STATE UNIVERSITY OF CAMPINAS – UNICAMP

Scientific Initiation Internship Report (MC040)

Application of Large Language Models for Explanations in Credit Risk Decisions

Author:
Larissa Ayumi Okabayashi

Affiliates:
Marcos Medeiros Raimundo

Thalita Biazzuz Veronese

1 Introduction

Nowadays, ML algorithms are present in our daily lives, from movie and music recommendation systems to high-risk areas such as healthcare, criminal justice, finance, and so on, assisting in decision-making.

One example of how this tool is used in the finance sector is through credit scoring models, which help lenders evaluate whether to approve a loan application based on the model's prediction of the applicant's ability to meet their financial obligations [1]. Such models are beneficial as they reduce the time required for loan approval processes, allow loan officers to focus only on a percentage of the applications, lead to cost savings, decrease human subjectivity, and reduce the risk of default [2]. There is an extensive body of literature on this topic, with various Machine Learning (ML) and Artificial Intelligence (AI) techniques proposed. These techniques can be outstanding in predictive power but are also known as "black-box"methods [9] because they do not provide explanations for their decisions, making it impossible for humans to interpret them. Therefore, it is highly unlikely that any financial expert would be willing to trust the predictions of a model without some form of justification [2].

Explainability plays a crucial role in understanding the reasons behind the decisions made by a model. It is the measure by which a human can understand the logic behind a given choice [9]. Regarding credit analysis, explainability becomes essential for debugging and auditing "black-box"models, enabling the understanding of the reasoning behind the predictions. Additionally, existing regulations, such as the "right to explanation"from GDPR and the ECOA [8], state that companies that deny credit to consumers must provide an accurate explanation, even if it was generated by an opaque algorithm.

However, explainability still presents a significant barrier to people not versed in Machine Learning, which is why it is necessary to use Natural Language Processing and Large Language Models (LLMs) to transform these explanations into intelligible text. In this way, individuals interested in understanding the reasoning behind a credit scoring model, but who lack the knowledge required to comprehend Explainable AI (XAI), such as a customer whose credit was denied by the bank or an external stakeholder, can access the information obtained from the explainability algorithms' results. Thus, the motivation behind this project is the expressed need for the development of natural language explanations to clarify decision-making in credit risk.

2 Objectives

Develop accessible natural language explanations for the field of credit scoring, especially aimed at individuals with accessibility challenges or limited familiarity with Machine Learning concepts. The goal is to make credit assessment information understandable and usable by a broader audience, promoting inclusion and facilitating the understanding of financial decisions.

3 Materials and Methods

3.1 Dataset

In this report, the application of data analysis and machine learning in credit risk assessment will be explored, followed by the application of interpretability methods to this model, using the German Credit dataset from UCI. This dataset, prepared by Prof. Hofmann, consists of 1,000 entries containing 20 categorical/symbolic attributes. These attributes include personal and financial information, credit history, and loan characteristics of each individual applying for credit from a banking institution. Each person is classified as either a good or bad credit risk based on their set of attributes.

3.2 Machine Learning Models for Credit Scoring: Definition and Evaluation

First, it is important to define what a machine learning (ML) model is. According to the popular definition by Arthur Samuel [14]:

> "The field of study that gives computers the ability to learn without being explicitly programmed."

According to the book Trustworthy Machine Learning [12]:

> "Machine learning is the study of algorithms that use data and information from observations and interactions as input and generalize from specific inputs to exhibit characteristics of human thought."

This report aims to interpret credit scoring models, which help determine whether a customer will be considered by the bank as a good or bad payer. To achieve this goal, the German Dataset was used, which is relatively small and does not have significant missing data. The data preprocessing included encoding categorical variables, such as Gender and Loan Purpose, using appropriate encoding techniques.

After preprocessing, some evaluation metrics were defined to measure the performance of ML models: AUC, Balanced Accuracy, Accuracy, Precision, Recall, F1, Specificity, and Sensitivity. These metrics are crucial to understand the overall performance of the models and ensure that the model does not introduce bias or unfairness in its predictions.

Particularly, the analysis of precision and recall for different subgroups (e.g., gender, race) is essential to identify if the model is favoring or disadvantaging any specific group. Evaluating Specificity and Sensitivity metrics by subgroup helps ensure that the model is not unfairly penalizing a specific group, promoting fairness in predictions.

The data was split into training and test sets, with 60

The choice to focus on XGBoost is justified by its balance across various performance metrics, making it a robust choice for credit scoring applications, where the ability to discriminate between good and bad payers is crucial. Additionally, interpretability techniques such as SHAP (SHapley Additive exPlanations) and PDP (Partial Dependence Plot) will be applied to provide explanations for the model's decisions, promoting transparency and trust in machine learning-based decision-making.

3.3 Interpretability Methods

3.3.1 Partial Dependence Plot

Partial dependence plots (PDP) are a visualization tool used in the field of machine learning interpretability. They are designed to help understand how a specific variable affects the model's prediction while holding other variables constant [9]. Intuitively, we can interpret partial dependence as the expected target response as a function of the input features of interest.

In this study, PDP results were generated using the XGBoost algorithm, trained with data from the German dataset. These results were used to generate accessible insights, allowing for natural language explanations to interpret the partial dependence plots. The goal is to enable users to understand whether changes in certain features of a loan application will affect credit approval, as well as to understand the relationship between each variable and the target variable, which is credit approval or denial.

Through these natural language explanations, the aim is to promote a deeper and more transparent understanding of the credit risk decision-making processes, contributing to the development of more ethical and interpretable credit evaluation methods. Furthermore, this approach seeks to enhance end-user trust in machine learning systems applied to credit evaluation by providing clear and accessible insights into how decisions are influenced by different loan application characteristics.

3.3.2 SHAP (SHapley Additive exPlanations)

SHAP values (SHapley Additive exPlanations) represent a powerful approach for explaining the outputs of any machine learning model. This methodology is based on game theory concepts to quantify each "player's"(or feature's) contribution to the model's final outcome. In the context of machine learning, each feature is assigned an importance value (SHAP value) that reflects its specific contribution to the model's output [9]. Features with positive SHAP values indicate a positive influence on the prediction, while negative values indicate a negative influence. The magnitude of these values expresses the intensity of each feature's effect on the prediction.

In this project, SHAP values were used to infer the influence of specific features on the prediction of whether a customer will be classified as a good or bad payer. In

other words, when analyzing a scenario where a customer's credit was denied, SHAP values allow us to identify which features most contributed to this decision. This analysis provides a detailed understanding of the relationships between customer attributes and the credit decision, offering critical insights for interpreting and explaining the model's decision-making processes.

By using SHAP values, we can not only identify the main determinants behind a credit decision but also quantify their relative importance. This information is fundamental for promoting transparency and interpretability in credit evaluation systems, enabling a more informed and ethical analysis of credit granting decisions.

3.4 Large Language Models

A Large Language Model (LLM) represents an advanced category of language models trained with deep learning techniques on vast textual datasets.

These models are designed to learn patterns and relationships between entities within a given language. Capable of generating human-like text, LLMs are versatile enough to perform a variety of Natural Language Processing (NLP) tasks. When presented with new text input, an LLM strives to predict or generate the most likely continuation based on the knowledge acquired during training [17].

Most LLMs are built using deep learning techniques, often through the Transformer architecture. This neural network architecture is specialized in capturing context and meaning, allowing it to learn relationships in sequential data, such as words in a sentence [17].

3.4.1 Falcon

The Falcon LLM, developed by the Technology Innovation Institute based in Abu Dhabi, represents a significant breakthrough in the field of AI language processing. This model, part of the Falcon family, includes three main variations: Falcon-180B, Falcon-40B, and Falcon-7B [10].

Notably, the Falcon-40B is the first truly open model to rival many existing proprietary models. Licensed under Apache 2.0, it is available for free for both commercial and research use [10].

All models were trained using the RefinedWeb dataset, which underwent rigorous filtering and deduplication processes to ensure the quality of the training data. Additionally, the models are easily accessible and adjustable through the Hugging Face ecosystem. Tools like Text Generation Inference and PEFT (Progressive Early Fertilization Training) are available to run and refine the models as needed.[10]

3.4.2 LLaMA 3

LLaMA 3, part of the language model family developed by Meta Inc., was designed to foster an open and responsible AI ecosystem. This innovation provides developers with a comprehensive range of advanced features, innovative design, user-friendly interface, as well as high speed and reliability. Trained on a large scale, LLaMA 3 has access to over 15 trillion publicly available data tokens, covering various domains such as code, historical knowledge, and multiple languages. This extensive and diverse training data, combined with Meta's advancements in pre-training and fine-tuning, resulted in a model that exhibits state-of-the-art performance across a wide variety of industry benchmarks and real-world scenarios. [11]

3.5 Prompt Engineering

Prompt engineering is the process of designing and optimizing the input text for Large Language Models (LLMs) [15]. Its goal is to maximize the model's potential, guiding it to generate precise and relevant responses. In practical applications, the structure and content of prompts can make a significant difference in the outcomes. Changes in length, arrangement of instances, and formulation, as well as the choice of examples and guidelines, substantially influence the models' outputs. Studies show that both the formulation and the sequence of examples in prompts affect LLM behavior [15].

The field has evolved from a basic practice of shaping prompts toward desired outputs to a structured area of research. Prompt engineering involves the systematic design and optimization of prompts to ensure accuracy, relevance, and coherence in LLM responses. This process is crucial for unlocking the full potential of the models, making them applicable in various fields.

Contemporary Prompt Engineering utilizes a variety of techniques, from fundamental approaches like role-prompting to more advanced methods such as "chain of thought"prompting. Its importance lies in its ability to direct the model's responses, enhancing its versatility and relevance in different sectors. A well-designed prompt can even mitigate challenges such as machine hallucinations [15].

In this report, we employed two specific techniques and one application of prompt engineering: one-zero-shot, chain-of-thought, and function calling, respectively.

- The one-zero-shot technique involves providing the model with a few examples before asking the desired question. By presenting specific examples, the model is better able to contextualize the subsequent question, resulting in more accurate and relevant answers. [15]
- The chain-of-thought technique consists of methodologically chaining information to help the model build knowledge. This technique helps the LLM develop step-by-step reasoning, allowing for more detailed and coherent answers. [15]

- The function calling technique refers to the ability of some models to connect to external functions. If the model identifies the need to call an external API based on the user's input, it will communicate with this API, which will execute its function and return a JSON string to the model. This string is then translated into a new message with the function results, summarized according to the user's question. [3]

These techniques exemplify how prompt engineering can significantly improve interactions with LLMs, ensuring that the generated responses are more accurate, relevant, and useful. Later in this report, the Prompt Engineering techniques relevant to the problem in question will be explored in greater detail and applied.

4 Case Studies

This report focuses on several case studies to make credit decisions more accessible to individuals with limited familiarity with machine learning concepts. Each case study addresses different aspects of interpretability and applicability of language models for generating explanations in credit decisions.

The first case study aims to explain to the user how a specific feature relates to the target variable. For example, the feature "Loan Amount"can increase the likelihood of a customer being classified as a bad payer, as higher loan amounts correspond to a higher probability of default. This relationship is illustrated in the Partial Dependence Plot (PDP) graph in Figure 5, where it is observed that the probability of the customer being classified as a bad payer is almost directly proportional to the loan amount.

Another example is "Loan Duration,"which also has a directly proportional relationship with the probability of default. In the PDP graph in Figure 5, it is noted that the probability of default increases with the loan duration, but after 46 months, this relationship changes, indicating different behavior for long-term loans. Additionally, the feature "Age"shows that the older the customer, the higher the likelihood of being classified as a good payer. This relationship can be seen in the PDP graph in Figure 5, where older customers tend to have lower default rates.

This information is crucial for end users to understand how the variables that compose their loan application influence the credit analysis outcome. Tools like Partial Dependence Plot graphs help illustrate these relationships clearly and accessibly, promoting greater transparency in credit decisions.

The second case study, related to the first, addresses how variation in a variable can affect the final decision of whether a customer is considered a good or bad payer. However, in a real scenario, care must be taken to ensure the system does not become vulnerable to adversarial attacks, which could threaten the privacy and security of the bank and its clients.

The third case study addresses the identification of the most important variables for credit approval or denial of a customer. This information is valuable for

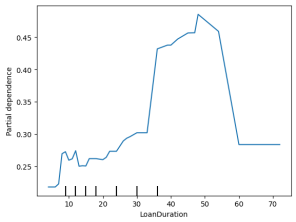

Figura 1: Effect of Loan Duration on Credit Decisions

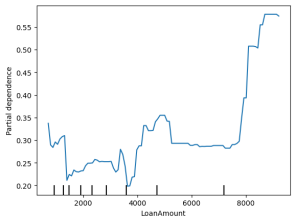

Figura 2: Impact of Loan Amount on Credit Decision

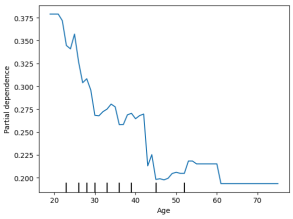

Figura 3: Influence of Age on Credit Approval

external stakeholders, such as regulators and auditors, who need to understand credit decisions without requiring in-depth technical knowledge of machine learning. For example, analyzing the customer with ID: '2', the Force Plot SHAP graphs in Figure 5 reveal the factors that contributed most to the customer being classified as a good payer by the credit scoring model. Notably, "Age"was the variable that contributed most to this classification. Although age is a sensitive variable [7] and should not significantly influence the credit decision, it is observed that at 45 years old, the customer has a lower probability of default, as shown in the graph in Figure 5.

The second variable that most influenced the positive classification was the "Loan Amount"of 3,049. The PDP graph for this variable (Figure 5) confirms that this amount is associated with loan approval. A variable that negatively contributed to the customer's score but not enough to classify them as a bad payer was "Years in Current Job,"with only 1 year. The graph in Figure 5 summarizes these observations, presenting the most important variables for the credit score of customer ID 2 in a simplified manner.

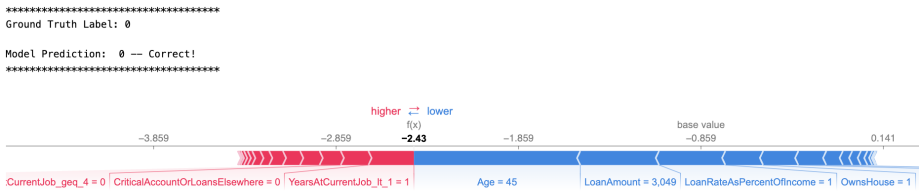

Figura 4: SHAP Force Plot Results for Customer 2

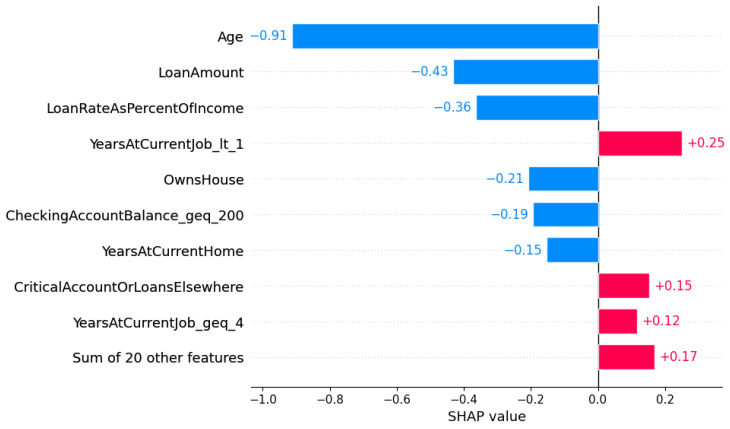

Figura 5: Most Important Features for Customer 2

5 Results

Initially, the Falcon-7B and Falcon-40B models were employed as LLMs (Large Language Models) to guide the research. These models were chosen due to their reputation as some of the best open-source models available on Hugging Face [4]. In the early stage, Falcon was used with a simple question model of the ”Question + Example”type. However, the answers obtained, as seen in Figures 6 and 7, were generalist, based on Falcon's prior knowledge.

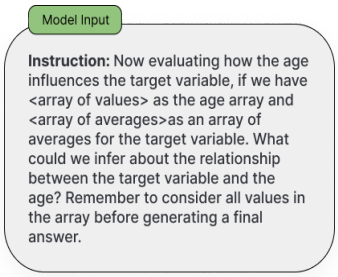

Figura 6: Model Input

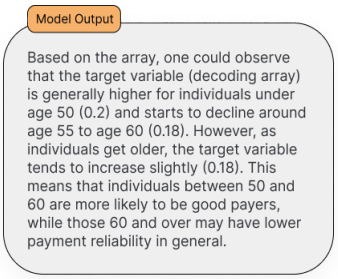

Figura 7: Model Output

As a result, prompt engineering techniques were introduced to enhance the model's performance in solving complex problems in a simplified way using the existing model [15]. However, no improvement was observed in the response quality, as the *Chain of Thought Prompting* technique only improves performance when used with models containing around 100 billion parameters [13]. Smaller models tend to produce illogical chains of thought, resulting in a reduction in accuracy compared to standard

prompts. Typically, the improvements obtained in prompts using the CoT technique are proportional to the model size [13]. However, using a model with more than 100 billion parameters, such as Falcon-180B, requires significantly more computational resources, leading us to consider other alternatives to generate higher-quality, detailed, and stable responses. In this regard, we opted to use Meta AI's Llama 3, widely recognized as one of the most advanced open-source models available on the market [4]. This model is capable of supporting a wide range of use cases and offers sophisticated features, including advanced reasoning capabilities [11]. Additionally, Llama 3 benefits from a robust and active online community, providing significant support compared to Falcon.

Thus, tests were conducted using prompt engineering techniques: *Generated Knowledge*, *One Zero Shot*, and *Chain of Thought*, focusing on the *LoanDuration* variable in case study 1. The *Generated Knowledge* technique involves structuring the model to first generate relevant facts necessary for the model to complete the prompt, often resulting in higher quality conclusions since the model is conditioned by relevant facts [15]. The *One Zero Shot* technique consists of providing the model with only a single instance as an example along with the task to evaluate how the model generalizes from limited information [15]. The *Chain of Thought* technique guides the model through a step-by-step chain of reasoning to reach a logical conclusion, providing a more detailed and robust explanation [15]. The results of these techniques can be seen in Figures 8, 9, and 10.

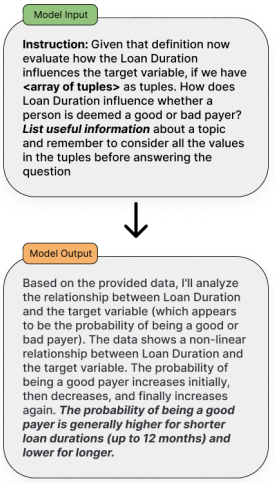

Figura 8: Generated Knowledge for LoanDuration

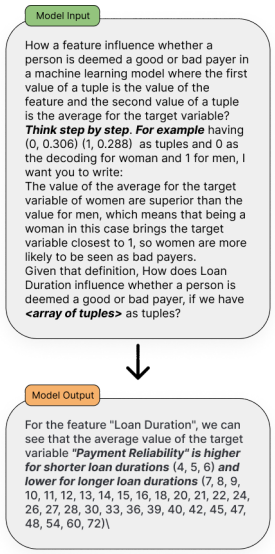

Figura 9: One Zero Shot for LoanDuration

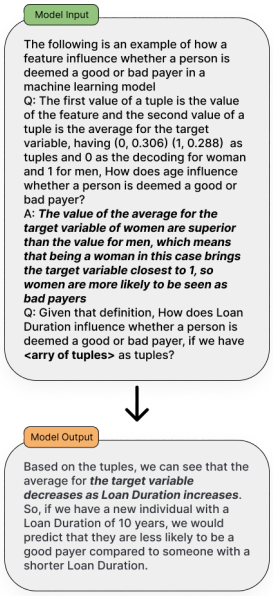

Figura 10: Chain of Thought for LoanDuration

However, the results of these techniques were unsatisfactory due to incom-

plete and insufficiently detailed responses, leading us to seek another technique. Consequently, the *Function Calling* technique was adopted [3]. In the context of the problem, the need to integrate the LLM with an API was identified. This is due to the requirement to apply interpretability methods to the data predicted by the XGBoost model. To do so, it is crucial to access the dataset, perform training and testing using XGBoost, and subsequently generate results for PDP (Partial Dependence Plot) and SHAP (SHapley Additive exPlanations). To meet this demand, the *Function Calling Prompt* method emerged as an ideal solution, allowing the creation of a robust connection between the LLM and multiple external APIs. One of these APIs is responsible for calculating PDP results, while another calculates SHAP results.

A detailed visualization of the system built can be seen in Figure 11, which illustrates the complete flow of the process, from data input to generating interpretable explanations.

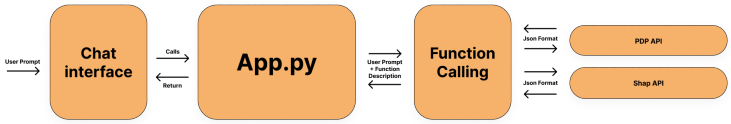

Figura 11: System Flowchart

By using the *Function Calling Prompt* method together with Llama 3-70B, satisfactory results were obtained, even in Portuguese, providing valuable insights for clients about how specific variables, such as loan amount and loan purpose, directly influence the credit profile. This resolution was developed with case study 1 in mind. These insights help the model determine the likelihood of a client being considered a good or bad payer. The analysis of the model-generated responses for these variables can be visualized in Figures 12 and 13.

Considering case study 2, which involves analyzing how variations in a variable can affect the final decision of whether a customer is considered a good or bad payer, satisfactory results were also obtained. This can be seen in Figure 14, which shows that the model was able to detect the particularity of the PDP graph in Figure **??**, contradicting the common assumption that as loan duration increases, so does the probability of the client being considered a bad payer. In fact, between 40 and 60 months, the opposite occurs: the probability of default drops from 0.407 to about 0.232.

Finally, concerning case study 3, which deals with identifying the most important variables for approving or denying a client's credit, the analysis was performed for a specific client with ID 2. The model-generated response can be visualized in Figure 15, which illustrates the most relevant variables for the credit decision of this client.

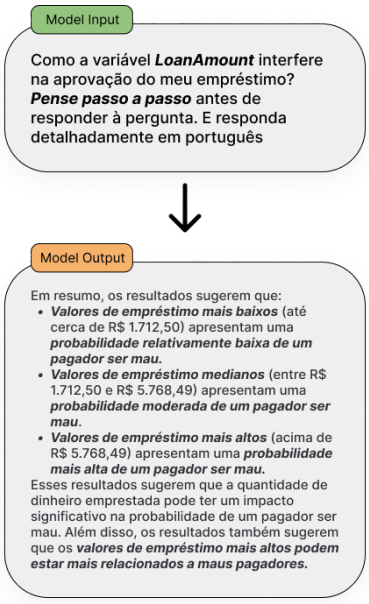

Figura 12: Partial Dependence Plot for Loan Amount

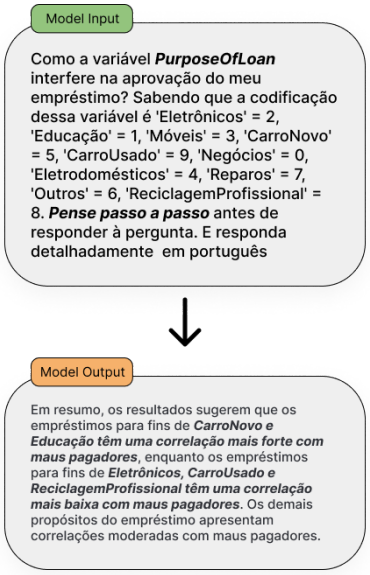

Figura 13: Partial Dependence Plot for Loan Purpose

5.1 UI Interface

To facilitate the use of the built application and make it more accessible, a user interface (UI) was developed using Gradio [6]. This interface was designed to guide the user in prompt choices, limiting the

need for prior technical knowledge of the domain. A detailed view of the interface can be seen in Figure 16, with functionalities that allow users to explore data analysis and prediction interpretations in a simplified manner, with automatic generation of graphics and tables to help guide decisions.

5.2 System Limitations

Despite the advancements and usefulness of the developed system, it is important to acknowledge some inherent limitations in its implementation and functioning. First, the system's effectiveness depends on the quality and accuracy of the LLM (Large Language Models) used, such as Llama 3-70B. If these models exhibit biases or limitations in their training data, it can negatively impact the results generated by the system [5].

Another relevant limitation is scalability. The use of large models, such as Llama 3-70B, requires considerable computational power. This can limit the system's

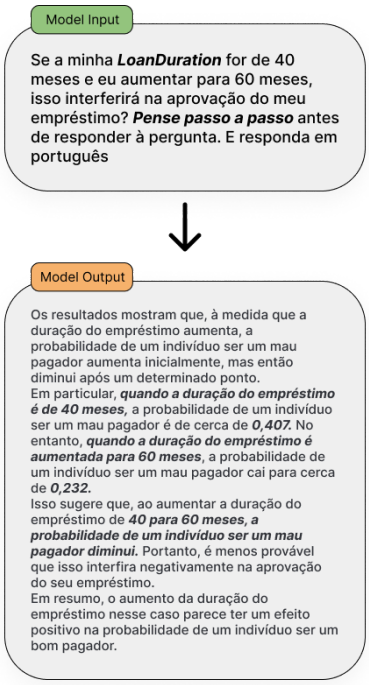

Figura 14: How LoanDuration Variation Can Affect the Target Variable

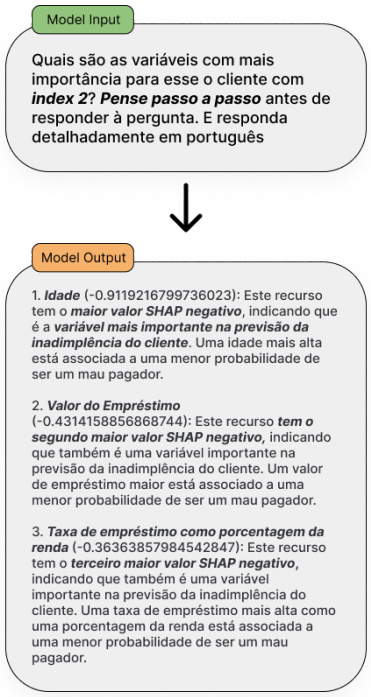

Figura 15: Most Important Variables for Credit Decision for Client with ID 2

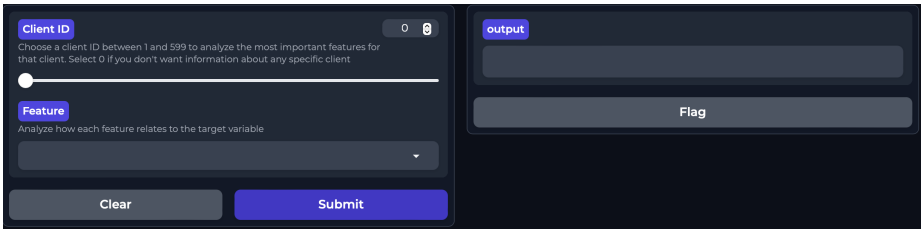

Figura 16: User Interface Developed in Gradio

scalability, especially in scenarios requiring real-time or large-scale analysis.

Machine learning systems, including those using interpretability techniques, can be vulnerable to adversarial attacks. Attackers can manipulate inputs to produce favorable or harmful outputs, compromising the integrity of credit decisions. Furthermore, the system's accuracy is directly related to the quality of the XGBoost model's training data. Low-quality, outdated, or biased data can lead to incorrect predictions and erroneous interpretations of credit variables [16].

The implementation of the *Function Calling Prompt* with multiple external APIs for PDP and SHAP calculations adds a layer of complexity to system maintenance. Any changes or failures in the APIs can disrupt the correct functioning of the system. Although the system provides valuable insights into the credit profile of specific clients, the ability to generalize these insights to other domains or contexts may be limited. The system is highly specialized and may not be applicable beyond its defined scope.

Finally, the use of sensitive data, such as age and credit history, raises legal and ethical concerns. Ensuring compliance with data protection regulations and ethical practices is crucial in the development and implementation of the system [8]. Recognizing these limitations is essential for continuing to improve the system, ensuring its robustness and reliability, and exploring new approaches that can mitigate these constraints.

6 Conclusion

This report presented a detailed approach to the use of Large Language Models (LLMs) for generating explanations in decision-making processes within the context of credit risk. By utilizing the Falcon-7B and Falcon-40B models [10], and later the Llama 3-70B [11], it was demonstrated how advanced prompt engineering techniques can enhance the interpretability and accuracy of the responses generated by the models, providing valuable insights for credit analysis [15].

The case studies illustrated how specific variables, such as loan amount and loan purpose, directly influence a client's credit profile. Through integration with external APIs for PDP and SHAP calculations, it was possible to provide detailed and contextual explanations about credit decisions, facilitating understanding for end users, regulators, and auditors.

The implementation of an intuitive user interface, built with Gradio [6], simplified client interaction with the system by limiting the user's prompt choices and offering clear options for analyzing specific features. This interface proved to be an effective tool for making credit decisions more accessible, even to those unfamiliar with machine learning concepts.

However, several limitations were identified in the system, including dependence on high-quality LLMs, scalability challenges, vulnerability to adversarial attacks, and ethical and legal issues related to the use of sensitive data [8]. These limitations highlight the ongoing need for improvement and the consideration of new approaches to mitigate these challenges.

In summary, the use of LLMs and interpretability techniques represents a significant advancement in the transparency and explainability of credit decisions. The application of methods such as the *Function Calling Prompt* [3] and integration with external APIs show promise for future research and the development of more robust and reliable systems. Continuing to explore and develop these technologies is crucial

for improving automated decision-making and ensuring fair and transparent practices in the financial sector.

Referências

[1] Irina Barakova, Dennis Glennon, and Ajay A. Palvia. Sample Selection Bias in Acquisition Credit Scoring Models: An Evaluation of the Supplemental-Data Approach. *SSRN Electronic Journal*, April 2013. Publisher: Elsevier BV. page.11

[2] Lara Marie Demajo, Vince Vella, and Alexiei Dingli. Explainable ai for interpretable credit scoring. *arXiv preprint arXiv:2012.03749*, 2020. page.11

[3] Llama API Documentation. Function calling - llama api. https://docs.llama-api.com/essentials/function, 2024. Accessed: 2024-07-12. page.66, page.1010, page.1313

[4] Clémentine Fourrier, Nathan Habib, Alina Lozovskaya, Konrad Szafer, and Thomas Wolf. Open llm leaderboard v2. https://huggingface.co/spaces/open-llm-leaderboard/open$_{l}lm_{l}eaderboard$, 2024. $page$.88, $page$.99

[5] Isabel O Gallegos, Ryan A Rossi, Joe Barrow, Md Mehrab Tanjim, Sungchul Kim, Franck Dernoncourt, Tong Yu, Ruiyi Zhang, and Nesreen K Ahmed. Bias and fairness in large language models: A survey. *Computational Linguistics*, pages 1–79, 2024. page.1111

[6] Gradio Inc. Gradio: Build and deploy machine learning models in python, 2024. Accessed: 2024-07-12. page.1111, page.1313

[7] Niki Kilbertus, Adrià Gascón, Matt Kusner, Michael Veale, Krishna Gummadi, and Adrian Weller. Blind justice: Fairness with encrypted sensitive attributes. In *International Conference on Machine Learning*, pages 2630–2639. PMLR, 2018. page.77

[8] Ministério da Defesa. Proteção de dados - lgpd, 2020. https://www.gov.br/defesa/pt-br/acesso-a-informacao/lei-geral-de-protecao-de-dados-pessoais-lgpd, Last accessed on 2022-09-26. page.11, page.1313

[9] Christoph Molnar. *Interpretable Machine Learning*. Lulu, 2 edition, 2022. page.11, page.33

[10] Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. The RefinedWeb dataset for Falcon LLM: outperforming curated corpora with web data, and web data only. *arXiv preprint arXiv:2306.01116*, 2023. page.44, page.1313

[11] Hugo Touvron, Louis Martin, et al. Llama 3: Open and efficient foundation language models, 2023. arXiv preprint arXiv:2307.01423. page.55, page.99, page.1313

[12] Kush R. Varshney. *Trustworthy Machine Learning*. Independently Published, Chappaqua, NY, USA, 2022. page.22

[13] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*, 2022. page.88, page.99

[14] Gio Wiederhold and John McCarthy. Arthur samuel: Pioneer in machine learning. *IBM Journal of Research and Development*, 36(3):329–331, 1992. page.22

[15] Cameron R. Wolfe. Advanced prompt engineering: What to do when few-shot learning isn't enough.... *Towards Data Science*, Aug 2023. Published online, 17 min read. page.55, page.88, page.99, page.1313

[16] Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. Mitigating unwanted biases with adversarial learning. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 335–340, 2018. page.1212

[17] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Z. Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jianyun Nie, and Ji rong Wen. A survey of large language models. *ArXiv*, abs/2303.18223, 2023. page.44