurate and reliable information. |
| **General** | Applies across various scenarios. |
| **Probable** | Aligns with human expectations of causality. |

## 2.3. MODEL-AGNOSTIC METHODS

Model-agnostic methods are typically applied post-hoc and can be used with any type of ML model. These methods can be applied to any model, regardless of its architecture (model flexibility), and allow for different forms of explanations (explanation flexibility), including only positive evidence (highlighting features responsible for a prediction) or the contrary. (Ribeiro, Singh, and Guestrin, 2016).

### 2.3.1. Partial Dependence Plot

The Partial Dependence Plot (PDP) is a global method, which shows the marginal effect of one or two features on a ML model's predicted outcome, using the Monte Carlo method. It reveals the relationship between a target and a feature, which can be linear, monotonic, or more complex. It is intuitive and straightforward to implement, making it a useful tool for understanding the main effects of features on predictions. PDPs work for both numerical and categorical features, providing valuable insights into their impact. However, they have notable limitations. PDPs assume feature independence, which can lead to unrealistic outcomes when features are correlated. They only display average effects, potentially obscuring variations within the data. Additionally, they are limited to visualizing one or two features at a time due to dimensional constraints and focus solely on main effects, ignoring interactions between features (Molnar, 2025).

Alternatives to PDPs, such as Accumulated Local Effects (ALE) plots and Individual Conditional Expectation (ICE) plots, address some of their limitations but also have their own drawbacks. ALE plots offer a faster, unbiased approach by calculating differences in predictions rather than averages, which helps isolate feature effects and better handle correlated features. However, they can still produce misleading results when features are strongly correlated, and analyzing second-order effects may prove challenging. ICE plots, on the other hand, focus on individual predictions to show how variations in a single feature impact outcome. They are effective for identifying interactions and heterogeneous effects. Nonetheless, ICE plots are limited to displaying only one feature at a time, can become cluttered when too many lines are plotted, and may create unrealistic data points when features are correlated. Both methods provide valuable alternatives, but they come with trade-offs that should be considered (Molnar, 2025).

### 2.3.2. Permutation Feature Importance

Permutation Feature Importance evaluates a feature's significance by measuring the increase in prediction error when its values are randomly shuffled. Higher errors after shuffling indicate greater importance, while unchanged errors suggest the feature is unimportant. This method calculates the difference or ratio between permuted and original errors, ranking features by their importance scores (Molnar, 2025).

The primary advantage of this approach is its ability to provide a global view of model behavior, capturing both main effects and feature interactions without requiring model retraining, which saves time. However, it has limitations. The randomness of shuffling can lead to inconsistent results, and the method may be biased when features are correlated, creating unrealistic instances and skewing scores. Additionally, correlated features can dilute each other's importance, complicating interpretation, and potentially obscuring relationships (Molnar, 2025).

### 2.3.3. Counterfactual Explanations

Counterfactual explanations identify the minimal changes in input features needed to alter a ML model's prediction, illustrating alternative scenarios. They are selective, focusing on key changes, and contrastive, highlighting differences between current and alternative outcomes. Counterfactuals are intuitive and interpretable, requiring only access to the model’s prediction function, which makes them useful for protecting trade secrets and adhering to data protection rules (Molnar, 2025).

The advantages of counterfactual explanations include their clarity, user-friendly nature, and ability to generate diverse alternatives, offering flexibility in decision-making. However, they have drawbacks. The Rashomon effect, where multiple explanations lead to the same outcome, can overwhelm users. Additionally, ensuring the quality of counterfactuals requires balancing similarity to the original instance, proximity to the target prediction, and realism of feature values. This complexity can pose challenges in generating and interpreting effective explanations (Molnar, 2025).

### 2.3.4. Local Interpretable Model-Agnostic Explanations

Local Interpretable Model-Agnostic Explanations (LIME), introduced by Ribeiro et al. in 2017, interprets predictions of black-box ML models by creating local surrogate models that approximate the model’s behavior around a specific instance (Ribeiro et al., 2016). By generating a dataset of perturbed samples and weighting them based on proximity to the target instance, LIME trains an interpretable model to explain local predictions. It works with various data types, such as tabular data, where features are perturbed, and text data, where words are randomly removed to assess their importance (Ribeiro et al., 2016).

LIME's advantages include its flexibility to explain predictions for any model architecture and its ability to use interpretable features, even when the original model relies on complex or abstract features. It also provides a fidelity measure to evaluate how well the surrogate model reflects the original model's local behavior (Molnar, 2025).

However, LIME has drawbacks. Defining the neighborhood around an instance is challenging due to the difficulty of optimizing the kernel width. If it's too small, it limits relevant data, while if it's too large, it includes distant instances that can affect explanation accuracy. The assumption of feature independence during sampling ignores correlations between features, which can lead to unrealistic data points, ultimately compromising the quality of the explanations. Additionally, LIME is prone to instability, with explanations varying for nearby data points or across repeated sampling runs (Molnar, 2025).

### 2.3.5. SHapley Additive exPlanations

Shapley values, based on cooperative game theory, offer a model-agnostic method to fairly attribute a prediction to individual features by calculating their marginal contributions across all possible feature combinations. They ensure a fair and reliable distribution of contributions, adhering to principles like efficiency, symmetry, dummy, and additivity. This makes Shapley values particularly valuable for providing detailed explanations of model predictions, enabling meaningful comparisons across data points or subsets (Molnar, 2025).

The advantages of Shapley values include their fairness in distributing prediction differences and their theoretical foundation, which enhances explanation reliability. They work well with complex models by using approximations like Monte Carlo sampling to manage computational costs and provide actionable insights into feature contributions. However, Shapley values have obstacles. Exact computation is computationally intensive, and approximations can be challenging to optimize. They are not ideal for sparse explanations, as they account for all features regardless of their relevance. Additionally, they require access to the original data, limiting their applicability compared to methods that rely only on the prediction function. The assumption of feature independence can produce unrealistic samples when features are correlated, and while conditional sampling can address this, it may violate key axioms (efficiency, symmetry, dummy, additivity), complicating interpretation further (Molnar, 2025).

SHAP (SHapley Additive exPlanations), introduced by Lundberg and Lee in 2017, based on game theory, explains individual predictions in ML models. It clarifies each feature's contribution to a model's prediction by treating feature values as players in a coalition game, in a consistent, additive manner. It ensures properties like local accuracy, missingness, and consistency, making explanations reliable and interpretable. SHAP employs two primary methods: KernelSHAP, which works broadly across model types but is computationally intensive, and TreeSHAP, specialized for tree-based models, offering greater efficiency but sometimes generating unintuitive attributions (Molnar, 2025).

Also, SHAP provides versatile visualization tools, including SHAP Feature Importance, which highlights impactful features, SHAP Summary Plots, which combine importance with feature effects, and SHAP Dependence Plots, which reveal feature relationships and interactions (Molnar, 2025).

In terms of advantages, SHAP stands out for its strong theoretical foundation, its ability to balance local and global perspectives consistently, and its capacity to capture feature interactions and contrastive insights. However, it also has limitations: KernelSHAP is computationally expensive and struggles with correlated features, TreeSHAP can produce unintuitive results, and SHAP generally requires access to the underlying data, which may raise practical and interpretability challenges in complex scenarios (Molnar, 2025).

### 2.3.6. Comparison of the methods

Table 2.2 - Comparison of post-hoc explainers' methods

| Method | Local vs Global | Explanation | Strengths | Weaknesses |
|---|---|---|---|---|
| **Partial Dependence Plot (PDP)** | Global | Displays the relationship between a feature and the predicted outcome, averaging over other features. | Easy to interpret, and visualizes average effect of features on predictions | Can be biased with correlated features, issue with the assumption of independence, may not capture interactions and heterogeneous effects |
| **Accumulated Local Effects (ALE) Plot** | Global | Shows the average effect of a feature on predictions, accounting for local correlations. | Unbiased with correlated features and faster to compute than PDP | Interpretation can be misleading with strong correlations, more complex and unsteady plots |
| **Permutations Feature Importance** | Global | Evaluates feature importance by measuring performance drop after permuting feature values. | Simple to implement; straightforward interpretation | Computationally expensive for large datasets and sensitive to feature correlation |
| **Individual Conditional Expectation (ICE)** | Local | Plots the predicted outcome for individual instances as a function of feature values, showing variability. | Reveals heterogeneity in feature effects, and reveals the potential magnitude of variation | Computationally intensive, harder to aggregate results and independence assumption for a single feature |
| **Counterfactual Explanations** | Local | Provides scenarios of what would happen if certain feature values were changed, illustrating decision boundaries. | Provides specific explanations for individual instances and shows how to achieve different outcomes | Can be computationally intensive, may not generalize well across high-dimensional data, and may not be intuitive |
| **Local Interpretable Model-Agnostic Explanations (LIME)** | Local | Fits a simple, interpretable model locally around the prediction of interest to explain the complex model's output. | Provides local interpretability for complex models and simple intuitive results | Limited to local approximations, can be sensitive to perturbations, and incapable of explaining models with non-linear decision boundaries |
| **Shapley Additive Explanations (SHAP)** | Local/ Global | Generalizes Shapley values, allowing for clear and consistent interpretations. | Consistent and additive feature importance and strong theoretical grounding | Can be complex to implement and may be computationally expensive |

The table demonstrates that post-hoc methods have various strengths and weaknesses depending on their application. Despite their usefulness, these methods have significant limitations that raise concerns, particularly in the case of methods like SHAP, which have gained popularity as the primary choice among post hoc explainers. Among these limitations, the tendency to produce counterintuitive explanations and the substantial technical effort required can make the interpretation process more complex and less accessible (Slack et al., 2019). In practice, this can lead to miscommunication or increased cognitive load for end users (Molnar, 2025). These challenges emphasize the need for novel interpretability methods that improve reliability and clarity in practical applications. One promising direction is the use of LLMs, which offer the potential for providing clear, text-based explanations to stakeholders, thus addressing some of the limitations of traditional methods.

## 2.4. LARGE LANGUAGE MODELS

LLMs, such as GPT-4.5 (OpenAI), LLaMA-2 (Meta), Bard (Google), and Claude-2 (Anthropic), represent a significant advancement in modern AI research and applications. These transformer-based neural networks, containing tens to hundreds of billions of parameters, are pre-trained on extensive text data, excelling at tasks like language understanding, generation, and explanation. Compared to earlier models like BERT, LLMs demonstrate enhanced capabilities in machine translation, question-answering, and other natural language tasks (Singh et al., 2024).

A decisive factor in their success is their ability to understand long-range dependencies and contextual nuances by self-attention mechanisms. These advancements enable LLMs to transform complex information into accessible language, improving decision-making and learning experiences for both technical and non-technical users (Mavrepis et al., 2024). Models like ChatGPT stand out due to features such as their open API, diverse training data, and versatility, which make them highly effective across a range of specialized tasks.

Another advantage of LLMs lies in their scalability. Scaling laws demonstrate that increasing model size, data volume, and computational resources significantly enhances their performance. Typically, these models undergo a two-phase process: pre-training followed by instruction fine-tuning and alignment with human preferences. This process further improves their ability to follow instructions and engage interactively. Methods like prompting, including few-shot prompting, allow LLMs to adapt to specific tasks by providing task-specific input, showcasing their versatility and application breadth (Zhao et al., 2024).

### 2.4.1. Families of LLMs

This section introduces some of the most widely used families of LLMs: OpenAI's GPT models, Anthropic's Claude models, Meta's LLaMA models, and DeepSeek's models. These LLMs differ in architecture, training data, and design goals, but all play a major role in current AI research and applications. Understanding their core features and development helps frame the broader context of how LLMs are used in data science.

#### 2.4.1.1. GPT models

The GPT (Generative Pre-trained Transformer) family, developed by OpenAI, is one of the most widely recognized series of LLMs built on a decoder-only architecture and trained using an autoregressive approach with masked self-attention mechanisms. These models are particularly well-suited for tasks such as text generation, conversational AI, summarization, translation, and content creation. Over time, the GPT family has evolved significantly in terms of scale, training strategies, and capabilities (Kampelopoulos etal., 2025).

The GPT series advanced significantly from GPT-1 to GPT-3, with GPT-3 reaching 175 billion parameters and offering strong language generation capabilities. However, GPT-3 struggled with code reasoning, instruction following, and safety. To address these issues, OpenAI introduced Codex, a 12B-parameter model trained on code, and developed GPT-3.5 models like text-davinci-002, which were fine-tuned to enhance reasoning and reduce harmful outputs. (Kalyan et al., 2024).

ChatGPT, introduced in late 2022, was based on GPT-3.5-turbo and later extended to GPT-4. It was optimized for dialogue and trained using human-annotated conversational data, contributing to improved performance in multi-turn interactions and safer responses (Kalyan et al., 2024).

OpenAI introduced GPT-4 in March 2023, further improving performance across a range of tasks. Unlike its predecessors, GPT-4 supports multimodal inputs (text and images), exhibits stronger reasoning abilities, and generates safer, more aligned responses. Its development included a six-month safety alignment phase and interventions such as red teaming and the use of an additional safety reward signal during RLHF (Reinforcement Learning from Human Feedback) training (Zhao et al., 2023). GPT-4 was also based on a Mixture of Experts (MoE) architecture with 1.7 trillion parameters, where only a fraction is active per inference (Kampelopoulos etal., 2025).

Later, OpenAI launched *GPT-4 Turbo*, which provided enhanced capabilities, including a longer context window (up to 128k tokens), improved efficiency, and support for tools like function calling and text-to-speech. GPT-4 Turbo also supports advanced multimodal interactions including vision and speech, as well as integration through the Assistants API, enabling goal-oriented agent development (Zhao et al., 2023).

Currently, *GPT-3.5 Turbo*, *GPT-4*, and *GPT-4 Turbo* remain core components of OpenAI's deployed models, each offering different trade-offs in terms of capability, speed, and cost. These models reflect an ongoing evolution in balancing scale, safety, usability, and specialization, particularly with emerging use cases in both general and domain-specific AI systems.

#### 2.4.1.2. Anthropic models

The Claude family of models, developed by Anthropic, follows a decoder-only architecture optimized for safety, alignment, and commercial usability. The models are well-suited for tasks such as long-document processing, open-ended dialogue, coding, summarization, and multimodal analysis. Over its evolution, from Claude 1 to Claude 3, Anthropic has made significant improvements in reasoning, context window length (up to 200,000 tokens), and multilingual support (Kampelopoulos etal., 2025).

Claude 3, released in March 2024, represents a major advancement. The family includes three sub-models, Claude 3 Haiku, Claude 3 Sonnet, and Claude 3 Opus, each optimized for different trade-offs in speed, cost, and performance. Opus is the most powerful, excelling in complex reasoning, math, and coding tasks, Sonnet balances performance and efficiency, and Haiku is the fastest and most cost-effective option, still supporting multimodal input such as charts, images, and graphs ((Anthropic, 2024).

The Claude 3 models are trained using techniques like unsupervised learning and Constitutional AI, which aligns model behavior with human values by applying principles derived from sources like the UN Declaration of Human Rights. An additional principle was introduced in Claude 3 to reinforce respect for disability rights. These models also underwent reinforcement learning from human feedback (RLHF) and extensive red-teaming to reduce harmful outputs (Anthropic, 2024).

Claude 3 models support function calling (tool use) and integrate well into applications via platforms like Claude.ai, Amazon Bedrock, and Google Vertex AI. They are trained on a mix of public, licensed, and internally curated datasets, with a knowledge cutoff in August 2023. They do not retain user interaction data and adhere to ethical crawling and data usage practices (Anthropic, 2024).

While Claude models show strong performance across multiple domains and languages, limitations remain, such as potential for hallucination, reduced precision with low-resolution images, and challenges with low-resource languages. Anthropic acknowledges these areas for improvement and continues iterative development toward more robust and aligned AI systems (Anthropic, 2024).

#### 2.4.1.3. Meta models

The LLaMA (Large Language Model Meta AI) family, developed by Meta AI, consists of open-weight decoder-only models designed for efficiency, performance, and accessibility. Since its initial release in 2023, the LLaMA series has evolved through multiple versions, LLaMA-1, LLaMA-2, LLaMA-3, and most recently LLaMA-3.1, with model sizes ranging from 7 billion to 405 billion parameters. LLaMA 3 models notably support an extended context window of up to 128,000 tokens and were pretrained on 15 trillion tokens, greatly enhancing their reasoning and generation capabilities (Kampelopoulos etal., 2025; Zhao et al., 2023)

The LLaMA 3 family includes the 70B model, which offers strong performance in a wide range of language tasks and has become one of the most popular open-weight alternatives to proprietary LLMs (Zhao et al., 2023).

Due to their open nature, relatively lower computational cost, and flexibility for fine-tuning, LLaMA models have become foundational in both academic research and real-world deployment. The local availability of models like LLaMA 3 70B enables experimentation in privacy-sensitive or resource-constrained environments, expanding access to high-performance LLMs.

#### 2.4.1.4. DeepSeek models

The DeepSeek-R1 series, developed as part of the open-source DeepSeek initiative, represents a significant step forward in enhancing reasoning capabilities in LLMs through large-scale reinforcement learning (RL). Unlike traditional pipelines that rely heavily on supervised fine-tuning (SFT), DeepSeek-R1 introduces an RL-first approach that enables complex reasoning, chain-of-thought (CoT) generation, and self-verification without requiring prior SFT. A notable variant, DeepSeek-R1-Zero, achieves this purely through RL, marking a novel contribution to LLM development (DeepSeek-AI et al., 2025).

The training pipeline of DeepSeek-R1 integrates multiple stages: cold-start SFT with CoT examples, reasoning-focused RL, rejection sampling for response refinement, and a final RL alignment phase to optimize for helpfulness and safety. This approach leads to improvements in a broad range of tasks including math, logic, factual QA, and creative writing. The model generates "thinking tokens" to explicitly trace its reasoning process, improving transparency and problem-solving clarity (Neha et al., 2025).

Despite these strengths, current limitations include sensitivity to prompt formats (favoring zero-shot over few-shot), mixed-language handling issues (mainly optimized for English and Chinese), and slower RL scalability for software engineering tasks. Future work aims to refine these areas through better prompt engineering, asynchronous RL, and targeted evaluation improvements (DeepSeek-AI et al., 2025).

### 2.4.2. Opportunities and Challenges in LLM Interpretation

LLMs provide unique opportunities for interpretation by offering natural-language interfaces that explain complex patterns intuitively and interactively. Their ability to generate explanations at varying levels of detail, supported by evidence or counterfactuals, makes them accessible to both technical and non-technical users. Interactive capabilities further enhance this accessibility, allowing users to ask follow-up questions and explore customized examples. These features are particularly valuable in high-stakes fields, where improved decision-making depends on clear and detailed explanations (Singh et al., 2024). Additionally, LLMs can explain smaller models and even their own operations, using techniques like fine-tuning and in-context learning to reduce their black-box nature (Singh et al., 2024).

Despite these interpretability strengths, significant challenges remain. LLMs are prone to hallucinations, generating unsupported or incorrect explanations. Strategies like self-verification, iterative prompting, and automated consistency checks are necessary to mitigate these issues. Furthermore, the complexity and size of LLMs make them difficult to interpret, particularly when full model access is unavailable (Wu et al., 2024).

In addition, specific interpretability techniques still face limitations. Attention mechanisms, which visualize token contributions, are helpful but remain debated in their effectiveness. Post-hoc feature attributions and natural-language explanations can clarify model predictions but are often prone to inaccuracies. Advanced reasoning techniques, such as chain-of-thought prompting, tree-of-thoughts, and graph-of-thoughts, as well as retrieval-augmented generation (RAG), seek to improve explanation clarity (Singh et al., 2024).

At a broader level, global explanations seek to address critical concerns such as bias, privacy, safety, and efficiency, which are essential for building trust in LLMs. Techniques like probing are widely used to analyze model representations, including attention heads, embeddings, and neurons. However, scaling these methods to large LLMs presents challenges, prompting innovations such as semi-automated techniques and the use of smaller LLMs for testing specific phenomena like attention response or in-context learning (Singh et al., 2024).

To overcome these challenges, research must prioritize improving explanation reliability by grounding outputs in factual data. Approaches like self-verification, iterative prompting, and automated consistency checks should be explored further. Additionally, creating interactive explanations that allow users to audit and refine model outputs in real time can enhance usability. As the field advances, the integration of richer contextual information and more robust evaluation frameworks will be critical to ensuring that LLMs deliver meaningful and reliable explanations at scale.

## 2.5. PROMPT DESIGN

LLMs enhance both interpretability and usability through methods that include partially interpretable models and interactive explanations. Partially interpretable models often employ chains of prompts, enabling decomposition of decision-making processes to analyze dataset patterns. These chains, whether human-constructed or generated on-the-fly, enable tasks like dataset explanation by creating more transparent processes. For instance, explanation trees or self-verification chains improve model reliability by prompting LLMs to recheck their outputs. Additionally, retrieval-based decision-making and tool integration further enhance transparency by incorporating external steps, such as retrieving context from a database or performing arithmetic calculations (Singh et al., 2024).

Prompting is the primary method for utilizing LLMs to address a diverse range of tasks, with the quality of prompts being essential in determining performance. Effective prompt creation, often referred to as prompt engineering, involves crafting clear, task-specific instructions. These prompts include detailed input and output specifications, contextual information, and

relevant constraints. For specialized data inputs, such as knowledge graphs or tables, structured data can be linearized or transformed into programmatic formats to improve LLM readability and processing (Zhao et al., 2024).

Zytek et al. (2024) illustrate the importance of tailored prompts in generating high-quality narratives. Initially, structured prompts produced accurate but generic outputs that lacked style customization. Incorporating hand-written exemplars improved narrative style and customization. To address the scalability challenges of manually written examples, bootstrapped few-shot techniques were introduced, allowing the generation of high-quality exemplars. This approach significantly enhanced narrative fluency and alignment with user preferences, highlighting the importance of carefully designed prompts in optimizing LLM outputs.

Critical design principles emphasize clarity in task goals to reduce ambiguity and guide the model effectively. Decomposing complex tasks into simpler steps, providing high-quality few-shot demonstrations, and adhering to model-friendly formats such as structured separators are key strategies for successful prompt design. Moreover, using role-playing capabilities within prompts can enhance performance in domain-specific tasks. These principles collectively optimize LLM understanding and usability, enabling effective solutions across a wide range of applications (Zhao et al., 2024).

### 2.5.1. In-Context Learning

Emergent abilities in LLMs refer to capabilities that are absent in smaller models but manifest in larger ones, often with a sudden and significant improvement in performance once the model reaches a certain scale. This phenomenon highlights the distinctive nature of LLMs compared to smaller pre-trained language models (PLMs). These abilities are particularly evident in complex, general tasks, enabling LLMs to handle a wide variety of applications (Wu et al., 2024).

One of the most prominent emergent abilities is in-context learning (ICL), introduced by GPT-3. ICL enables LLMs to process natural language instructions and task demonstrations embedded directly in a prompt, allowing the model to generate desired outputs without additional training or gradient updates (Wu et al., 2024).

ICL allows LLMs to adapt to new tasks by embedding a few task-specific examples directly into prompts, bypassing the need for retraining or fine-tuning. This capability leverages the reasoning and generalization strengths of LLMs, enabling their application across diverse domains (Kroeger et al., 2024).

### 2.5.2. Chain-of-Thought Prompting

Chain-of-Thought (CoT) prompting is a technique designed to enhance the reasoning abilities of LLMs by guiding them through intermediate steps in complex tasks such as arithmetic,

commonsense reasoning, and symbolic problem-solving. Unlike traditional prompt-based learning, which relies on simple input-output pairs, CoT prompting encourages LLMs to generate intermediate reasoning steps, thereby improving their ability to connect inputs to outputs effectively (Zhao et al., 2024).

While textual CoTs work well for many tasks, they may lack precision in handling more complex reasoning. In such cases, code-based CoTs provide a more structured and logical framework. Some advanced methods even combine text and code, leveraging the strengths of both approaches to break down tasks into manageable parts and further enhance reasoning capabilities (Zhao et al., 2024).

To improve the performance and stability of CoT prompting, several strategies have been developed. One key approach is designing better prompts that include diverse reasoning paths, or multiple CoTs, offering various ways of solving the same problem. This diversity leads to more robust and accurate reasoning outcomes (Zhao et al., 2024).

For generating effective CoTs, two primary methodologies have emerged:

1. Sampling-based methods: These involve generating multiple reasoning paths and selecting the most consistent one through majority voting (Zhao et al., 2024).
2. Verification-based methods: These focus on ensuring the correctness of reasoning steps, using trained verifiers or the LLM itself to validate outputs. Some approaches also incorporate backward reasoning to check the consistency of final answers (Zhao et al., 2024).

For more complex tasks, basic linear CoT structures are extended into tree-based or graph-based reasoning frameworks. Tree-structured methods, such as Tree of Thoughts, explore multiple reasoning paths in parallel, allowing for backtracking and lookahead. Graph-based methods, like Graph of Thoughts, capture intricate relationships between reasoning paths but come with higher computational costs (Zhao et al., 2024).

The effectiveness of CoT prompting is largely attributed to its structured approach, particularly in models trained on code, which provides a stronger logical foundation. Research highlights the importance of symbols (e.g., numbers in arithmetic), patterns (e.g., equations), and text (e.g., tokens) in facilitating the reasoning process. These elements combine to make CoT prompting a powerful tool for improving LLM reasoning capabilities (Zhao et al., 2024).

## 2.6. Related Work

Recent advancements in LLMs highlight their potential for generating post-hoc explanations of black-box ML models, although this area remains underexplored (Kroeger et al., 2024). To address this gap, researchers have developed strategies for ICL to analyze and explain ML predictions by focusing on local neighborhoods of input data. Two approaches, Perturb ICL (P-ICL) and Perturb+Guide ICL (PG-ICL), employ structured prompts to identify features

influencing model outputs. P-ICL uses task-specific prompts, local input-output examples, and guiding questions to engage LLMs in chain-of-thought reasoning for feature ranking, while PG-ICL extends this with step-by-step instructions to systematically evaluate feature importance. Supported by neighborhood sampling, these methods approximate local model behavior efficiently and generate natural language explanations that balance interpretability with computational scalability, offering a promising alternative to traditional methods like LIME and SHAP (Kroeger et al., 2024).

Building on this idea of translating complex model reasoning into more user-friendly explanations, Zytek et al. (2024) proposed an automated grading system that improves the clarity and accessibility of ML explanations. Their framework transforms SHAP values into readable narratives using two subsystems: NARRATOR, which leverages LLMs to generate natural language explanations, and GRADER, which evaluates them against accuracy, completeness, fluency, and conciseness. This pipeline not only improves accessibility but also scales evaluation by reducing dependence on human reviewers. Nonetheless, their findings also highlight challenges related to missing contextual information, which can limit the depth and usefulness of generated narratives, for instance, when explaining outcomes that depend on broader feature distributions or domain-specific benchmarks. Future research can address these limitations by providing richer context to improve the fluency and clarity of the generated narratives, ultimately aiding users in better comprehending the factors influencing model predictions.

In parallel, Fredes et al. (2024) explored combining causal reasoning and counterfactual explanations with LLMs to enhance interpretability in ML. Counterfactuals provide contrastive "what-if" explanations, but raw examples are difficult for end-users to interpret. To address this, the authors proposed a multi-step pipeline where LLMs identify causal factors, validate their relevance with program-aided prompts, and generate actionable natural language explanations. Experiments with tabular classifiers showed that larger sets of counterfactuals improve explanation validity, while prompting strategies like Zero-Shot and Tree-of-Thought enhance reasoning quality. They also introduced closed-loop evaluation, testing whether LLMs can regenerate valid counterfactuals from their own explanations. Overall, this work demonstrates the promise of LLM-based synthesis for making counterfactuals more understandable, while emphasizing the need for human-centered evaluation of explanation quality and trustworthiness.

Finally, complementing these technical advances, Suh et al. (2025) take a critical perspective on the role of LLMs in XAI. They caution against using LLMs solely as translators of technical outputs into natural language, as this may increase readability without fostering genuine understanding and can even lead to overreliance. Instead, they propose reframing LLMs as "devil's advocates" that actively interrogate model explanations by surfacing uncertainties, alternative interpretations, biases, and limitations. Their position emphasizes strategies such as adversarial prompting, varying explanation depth to user expertise, and encouraging

counterfactual reasoning to foster critical engagement. This perspective highlights the need to move beyond agreeable narratives toward explanations that foster skepticism and deeper reflection, aligning with emerging approaches that use LLMs not only for synthesis but also for reasoning and contestation, and to complement the constructive pipelines proposed by Kroeger, Zytek, and Fredes.

# 3. METHODOLOGY

This section provides a detailed explanation of the methods used to generate textual explanations based on the outputs of post-hoc explainers. The chapter begins with an overview of design science research and the CRISP-DM methodology in the context of this project, followed by a description of the final architecture, which integrates multiple methods into a unified pipeline. This approach follows a quantitative methodology, encompassing several key stages: data collection, data preprocessing, model development, application of post-hoc explainability techniques, and the design of a prompt framework for generating textual explanations.

## 3.1. RESEARCH DESIGN

This study employs a combination of the design science research methodology and the CRISP-DM framework to develop a method using LLMs for generating post-hoc explanations of ML predictions. These methodologies will be detailed in the following subsections, providing a comprehensive overview of their application in this research.

### 3.1.1. Design Science Research (DSR)

Design Science Research (DSR) is a problem-solving approach aimed at enhancing human knowledge through the creation of innovative artifacts and the generation of design knowledge (DK) (Hevner et al., 2004). This framework consists of three key components. First, the environment defines the problem space, including people, organizations, and technological infrastructures, by identifying real-world needs to ensure research relevance. Second, the knowledge base provides theories, models, methods, and prior research findings, forming the foundation for guiding the research process. Finally, design activities follow an iterative process with two core stages: the build phase, where artifacts are created using existing and new DK, and the evaluate phase, where their effectiveness and utility are assessed. DSR addresses real-world problems by linking research to stakeholder needs and leveraging existing knowledge bases. When novel solutions are needed, DSR combines, revises, and extends existing knowledge to develop innovative artifacts. The iterative cycle of building and evaluating ensures research outcomes contribute to both theoretical advancements and practical applications.

The main objective of this framework is to support the development of a new method using LLMs to generate post-hoc explanations for ML predictions. This project can be structured as follows:

1. **Environment**: It is shaped by the needs of technical stakeholders, such as data scientists, who require a more efficient and intuitive tool for reporting ML model outputs. Additionally, in critical fields like finance and healthcare, non-technical

stakeholders can benefit from this tool by gaining a deeper understanding of ML, fostering trust and fairness within their respective organizations.

2. **Knowledge Base**: It is built on prior research that has explored the use of LLMs to explain post-hoc explainers' outputs (SHAP and Counterfactuals), serving as a foundation for constructing this framework.
3. **Design Activities**: The first phase focuses on generating explanations using post-hoc explainers' outputs from various ML models applied to different problems in healthcare and finance. The second phase involves evaluating the quality and effectiveness of the explanations generated by the LLMs, ensuring their relevance and usability for stakeholders across different domains. This evaluation combines quantitative assessment using statistical and model-based metrics with qualitative feedback from domain stakeholders.

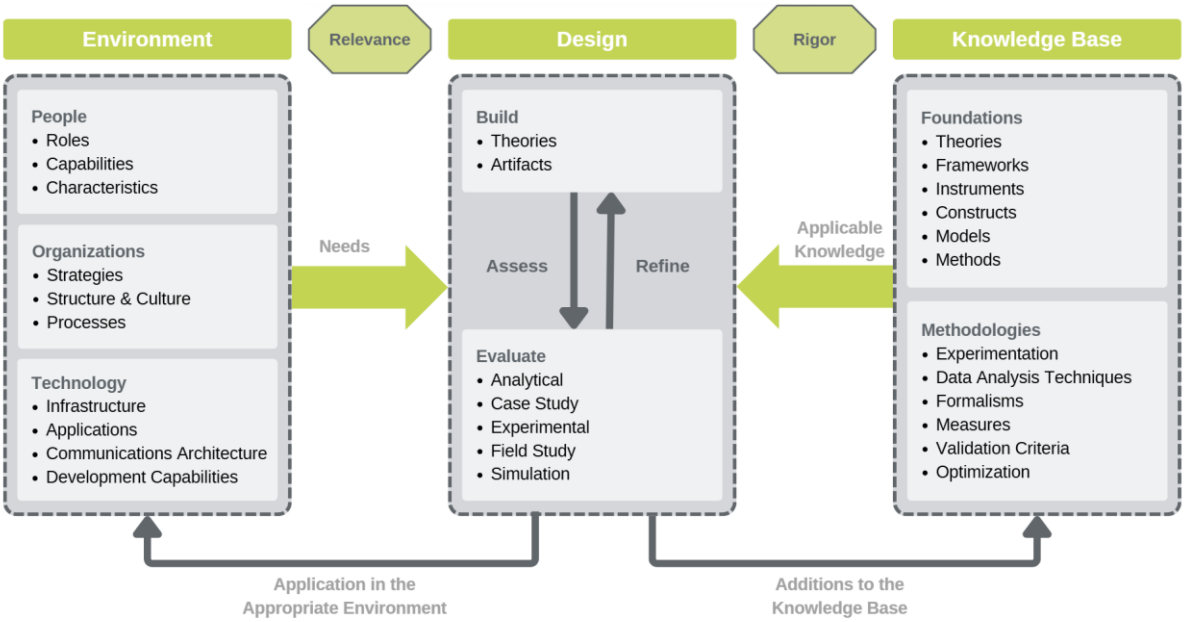



Figure 3.1 - Design Science Research Framework

### 3.1.2. Cross-Industry Standard Process for Data Mining (CRISP-DM)

CRISP-DM (Cross-Industry Standard Process for Data Mining) is a widely used process model that provides a structured approach for conducting data mining projects. It consists of six iterative phases that guide the workflow from business understanding to deployment.

- **Business Understanding:** Defines objectives, assesses resources, and establishes success criteria.
- **Data Understanding:** Collects, explores, and assesses data quality using statistical analysis to describe attributes and relationships.
- **Data Preparation:** Selects relevant data, cleans inconsistencies, and constructs necessary attributes for modeling.

- **Modeling:** Chooses modeling techniques, builds models, and evaluates them based on business goals and predefined criteria.
- **Evaluation:** Assesses model performance, interprets results, and ensures alignment with business objectives.
- **Deployment:** Implements final outputs as reports or software components, including planning, monitoring, and maintenance.

This structured framework provides a standardized approach to data science projects, ensuring consistency and efficiency. The next section explores its phases in greater detail, highlighting their application in the development of this project.

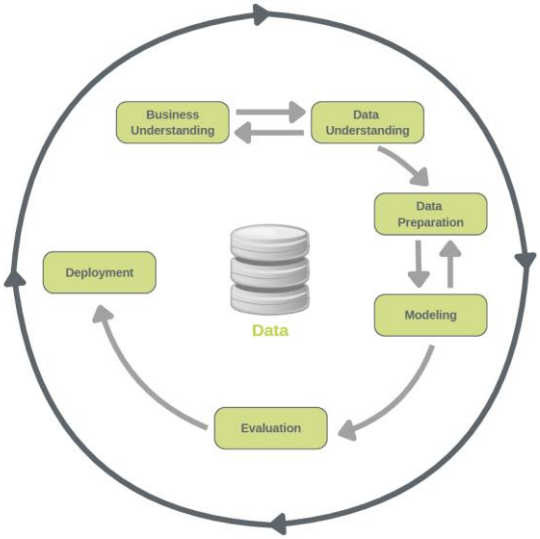



Figure 3.2 - Cross-Industry Standard Process for Data Mining Framework

### 3.1.3. Pipeline Architecture

The pipeline architecture comprises six distinct phases, integrating elements of both the DSR methodology and the CRISP-DM framework. These phases include data understanding and treatment, modeling, post-hoc explanation, prompt design, textual explanation, and evaluation. The evaluation phase comprises heuristic/LLMs metrics and a use case with human evaluation. As feedback evolved during human evaluation, an additional experiment with counterfactuals was conducted, following the same phases (prompt design, textual explanation, and evaluation) as SHAP. While CRISP-DM is inherently embedded within this structure due to its overlapping phases, this project necessitates the development of a tailored framework. Consequently, additional steps such as post-hoc explanation, prompt design, and textual explanation have been incorporated to address the specific requirements of this study.

Furthermore, it is important to highlight that the first two phases, differ in their sub steps due to the utilization of multiple datasets. These variations particularly impact the methods applied for data preprocessing, model selection, and parameter tuning.

The overall architecture is designed to establish a systematic and efficient process framework, ensuring a comprehensive understanding of diverse business cases among all stakeholders. This is ultimately achieved through the final textual output generated by the system.

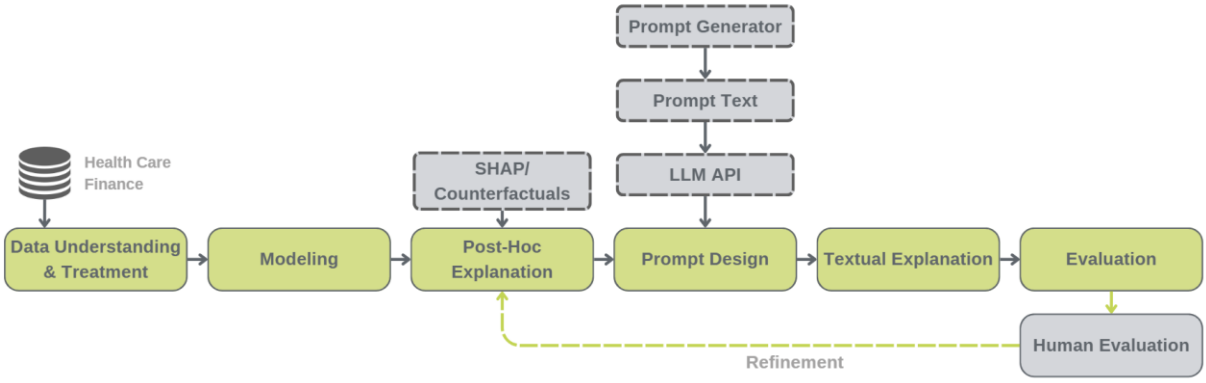



Figure 3.3 - Final Pipeline Architecture

## 3.2. Data Understanding and Treatment

The first step was to select two relevant themes in high-stakes fields where ML models play a crucial role in impactful decision-making. As mentioned in the introduction, XAI is essential for generating explanations that build trust by clarifying unexpected outcomes and addressing concerns about fairness (Slack et al., 2019). Two key domains where XAI is particularly relevant are finance and healthcare. Accordingly, two tabular datasets were chosen: Medical Information Mart for Intensive Care in Emergency Department (MIMIC-IV-ED) for healthcare (Johnson et al., 2023) and Bank Account Fraud (BAF) for finance (Jesus et al., 2022).

### 3.2.1. Medical Information Mart for Intensive Care Emergency Department Data

The Medical Information Mart for Intensive Care Emergency Department (MIMIC-IV-ED) dataset, which contains over 400,000 ED visit episodes from 2011 to 2019, is used to propose a public benchmark suite for emergency department (ED) prediction. It introduces three key ED-based outcomes for clinical prediction tasks: hospitalization, critical outcomes, and 72-hour ED reattendance.

Data processing follows the benchmarking approach of Xie et al. (2022). For master data generation, patients are identified using subjects_id, with each ED visit linked through stay_id. The data is merged into a standardized master dataset containing 448,972 ED visits from 216,877 patients. Variables were selected based on literature review and expert consultation, covering patient history, triage data, ED vitals, and outcomes, as outlined by Xie et al. (2022).

During data processing and benchmark dataset generation, patients under 18 years old and those without a primary triage classification were excluded. Outliers, such as Oxygen saturation ($SpO_2$) values greater than 100%, were identified based on expert

recommendations. The dataset was then split into 80% training data and a 20% test set (88,287 ED episodes from 65,169 patients) to ensure standardization for future benchmarking. Missing values were imputed using the median values from the training set.

International Classification of Diseases (ICD) codes are standardized codes used globally to classify diagnoses, symptoms, and procedures in healthcare. In the MMIC-IV dataset, each hospital admission is linked to a set of ICD diagnosis codes (from *diagnoses_icd.csv*), which reflect the patient's health conditions and comorbidities. To extract meaningful clinical information, the codes were converted into comorbidity features using the Charlson Comorbidity Index (CCI) and Elixhauser Comorbidity Index (ECI). The ICD codes were considered within a defined time window (e.g., five years) prior to each ED visit. Additionally, we implemented a neural network-based embedding approach, inspired by Med2Vec, to learn dense vector representations of ICD code sequences and enhance the modeling of patient comorbidities.

This benchmark provides a standardized framework for processing raw Electronic Health Record (EHR) data, ensuring reproducibility in ED prediction research.

### 3.2.2. Bank Account Fraud Data

The BAF suite consists of six datasets generated from a real-world online bank account fraud detection dataset; however, we will be using only the base dataset. This application is critical for Fair ML, as model predictions determine whether individuals are granted or denied financial services, potentially reinforcing social inequalities.

The dataset contains one million application instances with 30 features, including:

- Applicant-provided details (e.g., employment status)
- Derived attributes (e.g., phone number validity)
- Aggregated data (e.g., application frequency per ZIP code)

Spanning eight months, the dataset includes protected attributes such as age, income, and employment status. Each instance represents an individual online application, with fraud labels stored in the "is_fraud" column. Fraud prevalence ranges from 0.85% to 1.5% over different months, with higher rates observed in later months. The monthly application distribution varies between 9.5% and 15%, serving as a reference for sampling legitimate and fraudulent instances across dataset variants.

The dataset was already cleaned, as it contained no missing values or removable outliers. Since most variables were numerical, categorical features were transformed using One-Hot Encoding. In terms of correlation, only one variable, velocity_4w, was removed due to high correlation. To address the class imbalance, as only 10% of the data represented fraudulent cases, the scale_pos_weight option was used while training the models. Feature selection was conducted based on correlation values, Recursive Feature Elimination (RFE) using Random

Forest, and feature importance ranking from LightGBM. These steps ensured a balanced and optimized dataset for fraud detection modeling.

## 3.3. Machine Learning Framework

For the first dataset, the chosen model was based on the highest recall results among those used by Xie et al. (2022). Recall, which measures the proportion of actual positive cases correctly identified, was prioritized since missing a positive case can be costly. As a result, Gradient Boosting was selected, as it achieved the highest recall across all target classes. The model was then optimized using Optuna, a hyperparameter tuning framework that combines efficient searching with a pruning algorithm, significantly improving the cost-effectiveness of optimization. Additionally, Optuna's scalable and versatile design enhances its applicability (Akiba et al., 2019).

For the second dataset, the chosen model was XGBoost. As previously mentioned, recall was also selected as the evaluation metric, given the highly imbalanced nature of the data. Missing a fraudulent case (false negative) is more costly than incorrectly flagging a legitimate transaction as fraud (false positive). The optimization method chosen for this model was also Optuna.

## 3.4. Post Hoc Explainer Framework

In this step, the input to the framework was the output of the ML models used for predicting critical outcomes in the ED triage and for detecting fraud in bank account applications. The predictions from both datasets were explained using SHAP, specifically the Tree Explainer SHAP, since both models were tree-based. To complement the SHAP-based analysis with a deeper explanatory perspective, counterfactual explanations were additionally applied in the fraud detection task. This decision was guided by feedback from domain experts, who emphasized the value of counterfactual reasoning for interpretability in this context.

### 3.4.1. SHAP

To establish a foundation for the explanatory process, SHAP values were computed for each class, and the results were separated to ensure that each SHAP value was associated with the correct class. This separation was crucial for preparing the context for the subsequent explanation and avoiding potential errors. When SHAP values are applied to a model's output, they provide a global explanation for the entire dataset, and the direction of the SHAP value (positive or negative) may vary depending on the class of interest. To ensure consistency, the SHAP values were processed in such a way that they reflected the correct context for each class.

To evaluate how different input formats affect the explanations produced by the language model, SHAP values were presented alongside feature information in two distinct ways. The first format focused on technical detail and followed the structure:

- $(Feature\ Name, Feature\ Value, Prediction, SHAP\ Value)$

This version was intentionally minimal, providing only the raw numerical context without additional interpretation. The goal was to test whether the LLM would generate a more technical explanation when given purely quantitative inputs. The second format added a business-oriented layer of context and followed the structure:

- $(Feature\ Name, Feature\ Value, Feature\ Description, Prediction, SHAP\ Value)$

By including descriptive and domain-specific information, this format was designed to encourage the LLM to produce explanations that were more business-focused, reducing technical jargon and emphasizing interpretability for non-technical stakeholders. This additional layer of context is particularly valuable when datasets contain feature names that are not self-explanatory, a situation common in certain domains or organizations, ensuring the model has sufficient information to generate meaningful explanations. Presenting SHAP values in these two formats allowed for a structured comparison of how feature contributions to model predictions could be explained differently.

Finally, to avoid excessively large inputs and to ensure that only meaningful variables were included, a feature selection criterion was applied. Specifically, a z-score analysis was performed on the SHAP values for each instance. Variables were then selected based on their statistical significance, only those with z-scores indicating significant contributions to the model's prediction were retained. This approach helped streamline the input while preserving interpretability and relevance.

### 3.4.2. Counterfactuals Explanations

Counterfactuals were generated for the fraud detection dataset, with four counterfactual instances computed for each original observation. The desired class for each counterfactual was set as the opposite of the original prediction, allowing the analysis to identify the changes required for the model to predict the alternate outcome. This approach is intended to complement SHAP by focusing on actionable changes at the instance level.

All counterfactuals were stored in a dataframe, indexed to their corresponding original instances, to ensure traceability during subsequent comparisons and input preparation. The input format was structured as tuples of the form:

- $(feature, original\_value, counterfactual\_value, delta\_abs, delta\_pct)$, with one line per counterfactual prefixed by 'CF#<id>' and separated by newlines

Only features with modified values were included, enabling direct comparison of the changes present in each counterfactual.

To further connect counterfactual reasoning with model behavior, the results are stratified according to the four standard classification outcomes: true positives, true negatives, false positives, and false negatives. This division is designed to make it possible to examine counterfactuals in relation to performance outcomes that are particularly relevant for

stakeholders, such as identifying factors that may reduce false alarms or highlighting changes that could make missed fraud cases detectable. In this way, counterfactual explanations are linked to both model robustness and practical considerations in fraud detection.

## 3.5. PROMPT DESIGN

This section explores the process of transforming post-hoc explainers' values into readable narratives. The approach seeks to present model explanations in a more natural and accessible format, as opposed to traditional graphical representations. The objective is to create a more user-friendly and intuitive way for stakeholders to interpret ML model predictions. This is especially beneficial for non-technical users, who may find text-based explanations easier to understand.

The explanation framework consists of four key components:

1. Base Prompt: Defines the structure of the input that the model will process.
2. Input Component Structure: Outlines the format in which the post-hoc values will be provided.
3. Explanation Components: Consists in a prompt framework that follows the structure below:
    - Context: The prediction the model is making.
    - Explanation: The rationale behind the model's prediction.
    - Explanation Format: How the explanation should be conveyed (definition of the inputs given in explanation).
    - Narrative: A human-readable version of the mathematical explanation.
4. Output Instructions: Directs how the final narrative should be formatted.

Once the input data is structured into the required components, the prompt is sent to a language model, such as GPT-4o, which generates a narrative explanation of the model's predictions. In this study, the goal is to evaluate which model performs better and, consequently, provides more effective explanations. To achieve this, a comparative analysis will be conducted across several state-of-the-art API-based LLMs, including GPT models (GPT-4o, GPT-4 Turbo, GPT-3.5 Turbo) and Claude 3 Models (Haiku, Sonnet, and Opus) will be compared using the quality metrics presented in the evaluation sections. In addition to these, two locally deployable models, LLaMA 3 and DeepSeek, will be also included to assess their viability in privacy-sensitive environments. These local models are not dependent on external APIs, enabling secure on-premise deployment and better addressing privacy concerns by ensuring that sensitive information remains within organizational boundaries. This comparison allows for a comprehensive assessment of explanation quality across models with different architectural and deployment characteristics.

In the context of SHAP, the most influential features are selected and presented in a sorted order. These feature explanations are incorporated into the "Explanation" section of the framework. Then, in the ‘Explanation format’ ensures that the system can interpret the provided explanations by adding the definition of each input. Additionally, a brief description of how to interpret the explanation is provided in the "Narrative" component.

Counterfactual explanations follow the same pipeline, with the exception that features are not ordered by influence. Instead, only the features that change between the original instance and the counterfactual are presented, highlighting the minimal modifications required to alter the model’s prediction. This uniform formatting across SHAP and counterfactuals allows the framework to maintain flexibility while accommodating different types of ML explanations that can be expressed as text. The format allows for greater flexibility, allowing the framework to handle various types of ML model explanations that can be expressed as text.

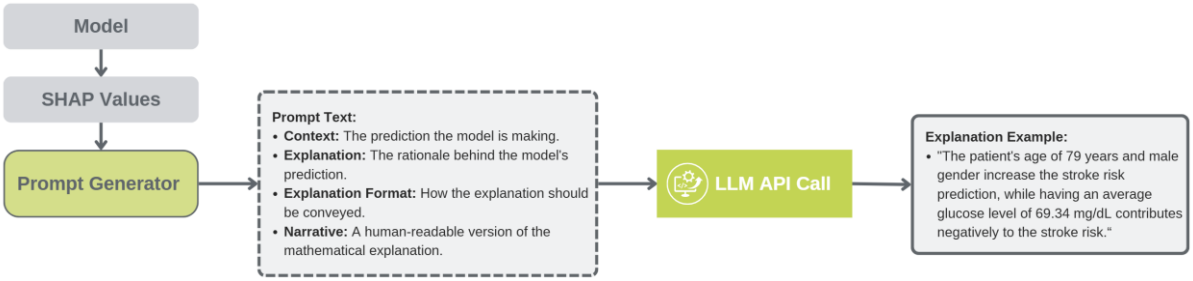



Figure 3.4 - Phases of the Prompting Design (SHAP example)

## 3.6. TEXTUAL EXPLANATION

To illustrate the functioning of the framework, consider an example involving a model trained to predict stroke risk. The following steps outline how a user might interact with the system:

1) **Selection of SHAP Explanations**: A set of N SHAP explanations is selected for the observations under examination. These explanations are structured according to the format described above and are included in the "Explanation" component of the framework.

2) **Manual Narrative Generation**: Initially, the user may manually craft a short narrative to describe the explanation in a desired style. For example, if the SHAP explanation includes features such as (Age, 79, 1, 0.3256), (Gender, Male, 1, 0.1291), and (Average Glucose Level, 69.34, 1, -0.0357), the user might write a narrative like: "The patient's age of 79 years and male gender increase the stroke risk prediction, while having an average glucose level of 69.34 mg/dL contributes negatively to the stroke risk."

3) **Automatic Narrative Generation**: After fine-tuning the model, the user can input further SHAP explanations, and the system will generate corresponding narrative versions. These narratives can then be shared with non-technical stakeholders, such

as doctors, nurses, or patients, to explain why a particular individual has a higher stroke risk.

Counterfactual explanations follow the same three-step process as SHAP. The only distinction is in the input and output format, which differs due to the characteristics of the counterfactual method.

For this study, four datasets will be used to generate explanations and evaluate the effectiveness of the narrative generation process. These datasets will help in providing the necessary examples for explaining and interpreting the model's predictions, helping to refine the system's ability to generate accurate and easily understandable explanations.

## 3.7. EVALUATION

This section examines the evaluation of final explanations generated by LLMs, with a focus on their effectiveness in communicating understanding to stakeholders. The goal is to assess whether these explanations genuinely contribute to stakeholder comprehension and facilitate the overarching objective of generating textual, explainable predictions from the models. To achieve this, a comprehensive set of ten evaluation metrics has been identified, capturing both the content and linguistic quality of the explanations, to evaluate a high-quality explanation. These metrics are designed to ensure that explanations are not only logically structured but also accessible, relevant, and readable for a wide range of users.

These evaluation metrics fall into two complementary categories: LLM-based evaluations and heuristic/statistical evaluations. LLM-based metrics rely on prompting language models to assess qualities such as factual accuracy, completeness, and interpretability through natural language rubrics, effectively simulating a human judgment perspective. In contrast, heuristic and statistical metrics apply rule-based or embedding-based techniques to evaluate features such as fluency, conciseness, and lexical diversity, dimensions that are more objectively quantifiable and less sensitive to the variability of generative models.

Employing both types of evaluation serves multiple purposes. It allows for a comparative analysis to explore whether one approach consistently produces more insightful or dependable assessments, thus revealing the unique strengths and limitations of each method. It also enables the capture of complementary aspects of explanation quality: while LLMs can better reflect semantic depth and logical coherence, heuristic methods are more precise in measuring structural and stylistic characteristics. This combination expands the scope of evaluation, ensuring that both interpretability and linguistic quality are thoroughly assessed. Moreover, using two distinct methodologies improves the overall robustness of the evaluation process by reducing the risk of bias or overfitting to a single framework. Given that LLM-based evaluation techniques are still evolving, integrating them with well-established heuristic approaches also provides a degree of methodological safety, helping to mitigate inconsistencies and validate findings through more traditional metrics. The selected evaluation metrics are as follows:

Table 3.1 - Metrics for evaluating LLMs

| Type | Metric | Description |
|---|---|---|
| **LLM-based Metrics** | Completeness | Measures whether the explanation fully covers all necessary aspects of the answer. |
| **LLM-based Metrics** | Accuracy | Verifies whether the information in the narrative is factually correct and aligns with the underlying explanation, particularly regarding feature values and contribution directions. |
| **LLM-based Metrics** | Fluency | Assesses the naturalness and human-likeness of the narrative's writing style. |
| **LLM-based Metrics** | Understandability | Assesses whether the explanation is accessible to lay users, such as those without a technical background. |
| **Heuristic Metrics** | Conciseness | Evaluates whether the explanation is efficiently worded, avoiding unnecessary verbosity. |
| **Heuristic Metrics** | Specificity | Ensures that explanations are detailed and tailored to the input rather than being vague or generic. |
| **Heuristic Metrics** | Readability | Estimates how easy the narrative is to read, based on conventional readability indices. |
| **Heuristic Metrics** | Flesch-Kincaid Grade Level | Calculates the approximate U.S. school grade level required to understand the explanation. |
| **Statistical / Embedding-based Metrics** | Interpretability | Evaluates how easy it is for a user to understand the explanation. |
| **Statistical / Embedding-based Metrics** | Coverage (Text Similarity-Based): | A secondary measure of completeness using reference alignment via ROUGE scoring. |

The evaluation metrics are selected based on a combination of linguistic heuristics, semantic similarity measures, and model-assisted evaluation protocols designed to approximate human reasoning. To ensure both scalability and validity, the study employs a two-tiered evaluation strategy. First, all explanations will be assessed automatically through well-established quantitative techniques, providing a consistent, reproducible, and large-scale evaluation pipeline. Second, to complement and validate these automated assessments, a sample of six stakeholders will be asked to evaluate a subset of explanations using the same criteria. Their ratings are intended to examine the extent to which the metrics align with human judgment, as well as to identify potential shortcomings or interpretability challenges that may not be fully captured by quantitative methods. This combined approach is designed to leverage the efficiency of automated evaluation while grounding the results in stakeholder perspectives, thereby strengthening both the robustness and the practical relevance of the findings.

For the first metric, Completeness, two complementary methods are applied. The GPT-based Completeness score is computed by prompting a language model to assess whether all features, values, and their contribution directions are mentioned in the explanation. Additionally, a ROUGE-L-based Coverage score measures the longest common subsequence (LCS) between the candidate and reference texts, calculating precision, recall, and F1 score to quantify content overlap (Ganesan, 2018). It is a traditional text similarity metric that

compares generated explanations with reference explanations, ensuring that key points are captured, similar to human-crafted explanations.

The accuracy metric evaluates whether the information in the narrative correctly reflects the input explanation. It checks that feature values and contribution directions (positive or negative) are accurate, based on the explanation (e.g., SHAP values). Narratives can omit features and still be considered accurate if no incorrect information is present. Scoring is done using an LLM-based grader, prompted to compare each feature in the narrative to the explanation. A rubric is applied, where a score of 4 indicates no errors and 0 indicates at least one error.

Fluency evaluates how natural and well-formed the language of the explanation is. This is judged either on its own or by comparison with exemplar narratives provided as stylistic references. A fluent narrative reads smoothly and resembles human-written text.

Understandability is judged by prompting a model to assess how accessible the narrative is to non-expert readers. Explanations that are free of jargon and use plain language are rated higher.

Conciseness is calculated heuristically by comparing the narrative's length to the number of features explained. The metric uses a configured threshold representing the ideal maximum number of words per feature. In Explingo, 20 words per feature is considered optimally concise. Longer narratives are penalized to discourage verbosity.

For Specificity, the N-gram Diversity metric will be used. Specifically, the Distinct-N score (with bigrams, i.e., Distinct-2) will evaluate the lexical diversity of the response by calculating the proportion of unique words or word pairs (n-grams) in the text. A higher Distinct-N score indicates greater diversity and less redundancy, ensuring that the explanation is detailed and not repetitive.

Readability is measured using the Flesch Reading Ease score from the textstat library. This formula uses word and sentence length to estimate how easy a text is to read, with scores scaled to a 0-4 range for evaluation consistency. Flesch Reading Ease produces values between 0 and 100, where higher scores indicate simpler texts that are accessible to younger readers. To interpret these scores in terms of education level, a conversion table is typically used.

Flesch-Kincaid Grade Level, by contrast, directly estimates the U.S. school grade level required to comprehend a text. For this study, the grade level is inverted and scaled so that simpler, more accessible texts receive higher scores, reflecting greater clarity. A higher score denotes ease of comprehension, approximating a fifth grade reading level, whereas a lower score reflects greater complexity, indicating that the text may only be readily understood by readers with graduate-level proficiency.

While both metrics are based on the same textual features, sentence length and word complexity, they apply different weightings and scales. Flesch Reading Ease is well suited for general interpretability, offering an absolute measure of difficulty, whereas Flesch-Kincaid Grade Level provides a direct educational benchmark, making it useful for assessing accessibility in terms of expected reader background.

Lastly, for Interpretability, the Perplexity metric will be used. This statistical indicator, computed via GPT-2, reflects how predictable a piece of text is for a language model. It measures the model's "surprise" when encountering new data, where a lower perplexity value indicates higher predictive accuracy. In this context, perplexity will be used to assess how predictable (and thus readable) the explanation is, ensuring that the explanation is clear and easy for users to comprehend.

By integrating ten metrics within a two-tiered evaluation strategy, the framework provides a robust and nuanced approach to assessing the quality of model-generated explanations. Automated methods, combining heuristic and LLM-based approaches, ensure both consistency and the ability to capture semantic and linguistic nuances. To complement these quantitative assessments, stakeholder feedback is incorporated, aligning evaluation outcomes with human judgment and uncovering interpretability challenges that may otherwise be overlooked. This combination enables a comprehensive analysis of explanation quality while highlighting the strengths and limitations of different evaluation methods. Taken together, these elements provide a balanced and comprehensive foundation for evaluation, reinforcing both the methodological rigor and the practical relevance of the findings.

# 4. RESULTS AND DISCUSSION

This section is structured into two parts to provide complementary perspectives on the problem's resolution. The first part takes a more analytical perspective, offering a global interpretation through the aggregation and analysis of quantitative evaluation metrics. The second part adopts a qualitative, human-centered approach, in which a sample of six stakeholders evaluates specific examples of model-generated explanations. Local examples are also presented in this chapter to illustrate the outputs of some models, helping readers understand how explanations appear in practice and how stakeholders assessed their clarity, usefulness, and plausibility. The goal of this dual analysis is to provide a comprehensive understanding of model performance from both interpretability and statistical standpoints.

## 4.1. QUANTITATIVE ANALYSIS

The quantitative and global interpretation approach focuses on the analysis of aggregated numerical results to derive broader insights from the conducted experiments. This section is structured around two post hoc explanation methods, SHAP and Counterfactuals, and is organized into three stages. First, it evaluates the performance metrics employed, assessing whether any particular metric consistently returns higher scores across models and what those scores imply in terms of explanation quality. This helps determine whether meaningful improvements have been achieved or if further enhancements are still possible. Second, it involves a comparative analysis of all the tested models to identify which, if any, demonstrate superior performance and to uncover any notable performance patterns or trade-offs. Finally, the analysis explores performance variations across different datasets to determine whether certain datasets produce better results, whether some models perform particularly well on specific datasets, and whether any evaluation metric stands out as especially informative across all or selected datasets. In addition, the comparison considers two distinct explanation formats, a technical version and a business-oriented version, to assess how the presentation of information influences evaluation outcomes. This comprehensive assessment intends to uncover systematic trends and deepen the understanding of the models' behavior in diverse scenarios.

### 4.1.1. SHAP

#### 4.1.1.1. Evaluation of Metrics

The evaluation metrics employed in this project exhibit considerable variation in both their methodological complexity and intended usage. Some metrics rely on straightforward statistical or heuristic approaches, while others use more advanced language models for evaluation. This dual approach was adopted to assess whether simpler or more sophisticated evaluation methods return more accurate or reliable assessments.

As detailed in the methodology section, metrics such as accuracy, fluency, completeness, and understandability were evaluated using GPT-4o, a large language model capable of providing nuanced judgments based on contextual and semantic understanding. These evaluations represent the more complex end of the methodological spectrum.

In contrast, metrics such as coverage, interpretability, specificity, readability, and the Flesch-Kincaid Grade Level were computed using heuristic or statistical techniques. These include established natural language processing tools and formulas such as ROUGE-L (for lexical overlap), perplexity (for syntactic plausibility), and Flesch-Kincaid readability scores (for linguistic complexity).

The results obtained are summed in the next table, which presents the mean and standard deviation of each evaluation metric used to assess the quality of the generated explanations.

Table 4.1 - Results of total score of SHAP generated explanations

| Metric | Mean | Standard Deviation |
|---|---|---|
| **Accuracy** | 3.02 | 1.01 |
| **Completeness** | 2.74 | 1.19 |
| **Fluency** | 2.61 | 0.76 |
| **Conciseness** | 3.86 | 0.40 |
| **Coverage** | 0.97 | 0.39 |
| **Interpretability** | 2.65 | 0.63 |
| **Specificity** | 3.67 | 0.27 |
| **Understandability** | 1.60 | 0.75 |
| **Readability** | 0.76 | 0.60 |
| **Flesch Kincaid Grade Level** | 0.27 | 0.34 |
| **Total Score** | 22.15 | 3.28 |

The highest-performing metric was Conciseness, with a mean score of 3.86 and a very low standard deviation of 0.40. This indicates that the generated explanations were consistently succinct and avoided unnecessary verbosity. Similarly, Specificity achieved a strong average score of 3.67 with a low standard deviation of 0.27, suggesting that the explanations were not only targeted and detailed but also stable across different instances. The metric Accuracy also performed relatively well, with a mean of 3.02 and a moderate standard deviation of 1.01, implying that most outputs were factually correct, although occasional inconsistencies appeared in edge cases or more ambiguous scenarios.

Metrics such as Fluency, Completeness, and Interpretability fall into the mid-range. Fluency, with a mean of 2.61, indicates that the explanations were generally grammatically correct and coherent, though not outstanding. Completeness ($\mu = 2.74$, $\sigma = 1.19$) displayed notable variability, reflecting inconsistency in how thoroughly information was covered across different explanations. Interpretability ($\mu = 2.65$, $\sigma = 0.63$) suggests that while some

explanations were understandable and logically structured, others fell short in providing clear reasoning or actionable insights.

At the lower end, several metrics highlighted areas for improvement. Understandability received a mean score of 1.60 ($\sigma = 0.75$), implying that many explanations were not easily comprehensible to end users . This may be due to complex phrasing, lack of clarity, or limited alignment with users' expectations. Readability ($\mu = 0.76$, $\sigma = 0.60$) and the Flesch-Kincaid Grade Level ($\mu = 0.27$, $\sigma = 0.34$) were also low, reinforcing the conclusion that the generated text may not be linguistically accessible to a broad audience. Additionally, the metric Coverage ($\mu = 0.97$, $\sigma = 0.39$) remained relatively weak, indicating limited lexical overlap or alignment with reference texts. This outcome may be partly explained by the explanation formats used as input, the technical format (Feature Name, Feature Value, Prediction, SHAP Value) and the business-oriented format (Feature Name, Feature Value, Feature Description, Prediction, SHAP Value). Both formats are highly structured and character-dense, which creates a substantial difference compared to natural textual narratives. As a result, metrics that rely on textual overlap or linguistic similarity, such as coverage, may produce lower scores, not necessarily because of poor explanation quality, but due to the inherent mismatch between structured input formats and narrative-style outputs.

The Total Score, computed as the sum of all individual metric values, granted a mean of 22.15 with a standard deviation of 3.28. This aggregate score provides a high-level summary of the system's performance and reflects a balance between metrics evaluated by large language models and those assessed through heuristic or statistical approaches.

These results highlight the strengths and limitations of the current system. The disparity between model-based and heuristic evaluations also emphasizes the importance of employing a mixed-method evaluation strategy. While GPT-4o-based metrics capture high-level semantic and contextual quality, heuristic metrics reveal structural and linguistic weaknesses that might otherwise be overlooked. An exception to this trend is found in Specificity and Interpretability, which, despite not being GPT-based, still achieved relatively strong results.

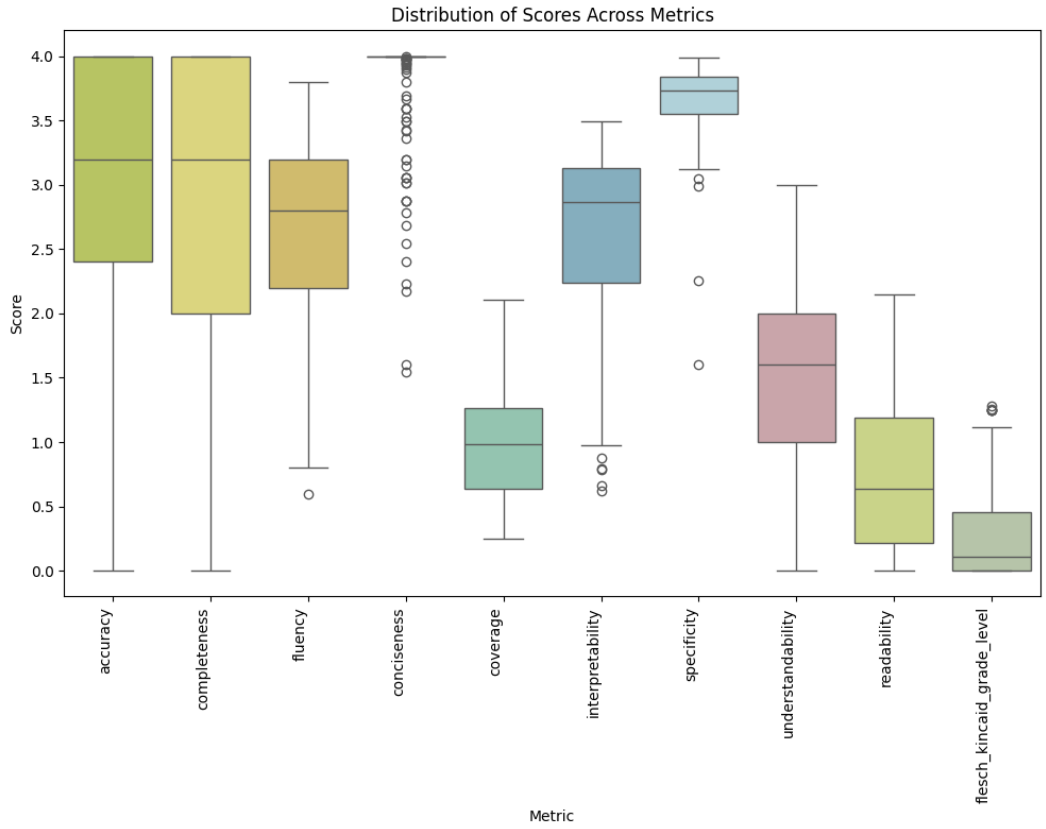



Figure 4.1 - Boxplot of score distributions across all evaluation metrics (SHAP explanations)

Finally, the boxplot provides a visual summary of the score distributions across all evaluation metrics. It reinforces the numerical analysis by highlighting the variability and consistency in the system's performance. Metrics such as conciseness and specificity display very tight interquartile ranges with high median scores, indicating consistently strong performance across explanations. Accuracy and completeness, in contrast, show wider distributions, reflecting variability in factual correctness and coverage depending on the instance. Fluency and interpretability fall in the mid-range, with moderate dispersion, suggesting generally acceptable but uneven performance. On the other hand, understandability, readability, and the Flesch-Kincaid Grade Level exhibit low medians and limited spread, reaffirming that many generated explanations are not easily accessible to a general audience and may require higher reading proficiency.

### 4.1.1.2. Comparative Analysis of the models

In this analysis, eight distinct language models were evaluated to determine their effectiveness in generating textual explanations for model predictions. The primary objective was to compare their performance across a comprehensive set of evaluation metrics and assess whether certain models consistently produce higher-quality explanations. By analyzing both API-based and locally hosted models, this comparison provides valuable insights into the strengths and limitations of each model in the context of interpretability and explanation generation.

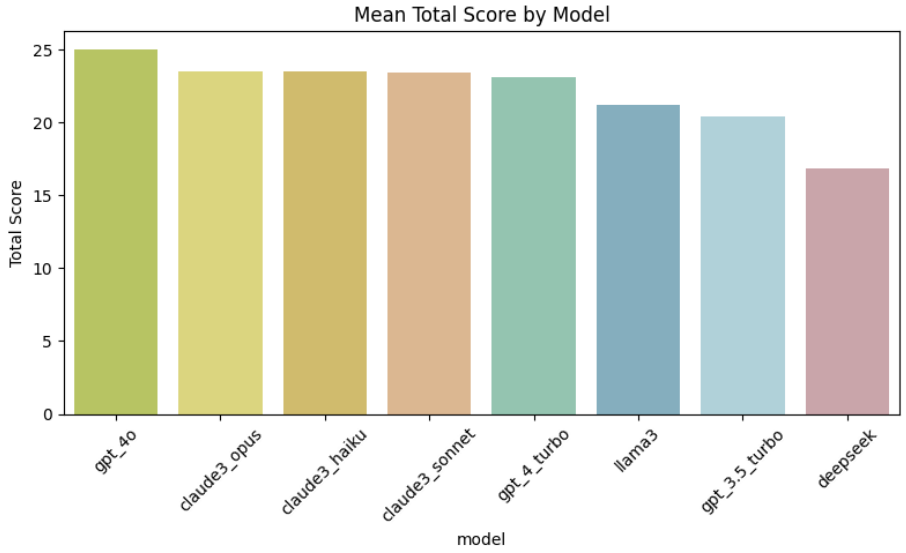



Figure 4.2 - Mean total score across LLMs of SHAP explanations

By looking at the graph, it is possible to compare the mean total score obtained by each language model evaluated in this study. Among all models, GPT-4o achieved the highest average score, followed by Claude 3 models, indicating superior overall performance in generating high-quality, accurate, and comprehensible explanations.

Models from the Claude 3 series (Haiku, Opus, and Sonnet) performed consistently well, with minimal differences in mean scores, suggesting robustness across versions. In contrast, GPT-3.5 Turbo, while still a capable model, scored noticeably lower than its GPT-4 counterparts, reflecting the performance gap between model generations.

The two locally hosted models, LLaMA 3 and DeepSeek, exhibited lower average scores compared to API-based models. DeepSeek received the lowest mean total score, indicating challenges in maintaining the same level of explanation quality. This performance discrepancy may be attributed to differences in model architecture, tuning, or access to broader training data in hosted models versus local deployments.

Overall, these results reinforce the advantage of more advanced, cloud-hosted language models for tasks requiring nuanced and high-quality textual outputs, particularly in multi-faceted evaluations. However, it is important to note that in high-stakes domains where data privacy and regulatory compliance are critical, the use of cloud-based models may raise significant concerns. In such cases, locally hosted models (e.g., Llama 3), while generally scoring lower, still demonstrate competitive performance and may offer a viable alternative when data confidentiality is a priority.

#### 4.1.1.3. Comparative Analysis in different datasets

To evaluate model performance across different real-world applications, each language model was tested on four distinct datasets spanning two high-stakes domains: healthcare (Triage

Datasets) and finance (Fraud Detection Dataset). The goal of this analysis is to assess how well each model generates explanatory outputs when faced with domain-specific challenges, and whether certain models generalize better across use cases. The table below summarizes the mean total scores achieved by each model on these datasets, reported separately for the technical format and the business-oriented format of the explanations.

Table 4.2 - Mean total scores by each model and dataset (SHAP explanations)

| Model | 72h Triage | 72h Triage Business | Critical Triage | Critical Triage Business | Hospital Triage | Hospital Triage Business | Fraud Detection | Fraud Detection Business |
|---|---|---|---|---|---|---|---|---|
| **Claude 3 Haiku** | 24.42 | 26.03 | 25.10 | 24.34 | 23.11 | 23.26 | 24.19 | 20.73 |
| **Claude 3 Opus** | 24.55 | 26.30 | 25.57 | 24.47 | 23.17 | 23.26 | 24.13 | 20.33 |
| **Claude 3 Sonnet** | 24.08 | 26.23 | 25.30 | 24.20 | 23.11 | 23.20 | 24.33 | 20.53 |
| **DeepSeek** | 18.26 | 16.33 | 16.68 | 19.86 | 17.02 | 17.76 | 19.10 | 13.16 |
| **GPT 3.5. Turbo** | 19.45 | 21.76 | 22.22 | 20.04 | 20.54 | 19.69 | 22.09 | 20.02 |
| **GPT 4 Turbo** | 25.18 | 24.96 | 25.25 | 23.04 | 23.39 | 19.70 | 22.86 | 20.40 |
| **GPT 4o** | 26.70 | 25.85 | 26.89 | 25.01 | 24.41 | 24.56 | 24.55 | 22.55 |
| **LlaMA 3** | 22.89 | 23.79 | 22.46 | 21.88 | 20.82 | 18.30 | 22.48 | 19.83 |

As shown in the results, GPT-4o consistently achieved the highest scores across all datasets and formats (e.g., 72h Triage: $\mu = 26.70$ technical, $\mu = 25.85$ business; Critical Triage: $\mu = 26.89$ technical, $\mu = 25.01$ business). This indicates superior generalization and robustness across domains. GPT-4 Turbo and the Claude 3 series also showed competitive performance, maintaining high scores in both medical and financial contexts.

In the healthcare domain, the best-performing models across the three datasets were again GPT-4o, followed by GPT-4 Turbo and Claude 3 models. These models consistently produced high-quality, interpretable explanations suited for clinical settings. In the finance domain, performance on Fraud Detection (technical format) was comparable to the healthcare datasets (e.g., GPT-4o: $\mu = 24.55$), showing that models can generate strong explanations even in complex financial contexts. However, the Fraud Detection Business dataset resulted in the lowest scores overall (e.g., GPT-4o: $\mu = 22.55$; Claude 3 Haiku: $\mu = 20.73$; DeepSeek: $\mu = 13.16$). This suggests that producing business-oriented explanations for fraud detection is particularly challenging, likely due to the ambiguity, variability, and context-dependence of financial decision-making, whereas healthcare triage tasks provide a more structured and constrained environment for explanation generation.

LLaMA 3 performed moderately across domains, with relatively stable results in 72h Triage ($\mu = 22.89$ technical, $\mu = 23.79$ business) and Critical Triage ($\mu = 22.46$ technical, $\mu = 21.88$ business), although still behind API-based models. DeepSeek, on the other hand, lagged

significantly across all datasets, particularly in Fraud Detection (μ = 19.10 technical, μ = 13.16 business), highlighting the difficulty of maintaining performance outside hosted LLMs. Nevertheless, LLaMA 3 shows promise as a privacy-conscious alternative with acceptable results in high-stakes domains.

A comparison of the two input formats reveals important differences. In several cases, business-oriented explanations improved overall scores (e.g., Claude 3 Haiku in 72h Triage: 24.42 to 26.03), suggesting that the added descriptive context makes outputs more interpretable and better aligned with evaluation metrics. However, in other cases, the technical format achieved higher performance (e.g., GPT-4 Turbo in Hospital Triage: 23.39 technical vs. 19.70 business), indicating that concise, structured inputs can sometimes produce clearer outputs than verbose business-oriented prompts. These results suggest that the effectiveness of explanation format depends both on the dataset and the model, underlining the importance of tailoring input representations to the domain and user needs.

Another important insight is that the datasets with the highest performance scores were those with fewer input features. This indicates that a reduced number of variables can make it easier for the model to generate accurate and coherent explanations, as the input becomes less complex and less cognitively demanding to process. Evidence for this comes from the Fraud Detection use case presented in the annexes, where the same dataset (Fraud Detection vs Fraud Detection VC), when provided with a smaller set of variables, resulted in higher scores. This suggests that reducing input dimensionality helps the model focus on the most relevant factors, thereby improving both interpretability and overall explanation quality.

### 4.1.2. Counterfactuals Explanation

#### 4.1.2.1. Evaluation of Metrics

As outlined in the SHAP evaluation, the same set of metrics was applied to assess the quality of the counterfactual-based explanations. The results are summarized in the following table, which presents the mean and standard deviation for each metric.

Table 4.3 - Results of total score of Counterfactuals generated explanations

| Metric | Mean | Standard Deviation |
|---|---|---|
| **Accuracy** | 1.89 | 1.07 |
| **Completeness** | 0.53 | 0.63 |
| **Fluency** | 1.31 | 0.87 |
| **Conciseness** | 3.83 | 0.59 |
| **Coverage** | 0.28 | 0.28 |
| **Interpretability** | 1.88 | 0.63 |
| **Specificity** | 3.71 | 0.59 |
| **Understandability** | 1.46 | 0.80 |
| **Readability** | 0.81 | 0.68 |
| **Flesch Kincaid Grade Level** | 0.22 | 0.29 |

| | | |
|---|---|---|
| **Total Score** | 15.91 | 3.38 |

Conciseness (μ = 3.83, σ = 0.59) and Specificity (μ = 3.71, σ = 0.59) once again emerged as the strongest metrics, indicating that counterfactual explanations were consistently succinct and provided targeted, detailed insights. However, the majority of other metrics showed weaker performance compared to SHAP. Accuracy (μ = 1.89) and Interpretability (μ = 1.88) were relatively low, suggesting that while some counterfactual explanations captured relevant decision logic, many lacked factual precision or failed to clearly articulate the reasoning behind changes in input features.

Completeness (μ = 0.53, σ = 0.63) was particularly poor, reflecting that counterfactual explanations often omitted important contextual details. Similarly, Coverage (μ = 0.28, σ = 0.28) and Fluency (μ = 1.31, σ = 0.87) indicate that explanations showed limited overlap with reference outputs and were not consistently coherent. Understandability (μ = 1.46, σ = 0.80) and Readability (μ = 0.81, σ = 0.68) reinforce this observation, highlighting that many outputs were difficult for users to follow, especially for non-technical audiences.

The Total Score of 15.91 (σ = 3.38) reflects an overall weaker performance of counterfactual explanations compared to SHAP (22.15). While counterfactuals provided concise and specific insights, their lack of completeness, readability, and factual accuracy limited their overall usefulness.

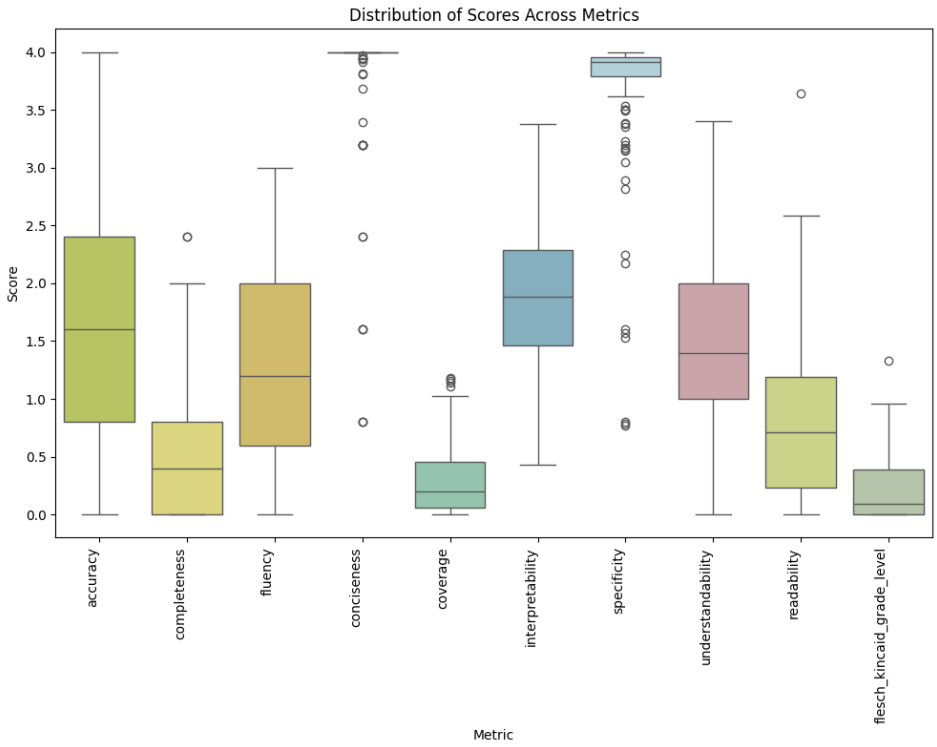



Figure 4.3 - Boxplot of score distributions across all evaluation metrics (Counterfactuals explanation)

The distribution of scores across metrics further illustrates these results. Conciseness and Specificity show very tight interquartile ranges with consistently high medians, confirming their stability as strong dimensions of counterfactual explanations. In contrast, Completeness and Coverage exhibit very low medians and narrow distributions clustered near zero, reinforcing the consistent weakness of these aspects. Accuracy and Interpretability display moderate medians but wide variability, suggesting that performance was highly case-dependent. Fluency and Understandability also show dispersed distributions, reflecting inconsistencies in grammatical correctness and ease of comprehension. Finally, Readability and Flesch-Kincaid Grade Level have low medians and compressed ranges, indicating that the explanations remained linguistically challenging across instances. Together, these patterns emphasize that while counterfactuals reliably produce concise and specific outputs, they fall short in delivering complete, accessible, and easily understandable narratives.

#### 4.1.2.2. Comparative Analysis of the models

The figure below presents the mean total scores of the eight language models when generating counterfactual explanations.

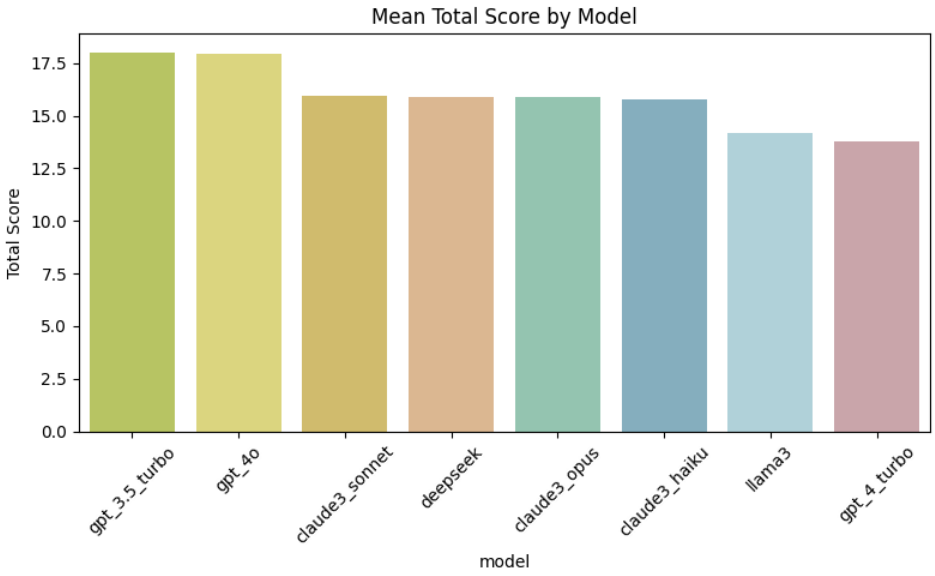



Figure 4.4 - Mean total score across LLMs of Counterfactuals Explanations

GPT-3.5 Turbo and GPT-4o achieved the highest performance, demonstrating strong ability to produce concise and targeted counterfactual outputs. Claude 3 Sonnet and DeepSeek followed in the mid-range, with DeepSeek showing competitive results relative to API-based models. Claude 3 Opus and Haiku trailed slightly behind, though their scores remained close to the middle group.

The lowest scores were obtained by LLaMA 3 and GPT-4 Turbo, with the latter performing significantly below the other GPT-4 variants. This gap highlights the variability within model

families and suggests that not all versions are equally well suited for counterfactual explanation tasks.

Overall, the results confirm the trend observed in the SHAP analysis: advanced API-based models, particularly GPT-3.5 Turbo and GPT-4o, consistently deliver higher-quality explanations, while locally hosted alternatives such as LLaMA 3 underperform. However, DeepSeek's mid-range performance indicates that locally deployed models can remain competitive, offering a viable option for sensitive domains.

#### 4.1.2.3. Comparative Analysis in different datasets

The table below summarizes the mean scores obtained by each model on the Fraud Detection dataset when explanations were generated using counterfactuals. Unlike the SHAP evaluation, which covered both healthcare and finance datasets, the counterfactual analysis was limited to the financial domain. This decision was based on feedback collected during the human evaluation of SHAP explanations, where participants highlighted the need for additional perspectives in finance-related tasks. The objective was therefore to provide a more complete assessment by extending the evaluation with counterfactuals in this specific domain.

Furthermore, the evaluation was not only conducted on the full dataset but also divided by classification outcome (false negatives, false positives, true negatives, and true positives). This breakdown links counterfactuals directly to the confusion matrix, making it possible to connect explanations with specific model weaknesses. For instance, counterfactuals for false positives indicate what minimal changes would have prevented legitimate transactions from being flagged, which is important for improving user trust and reducing false alarms. For false negatives, they show what adjustments would have enabled the model to detect missed fraud, supporting efforts to improve recall. True positives highlight how fraudulent cases could be altered to evade detection, which is relevant for robustness testing, while true negatives reveal the boundary between legitimate and fraudulent behaviour. This structure ensures that counterfactual explanations not only provide interpretability but also contribute directly to refining performance in fraud detection systems.

Table 4.4 - Mean total scores by each model and dataset (Counterfactuals explanations)

| Model | Fraud CF All | Fraud CF FN | Fraud CF FP | Fraud CF TN | Fraud CF TP |
|---|---|---|---|---|---|
| **Claude 3 Haiku** | 16.43 | 16.12 | 14.69 | 15.08 | 16.60 |
| **Claude 3 Opus** | 16.53 | 16.22 | 14.69 | 15.43 | 16.45 |
| **Claude 3 Sonnet** | 16.68 | 16.37 | 14.59 | 15.38 | 16.65 |
| **DeepSeek** | 16.28 | 15.68 | 16.15 | 15.82 | 15.64 |

| | | | | | |
|---|---|---|---|---|---|
| **GPT 3.5. Turbo** | 18.81 | 17.23 | 18.58 | 18.26 | 17.89 |
| **GPT 4 Turbo** | 13.69 | 13.40 | 14.31 | 15.16 | 12.31 |
| **GPT 4o** | 18.95 | 17.32 | 16.99 | 18.44 | 17.87 |
| **LlaMA 3** | 17.35 | 16.81 | 12.25 | 9.29 | 15.16 |

As shown in the results, GPT-4o achieved the highest overall performance (μ = 18.95), closely followed by GPT-3.5 Turbo (μ = 18.81). These "All" scores reflect the aggregate explanations across all classification outcomes (TP, FP, TN, FN), providing a general view of each model's ability to generate counterfactual explanations. Claude 3 models (Sonnet, Opus, Haiku) scored slightly lower, but remained stable across categories, while DeepSeek achieved a comparable overall score (μ = 16.28), showing competitive results despite being locally deployed.

LLaMA 3 achieved a relatively strong overall score (μ = 17.35), but the breakdown reveals sharp inconsistencies: it performed well on false negatives and true positives yet dropped significantly for false positives (μ = 12.25) and especially true negatives (μ = 9.29). GPT-4 Turbo obtained the lowest overall score (μ = 13.69), underperforming across nearly all categories and reinforcing its weaker suitability for explanation tasks.

The breakdown by classification outcome may shed some light on these differences. False positives and true negatives tended to receive lower scores across models, possibly because such counterfactuals involve subtle boundary conditions: explaining why legitimate cases were misclassified as fraud (FPs) or identifying what changes might make a legitimate transaction appear fraudulent (TNs). By contrast, explanations for true positives and false negatives often received higher scores. For TPs, counterfactuals may illustrate how fraudulent cases could be minimally altered to evade detection, while for FNs they may suggest what changes could have enabled the model to detect missed fraud. These comparatively clearer decision signals might make TP and FN explanations more accessible and informative, which could help explain why model performance appears stronger in those categories and in the overall 'All' scores. However, this interpretation would benefit from further examination.

Overall, the analysis shows that GPT-4o and GPT-3.5 Turbo consistently deliver the strongest counterfactual explanations in finance. Claude models and DeepSeek perform competitively but with lower aggregate scores, while LLaMA 3's instability and GPT-4 Turbo's overall weakness highlight the variability across models. Importantly, viewing results both in aggregate ("All") and by confusion matrix categories ensures that counterfactual explanations are not only interpretable but also diagnostic of model strengths and weaknesses in fraud detection.

## 4.2. QUALITATIVE ANALYSIS

The qualitative and local interpretation approach focuses on the individual analysis of the explanations generated by the models. This perspective allows for an assessment of whether

the outputs produced by the models are coherent and aligned with the input data provided. The analysis is divided into two parts: first, the outputs generated with SHAP, and then the outputs generated with counterfactuals. In the SHAP analysis, a human evaluation involving a group of fraud stakeholders provides insights that complement and support the quantitative metrics. In both cases, a selection of explanations was made from across the datasets, specifically those associated with the highest overall performance scores. This selection enables a qualitative evaluation to determine, from a business perspective, whether differences in textual explanations have a significant impact. Furthermore, this process helps assess whether the evaluation metrics accurately reflect explanation quality. If strong alignment is observed, it reinforces the reliability of the chosen metrics and, by extension, the trustworthiness of the evaluation framework used in this project.

### 4.2.1. SHAP

#### 4.2.1.1. Human Case Evaluation

To complement the metric-based evaluation of SHAP explanations in the fraud detection dataset, a human case study was conducted in which domain fraud experts scored explanations using the same evaluation dimensions, adapted to a business-oriented perspective. The questionnaire used in this case study is provided in the Annexes to offer a clearer view of its structure. The table above summarizes the results, showing minimum, maximum, mean, and standard deviation of total scores across 20 evaluated explanations, along with their corresponding model outputs and the delta compared to metric-based evaluations.

Table 4.5 - Human Evaluation Results

| Explanation ID | Min Total Score | Max Total Score | Average Total Score | Standard Deviation Total Score | Average Total Score (Analytical Metrics) | Model Name | Delta |
|---|---|---|---|---|---|---|---|
| **17** | 25 | 36 | 31 | 4 | 26 | Claude Haiku | -5 |
| **7** | 26 | 35 | 31 | 3 | 23 | Claude Haiku | -8 |
| **9** | 27 | 33 | 30 | 2 | 27 | Claude Haiku | -4 |
| **20** | 22 | 33 | 30 | 4 | 18 | DeepSeek R1 | -11 |
| **18** | 19 | 34 | 29 | 6 | 27 | Claude Haiku | -1 |
| **5** | 22 | 34 | 28 | 4 | 26 | GPT 4o | -3 |
| **8** | 21 | 36 | 28 | 6 | 22 | Claude Haiku | -6 |
| **6** | 21 | 32 | 28 | 4 | 17 | Claude Haiku | -10 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| **2** | 20 | 33 | 28 | 5 | 26 | GPT 4o | -2 |
| **14** | 23 | 33 | 28 | 4 | 21 | DeepSeek R1 | -6 |
| **1** | 21 | 34 | 28 | 5 | 19 | GPT 4o | -8 |
| **12** | 18 | 34 | 26 | 7 | 21 | Llama 3 | -5 |
| **19** | 20 | 34 | 26 | 6 | 15 | Llama 3 | -11 |
| **13** | 16 | 33 | 26 | 6 | 21 | DeepSeek R1 | -5 |
| **4** | 20 | 30 | 26 | 4 | 24 | GPT 4o | -2 |
| **15** | 22 | 35 | 26 | 5 | 24 | GPT 4o | -1 |
| **16** | 18 | 33 | 25 | 5 | 25 | GPT 4o | 0 |
| **11** | 14 | 35 | 25 | 7 | 24 | Llama 3 | -1 |
| **3** | 16 | 31 | 24 | 5 | 21 | GPT 4o | -3 |
| **10** | 16 | 28 | 24 | 5 | 22 | Claude Haiku | -2 |

Overall, humans assigned higher scores ($\mu = 27$) than the metric-based evaluation ($\mu = 23$), suggesting that automatic metrics tend to underestimate the perceived usefulness of SHAP explanations when presented in a business context. This reinforces the importance of human-in-the-loop evaluation, particularly in domains like finance where interpretability must align with decision-maker needs.

Claude 3 Haiku achieved some of the strongest human ratings (e.g., Explanations 17, 9, 7, all presented an average between 30–31), though these were systematically scored lower by metrics (deltas of –3 to –7). This pattern indicates that while metrics penalized certain features, such as verbosity or reduced lexical overlap, human evaluators valued the same outputs for their clarity and contextual alignment. GPT-4o also received competitive human ratings, with deltas closer to zero (–1 to –2), showing that its outputs align more closely with both metric-based and human criteria.

LLaMA 3 and DeepSeek, however, displayed larger mismatches. For instance, LLaMA 3 (Explanation 15, $\mu = 26$) was rated well by humans but received a much lower metric score (15), creating the largest negative delta (–11). Similarly, some DeepSeek outputs (e.g., IDs 20 and 14) showed differences of 5–11 points between human and metric scores. These cases illustrate how certain models generate explanations that, while not optimized for automated measures like lexical overlap or readability indices, are still judged as meaningful by humans.

The variability in human ratings is also informative. Some explanations showed strong consensus (e.g., ID 9, $\sigma = 2$), while others revealed disagreement (e.g., IDs 13 and 19, $\sigma = 7$). This spread highlights the subjective nature of interpretability and the influence of individual evaluator expectations in a business-oriented setting.

Furthermore, feedback from the human evaluation highlighted a preference for explanations that include feature contributions with values, as these provide clearer context. The level of

technical detail should depend on the audience: technical users value detailed outputs, while business users benefit from simpler narratives. Additional suggestions included starting with a direct statement of the outcome, avoiding conditional phrasing that reduces confidence, clarifying the scale of contributions, and concluding with a concise summary. It was further suggested to conduct an experiment employing counterfactuals to examine how alterations in specific input variables might influence the model's predictions (as shown in the previous chapter).

In summary, the human evaluation of fraud SHAP explanations highlights three key insights:

1. Human evaluators systematically rated explanations higher than metric-based methods, showing that automatic scores may undervalue business-oriented clarity.

2. Claude 3 Haiku and GPT-4o produced the most human-preferred explanations, though the gap with metrics was larger for Claude.

3. LLaMA 3 and DeepSeek were penalized more heavily by metrics than by humans, indicating that some explanations resonate with users even if they diverge from standard linguistic or structural benchmarks. Moreover, because both are locally deployable, they represent valid options for organizations working with sensitive or proprietary data, where privacy constraints limit the use of cloud-based APIs.

These findings emphasize that while metric-based evaluations provide valuable consistency, they cannot fully substitute for human judgment, especially in high-stakes domains like finance where clarity, trust, and contextual alignment are critical.

#### 4.2.1.2. Explanations Examples

The qualitative analysis of SHAP explanations highlights how outputs vary in clarity, completeness, and business relevance across datasets and models.

The first example comes from Claude 3 Haiku on the Fraud Detection Business dataset, which achieved a high score ($\mu = 26.82$), the best score in human evaluation:

- *"The model predicts this customer is likely to commit bank account fraud. The strongest factor is their housing situation, which is in the "BA" category - a common, stable arrangement that suggests lower fraud risk. However, there are a few red flags. They are using an unusual "other" operating system which could indicate a fraud scheme. They also have a relatively high credit risk score of 278, meaning the bank sees them as more likely to default or commit fraud. Lastly, there have been a high number of recent applications (1052) from their zip code, which might be a fraud hotspot. So while their housing points to low risk, the rare OS, high risk score, and many local applications lead the model to predict potential fraud."*

This explanation is fluent and business-oriented, weaving technical factors into a clear narrative. It highlights both positive and negative contributions and contextualizes them in real-world terms ("fraud hotspot," "unusual OS"), making it accessible for decision-makers.

The second example, also from Claude 3 Haiku (μ = 25.62), the third-best score in human evaluation, and from Fraud Detection Business dataset, offers a sharper contrast:

- *"The model predicts this customer is likely committing bank account fraud. The strongest factors increasing fraud risk are that their housing status is BA (0.94 contribution) and they use a Windows device (0.59 contribution). Factors decreasing fraud risk include the customer's young age of 20 (-0.29), high 4-week velocity of $4,297 (-0.16), having transactions from 656 zip codes in the last 4 weeks (-0.11), and having 13 distinct email addresses associated with their date of birth in the last 4 weeks (-0.11)."*

Here, the explanation is more technical and less contextual. It lists contributions and directions but provides limited reasoning about why these variables matter. Compared to the previous example, it illustrates how the same model can alternate between narrative-rich and list-like outputs, affecting interpretability depending on stakeholder needs.

The third example comes from LLaMA 3 on the 72h Triage Business dataset (μ = 24.68):

- *"The ML model predicts a 72-hour ED reattendance after discharge based on the following factors. Having 16 ED visits in the past 90 days contributed positively to the prediction, indicating medium-term high ED utilization. Additionally, having 14 ED visits in the past 30 days contributed positively, suggesting frequent short-term ED use may be due to acute illness or poor follow-up. The model also considered that you had 50 ED visits in the past year, which captures long-term ED usage patterns and contributed positively. However, a triage acuity score of 4.0, derived from the standardized triage severity score at your initial visit, contributed negatively to the prediction. Finally, an abnormal body temperature of 35.83 degrees Celsius during your last recorded ED stay may signal unresolved conditions likely to drive return, which also contributed negatively to the prediction."*

This explanation is well structured, combining both technical features and domain framing ("poor follow-up," "unresolved conditions"). It balances detail with readability, though the volume of numbers of risks making it overwhelming. Nevertheless, the contextual language makes the explanation suitable for business-oriented stakeholders, particularly in healthcare.

The fourth example, from GPT-4o on the Critical Triage dataset (μ = 28.03), shows one of the strongest SHAP explanations:

- *"The model predicts a critical illness based on several factors. The most significant factor is the triage acuity, which has a value of 2.0 and strongly contributes to the*

*prediction. The patient's history of ICU admissions in the past year, with 2 occurrences, also significantly influences the prediction. Additionally, the patient's respiratory rate at triage is 25.0, which is a notable factor. The systolic blood pressure at triage is 101.0, and the patient's age is 84.0, both of which contribute to the prediction. The heart rate at triage is 109.0, adding to the likelihood of critical illness. However, the number of emergency department visits in the past year, which is 8, slightly decreases the prediction of critical illness."*

This explanation is fluent, precise, and interpretable for both technical and non-technical audiences. It organizes features by importance and provides clinically meaningful variables (acuity, ICU history, vitals). However, like other SHAP-based outputs, it remains descriptive rather than interpretive, explaining what influenced the prediction but not why these features matter clinically.

Finally, a Claude 3 Opus example from the Hospital Triage Business dataset (μ = 27.15) offers another perspective:

- *"The model predicts this patient is likely to be hospitalized following their ED visit. The key factors driving this prediction are the patient's history of 5 hospital admissions in the past year, indicating high inpatient utilization, and their elevated temperature of 37.9°C (100.2°F) at triage, suggesting a possible infection. The patient's triage acuity level of 3 slightly reduces hospitalization risk. However, their 2 hospitalizations in the past 90 days, comorbid fluid and electrolyte disorders, chief complaint of abdominal pain, and older age of 64 years all further increase the likelihood of admission. The patient's 6 ED visits in the past year and heart rate of 73 bpm at triage provide a small reduction in hospitalization risk."*

This explanation is accessible, structured, and provides a reasonable balance between quantitative and qualitative reasoning. It integrates technical features with clinically relevant framing, making it a strong example of business-oriented SHAP interpretation.

### 4.2.2. Counterfactuals

Following the approach of the previous subchapter, this section presents selected explanations, now complemented by counterfactuals. The first part of the qualitative analysis examines counterfactual explanations from models and different datasets. The first example comes from GPT-3.5 Turbo, which achieved one of the highest overall scores (μ = 18.95) on the Fraud CF All dataset for true positives.

- "*The ML model predicts bank account fraud based on customer and account features. In a scenario where the model correctly predicted fraud (True Positives), counterfactual analyses were conducted to determine changes that would make a fraudulent case*

*appear non-fraudulent to the model. For instance, increasing the 'name_email_similarity' feature by 0.06 or decreasing 'prev_address_months_count' by 26 would lead to a change in prediction from fraud to non-fraud. Similarly, adjusting 'current_address_months_count' by 39 or 'income' by 0.4 could also impact the model's prediction. These insights are valuable for simulating potential attacks and testing the model's robustness against fraudulent activities."*

This explanation is detailed and business relevant. It clearly frames the context (true positives) in a dataset with all cases, which presents that the context given in the prompt is working correctly. It also specifies how minimal changes could flip the model's prediction, which is highly useful for robustness testing. The inclusion of numerical adjustments enhances precision, and it is also accessible for non-technical audiences to understand. Its strength lies in explicitly linking feature modifications to fraud evasion, so balancing interpretability with practical utility.

The second example is from Claude 3 Sonnet, which achieved an even higher total score ($\mu$ = 25.96).

- *"The model predicted this case as fraudulent. However, the model would have predicted it as non-fraudulent if certain features were different. In the first counterfactual (CF#1), reducing the name-email similarity from 0.62 to 0.28 (-54.8%), decreasing the current address months count from 83 to 37 (-55.4%), lowering the zip count in the last 4 weeks from 1638 to 901 (-45%), slightly decreasing the velocity in the last 4 weeks from 4228.27 to 4140.25 (-2.1%), reducing the credit risk score from 192 to 160 (-16.7%), and decreasing the bank months count from 2 to 1 (-50%) would flip the prediction to non-fraudulent."*

This output is more exhaustive than GPT-3.5, presenting one of 4 counterfactuals that were presented in the input with precise percentage changes. The structured format improves clarity and provides clear thresholds for flipping predictions. However, the sheer number of variables and the heavy use of numerical detail may overwhelm non-technical stakeholders. Still, the explanation is rigorous and reflects why Claude 3 models consistently performed well in evaluations: they produce systematic, granular outputs that balance linguistic fluency with analytic detail.

The third example is drawn from GPT-4o, which obtained a strong score ($\mu$ = 24.88) on false positives.

- "*The model predicts bank account fraud based on various customer and account features. In cases where the model incorrectly predicts fraud (False Positives), we can identify minimal changes that would have prevented these legitimate cases from being flagged. For instance, in one scenario, reducing the similarity between the customer's name and email from 0.16 to 0.07, decreasing the number of months at the previous address from 32 to 6, and reducing the number of months at the current address from*

*101 to 0 would have changed the prediction. Additionally, lowering the number of zip code changes in the last four weeks from 1626 to 1161, decreasing the transaction velocity over four weeks from 4484.93 to 4255.4, reducing the credit risk score from 130 to -149, and slightly lowering the proposed credit limit from 200 to 190 would also have prevented the false positive. Other scenarios suggest that increasing the income from 0.1 to 0.5, extending the current address duration from 101 to 140 months, and adjusting the credit risk score from 130 to 158 could also avert a false fraud alert. These insights help in understanding the model's decision-making process and in reducing false alarms, thereby enhancing user trust."*

This explanation stands out for its breadth, it not only lists multiple counterfactual scenarios but also contextualizes their relevance (reducing false positives to improve trust). The narrative is accessible, and while the numeric details are dense, they remain interpretable. GPT-4o's strength here lies in striking a balance between technical accuracy and business applicability.

Finally, the DeepSeek example focuses on false negatives and received a total score of 22.07.

- "*The model predicts whether a bank account transaction is fraudulent (1) or not (0). This explanation focuses on instances it correctly identified as non-fraudulent, but which were actually fraudulent. To understand why these specific frauds might have been missed and how the prediction could be improved for similar cases, we can consider small hypothetical changes to the transaction details that would likely alter the model's outcome from 0 (non-fraud) to 1 (fraud). For example, slightly reducing the similarity score between a name and email address by about -25% or -34%, decreasing the count of current addresses significantly by around -41% or -13%, lowering the zip code frequency count moderately by approximately -3% or -8%, increasing transaction velocity substantially by nearly +14% or +20%, or slightly improving a credit risk score by about -7% could have potentially triggered the fraud detection in these cases, highlighting specific features and magnitudes of change that might be critical for adjusting model behavior to improve recall."*

This explanation is informative and highlights how missed fraud cases could have been detected, which is highly relevant for improving recall. It combines percentage ranges with qualitative interpretation, offering insight into how both subtle and larger shifts affect predictions, though the numerical detail makes the narrative somewhat dense. The language is also less polished than GPT-based outputs, and parts of the explanation are repetitive.

# CONCLUSIONS AND FUTURE WORKS

This study set out to explore how LLMs can enhance the interpretability of ML models predictions models through the generation of natural language explanations. By integrating XAI techniques, particularly SHAP and counterfactuals, with multiple LLMs and two explanation styles (technical and business-oriented), a framework was proposed to produce readable, relevant, and domain-adaptable explanations. This project was evaluated by combining a quantitative, metric-based analysis with a qualitative, human-centered assessment by fraud stakeholders, providing both global and local views of explanation quality. The resulting narratives intended to bridge the gap between technical model outputs and the interpretative needs of non-technical stakeholders.

The empirical results demonstrated that LLMs can effectively transform complex model explanations into concise, structured, and intelligible narratives. GPT-4o achieved the highest overall scores across datasets and formats for SHAP explanations, while GPT-3.5 Turbo and GPT-4o led in counterfactuals. Claude 3 models showed consistent, competitive performance, with Claude 3 Haiku often preferred by human evaluators. Locally deployable models (DeepSeek, LLaMA 3) trailed in aggregate metrics but were judged more favourably by humans in several cases and remain attractive where data privacy or regulatory constraints preclude cloud APIs.

Despite these strengths, results revealed persistent limitations: lower understandability, readability, and completeness, especially when explanations must bridge structured inputs and narrative outputs. The counterfactual analysis, motivated by feedback from the SHAP human study, was restricted to finance and, when broken down by TP/FP/TN/FN, showed diagnostic value (e.g., insights to reduce false alarms or improve recall) but also highlighted variability in coherence and accuracy. Overall, LLMs materially advance post-hoc interpretability and are a promising tool for enhancing explainability, nevertheless further refinement is required to ensure explanations are consistently accessible, complete, and decision-useful in high-stakes contexts.

This research was guided by five main questions, all of which have been addressed through the proposed methodology and empirical analysis:

1. **Develop a framework that investigates and compares different LLMs (e.g., GPT, Claude, DeepSeek, LLaMA) for generating post-hoc explanations of ML predictions.**

The research successfully developed a modular framework that couples black-box predictions with post-hoc explainers and LLM narration. Eight models (GPT-4o/4 Turbo/3.5, Claude 3 Haiku/Opus/Sonnet, DeepSeek, LLaMA 3) were integrated under a common prompting and evaluation pipeline. Results show consistent rank ordering, with GPT-4o leading overall for SHAP, and GPT-3.5 + GPT-4o leading in counterfactuals, demonstrating its capacity to generate interpretable outputs from black-box models.

2. **Incorporate established explainability methods, such as SHAP, and extend the framework with counterfactual explanations to provide complementary perspectives on model interpretability.**

The framework supports SHAP across healthcare and finance, and counterfactuals in finance (added in response to SHAP human-evaluation feedback. Counterfactuals were analyzed both in aggregate and by TP/FP/TN/FN, enabling diagnostic insights (e.g., reducing false positives, improving recall, probing decision boundaries, and robustness testing). While counterfactuals produced concise and specific narratives, they underperformed SHAP on completeness, readability, and accuracy, indicating the two methods are complementary.

3. **Design two explanation styles, technical and business-oriented, to address different stakeholder groups, so as to simplify and enhance trust, transparency, and decision-making.**

Both styles were instantiated and evaluated across datasets. Human feedback from fraud stakeholders preferred feature-contribution statements with values (technical) but emphasized audience-dependent detail and requested clear outcome first, limited conditional phrasing, scale clarification, and a brief concluding summary. Quantitatively, format effects were dataset/model-dependent (e.g., business style improved some Claude results in triage, while technical style benefited GPT-4 Turbo in hospital triage), underscoring the value of offering both styles.

4. **Test the effectiveness of the proposed LLM-based framework across diverse datasets (e.g., tabular) and in critical domains such as healthcare and finance)**.

The framework was tested on two distinct, high-impact tabular datasets from healthcare and finance domains. Results showed robust generalization capabilities across different datasets, with particularly strong performance observed in lower-dimensional datasets, confirming the adaptability and scalability of the approach.

5. **Evaluate the quality and impact of LLM-based explanations through heuristic assessment, LLM-based metrics, and a domain-specific use case in the finance sector.**

A mixed-method evaluation combined heuristic/NLP metrics (e.g., ROUGE-L, perplexity, readability indices, specificity) with LLM-based ratings (accuracy, fluency, completeness, understandability) and a human study in finance. SHAP explanations reached a Total Score on average of 22.15, with strengths in conciseness and specificity, but weaknesses in readability/understandability/completeness. Counterfactuals scored lower overall (approximately 15.91) but provided strong conciseness/specificity diagnostics. Human evaluators in fraud rated explanations higher than automated metrics, especially for Claude 3 Haiku and GPT-4o, and confirmed the importance of contribution values, audience-tuned detail, outcome-first structure, and summary closure.

## 4.3. Work Contribution

This research makes several contributions to the field of XAI. It introduces a comprehensive framework that integrates post-hoc explanation methods, namely SHAP and counterfactuals, with Large Language Models to generate textual narratives that are accessible to both technical and business-oriented stakeholders. By producing dual explanation styles, the framework addresses the interpretability and communication gap that is particularly critical in high-stakes domains such as healthcare and finance. The study further contributes a systematic comparison of eight LLMs, encompassing both API-based and locally deployable models, thereby highlighting the trade-offs between performance, accessibility, and privacy. A robust evaluation methodology was developed, combining heuristic NLP metrics, LLM-based semantic assessments, and human-in-the-loop evaluation in a financial use case. This mixed-method approach demonstrated that while automated metrics provide structure and consistency, they often underestimate the clarity and contextual value recognized by human stakeholders. Finally, the inclusion of counterfactual explanations, analyzed across confusion matrix categories, illustrates how explainability can support not only interpretability but also model refinement, robustness testing, and trust-building.

## 4.4. Constraints and Limitations

While the findings of this study are promising, several limitations must be acknowledged. The framework is inherently dependent on the capabilities of specific LLMs, many of which are API-based and thus subject to cost, accessibility, and data privacy concerns. Although locally deployable models such as LLaMA 3 and DeepSeek were included, their performance remained inconsistent compared to hosted alternatives and is computationally expensive. Another limitation lies in the reliance on automated metrics for most evaluations, which, while systematic, do not fully capture subjective dimensions such as trust, clarity, and perceived usefulness. Human evaluation was conducted only in the finance domain, limiting broader generalization. Furthermore, the study focused exclusively on structured, tabular data, leaving its applicability to unstructured or multimodal data untested. Finally, explanation quality was shown to be sensitive to prompt design and model alignment, introducing variability in performance and underscoring the challenge of reproducibility in LLM-based XAI systems.

## 4.5. Future Work

This study opens several avenues for future research and development. A priority is the expansion of human-centered evaluation, involving broader groups of end-users such as clinicians, fraud analysts, and decision-makers. Their feedback will provide deeper insights into the alignment between generated explanations and real-world decision-making needs.

Another promising avenue is the extension to multimodal data, where LLM-based explanation frameworks could be applied to text-rich EHR, medical imaging, or hybrid financial data streams. In parallel, automated prompt optimization techniques (e.g., RL or meta-prompting)

should be explored to reduce variability and improve explanation quality across different contexts.

Another critical area is the analysis of bias and fairness in LLM-generated narratives, particularly in sensitive applications where interpretability intersects with ethical concerns. Moreover, improving the performance of open-source, locally deployable models such as LLaMA 3 and DeepSeek could enable privacy-preserving implementations, especially in regulated environments where data cannot leave institutional boundaries.

Lastly, while the current study demonstrated strengths in conciseness, specificity, and accuracy, weaknesses in readability, understandability, and completeness remain. Future work should aim to enhance explanations by not only describing which features influenced a prediction but also clarifying why those features are domain-relevant as domain-specialist would do, thus moving closer to causal, context-aware reasoning.

In summary, this thesis highlights the potential of LLMs to advance XAI by turning complex model outputs into narratives that are clearer, more accessible, and more actionable. While challenges in completeness, readability, and generalizability remain, the framework developed here represents a step toward AI systems that are not only more transparent and trustworthy, but also better aligned with the needs of decision-makers in high-stake domains. Ultimately, the value of AI will be measured not only by accuracy but also by clarity and trustworthiness, and this work lays a foundation for future research to build more rigorous and human-centered approaches to XAI.

## BIBLIOGRAPHICAL REFERENCES


Ai, Q., & Narayanan, L. R. (2021). Model-agnostic vs. model-intrinsic interpretability for explainable product search. *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. https://doi.org/10.1145/3459637.3482276

Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. (2019). Optuna: A Next-generation Hyperparameter Optimization Framework. *ArXiv (Cornell University)*. https://doi.org/10.48550/arXiv.1907.10902

Ali, S., Abuhmed, T., El-Sappagh, S., Muhammad, K., Alonso-Moral, J. M., Confalonieri, R., Guidotti, R., Del Ser, J., Díaz-Rodríguez, N., & Herrera, F. (2023). Explainable artificial intelligence (XAI): What we know and what is left to attain Trustworthy Artificial Intelligence. *Information Fusion*, *99*, 101805. https://doi.org/10.1016/j.inffus.2023.101805

Anthropic. (2024). *The Claude 3 model family: Opus, Sonnet, Haiku*. Anthropic. https://www-cdn.anthropic.com/de8ba9b01c9ab7cbabf5c33b80b7bbc618857627/Model_Card_Claude_3.pdf

Bansal, N., Agarwal, C., & Nguyen, A. (2020). SAM: The sensitivity of attribution methods to hyperparameters. *arXiv (Cornell University)*. https://doi.org/10.48550/arXiv.2003.08754

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., . . . Amodei, D. (2020). Language Models are Few-Shot Learners. *arXiv (Cornell University)*. https://doi.org/10.48550/arXiv.2005.14165

Burton, J., Moubayed, N. A., & Enshaei, A. (2023). Natural Language Explanations for Machine Learning classification decisions. *2022 International Joint Conference on Neural Networks (IJCNN)*, 1–9. https://doi.org/10.1109/IJCNN54540.2023.10191637

DeepSeek-AI, Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., Bi, X., Zhang, X., Yu, X., Wu, Y., Wu, Z. F., Gou, Z., Shao, Z., Li, Z., Gao, Z., . . . Zhang, Z. (2025). DeepSeek-R1: Incentivizing reasoning capability in LLMs via Reinforcement Learning. *arXiv (Cornell University)*. https://doi.org/10.48550/arXiv.2501.12948

Doshi-Velez, F., & Kim, B. (2017). Towards a rigorous science of interpretable machine learning. *arXiv (Cornell University)*. https://doi.org/10.48550/arXiv.1702.08608

Fredes, A., & Vitria, J. (2024). Using LLMs for explaining sets of counterfactual examples to final users. *arXiv (Cornell University)*. https://doi.org/10.48550/arXiv.2408.15133

Ganesan, K. (2018). ROUGE 2.0: Updated and Improved measures for evaluation of summarization tasks. *arXiv (Cornell University)*. https://doi.org/10.48550/arXiv.1803.01937

Hassija, V., Chamola, V., Mahapatra, A., Singal, A., Goel, D., Huang, K., Scardapane, S., Spinelli, I., Mahmud, M., & Hussain, A. (2023). Interpreting Black-Box Models: A Review on Explainable Artificial Intelligence. *Cognitive Computation*, *16*(1), 45–74. https://doi.org/10.1007/s12559-023-10179-8

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in Information Systems Research. *MIS Quarterly*, *28*(1), 75. https://doi.org/10.2307/25148625

Jesus, S., Pombal, J., Alves, D., Cruz, A., Saleiro, P., Ribeiro, R. P., Gama, J., & Bizarro, P. (2022). Turning the tables: biased, imbalanced, dynamic tabular datasets for ML evaluation. *arXiv (Cornell University)*.https://doi.org/10.48550/arXiv.2211.13358

Johnson, A., Bulgarelli, L., Pollard, T., Celi, L. A., Mark, R., & Horng, S. (2023). MIMIC-IV-ED (version 2.2) [Dataset]. *PhysioNet*. https://doi.org/10.13026/5ntk-km72

Kalyan, K. S. (2024). A survey of GPT-3 family large language models including ChatGPT and GPT-4. *Natural Language Processing Journal*, *6*, 100048. https://doi.org/10.1016/j.nlp.2023.100048

Kampelopoulos, D., Tsanousa, A., Vrochidis, S., & Kompatsiaris, I. (2025). A review of LLMs and their applications in the architecture, engineering and construction industry. *Artificial Intelligence Review*, *58*(8). https://doi.org/10.1007/s10462-025-11241-7

Kroeger, N., Ley, D., Krishna, S., Agarwal, C., & Lakkaraju, H. (2024). In-context explainers: Harnessing LLMs for explaining black box models. *arXiv (Cornell University)*. https://doi.org/10.48550/arXiv.2310.05797

Lundberg, S. M., & Lee, S. (2017). A unified approach to interpreting model predictions. *arXiv (Cornell University)*. https://doi.org/10.48550/arXiv.1705.07874

Mavrepis, P., Makridis, G., Fatouros, G., Koukos, V., Separdani, M. M., & Kyriazis, D. (2024). XAI for All: Can Large Language Models Simplify Explainable AI? *arXiv (Cornell University)*. https://doi.org/10.48550/arXiv.2401.13110

Molnar, C. (2025). *Interpretable machine learning: A guide for making black box models explainable* (3rd ed.). GitHub. https://christophm.github.io/interpretable-ml-book/

Moreira, C., Chou, Y., Hsieh, C., Ouyang, C., Pereira, J., & Jorge, J. (2024). Benchmarking Instance-Centric Counterfactual Algorithms for XAI: From white box to black box. *ACM Computing Surveys*. https://doi.org/10.1145/3672553

Neha, F., & Bhati, D. (2025). A Survey of DeepSeek Models. *TechRxiv*. https://doi.org/10.36227/techrxiv.173896582.25938392/v1

Nori, H., Jenkins, S., Koch, P., & Caruana, R. (2019). InterpretML: a unified framework for machine learning interpretability. *arXiv (Cornell University)*. https://doi.org/10.48550/arXiv.1909.09223

Radford, A., Jozefowicz, R., & Sutskever, I. (2017). Learning to generate reviews and discovering sentiment. *arXiv (Cornell University)*. https://doi.org/10.48550/arXiv.1704.01444

Reddy, G. P., & Kumar, Y. V. P. (2023). Explainable AI (XAI): Explained. *2023 IEEE Open Conference of Electrical, Electronic and Information Sciences (eStream)*, 1–6. https://doi.org/10.1109/ESTREAM59056.2023.10134984

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should I trust you?": explaining the predictions of any classifier. *arXiv (Cornell University)*. https://doi.org/10.48550/arXiv.1602.04938

Rieg, T., Frick, J., Baumgartl, H., & Buettner, R. (2020). Demonstration of the potential of white-box machine learning approaches to gain insights from cardiovascular disease electrocardiograms. *PLoS ONE*, *15*(12), e0243615. https://doi.org/10.1371/journal.pone.0243615

Singh, C., Inala, J. P., Galley, M., Caruana, R., & Gao, J. (2024). Rethinking interpretability in the era of large language models. *arXiv (Cornell University)*. https://doi.org/10.48550/arXiv.2402.01761

Situ, X., Zukerman, I., Paris, C., Maruf, S., & Haffari, G. (2021). Learning to explain: Generating stable explanations fast. In F. Xia, W. Li, & R. Navigli (Eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)* (pp. 5340–5355). Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.acl-long.415

Slack, D., Hilgard, S., Jia, E., Singh, S., & Lakkaraju, H. (2019). Fooling LIME and SHAP: Adversarial Attacks on Post hoc Explanation Methods. *arXiv (Cornell University)*. https://doi.org/10.48550/arXiv.1911.02508

Suh, A., Alperin, K., Li, H., & Gomez, S. R. (2025). Don't just translate, agitate: Using large language models as devil's advocates for AI explanations. *Zenodo*. https://doi.org/10.5281/zenodo.15170455

Wu, X., Zhao, H., Zhu, Y., Shi, Y., Yang, F., Liu, T., Zhai, X., Yao, W., Li, J., Du, M., & Liu, N. (2024). Usable XAI: 10 Strategies Towards Exploiting Explainability in the LLM Era. *arXiv (Cornell University)*. https://doi.org/10.48550/arXiv.2403.08946

Xie, F., Zhou, J., Lee, J. W., Tan, M., Li, S., Rajnthern, L. S., Chee, M. L., Chakraborty, B., Wong, A. I., Dagan, A., Ong, M. E. H., Gao, F., & Liu, N. (2021). Benchmarking emergency department triage prediction models with machine learning and large public electronic health records. *arXiv (Cornell University)*. https://doi.org/10.48550/arXiv.2111.11017

Yeh, C., Hsieh, C., Suggala, A. S., Inouye, D. I., & Ravikumar, P. (2019). On the (In)fidelity and Sensitivity for Explanations. *arXiv (Cornell University)*. https://doi.org/10.48550/arXiv.1901.09392

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., & Artzi, Y. (2019). BERTScore: Evaluating Text Generation with BERT. *arXiv (Cornell University)*. https://doi.org/10.48550/arXiv.1904.09675

Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., Du, Y., Yang, C., Chen, Y., Chen, Z., Jiang, J., Ren, R., Li, Y., Tang, X., Liu, Z., . . . Wen, J. (2023). A survey of large language models. *arXiv (Cornell University)*. https://doi.org/10.48550/arXiv.2303.18223

Zytek, A., Pidò, S., & Veeramachaneni, K. (2024). LLMS for XAI: Future Directions for Explaining Explanations. *arXiv (Cornell University)*. https://doi.org/10.48550/arXiv.2405.06064

Zytek, A., Pido, S., Alnegheimish, S., Berti-Équille, L., & Veeramachaneni, K. (2024). Explingo: Explaining AI predictions using large language models. In *2024 IEEE International Conference on Big Data (BigData)* (pp. 1197–1208). IEEE. https://doi.org/10.48550/arXiv.2412.05145

# APPENDIX A

## A1) SHAP

| | accuracy | completeness | fluency | conciseness | coverage | interpretability | specificity | understandability | readability | flesch_kincaid_grade_level | total score | total_score_no_conciseness |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| count | 216.0000 | 216.0000 | 216.0000 | 216.0000 | 216.0000 | 216.0000 | 216.0000 | 216.0000 | 216.0000 | 216.0000 | 216.0000 | 216.0000 |
| mean | 3.0222 | 2.7444 | 2.6139 | 3.8580 | 0.9701 | 2.6454 | 3.6658 | 1.6037 | 0.7582 | 0.2675 | 22.1493 | 18.2913 |
| std | 1.0139 | 1.1878 | 0.7611 | 0.4022 | 0.3906 | 0.6219 | 0.2679 | 0.7466 | 0.6003 | 0.3441 | 3.2832 | 3.2583 |
| min | 0.0000 | 0.0000 | 0.6000 | 1.5482 | 0.2500 | 0.6246 | 1.6000 | 0.0000 | 0.0000 | 0.0000 | 6.6684 | 5.0684 |
| 25% | 2.4000 | 2.0000 | 2.2000 | 4.0000 | 0.6359 | 2.2402 | 3.5468 | 1.0000 | 0.2142 | 0.0000 | 20.1057 | 16.1928 |
| 50% | 3.2000 | 3.2000 | 2.8000 | 4.0000 | 0.9804 | 2.8664 | 3.7351 | 1.6000 | 0.6406 | 0.1110 | 22.6675 | 18.8148 |
| 75% | 4.0000 | 4.0000 | 3.2000 | 4.0000 | 1.2678 | 3.1261 | 3.8366 | 2.0000 | 1.1924 | 0.4591 | 24.5989 | 20.7528 |
| max | 4.0000 | 4.0000 | 3.8000 | 4.0000 | 2.1046 | 3.4897 | 3.9892 | 3.0000 | 2.1444 | 1.2771 | 27.8934 | 23.9118 |

Table A1.1 – Results of SHAP Explanation Metrics

| | accuracy | | | | completeness | | | | fluency | | | | conciseness | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| dataset | mean | std | min | max | mean | std | min | max | mean | std | min | max | mean | std | min | max |
| 72h Triage | 3.2667 | 0.8478 | 1.6000 | 4.0000 | 2.7667 | 1.1060 | 0.4000 | 4.0000 | 2.7583 | 0.7027 | 1.0000 | 3.8000 | 3.9118 | 0.2280 | 3.1446 | 4.0000 |
| 72h Triage Business | 3.5000 | 0.8108 | 1.6000 | 4.0000 | 2.9667 | 1.2024 | 0.4000 | 4.0000 | 2.7917 | 0.6814 | 0.8000 | 3.6000 | 3.9654 | 0.1631 | 3.2000 | 4.0000 |
| Critical Triage | 3.3667 | 1.0003 | 0.8000 | 4.0000 | 3.0833 | 0.9832 | 0.4000 | 4.0000 | 2.9083 | 0.7366 | 1.0000 | 3.8000 | 3.9903 | 0.0418 | 3.7959 | 4.0000 |
| Critical Triage Business | 3.6667 | 0.5231 | 2.4000 | 4.0000 | 3.1667 | 1.0072 | 0.8000 | 4.0000 | 2.7417 | 0.6965 | 1.2000 | 3.6000 | 3.9934 | 0.0238 | 3.8938 | 4.0000 |
| Fraud Detection | 3.6000 | 0.5779 | 2.4000 | 4.0000 | 3.3000 | 0.7199 | 1.6000 | 4.0000 | 2.9000 | 0.5242 | 1.4000 | 3.6000 | 3.1145 | 0.6023 | 1.5482 | 4.0000 |
| Fraud Detection Business | 2.0000 | 0.8506 | 0.8000 | 3.2000 | 2.5500 | 0.8733 | 0.0000 | 4.0000 | 1.6583 | 0.4809 | 0.6000 | 2.4000 | 3.9000 | 0.4899 | 1.6000 | 4.0000 |
| Fraud Detection VC | 2.1000 | 1.1018 | 0.0000 | 4.0000 | 1.6167 | 1.4660 | 0.0000 | 4.0000 | 2.4583 | 0.7857 | 1.2000 | 3.8000 | 3.8861 | 0.3716 | 2.2318 | 4.0000 |
| Hospital Triage | 3.1000 | 0.6379 | 1.6000 | 4.0000 | 2.7333 | 1.2182 | 0.4000 | 4.0000 | 2.7833 | 0.7642 | 0.8000 | 3.4000 | 3.9603 | 0.1634 | 3.2000 | 4.0000 |
| Hospital Triage Business | 2.6000 | 0.9510 | 0.8000 | 4.0000 | 2.5167 | 1.2342 | 0.4000 | 4.0000 | 2.5250 | 0.6771 | 1.2000 | 3.4000 | 4.0000 | 0.0000 | 4.0000 | 4.0000 |

| | interpretability | | | | specificity | | | | understandability | | | | readability | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| dataset | mean | std | min | max | mean | std | min | max | mean | std | min | max | mean | std | min | max |
| 72h Triage | 3.0007 | 0.6200 | 0.6584 | 3.4897 | 3.5179 | 0.2129 | 3.1433 | 3.8801 | 1.8417 | 0.8587 | 0.4000 | 3.0000 | 1.2009 | 0.5067 | 0.1274 | 2.0980 |
| 72h Triage Business | 2.9161 | 0.3772 | 1.9192 | 3.2988 | 3.6118 | 0.1931 | 3.1201 | 3.9407 | 1.8750 | 0.6848 | 0.8000 | 3.0000 | 0.7174 | 0.3467 | 0.2030 | 1.2498 |
| Critical Triage | 2.9992 | 0.3939 | 1.7921 | 3.3246 | 3.6737 | 0.1971 | 3.1597 | 3.9521 | 1.6667 | 0.8186 | 0.2000 | 3.0000 | 1.0169 | 0.5513 | 0.0000 | 2.0996 |
| Critical Triage Business | 2.7838 | 0.3708 | 1.8314 | 3.1908 | 3.7922 | 0.0834 | 3.5818 | 3.9548 | 1.2250 | 0.5542 | 0.6000 | 2.8000 | 0.2367 | 0.2591 | 0.0000 | 0.9221 |
| Fraud Detection | 2.2122 | 0.4727 | 0.7920 | 2.9738 | 3.6391 | 0.2007 | 3.2579 | 3.9414 | 1.4583 | 0.7283 | 0.2000 | 3.0000 | 1.2366 | 0.5971 | 0.3024 | 2.1444 |
| Fraud Detection Business | 1.8105 | 0.5376 | 0.6246 | 2.6747 | 3.7263 | 0.4660 | 1.6000 | 3.9892 | 1.7917 | 0.6420 | 0.8000 | 3.0000 | 0.8547 | 0.6018 | 0.0348 | 1.6705 |
| Fraud Detection VC | 2.2276 | 0.5958 | 0.9796 | 2.9160 | 3.6573 | 0.3474 | 2.2567 | 3.9721 | 1.4167 | 0.9173 | 0.0000 | 3.0000 | 1.1095 | 0.4987 | 0.0000 | 2.0415 |
| Hospital Triage | 2.9863 | 0.4224 | 1.3008 | 3.4177 | 3.6226 | 0.2849 | 2.9885 | 3.9716 | 1.5250 | 0.7685 | 0.0000 | 3.0000 | 0.3353 | 0.2856 | 0.0000 | 1.1227 |
| Hospital Triage Business | 2.8717 | 0.3329 | 2.0513 | 3.2935 | 3.7515 | 0.1619 | 3.4229 | 3.9728 | 1.6333 | 0.5096 | 0.6000 | 2.4000 | 0.1159 | 0.1297 | 0.0000 | 0.3189 |

| | flesch_kincaid_grade_level | | | | coverage | | | | total score | | | | total_score_no_conciseness | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| dataset | mean | std | min | max | mean | std | min | max | mean | std | min | max | mean | std | min | max |
| 72h Triage | 0.4796 | 0.3855 | 0.0000 | 1.2565 | 0.5709 | 0.2513 | 0.2774 | 1.1212 | 23.3152 | 3.3095 | 14.4813 | 27.8934 | 19.4034 | 3.3767 | 10.4813 | 23.9118 |
| 72h Triage Business | 0.1284 | 0.1930 | 0.0000 | 0.5806 | 1.4343 | 0.3510 | 0.5456 | 2.1046 | 23.9067 | 3.4072 | 15.0316 | 27.5116 | 19.9413 | 3.3360 | 11.0316 | 23.5116 |
| Critical Triage | 0.3180 | 0.3698 | 0.0000 | 1.2446 | 0.6594 | 0.2534 | 0.3656 | 1.2828 | 23.6825 | 3.2658 | 15.5099 | 27.6504 | 19.6922 | 3.2724 | 11.5099 | 23.6504 |
| Critical Triage Business | 0.0680 | 0.1989 | 0.0000 | 0.9339 | 1.1821 | 0.2280 | 0.7198 | 1.6451 | 22.8562 | 2.2212 | 17.2568 | 26.2115 | 18.8629 | 2.2246 | 13.2568 | 22.2115 |
| Fraud Detection | 0.5775 | 0.3982 | 0.0000 | 1.2512 | 0.9291 | 0.2498 | 0.6199 | 1.7005 | 22.9672 | 2.0791 | 17.8827 | 25.6106 | 19.8527 | 2.2775 | 14.8645 | 22.7346 |
| Fraud Detection Business | 0.2419 | 0.2522 | 0.0000 | 0.6652 | 1.1582 | 0.2646 | 0.3691 | 1.6571 | 19.6915 | 3.2445 | 6.6684 | 23.5931 | 15.7915 | 2.8371 | 5.0684 | 19.5931 |
| Fraud Detection VC | 0.3922 | 0.3805 | 0.0000 | 1.2771 | 0.8967 | 0.3908 | 0.2500 | 1.5333 | 19.7612 | 3.5477 | 10.7501 | 26.0082 | 15.8751 | 3.4068 | 8.5183 | 22.0343 |
| Hospital Triage | 0.1751 | 0.2784 | 0.0000 | 0.9334 | 0.7247 | 0.2912 | 0.4480 | 1.4396 | 21.9460 | 2.4529 | 15.2294 | 25.6781 | 17.9857 | 2.3654 | 12.0294 | 21.6781 |
| Hospital Triage Business | 0.0271 | 0.0465 | 0.0000 | 0.1638 | 1.1756 | 0.2529 | 0.6277 | 1.8076 | 21.2168 | 2.7631 | 16.0149 | 25.0346 | 17.2168 | 2.7631 | 12.0149 | 21.0346 |

Table A1.2 – Metrics Results for SHAP Explanations by Dataset

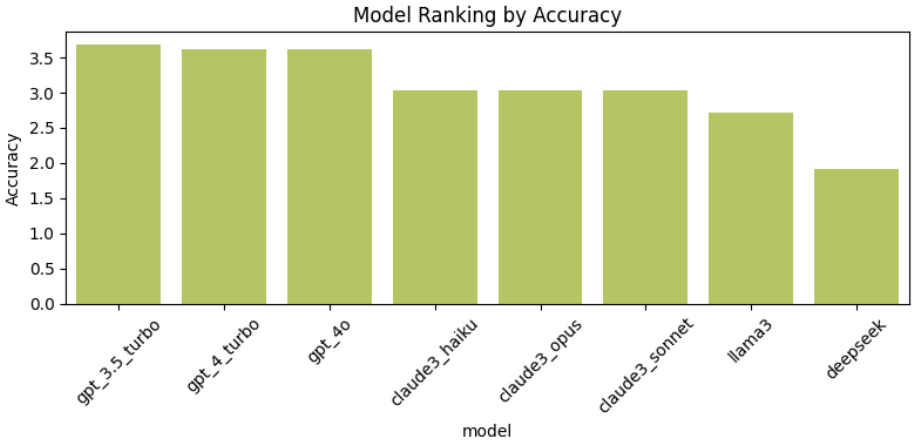
| | total score | | | | accuracy | | | | completeness | | | | fluency | | | | conciseness | | | | total_score_no_conciseness | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| model | mean | std | min | max | mean | std | min | max | mean | std | min | max | mean | std | min | max | mean | std | min | max | mean | std | min | max |
| claude3_haiku | 23.51 | 2.20 | 18.03 | 27.11 | 2.99 | 1.01 | 0.80 | 4.00 | 3.20 | 0.97 | 0.00 | 4.00 | 2.90 | 0.50 | 2.00 | 3.60 | 3.91 | 0.28 | 2.88 | 4.00 | 19.60 | 2.25 | 14.03 | 23.11 |
| claude3_opus | 23.54 | 2.38 | 17.83 | 27.51 | 2.99 | 1.01 | 0.80 | 4.00 | 3.20 | 0.97 | 0.00 | 4.00 | 2.93 | 0.64 | 1.40 | 3.80 | 3.91 | 0.28 | 2.88 | 4.00 | 19.63 | 2.42 | 13.83 | 23.51 |
| claude3_sonnet | 23.46 | 2.31 | 17.83 | 27.51 | 2.99 | 1.01 | 0.80 | 4.00 | 3.20 | 0.97 | 0.00 | 4.00 | 2.85 | 0.58 | 1.80 | 3.60 | 3.91 | 0.28 | 2.88 | 4.00 | 19.55 | 2.37 | 13.83 | 23.51 |
| deepseek | 16.89 | 2.92 | 6.67 | 20.82 | 1.93 | 0.95 | 0.00 | 4.00 | 1.19 | 0.75 | 0.00 | 2.40 | 2.06 | 0.60 | 0.60 | 3.00 | 3.71 | 0.60 | 1.60 | 4.00 | 13.18 | 2.51 | 5.07 | 16.82 |
| gpt_3.5_turbo | 20.45 | 2.07 | 14.48 | 24.97 | 3.41 | 0.85 | 0.80 | 4.00 | 1.78 | 0.86 | 0.00 | 4.00 | 1.60 | 0.63 | 0.80 | 3.00 | 4.00 | 0.00 | 4.00 | 4.00 | 16.45 | 2.07 | 10.48 | 20.97 |
| gpt_4_turbo | 23.12 | 2.40 | 16.01 | 27.26 | 3.44 | 0.82 | 0.80 | 4.00 | 3.35 | 0.77 | 1.20 | 4.00 | 2.92 | 0.70 | 1.20 | 3.80 | 3.66 | 0.64 | 1.55 | 4.00 | 19.46 | 2.56 | 12.01 | 23.74 |
| gpt_4o | 25.01 | 1.50 | 21.97 | 27.89 | 3.61 | 0.56 | 2.40 | 4.00 | 3.75 | 0.43 | 2.40 | 4.00 | 3.08 | 0.46 | 1.60 | 3.80 | 3.86 | 0.39 | 2.55 | 4.00 | 21.15 | 1.51 | 17.97 | 23.91 |
| llama3 | 21.21 | 2.16 | 17.36 | 24.79 | 2.81 | 0.92 | 1.60 | 4.00 | 2.30 | 1.00 | 0.40 | 4.00 | 2.56 | 0.60 | 0.80 | 3.40 | 3.91 | 0.28 | 2.69 | 4.00 | 17.30 | 2.23 | 13.42 | 21.36 |

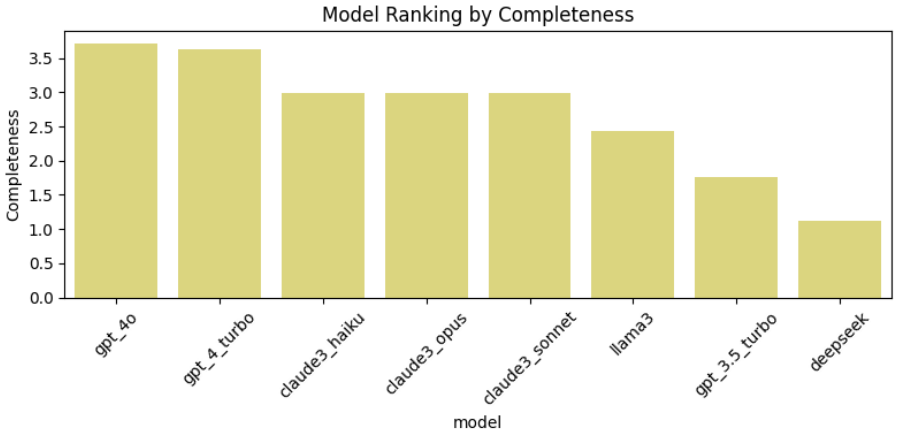
| | interpretability | | | | specificity | | | | understandability | | | | readability | | | | flesch_kincaid_grade_level | | | | coverage | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| model | mean | std | min | max | mean | std | min | max | mean | std | min | max | mean | std | min | max | mean | std | min | max | mean | std | min | max |
| claude3_haiku | 2.77 | 0.46 | 1.52 | 3.29 | 3.74 | 0.12 | 3.44 | 3.89 | 1.71 | 0.66 | 0.20 | 3.00 | 1.05 | 0.64 | 0.05 | 2.14 | 0.31 | 0.36 | 0.00 | 1.25 | 0.93 | 0.38 | 0.28 | 1.54 |
| claude3_opus | 2.77 | 0.46 | 1.52 | 3.29 | 3.74 | 0.12 | 3.44 | 3.89 | 1.71 | 0.66 | 0.20 | 3.00 | 1.05 | 0.64 | 0.05 | 2.14 | 0.31 | 0.36 | 0.00 | 1.25 | 0.93 | 0.38 | 0.28 | 1.54 |
| claude3_sonnet | 2.77 | 0.46 | 1.52 | 3.29 | 3.74 | 0.12 | 3.44 | 3.89 | 1.71 | 0.66 | 0.20 | 3.00 | 1.05 | 0.64 | 0.05 | 2.14 | 0.31 | 0.36 | 0.00 | 1.25 | 0.93 | 0.38 | 0.28 | 1.54 |
| deepseek | 1.88 | 0.52 | 0.62 | 2.50 | 3.69 | 0.56 | 1.60 | 3.99 | 1.28 | 0.56 | 0.00 | 2.40 | 0.31 | 0.29 | 0.00 | 1.00 | 0.09 | 0.14 | 0.00 | 0.42 | 0.75 | 0.28 | 0.25 | 1.24 |
| gpt_3.5_turbo | 2.46 | 0.85 | 0.66 | 3.28 | 3.74 | 0.24 | 3.16 | 3.99 | 2.12 | 0.76 | 0.20 | 3.00 | 0.43 | 0.50 | 0.00 | 2.10 | 0.07 | 0.18 | 0.00 | 0.74 | 0.85 | 0.36 | 0.28 | 1.70 |
| gpt_4_turbo | 2.85 | 0.35 | 1.88 | 3.41 | 3.53 | 0.18 | 3.25 | 3.89 | 1.32 | 0.50 | 0.40 | 2.40 | 0.65 | 0.48 | 0.00 | 1.62 | 0.35 | 0.32 | 0.00 | 1.08 | 1.06 | 0.28 | 0.63 | 1.87 |
| gpt_4o | 3.09 | 0.31 | 2.24 | 3.49 | 3.48 | 0.19 | 2.99 | 3.76 | 1.88 | 0.94 | 0.20 | 3.00 | 0.78 | 0.59 | 0.00 | 2.04 | 0.34 | 0.46 | 0.00 | 1.28 | 1.14 | 0.50 | 0.36 | 2.10 |
| llama3 | 2.56 | 0.63 | 1.25 | 3.30 | 3.66 | 0.18 | 3.25 | 3.95 | 1.10 | 0.66 | 0.00 | 2.40 | 0.75 | 0.49 | 0.00 | 1.70 | 0.37 | 0.35 | 0.00 | 1.12 | 1.19 | 0.36 | 0.57 | 1.80 |

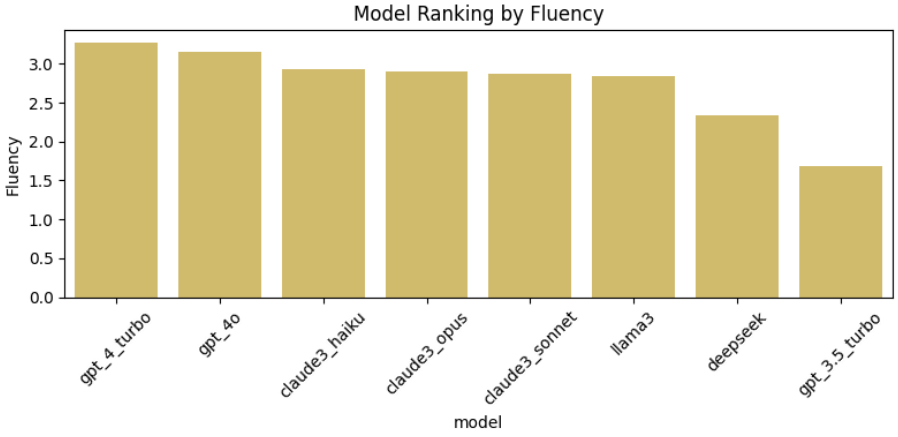
Table A1.3 - Metrics Results of SHAP Explanations by Model

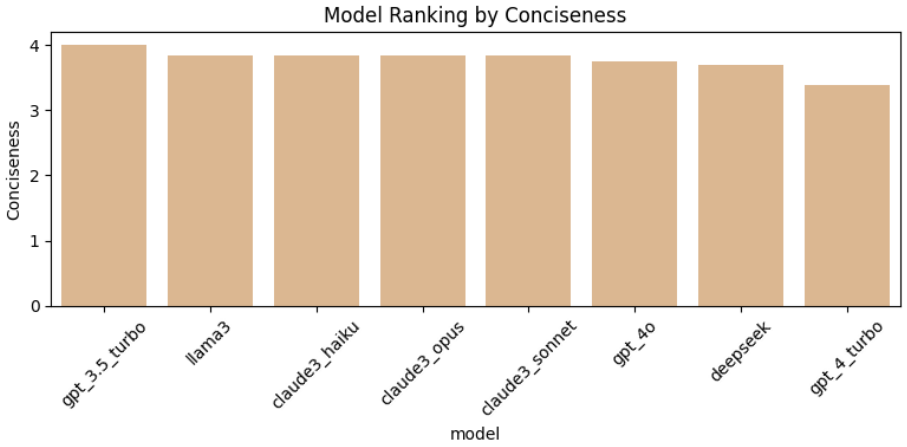
| model | 72h Triage | 72h Triage Business | Critical Triage | Critical Triage Business | Fraud Detection | Fraud Detection Business | Fraud Detection VC | Hospital Triage | Hospital Triage Business |
|---|---|---|---|---|---|---|---|---|---|
| claude3_haiku | 24.4153 | 26.0328 | 25.1010 | 24.3378 | 24.1943 | 20.7253 | 20.4304 | 23.1051 | 23.2630 |
| claude3_opus | 24.5487 | 26.2994 | 25.5677 | 24.4711 | 24.1277 | 20.3253 | 20.0971 | 23.1718 | 23.2630 |
| claude3_sonnet | 24.0820 | 26.2328 | 25.3010 | 24.2044 | 24.3277 | 20.5253 | 20.1638 | 23.1051 | 23.1963 |
| deepseek | 18.2626 | 16.3300 | 16.6812 | 19.8637 | 19.1041 | 13.1623 | 13.8549 | 17.0153 | 17.7632 |
| gpt_3.5_turbo | 19.4505 | 21.7588 | 22.2159 | 20.0356 | 22.0900 | 20.0200 | 18.2679 | 20.5380 | 19.6864 |
| gpt_4_turbo | 26.1767 | 24.9620 | 25.2480 | 23.0397 | 22.8638 | 20.3968 | 22.2612 | 23.3939 | 19.7037 |
| gpt_4o | 26.6954 | 25.8458 | 26.8880 | 25.0148 | 24.5536 | 22.5475 | 24.5872 | 24.4124 | 24.5605 |
| llama3 | 22.8905 | 23.7919 | 22.4568 | 21.8828 | 22.4767 | 19.8299 | 18.4272 | 20.8261 | 18.2982 |

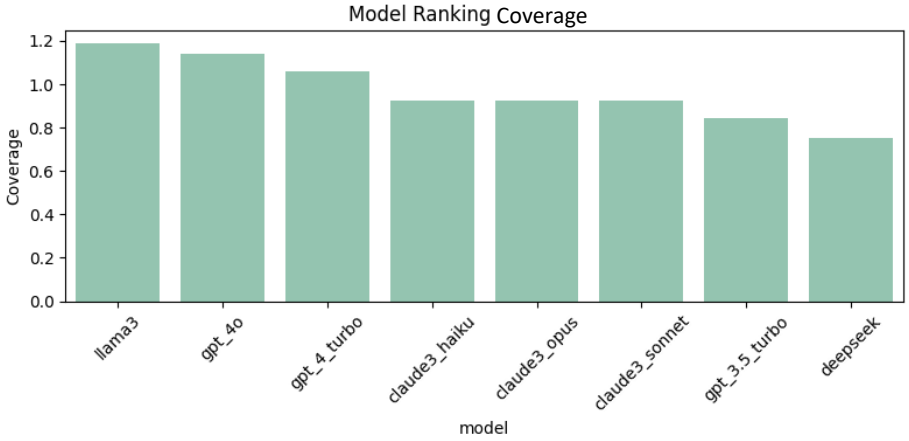
Table A1.4 - Total Scores of SHAP Explanations Across Models and Datasets

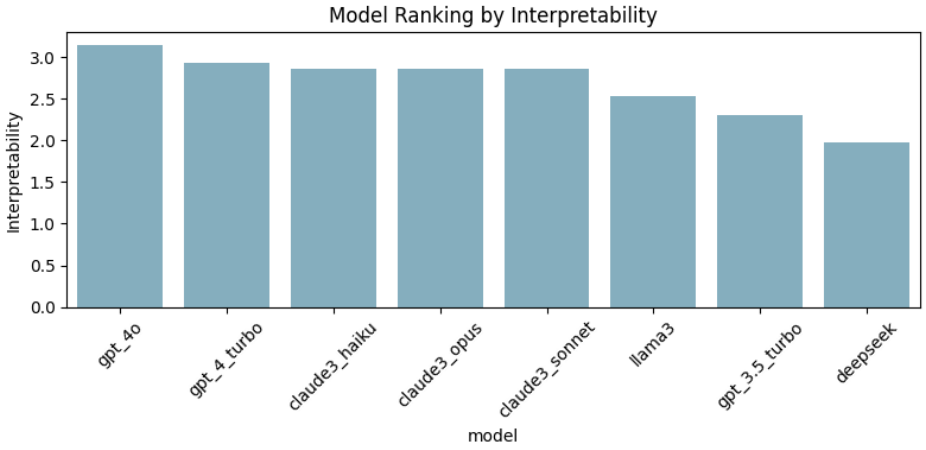

Model Ranking by Accuracy
Accuracy
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
gpt_3.5_turbo
gpt_4_turbo
gpt_4o
claude3_haiku
claude3_opus
claude3_sonnet
llama3
deepseek
model

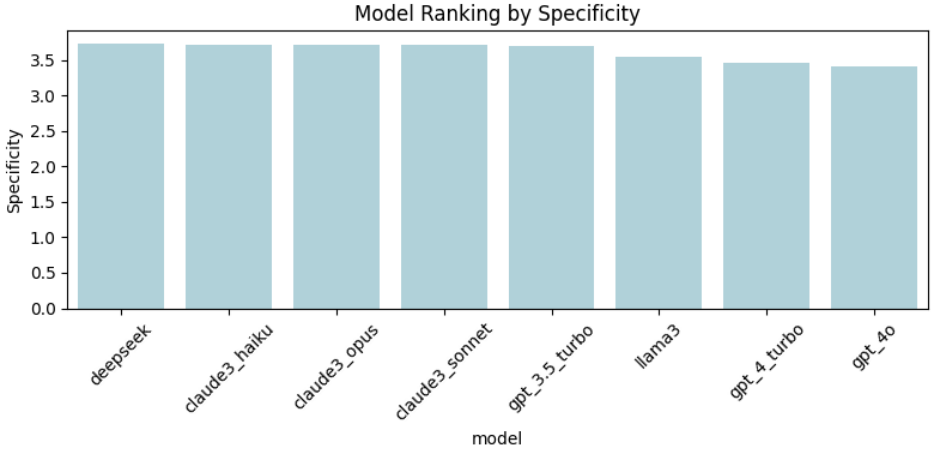

Model Ranking by Completeness
Completeness
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
gpt_4o
gpt_4_turbo
claude3_haiku
claude3_opus
claude3_sonnet
llama3
gpt_3.5_turbo
deepseek
model

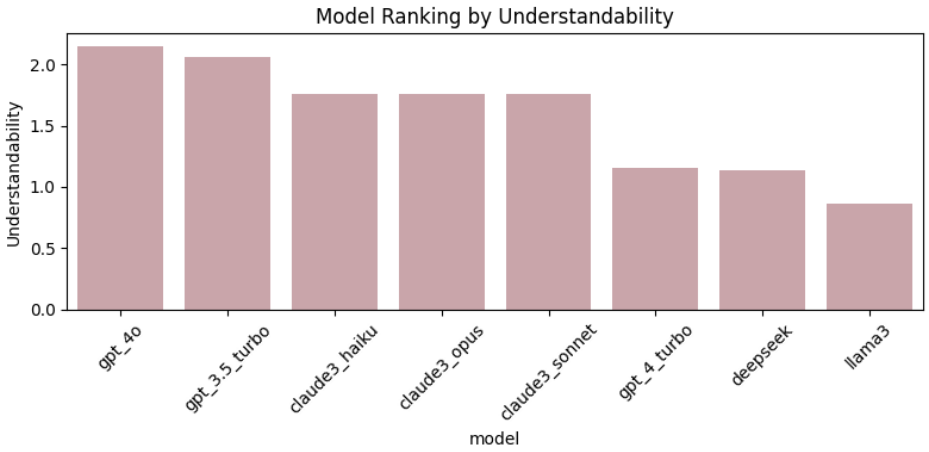

Model Ranking by Fluency
Fluency
0.0
0.5
1.0
1.5
2.0
2.5
3.0
gpt_4_turbo
gpt_4o
claude3_haiku
claude3_opus
claude3_sonnet
llama3
deepseek
gpt_3.5_turbo
model

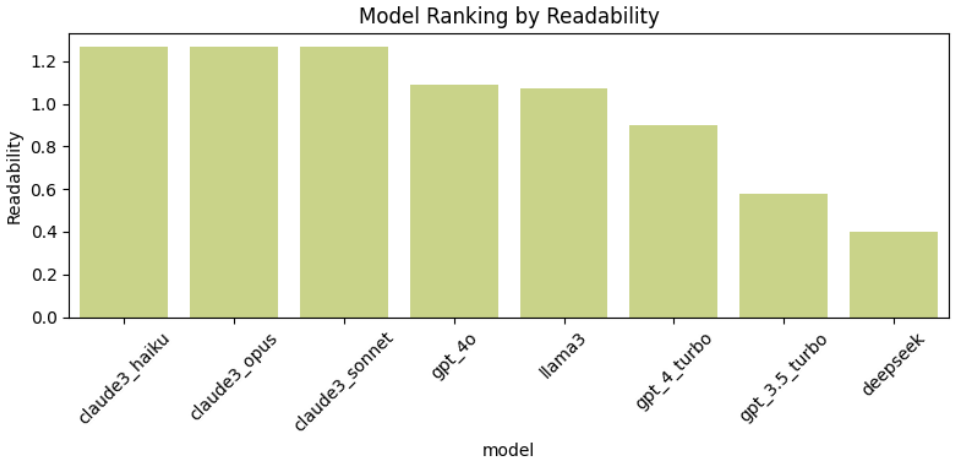

Model Ranking by Conciseness
Conciseness
0
1
2
3
4
gpt_3.5_turbo
llama3
claude3_haiku
claude3_opus
claude3_sonnet
gpt_4o
deepseek
gpt_4_turbo
model

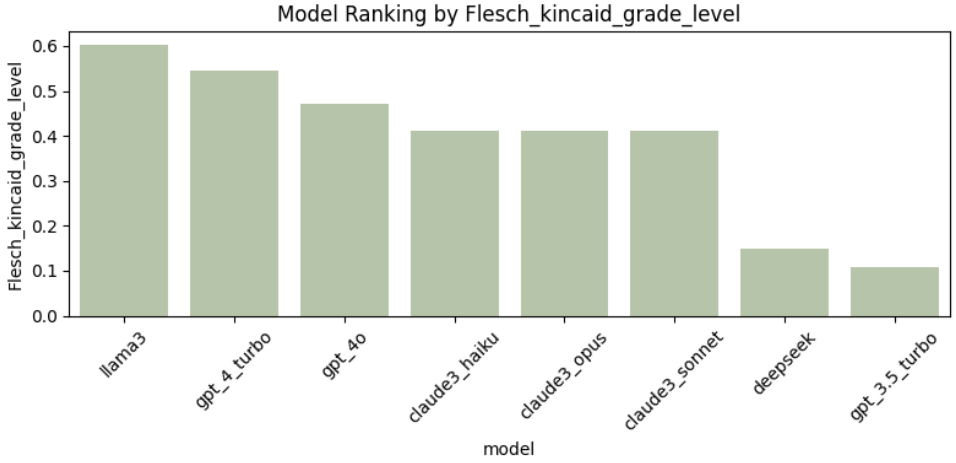

Model Ranking Coverage
Coverage
0.0
0.2
0.4
0.6
0.8
1.0
1.2
llama3
gpt_4o
gpt_4_turbo
claude3_haiku
claude3_opus
claude3_sonnet
gpt_3.5_turbo
deepseek
model

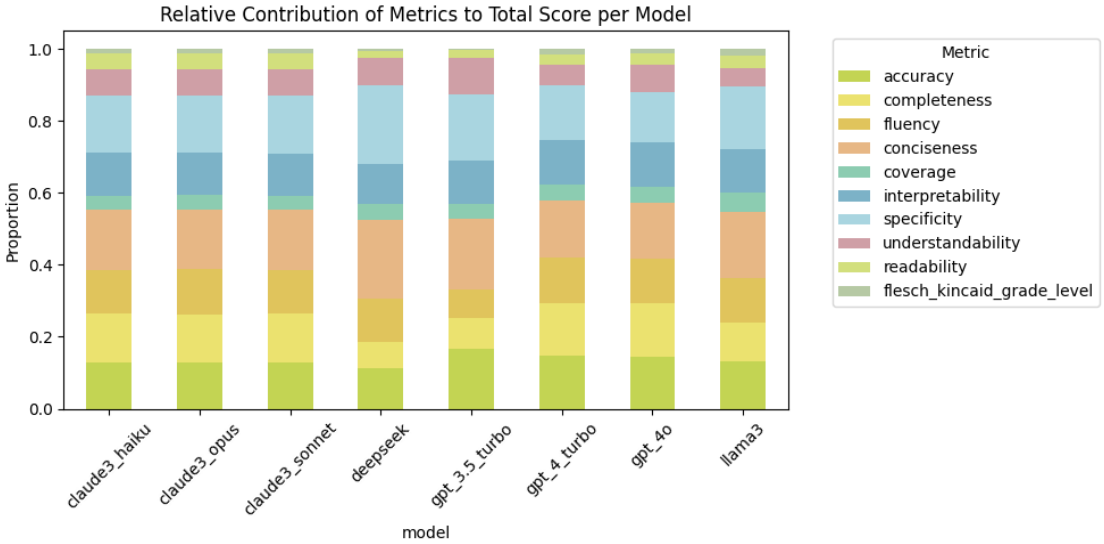

Model Ranking by Interpretability
Interpretability
3.0
2.5
2.0
1.5
1.0
0.5
0.0
gpt_4o
gpt_4_turbo
claude3_haiku
claude3_opus
claude3_sonnet
llama3
gpt_3.5_turbo
deepseek
model


Model Ranking by Specificity
Specificity
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
deepseek
claude3_haiku
claude3_opus
claude3_sonnet
gpt_3.5_turbo
llama3
gpt_4_turbo
gpt_4o
model


Model Ranking by Understandability
Understandability
2.0
1.5
1.0
0.5
0.0
gpt_4o
gpt_3.5_turbo
claude3_haiku
claude3_opus
claude3_sonnet
gpt_4_turbo
deepseek
llama3
model


Model Ranking by Readability
Readability
1.2
1.0
0.8
0.6
0.4
0.2
0.0
claude3_haiku
claude3_opus
claude3_sonnet
gpt_4o
llama3
gpt_4_turbo
gpt_3.5_turbo
deepseek
model


Model Ranking by Flesch_kincaid_grade_level
Flesch_kincaid_grade_level
0.6
0.5
0.4
0.3
0.2
0.1
0.0
llama3
gpt_4_turbo
gpt_4o
claude3_haiku
claude3_opus
claude3_sonnet
deepseek
gpt_3.5_turbo
model


Relative Contribution of Metrics to Total Score per Model
Proportion
1.0
0.8
0.6
0.4
0.2
0.0
claude3_haiku
claude3_opus
claude3_sonnet
deepseek
gpt_3.5_turbo
gpt_4_turbo
gpt_4o
llama3
model
Metric
accuracy
completeness
fluency
conciseness
coverage
interpretability
specificity
understandability
readability
flesch_kincaid_grade_level

## A2) Human Evaluation (Case-Study)

| | | RATING | | | | |
|---|---|---|---|---|---|---|
| METRIC | EXPLANATION | 0 | 1 | 2 | 3 | 4 |
| **Accuracy** | Factual correctness + consistency with underlying values. E.g. *How accurate or correct do you find this explanation?* | Contains several errors | Contains some errors | Mostly correct, with minor errors | Almost entirely correct | Entirely accurate |
| **Completeness** | Measures coverage of necessary aspects. E.g. *Does this explanation cover all the important points you would expect?* | Misses many key points | Covers a few important points | Covers some important points | Covers most important points | Fully comprehensive |
| **Fluency** | Naturalness of writing style. E.g. *How natural and well-written does this explanation sound?* | Very unnatural | Somewhat unnatural | Acceptable but not smooth | Mostly fluent | Very fluent and human-like |
| **Conciseness** | Avoids verbosity. E.g. *Is this explanation concise and to the point?* | Very wordy/repetitive | Somewhat wordy | Acceptable lenght | Concise | Very concise and efficient |
| **Coverage** | Alignment with the reference. E.g. *Compared to what you would expect in a good explanation, how complete is this one?* | Very incomplete | Incomplete | Moderately complete | Almost complete | Fully complete |
| **Interpretability** | How easy is to interpret. E.g. *How well does this explanation help you understand the reasoning behind the answer?* | Not interpretable at all | Poor interpretability | Moderately interpretable | Easy to interpret | Very interpretable and clear |
| **Specificity** | Detailed and tailored or vague. E.g. *Does this explanation feel specific to the input (not generic)?* | Very vague | Somewhat vague | Moderately understandable | Quite specific | Very specific and tailored |
| **Understandability** | Accessibility to lay users. E.g. *How easy is this explanation to understand for somone without a technical background?* | Very hard to understand | Difficult to understand | Moderately specific | Easy to understand | Very easy to understand |
| **Readability** | Readability Index. E.g. *How easy is this explanation to read and follow?* | Very hard to read | Somewaht hard to read | Moderately easy to read | Easy to read | Very easy to read |

Table A2.1 – Metrics for Human Evaluation of SHAP Explanations

| ID | EXPLANATION |
|---|---|
| 1 | "The model predicts bank account fraud by analyzing various features. A higher-than-average proposed credit limit of 2000.0 suggests a higher risk or aggressive credit seeking, contributing positively to the fraud prediction. The housing status, categorized as BA, indicates a less stable arrangement compared to the most common category, BC, which correlates with a higher fraud risk. The use of the Windows operating system is noted, as unusual or rare systems can be associated with fraud schemes, contributing positively to the prediction. The email domain is not free, which slightly reduces the fraud risk since free domains are easier to create for fraudulent purposes. Lastly, the velocity of transactions over four weeks is 4367.48, indicating high-volume activity that could be part of organized fraud patterns, but it contributes negatively to the fraud prediction, suggesting it is within a less suspicious range." |
| 2 | "The model predicts bank account fraud by analyzing various features of the customer and their account. The operating system used by the customer is Windows, which is significant because unusual or rare systems might indicate fraudulent activity. The customer's age is 60, which is considered because age can influence the likelihood of fraud. The credit risk score is 285, suggesting a moderate risk of default or fraud. The customer has lived at their current address for 67 months, indicating stability, which typically suggests a lower risk of fraud. The similarity between the customer's name and email is low, which might suggest a fake or suspicious account. The customer remained logged in during the session, which could indicate automated behavior, but in this case, it negatively impacts the fraud prediction. The housing status is categorized as "BC," which is common and suggests stability, thus reducing the fraud risk. Lastly, the velocity of transactions over four weeks is 4285.96, which is within a range that might indicate organized fraud patterns, but in this case, it slightly reduces the fraud prediction." |
| 3 | "The model predicts bank account fraud by analyzing various features. It considers the operating system used, with Windows potentially indicating unusual systems often linked to fraud. The customer's age is also a factor, as older individuals might be seen as less likely to commit fraud. A high credit risk score suggests a greater chance of fraud or default. The length of time a customer has lived at their current address is considered, with longer stays indicating stability and a lower fraud risk. The similarity between the customer's name and email is checked to detect fake accounts. Whether the user stays logged in is monitored, as it can indicate automated behavior. The housing status, anonymized as "BC," suggests a stable living situation, which typically correlates with lower fraud risk. Lastly, the model looks at the velocity of transactions over four weeks, with high volumes potentially signaling organized fraud." |
| 4 | "The model predicts bank account fraud by analyzing various features of the customer and account. The housing status, represented by the anonymized category "BA," suggests a stable housing situation, which typically correlates with a lower fraud risk. This feature has a high contribution value of 0.9357. The device operating system used is Windows, which is a common system and contributes 0.5882 to the prediction, indicating it is not unusual or rare. The customer's age is 20, which slightly decreases the fraud prediction with a contribution of -0.2903, as younger ages might be associated with higher risk. The velocity of transactions over four weeks is 4297.04, which is within the normal range and contributes -0.1640, suggesting no significant long-term high-volume activity. The zip count over four weeks is 656, indicating a moderate concentration of applications in a specific area, contributing -0.1115 to the prediction. Lastly, the use of 13 distinct emails with the same date of birth over four weeks slightly decreases the fraud prediction with a contribution of -0.1085, as it is not excessively high to suggest repeated use of fake IDs." |
| 5 | "The model predicts bank account fraud by analyzing various features of the customer's profile. The applicant's housing status, indicated as "BA," suggests a stable housing situation, which typically correlates with a lower risk of fraud. The use of the Windows operating system is noted, as unusual or rare systems might be associated with fraudulent activities. The model also considers the number of distinct emails associated with the applicant's date of birth over the past four weeks, with a count of 2, which could indicate the reuse of fake identities. The applicant's account history with the bank is relatively short, at 1 month, which might suggest less stability and trust. Additionally, the model examines the concentration of applications from the applicant's zip code, with a count of 867, to identify potential fraud hotspots. Lastly, the payment type "AB" is noted, which is a common plan and may reflect standard repayment behavior, reducing the likelihood of fraud." |

Table A2.2 - Example Explanations Generated by SHAP used in the Human Evaluation (1-5)

| ID | EXPLANATION |
|---|---|
| 6 | "The model predicts this transaction is likely fraudulent, with key factors being the customer's housing situation (category "BA", which is common but still associated with higher fraud), the use of a Windows device (potentially unusual), and the relatively young age of 20. There are also signs of organized fraud, with high transaction volumes over the last 4 weeks, many applications from the same zip code, and the same birthdate used across 13 different email addresses recently. While each factor alone may not be conclusive, together they paint a suspicious pattern that warrants flagging this transaction for further review." |
| 7 | "The model predicts this customer is at higher risk for bank fraud. Key factors include: The customer is using a Windows operating system, which is more unusual and could indicate fraud. At age 60, they are in an age range with elevated fraud risk. Their credit risk score of 285 is on the higher end, suggesting greater chance of fraud or default. The customer has lived at their current address for 67 months, a medium length of time that provides some stability but is not exceptionally long. There is little similarity between the customer's name and email address, a possible sign of a fake or suspicious account. They remained logged in during the session, which can occur with automated fraudulent activity. Their housing status is a common category, so does not strongly indicate risk either way. The account has seen high transaction volumes over the past 4 weeks, which fits patterns of organized fraud. Taken together, there are multiple signals pointing to increased fraud risk for this account." |
| 8 | "The model predicts that this customer is likely committing bank account fraud. The strongest indicator is their housing status of "BA", which suggests an unstable living situation that is often associated with higher fraud risk. They are also using a Windows operating system, which is sometimes favored by fraudsters over more common options. There have been 2 distinct email addresses associated with this date of birth in the past 4 weeks, a tactic used by fraudsters to reuse fake identities. On the positive side, they have had an account with the bank for 1 month, which is a relatively short history that doesn't strongly indicate trust and stability yet. The zip code on the application has had a high number of 867 applications in the past 4 weeks, which could be a sign of a geographic fraud hotspot. Finally, they selected payment plan "AB", the most common plan, which is fairly neutral in assessing their risk." |
| 9 | "The model predicts this customer is likely to commit bank account fraud. The strongest factor is their housing situation, which is in the "BA" category - a common, stable arrangement that suggests lower fraud risk. However, there are a few red flags. They are using an unusual "other" operating system which could indicate a fraud scheme. They also have a relatively high credit risk score of 278, meaning the bank sees them as more likely to default or commit fraud. Lastly, there have been a high number of recent applications (1052) from their zip code, which might be a fraud hotspot. So while their housing points to low risk, the rare OS, high risk score, and many local applications lead the model to predict potential fraud." |
| 10 | "The model predicts this transaction is likely fraudulent, with a SHAP value of 0.9356986880302429 for the customer's housing status being in the "BA" category. This anonymized category represents the applicant's housing situation, and "BA" being the most common could indicate more stable arrangements that correlate with lower fraud risk. The customer using a Windows operating system (OS) on their device also contributes to the fraud prediction, with a SHAP value of 0.5882472991943359. Windows is a common OS, but detecting unusual or rare systems can help identify potential fraud schemes. The customer's young age of 20 years old decreases the fraud likelihood, with a negative SHAP contribution of -0.2902843654155731. Younger applicants may have lower creditworthiness and higher fraud risk. The velocity of transactions over the past 4 weeks, at 4297.043108140676, is within the typical range of [2779, 7043] but still decreases the fraud prediction by -0.16399365663528442. The number of distinct zip codes used in the last 4 weeks, 656, is on the lower end of the [1, 5767] range and contributes -0.11154194176197052 to the prediction, suggesting there is no unusual concentration of applications in specific areas that could be fraud hotspots. Finally, the number of distinct email addresses used with this date of birth in the past 4 weeks is 13, which falls within the [0, 42] range but decreases the fraud prediction by -0.10845059156417847, indicating the birth date is not being repeatedly used across many different email addresses as a tactic to reuse fake IDs." |

Table A2.3 - Example Explanations Generated by SHAP used in the Human Evaluation (6-10)

| ID | EXPLANATION |
|---|---|
| 11 | "The ML model predicts bank account fraud based on customer and account characteristics. Higher-than-average credit requests may signal higher risk or aggressive credit seeking, which contributes positively to the prediction (1.06). Housing stability, as represented by the category "BA", also correlates with lower fraud risk, contributing 0.9 to the prediction. The use of a Windows device can help detect unusual or rare systems that may be used in potential fraud schemes, adding 0.66 to the prediction. On the other hand, using a free email domain (e.g., Gmail) is more common but also easier to create for fraudulent purposes, contributing negatively (-0.35). Finally, long-term high-volume activity, as identified by velocity_4w, can help detect organized fraud patterns, but this feature contributes slightly negatively (-0.15) to the prediction." |
| 12 | "The ML model predicts bank account fraud based on customer and account characteristics. For this prediction, the housing situation being "BA" (owned or rented) contributes positively to the likelihood of fraud, with a significant contribution score of 0.9646. This suggests that individuals with stable housing arrangements are less likely to engage in fraudulent activities. On the other hand, using an unusual device operating system like "other" negatively impacts the prediction, with a contribution score of -0.4876. This could indicate that fraudsters often use rare or uncommon systems to carry out their schemes. The credit risk score of 278 also contributes positively to the prediction, suggesting that individuals with higher credit risk scores may be more likely to engage in fraudulent activities. Finally, an unusual concentration of applications in a specific area (zip count) negatively impacts the prediction, indicating that fraud hotspots are less common in this case." |
| 13 | "The model predicts whether a bank account transaction might be fraudulent based on various characteristics of the customer and their device. Features like `device_os` being Windows or Macintox suggest potential involvement in fraud schemes that utilize these systems more unusually. A higher `credit_risk_score`, such as 285, indicates increased risk for both defaulting on credit and engaging in fraud. The applicant's age of 60 is considered, with older ages potentially influencing fraud likelihood due to various factors like less tech-savviness or different spending patterns. Their long-term stability at the current address (67 months) generally points towards a lower fraud risk. A low `name_email_similarity` score (0.108) raises suspicion about fake accounts, as it suggests little match between name and email. An active `keep_alive_session` indicates logged-in behavior that might be automated or unusual for fraud detection. The housing status category "BC" is noted; while stability often reduces risk, its commonality doesn't necessarily indicate low fraud potential here. Finally, the high long-term transaction velocity (`velocity_4w`) of 4285 suggests significant ongoing activity which can sometimes correlate with organized fraud patterns." |
| 14 | "Based on customer characteristics, this model predicts whether a bank account application is fraudulent. For instance, an applicant's housing status being 'BA' (an anonymized category likely representing stable arrangements) contributed positively to the prediction, suggesting lower fraud risk associated with that specific housing situation. The operating system of their device, Windows, also showed positive contribution as it represents a common system type. However, other features indicated higher potential for fraud: the applicant's age being 20 had a negative contribution; high long-term transaction velocity (4297.0431) and unusual application concentration in one zip code area (656) both showed negative contributions; finally, repeated use of birth dates across different emails (13 distinct values) also contributed negatively to the prediction, implying potential fraud through fake ID reuse." |
| 15 | "The model predicts bank account fraud with a high level of confidence due to several factors. The customer's housing status significantly increases the likelihood of fraud, contributing strongly to the prediction. Additionally, the use of a Windows operating system on the customer's device also raises the fraud risk. The presence of two distinct email addresses associated with the date of birth within four weeks further supports the fraud prediction. However, the model considers the customer's short banking history of one month as a factor that slightly decreases the fraud likelihood. Similarly, the high number of zip codes used in the past four weeks and the payment type contribute to a lesser extent in reducing the fraud prediction. Overall, the positive contributions outweigh the negative ones, leading to a strong prediction of fraud." |

Table A2.4 - Example Explanations Generated by SHAP used in the Human Evaluation (11-15)

| ID | EXPLANATION |
|---|---|
| 16 | "The model predicts that there is a high likelihood of bank account fraud based on several factors. The customer's housing status is a significant indicator, contributing strongly to the prediction. The use of a Windows operating system also adds to the likelihood of fraud, though to a lesser extent. The fact that there are two distinct email addresses associated with the date of birth in the last four weeks is another factor that increases the risk. However, having only one month of banking history slightly reduces the fraud prediction. Additionally, the number of zip codes associated with the account in the last four weeks and the type of payment method used (AB) both slightly decrease the likelihood of fraud. Overall, the combination of these factors leads the model to predict a high risk of fraud." |
| 17 | "The model predicts this customer is likely committing bank account fraud. The strongest factors increasing fraud risk are that their housing status is BA (0.94 contribution) and they use a Windows device (0.59 contribution). Factors decreasing fraud risk include the customer's young age of 20 (-0.29), high 4-week velocity of $4,297 (-0.16), having transactions from 656 zip codes in the last 4 weeks (-0.11), and having 13 distinct email addresses associated with their date of birth in the last 4 weeks (-0.11)." |
| 18 | "The model predicts that this customer is likely to commit bank account fraud. The most important factor leading to this prediction is that the customer has a proposed credit limit of $2000, which increases the fraud risk by 1.06. The customer's housing status of 'BA' and the fact that they use a Windows operating system on their device also contribute to the fraud prediction, increasing the risk by 0.90 and 0.66 respectively. On the other hand, the customer having a non-free email address slightly reduces the predicted fraud risk by 0.35. Finally, the customer's velocity over the past 4 weeks of $4367.48 has a small negative impact of 0.15 on the fraud prediction." |
| 19 | "The ML model's prediction of bank account fraud is influenced by several factors. A proposed credit limit of $2000 has a significant positive impact on the likelihood of fraud, suggesting that higher credit limits may be associated with increased risk. Additionally, having a housing status of "BA" also contributes positively to the prediction, implying that individuals with higher education levels may be more likely to engage in fraudulent activities. The use of Windows as the device operating system has a moderate positive effect on the prediction, possibly indicating that users of this OS are more prone to fraud. On the other hand, having an email account that is free (email_is_free = 0) has a negative impact on the likelihood of fraud, suggesting that individuals with paid email accounts may be less likely to engage in fraudulent behavior. Finally, velocity_4w, which measures some type of financial activity over a four-week period, has a small negative effect on the prediction, possibly indicating that stable financial transactions are less likely to be fraudulent." |
| 20 | "The machine learning model predicts that this account might be involved in fraudulent activity, likely due to several factors related to the customer's device and online behavior. Specifically, it suggests that using a Windows operating system on the device (with a value of 1), combined with other characteristics like being over 60 years old (value: 60) or having a low credit risk score (285), could be contributing positively to this prediction. However, there are also factors pointing away from fraud; for instance, the customer has lived at their current address for many months (67.0), and while their name doesn't strongly match any email used in transactions (similarity: 0.109), these seem less significant than other positive indicators according to the model's analysis. The presence of a keep-alive session (value: 1) is noted as having a negative contribution, meaning it might slightly reduce the likelihood of fraud compared to some other features, but overall the contributions lean towards identifying potential risk." |

Table A2.5 - Example Explanations Generated by SHAP used in the Human Evaluation (16-20)

## A3) Counterfactuals

| | accuracy | completeness | fluency | conciseness | coverage | interpretability | specificity | understandability | readability | flesch_kincaid_grade_level | total score | total_score_no_conciseness |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| count | 153.0000 | 153.0000 | 153.0000 | 153.0000 | 153.0000 | 153.0000 | 153.0000 | 153.0000 | 153.0000 | 153.0000 | 153.0000 | 153.0000 |
| mean | 1.8876 | 0.5255 | 1.3085 | 3.8268 | 0.2809 | 1.8835 | 3.7071 | 1.4562 | 0.8131 | 0.2207 | 15.9099 | 12.0831 |
| std | 1.0687 | 0.6266 | 0.8721 | 0.5901 | 0.2793 | 0.6263 | 0.5936 | 0.8011 | 0.6778 | 0.2877 | 3.3758 | 2.9838 |
| min | 0.0000 | 0.0000 | 0.0000 | 0.8000 | 0.0000 | 0.4307 | 0.7701 | 0.0000 | 0.0000 | 0.0000 | 2.2387 | 1.4387 |
| 25% | 0.8000 | 0.0000 | 0.6000 | 4.0000 | 0.0625 | 1.4655 | 3.7929 | 1.0000 | 0.2307 | 0.0000 | 14.4326 | 10.5823 |
| 50% | 1.6000 | 0.4000 | 1.2000 | 4.0000 | 0.1970 | 1.8816 | 3.9139 | 1.4000 | 0.7155 | 0.0925 | 16.2831 | 12.3149 |
| 75% | 2.4000 | 0.8000 | 2.0000 | 4.0000 | 0.4549 | 2.2851 | 3.9560 | 2.0000 | 1.1887 | 0.3930 | 17.8942 | 13.8942 |
| max | 4.0000 | 2.4000 | 3.0000 | 4.0000 | 1.1857 | 3.3801 | 4.0000 | 3.4000 | 3.6411 | 1.3333 | 23.8732 | 19.8732 |

Table A3.1 – Results of Counterfactuals Explanation Metrics

| | accuracy | | | | completeness | | | | fluency | | | | conciseness | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| dataset | mean | std | min | max | mean | std | min | max | mean | std | min | max | mean | std | min | max |
| Fraud CF All | 2.0414 | 1.2016 | 0.8000 | 4.0000 | 0.5517 | 0.6451 | 0.0000 | 2.4000 | 1.6414 | 0.9109 | 0.2000 | 3.0000 | 3.8269 | 0.6102 | 0.8000 | 4.0000 |
| Fraud CF FN | 2.0138 | 0.8717 | 0.0000 | 4.0000 | 0.6345 | 0.7212 | 0.0000 | 2.0000 | 0.8966 | 0.6647 | 0.0000 | 2.4000 | 3.8621 | 0.5267 | 1.6000 | 4.0000 |
| Fraud CF FP | 1.6828 | 1.0764 | 0.0000 | 4.0000 | 0.4966 | 0.6560 | 0.0000 | 2.4000 | 1.3862 | 0.7927 | 0.2000 | 3.0000 | 3.8526 | 0.4801 | 1.6000 | 4.0000 |
| Fraud CF TN | 2.0121 | 1.1335 | 0.0000 | 4.0000 | 0.4242 | 0.5651 | 0.0000 | 2.0000 | 1.3030 | 0.9316 | 0.0000 | 3.0000 | 3.8033 | 0.6930 | 0.8000 | 4.0000 |
| Fraud CF TP | 1.6970 | 1.0346 | 0.0000 | 4.0000 | 0.5333 | 0.5715 | 0.0000 | 2.0000 | 1.3152 | 0.9070 | 0.0000 | 3.0000 | 3.7964 | 0.6328 | 0.8000 | 4.0000 |

| | interpretability | | | | specificity | | | | understandability | | | | readability | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| dataset | mean | std | min | max | mean | std | min | max | mean | std | min | max | mean | std | min | max |
| Fraud CF All | 1.9920 | 0.4918 | 0.5907 | 2.9097 | 3.7071 | 0.6113 | 0.7701 | 3.9744 | 1.3172 | 0.7320 | 0.0000 | 2.8000 | 0.9551 | 0.6537 | 0.0000 | 2.4164 |
| Fraud CF FN | 2.1620 | 0.5788 | 1.0365 | 3.0680 | 3.7012 | 0.5310 | 1.5700 | 3.9560 | 1.5862 | 0.8348 | 0.0000 | 3.0000 | 0.8349 | 0.7017 | 0.0000 | 2.3649 |
| Fraud CF FP | 1.6553 | 0.6453 | 0.6995 | 2.9986 | 3.7558 | 0.5005 | 1.5283 | 4.0000 | 1.3586 | 0.8287 | 0.0000 | 3.4000 | 0.6829 | 0.5425 | 0.0000 | 2.5863 |
| Fraud CF TN | 1.7721 | 0.7200 | 0.5817 | 3.3801 | 3.7069 | 0.6884 | 0.7886 | 4.0000 | 1.5818 | 0.8267 | 0.0000 | 2.8000 | 0.5087 | 0.4636 | 0.0000 | 2.3061 |
| Fraud CF TP | 1.8555 | 0.5778 | 0.4307 | 2.9273 | 3.6698 | 0.6351 | 0.8000 | 4.0000 | 1.4242 | 0.7918 | 0.0000 | 2.6000 | 1.0878 | 0.8353 | 0.0000 | 3.6411 |

| | flesch_kincaid_grade_level | | | | coverage | | | | total score | | | | total_score_no_conciseness | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| dataset | mean | std | min | max | mean | std | min | max | mean | std | min | max | mean | std | min | max |
| Fraud CF All | 0.2981 | 0.2930 | 0.0000 | 0.8137 | 0.3604 | 0.3036 | 0.0061 | 1.0221 | 16.6913 | 3.3692 | 3.9371 | 21.8664 | 12.8644 | 2.9639 | 3.1371 | 18.0536 |
| Fraud CF FN | 0.1568 | 0.2528 | 0.0000 | 0.9575 | 0.2540 | 0.2855 | 0.0000 | 1.1857 | 16.1021 | 2.6983 | 6.1395 | 20.9748 | 12.2400 | 2.2891 | 4.5395 | 16.9748 |
| Fraud CF FP | 0.1003 | 0.2049 | 0.0000 | 0.9468 | 0.2714 | 0.3207 | 0.0000 | 1.1741 | 15.2425 | 3.3295 | 6.1240 | 19.7156 | 11.3899 | 3.0915 | 4.5240 | 15.9067 |
| Fraud CF TN | 0.1042 | 0.1680 | 0.0000 | 0.8736 | 0.2315 | 0.1922 | 0.0275 | 0.6841 | 15.4479 | 3.7327 | 3.2472 | 23.8732 | 11.6446 | 3.2762 | 2.4472 | 19.8732 |
| Fraud CF TP | 0.4313 | 0.3395 | 0.0000 | 1.3333 | 0.2922 | 0.2873 | 0.0000 | 1.1431 | 16.1028 | 3.5980 | 2.2387 | 22.6280 | 12.3063 | 3.1081 | 1.4387 | 18.6280 |

Table A3.2 – Metrics Results for Counterfactuals Explanations by Dataset

| | total score | | | | accuracy | | | | completeness | | | | fluency | | | | conciseness | | | | total_score_no_conciseness | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| model | mean | std | min | max | mean | std | min | max | mean | std | min | max | mean | std | min | max | mean | std | min | max | mean | std | min | max |
| claude3_haiku | 15.78 | 2.47 | 10.67 | 20.09 | 1.76 | 0.88 | 0.00 | 3.20 | 0.44 | 0.56 | 0.00 | 1.60 | 0.87 | 0.83 | 0.00 | 2.60 | 4.00 | 0.01 | 3.95 | 4.00 | 11.79 | 2.48 | 6.67 | 16.15 |
| claude3_opus | 15.89 | 2.44 | 10.07 | 20.09 | 1.74 | 0.87 | 0.00 | 3.20 | 0.43 | 0.53 | 0.00 | 1.60 | 0.96 | 0.84 | 0.00 | 2.60 | 4.00 | 0.01 | 3.95 | 4.00 | 11.89 | 2.44 | 6.07 | 16.15 |
| claude3_sonnet | 15.93 | 2.47 | 10.67 | 20.49 | 1.76 | 0.88 | 0.00 | 3.20 | 0.44 | 0.56 | 0.00 | 1.60 | 1.02 | 0.77 | 0.00 | 3.00 | 4.00 | 0.01 | 3.95 | 4.00 | 11.94 | 2.47 | 6.67 | 16.55 |
| deepseek | 15.92 | 1.14 | 13.43 | 17.27 | 1.55 | 1.02 | 0.00 | 3.20 | 0.88 | 0.74 | 0.00 | 2.00 | 2.29 | 0.45 | 1.40 | 2.80 | 3.92 | 0.22 | 3.20 | 4.00 | 12.00 | 1.05 | 9.98 | 13.42 |
| gpt_3.5_turbo | 17.99 | 1.24 | 15.99 | 19.48 | 2.99 | 0.83 | 1.60 | 4.00 | 0.88 | 0.48 | 0.00 | 1.60 | 2.01 | 0.62 | 1.00 | 3.00 | 4.00 | 0.00 | 4.00 | 4.00 | 13.99 | 1.24 | 11.99 | 15.48 |
| gpt_4_turbo | 13.77 | 4.39 | 2.24 | 17.92 | 2.28 | 1.14 | 0.00 | 4.00 | 0.16 | 0.30 | 0.00 | 0.80 | 0.75 | 0.40 | 0.00 | 1.40 | 3.48 | 1.08 | 0.80 | 4.00 | 10.30 | 3.39 | 1.44 | 13.92 |
| gpt_4o | 17.92 | 2.86 | 13.26 | 23.87 | 2.00 | 1.05 | 0.00 | 4.00 | 0.76 | 0.86 | 0.00 | 2.40 | 1.48 | 0.76 | 0.40 | 3.00 | 3.95 | 0.14 | 3.39 | 4.00 | 13.97 | 2.86 | 9.26 | 19.87 |
| llama3 | 14.17 | 5.95 | 3.25 | 22.63 | 1.07 | 1.16 | 0.00 | 4.00 | 0.40 | 0.59 | 0.00 | 1.60 | 1.76 | 0.81 | 0.20 | 3.00 | 3.09 | 1.08 | 0.80 | 4.00 | 11.08 | 4.93 | 2.45 | 18.63 |

| | interpretability | | | | specificity | | | | understandability | | | | readability | | | | flesch_kincaid_grade_level | | | | coverage | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| model | mean | std | min | max | mean | std | min | max | mean | std | min | max | mean | std | min | max | mean | std | min | max | mean | std | min | max |
| claude3_haiku | 1.55 | 0.41 | 0.70 | 2.47 | 3.92 | 0.07 | 3.78 | 4.00 | 1.81 | 0.53 | 0.40 | 2.60 | 1.02 | 0.50 | 0.17 | 2.15 | 0.27 | 0.28 | 0.00 | 0.78 | 0.15 | 0.15 | 0.00 | 0.51 |
| claude3_opus | 1.53 | 0.39 | 0.70 | 2.47 | 3.92 | 0.07 | 3.78 | 4.00 | 1.84 | 0.51 | 0.40 | 2.60 | 1.02 | 0.54 | 0.17 | 2.15 | 0.30 | 0.29 | 0.00 | 0.78 | 0.15 | 0.15 | 0.00 | 0.51 |
| claude3_sonnet | 1.55 | 0.41 | 0.70 | 2.47 | 3.92 | 0.07 | 3.78 | 4.00 | 1.81 | 0.53 | 0.40 | 2.60 | 1.02 | 0.50 | 0.17 | 2.15 | 0.27 | 0.28 | 0.00 | 0.78 | 0.15 | 0.15 | 0.00 | 0.51 |
| deepseek | 2.05 | 0.39 | 1.52 | 2.83 | 3.79 | 0.24 | 3.04 | 3.96 | 0.56 | 0.46 | 0.00 | 1.60 | 0.16 | 0.21 | 0.00 | 0.65 | 0.04 | 0.09 | 0.00 | 0.28 | 0.67 | 0.30 | 0.37 | 1.19 |
| gpt_3.5_turbo | 2.03 | 0.29 | 1.35 | 2.45 | 3.92 | 0.04 | 3.85 | 3.99 | 1.23 | 0.25 | 0.80 | 1.60 | 0.47 | 0.36 | 0.09 | 1.37 | 0.09 | 0.14 | 0.00 | 0.43 | 0.38 | 0.11 | 0.26 | 0.61 |
| gpt_4_turbo | 2.30 | 0.77 | 0.43 | 3.05 | 3.42 | 1.06 | 0.77 | 3.98 | 1.14 | 0.44 | 0.20 | 2.20 | 0.13 | 0.19 | 0.00 | 0.65 | 0.02 | 0.06 | 0.00 | 0.27 | 0.10 | 0.09 | 0.00 | 0.30 |
| gpt_4o | 2.27 | 0.71 | 0.62 | 3.38 | 3.75 | 0.24 | 3.19 | 4.00 | 2.34 | 0.61 | 0.80 | 3.40 | 0.88 | 0.57 | 0.00 | 1.86 | 0.20 | 0.26 | 0.00 | 0.87 | 0.30 | 0.29 | 0.00 | 0.91 |
| llama3 | 2.05 | 0.83 | 0.58 | 3.01 | 2.78 | 0.91 | 0.79 | 3.72 | 0.17 | 0.28 | 0.00 | 0.80 | 1.69 | 0.89 | 0.55 | 3.64 | 0.56 | 0.38 | 0.08 | 1.33 | 0.59 | 0.33 | 0.12 | 1.16 |

Table A3.3 - Metrics Results of Counterfactuals Explanations by Model

| model | Fraud CF All | Fraud CF FN | Fraud CF FP | Fraud CF TN | Fraud CF TP |
|---|---|---|---|---|---|
| claude3_haiku | 16.4305 | 16.1237 | 14.6916 | 15.0805 | 16.5982 |
| claude3_opus | 16.5305 | 16.2237 | 14.6916 | 15.4305 | 16.4482 |
| claude3_sonnet | 16.6805 | 16.3737 | 14.5916 | 15.3805 | 16.6482 |
| deepseek | 16.2824 | 15.6831 | 16.1515 | 15.8176 | 15.6415 |
| gpt_3.5_turbo | 18.0075 | 17.2291 | 18.5796 | 18.2632 | 17.8932 |
| gpt_4_turbo | 13.6888 | 13.4028 | 14.3112 | 15.1556 | 12.3140 |
| gpt_4o | 18.9532 | 17.3220 | 16.9902 | 18.4428 | 17.8720 |
| llama3 | 17.3479 | 16.8135 | 12.2450 | 9.2853 | 15.1573 |

Table A3.4 - Total Scores of SHAP Explanations Across Models and Datasets

## A4) Ethics Committee Report – DSCI2025-9-47766


This is to certify that
Project No.: **DSCI2025-9-47766**
Project Title: **Explainable AI: Enhancing Machine Learning Model Interpretability with Generative AI**
Principal Researcher: **Inês Castelhano**

according to the regulations of the Ethics Committee of NOVA IMS and MagIC Research Center this project was considered to meet the requirements of the NOVA IMS Internal Review Board, being considered **APPROVED** on 9/4/2025.

It is the Principal Researcher's responsibility to ensure that all researchers and stakeholders associated with this project are aware of the conditions of approval and which documents have been approved.

The Principal Researcher is required to notify the Ethics Committee, via amendment or progress report, of
- Any significant change to the project and the reason for that change;
- Any unforeseen events or unexpected developments that merit notification;
- The inability of the Principal Researcher to continue in that role or any other change in research personnel involved in the project.

Lisbon, 9/4/2025

NOVA IMS Ethics Committee
ethicscommittee@novaims.unl.pt


**NOVA Information Management School**
**Instituto Superior de Estatística e Gestão de Informação**

Universidade Nova de Lisboa