



UNICAMP

UNIVERSIDADE ESTADUAL DE CAMPINAS –
UNICAMP

Relatório do Estágio De Iniciação Científica (MC040)

**Aplicação de Grandes Modelos de Linguagem
para Explicações em Decisões de Risco de
Crédito**

Autora:

Larissa Ayumi Okabayashi

Associados:

Marcos Medeiros Raimundo

Thalita Biazuz Veronese

1 Introdução

Atualmente algoritmos de ML estão presentes no nosso cotidiano, desde sistemas de recomendação de filmes e músicas até áreas de alto risco como saúde, justiça criminal, finanças e assim por diante, auxiliando na tomada de decisões.

Um exemplo de como essa ferramenta é usada na área de finanças é através de modelos de pontuação de crédito que auxiliam na avaliação dos credores para decidir se aceitam ou não uma aplicação de empréstimo com base na expectativa do modelo em relação à capacidade do requerente de cumprir ou não suas obrigações financeiras [1]. Tais modelos são benéficos, uma vez que reduzem o tempo necessário para o processo de aprovação de empréstimos, permitem que os oficiais de empréstimo se concentrem apenas em uma porcentagem das aplicações, resultam em economia de custos, diminuem a subjetividade humana e reduzem o risco de inadimplência [2]. Existe uma vasta literatura sobre o tema, com várias técnicas de Aprendizado de Máquina (ML) e Inteligência Artificial (IA) propostas. Essas técnicas podem ser excepcionais em poder preditivo, mas também são conhecidas como métodos de "caixa-preta" [9], pois não fornecem explicações para suas decisões, tornando os humanos incapazes de interpretá-las. Portanto, é altamente improvável que qualquer especialista financeiro esteja disposto a confiar nas previsões de um modelo sem qualquer tipo de justificativa [2].

A explicabilidade desempenha um papel fundamental na compreensão das razões por trás das decisões tomadas por um modelo. É a medida pela qual um ser humano consegue entender a lógica por trás de uma determinada escolha [9]. No que diz respeito a análise de crédito, a explicabilidade se torna relevante para possibilitar a depuração e auditoria dos modelos de "caixa-preta" e assim poder entender a lógica por trás das previsões. Além disso, as regulamentações existentes, como o "direito à explicação" do GDPR e o ECOA [8] afirmam que empresas que negam crédito aos consumidores devem fornecer uma explicação precisa, mesmo que tenha sido gerada por um algoritmo opaco.

Contudo, a explicabilidade ainda possui uma barreira de entrada significativa para pessoas não letradas em Machine Learning, por isso, se faz necessário o uso do Processamento de Linguagem Natural e dos Grandes Modelos de Linguagem (LLMs) para transformar essas explicações em um texto inteligível. E assim, pessoas interessadas na explicação de um modelo de pontuação de crédito, mas que não possuem o conhecimento necessário para entender IA explicável (XAI), tais como um cliente que teve seu crédito negado pelo banco ou um stakeholder externo, podem ter acesso às informações obtidas no resultado da aplicação dos algoritmos de explicabilidade. Assim, a motivação por trás deste projeto é a necessidade declarada do desenvolvimento de explicações em linguagem natural para explicar as tomadas de decisão em risco de Crédito.

2 Objetivos

Desenvolver explicações acessíveis em linguagem natural para o campo do pontuação de crédito, especialmente destinadas a pessoas com dificuldades de acessibilidade ou com pouca familiaridade em conceitos de Machine Learning. A proposta visa tornar as informações sobre avaliação de crédito compreensíveis e utilizáveis por um público mais amplo, promovendo inclusão e facilitando o entendimento das decisões financeiras.

3 Materiais e Métodos

3.1 Dataset

Neste relatório, será explorada a aplicação da análise de dados e do aprendizado de máquina na avaliação de risco de crédito, para posteriormente aplicar métodos de interpretabilidade nesse modelo, utilizando o conjunto de dados de crédito German da UCI. Este conjunto de dados, preparado pelo Prof. Hofmann, é composto por 1000 entradas contendo 20 atributos categóricos/simbólicos. Esses atributos englobam informações pessoais e financeiras, histórico de crédito e características do empréstimo de cada indivíduo que solicita crédito a uma instituição bancária. Cada pessoa é classificada como um risco de crédito bom ou ruim de acordo com o conjunto de atributos.

3.2 Modelos de Aprendizado de Máquina para Pontuação de crédito: Definição e Avaliação

Inicialmente, é importante definirmos o que é um modelo de aprendizado de máquina (ML). Segundo a definição popularizada por Arthur Samuel [14]:

“A área de estudo que dá aos computadores a habilidade de aprender sem que sejam explicitamente programados”

De acordo com o livro Trustworthy Machine Learning [12]:

”O aprendizado de máquina é o estudo de algoritmos que utilizam dados e informações provenientes de observações e interações como entrada e generalizam a partir de entradas específicas para exibir características do pensamento humano.”

Este relatório busca interpretar modelos de pontuação de crédito, os quais ajudam a determinar se um cliente será considerado pelo banco um bom ou mau pagador. Para alcançar esse objetivo, foi utilizado o Dataset German, que é relativamente

pequeno e não possui dados faltantes significativos. O pré-processamento dos dados incluiu a codificação de variáveis categóricas, como Gênero e Finalidade do Empréstimo, utilizando técnicas de encoding adequadas.

Após o pré-processamento, algumas métricas de avaliação foram definidas para medir a performance dos modelos de ML: AUC, Balanced Accuracy, Accuracy, Precision, Recall, F1, Specificity, Sensitivity. Essas métricas são cruciais para entender o desempenho geral dos modelos e garantir que o modelo não introduza vieses ou injustiças em suas previsões.

Particularmente, a análise da precisão e do recall para diferentes subgrupos (por exemplo, gênero, raça) é essencial para identificar se o modelo está favorecendo ou desfavorecendo algum grupo específico. Avaliar as métricas de Specificity e Sensitivity por subgrupo ajuda a garantir que o modelo não esteja penalizando injustamente um grupo específico, promovendo a equidade nas previsões.

A divisão dos dados em conjuntos de treino e teste foi realizada, com 60% dos dados destinados ao treino e 40% ao teste. O treinamento foi realizado utilizando os modelos de Regressão Logística, Floresta Aleatória e XGBoost. No entanto, este relatório foca na aplicação dos métodos de interpretabilidade no XGBoost, devido ao seu desempenho superior, evidenciado por uma maior AUC, alta acurácia, boa precisão e especificidade, embora sua sensibilidade não seja a melhor entre os modelos comparados.

A escolha do XGBoost para aprofundamento se justifica por seu equilíbrio entre várias métricas de desempenho, tornando-o uma escolha robusta para aplicações de scoring de crédito, onde a capacidade de discriminar entre bons e maus pagadores é crucial. Além disso, técnicas de interpretabilidade, como SHAP (SHapley Additive exPlanations) e PDP (Partial Dependence Plot), serão aplicadas para fornecer explicações sobre as decisões do modelo, promovendo transparência e confiança na tomada de decisão baseada em aprendizado de máquina.

3.3 Métodos de Interpretabilidade

3.3.1 Partial Dependence Plot

Os gráficos de dependência parcial (PDP) é uma ferramenta de visualização utilizada no campo de interpretabilidade de modelos de aprendizado de máquina. Ele é projetado para ajudar a entender como uma variável específica afeta a previsão de um modelo, enquanto mantém as demais variáveis constantes [9]. Intuitivamente, podemos interpretar a dependência parcial como a resposta alvo esperada em função das características de entrada de interesse.

No âmbito deste estudo, geramos resultados do PDP utilizando o algoritmo XGBoost, treinado com dados do conjunto de dados investigado, o German. Esses resultados foram empregados para gerar insights acessíveis, permitindo uma explicação em linguagem natural para interpretar os gráficos de dependência parcial. O objetivo

é capacitar o usuário a compreender se a variação em determinadas características de um pedido de empréstimo afetará a aprovação de crédito, bem como compreender a relação de cada variável em relação à variável-alvo, que é a aprovação ou negação de crédito.

Por meio dessas explicações em linguagem natural, busca-se promover uma compreensão mais profunda e transparente dos processos de tomada de decisão de risco de crédito, contribuindo assim para o desenvolvimento de métodos de avaliação de crédito mais éticos e interpretáveis. Além disso, essa abordagem visa aprimorar a confiança dos usuários finais nos sistemas de aprendizado de máquina aplicados à avaliação de crédito, ao fornecer insights claros e acessíveis sobre como as decisões são influenciadas por diferentes características dos pedidos de empréstimo.

3.3.2 SHAP (SHapley Additive exPlanations)

Os valores SHAP (SHapley Additive exPlanations) representam uma poderosa abordagem para explicar as saídas de qualquer modelo de aprendizado de máquina. Essa metodologia se baseia em conceitos teóricos de jogos para quantificar a contribuição de cada "jogador" (ou recurso) para o resultado final do modelo. No contexto do aprendizado de máquina, cada recurso recebe um valor de importância (SHAP value) que reflete sua contribuição específica para a saída do modelo [9]. Recursos com valores SHAP positivos indicam uma influência positiva na previsão, enquanto valores negativos indicam uma influência negativa. A magnitude desses valores expressa a intensidade do efeito de cada recurso na predição.

No contexto deste projeto, os valores SHAP foram empregados para inferir a influência de determinadas características na predição da probabilidade de um cliente ser classificado como bom ou mau pagador. Em outras palavras, ao analisar um cenário em que um cliente teve seu crédito negado, os valores SHAP nos permitem identificar quais características (features) mais contribuíram para essa decisão. Essa análise proporciona uma compreensão detalhada das relações entre os atributos do cliente e a decisão de crédito, fornecendo insights críticos para interpretação e explicação dos processos de tomada de decisão do modelo.

Utilizando os valores SHAP, conseguimos não apenas identificar os principais determinantes por trás de uma decisão de crédito, mas também quantificar sua importância relativa. Essas informações são fundamentais para promover transparência e interpretabilidade nos sistemas de avaliação de crédito, possibilitando uma análise mais informada e ética das decisões de concessão de crédito.

3.4 Grandes Modelos de Linguagem

Um LLM, ou Modelo de Linguagem de Grande Porte, representa uma categoria avançada de modelos de linguagem treinados com técnicas de aprendizado profundo em vastos conjuntos de dados textuais.

Esses modelos são projetados para aprender padrões e relações entre entidades dentro de um determinado idioma. Capazes de gerar texto de forma semelhante ao humano, os LLMs são versáteis o suficiente para executar uma variedade de tarefas de Processamento de Linguagem Natural (PLN). Quando apresentados com uma nova entrada de texto, um LLM se esforça para prever ou gerar a continuação mais provável com base no conhecimento adquirido durante o treinamento [17].

A maioria dos modelos LLM é construída utilizando técnicas de Deep Learning, frequentemente por meio da arquitetura Transformer. Essa arquitetura de rede neural é especializada em capturar o contexto e o significado, permitindo o aprendizado de relações em dados sequenciais, como as palavras em uma frase [17].

3.4.1 Falcon

O Falcon LLM, desenvolvido pelo Instituto de Inovação Tecnológica com sede em Abu Dhabi, representa um avanço significativo no campo do processamento de linguagem de IA. Este modelo, parte da família Falcon, compreende três principais variações: Falcon-180B, Falcon-40B e Falcon-7B [10].

É digno de nota que o Falcon-40B é o primeiro modelo verdadeiramente aberto a rivalizar com muitos modelos proprietários existentes. Sendo licenciado sob a Apache 2.0, está disponível gratuitamente para uso tanto comercial quanto de pesquisa [10].

Todos os modelos foram treinados utilizando o conjunto de dados RefinedWeb, o qual foi submetido a rigorosos processos de filtragem e deduplicação para assegurar a qualidade dos dados de treinamento. Além disso, os modelos são facilmente acessíveis e ajustáveis através do ecossistema Hugging Face. Ferramentas como Inferência de Geração de Texto e PEFT (Treinamento Progressivo de Fertilização Precoce) estão disponíveis para executar e aprimorar os modelos conforme necessário.[10]

3.4.2 LLaMA 3

O LLaMA 3, parte da família de modelos de linguagem desenvolvida pela Meta Inc., foi concebido com o propósito de fomentar um ecossistema de IA aberto e responsável. Esta inovação proporciona aos desenvolvedores uma gama abrangente de recursos avançados, um design inovador, uma interface amigável, além de alta velocidade e confiabilidade. Treinado em larga escala, o LLaMA 3 tem acesso a mais de 15 trilhões de tokens de dados disponíveis publicamente, abrangendo diversos domínios, como código, conhecimento histórico e múltiplos idiomas. Essa extensa e diversificada base de dados de treinamento, aliada aos avanços da Meta em pré-treinamento e ajuste fino de instrução, resultou em um modelo que exibe um desempenho de ponta em uma ampla variedade de benchmarks do setor e cenários do mundo real. [11]

3.5 Engenharia de Prompt

A engenharia de prompt é o processo de projetar e otimizar o texto de entrada para Grandes Modelos de Linguagem (LLMs). [15] Seu objetivo é maximizar o potencial dos modelos, orientando-os para gerar respostas precisas e relevantes. Em aplicações práticas, a estrutura e o conteúdo dos prompts podem causar diferenças significativas nos resultados. Alterações no comprimento, disposição das instâncias e formulação, assim como a escolha de exemplos e diretrizes, influenciam substancialmente as saídas dos modelos. Estudos mostram que tanto a formulação quanto a sequência de exemplos nos prompts afetam o comportamento dos LLMs. [15]

A disciplina evoluiu de uma prática básica para moldar prompts em direção às saídas desejadas para uma área de pesquisa estruturada. A engenharia de prompt envolve o design sistemático e a otimização dos prompts para garantir precisão, relevância e coerência nas respostas dos LLMs. Esse processo é crucial para explorar todo o potencial dos modelos, tornando-os aplicáveis em diversas áreas.

A Engenharia de Prompt contemporânea utiliza uma variedade de técnicas, desde abordagens fundamentais, como o role-prompting, até métodos mais avançados, como o "chain of thought" prompting. Sua importância reside na capacidade de direcionar as respostas dos modelos, ampliando sua versatilidade e relevância em diferentes setores. Um prompt bem elaborado pode até mitigar desafios como alucinações de máquina. [15]

Neste relatório, utilizamos duas técnicas e uma aplicação específicas de engenharia de prompt: one-zero-shot, chain-of-thought, e function calling, respectivamente.

- A técnica one-zero-shot envolve fornecer ao modelo alguns exemplos antes de fazer a pergunta desejada. Ao apresentar exemplos específicos, o modelo é capaz de contextualizar melhor a pergunta subsequente, resultando em respostas mais precisas e relevantes. [15]
- A técnica chain-of-thought consiste em encadear informações metodologicamente para auxiliar o modelo na construção do conhecimento. Essa técnica ajuda o LLM a desenvolver um raciocínio passo a passo, permitindo respostas mais detalhadas e coerentes. [15]
- Já a técnica function calling refere-se à habilidade de alguns modelos de se conectar a funções externas. Se o modelo identificar a necessidade de chamar uma API externa a partir do input do usuário, ele se comunicará com essa API, que executará sua função e retornará para o modelo uma string JSON. Essa string será então traduzida em uma nova mensagem com os resultados da função, resumidos de acordo com a pergunta do usuário. [3]

Essas técnicas exemplificam como a engenharia de prompt pode melhorar significativamente a interação com LLMs, garantindo que as respostas geradas sejam mais precisas, relevantes e úteis. Mais adiante neste relatório, serão exploradas de ma-

neira mais detalhada e aplicada as técnicas de Engenharia de Prompt pertinentes à problemática em questão.

4 Casos de Estudo

Este relatório se concentra em alguns casos de estudo para tornar as decisões de crédito mais acessíveis a pessoas com pouca familiaridade com conceitos de Machine Learning. Cada caso de estudo aborda diferentes aspectos da interpretabilidade e aplicabilidade de modelos de linguagem para geração de explicações em decisões de crédito.

O primeiro caso de estudo visa explicar ao usuário como uma feature específica se relaciona com a variável alvo. Por exemplo, a feature "Valor do Empréstimo" pode aumentar a probabilidade de um cliente ser considerado um mau pagador, já que, quanto maior o valor do empréstimo, maior a probabilidade de inadimplência. Esta relação é ilustrada no gráfico de Partial Dependence Plot (PDP) na Figura 2, onde se observa que a probabilidade do cliente ser considerado um mau pagador é quase diretamente proporcional ao valor do empréstimo.

Outro exemplo é a "Duração do Empréstimo", que também possui uma relação diretamente proporcional com a probabilidade de inadimplência. No gráfico de PDP da Figura 1, nota-se que a probabilidade de inadimplência aumenta com a duração do empréstimo, mas após 46 meses essa relação muda, indicando um comportamento diferente para empréstimos de longa duração. Além disso, a feature "Idade" mostra que quanto mais idoso é o cliente, maior a probabilidade dele ser considerado um bom pagador. Esta relação pode ser observada no gráfico de PDP da Figura 3, onde clientes de idade mais avançada tendem a ter menores taxas de inadimplência.

Estas informações são cruciais para que o cliente final compreenda como as variáveis que compõem seu pedido de empréstimo influenciam o resultado da análise de crédito. Ferramentas como os gráficos de Partial Dependence Plot ajudam a ilustrar essas relações de maneira clara e acessível, promovendo uma maior transparência nas decisões de crédito.

O segundo caso de estudo está relacionado ao primeiro e trata de como a variação em uma variável pode afetar a decisão final de um cliente ser considerado um bom ou mau pagador. Contudo, em um cenário real, é necessário tomar cuidado para que o sistema não se torne vulnerável a ataques adversariais, ameaçando a privacidade e segurança do banco e de seus clientes.

O terceiro caso de estudo aborda a identificação das variáveis mais importantes para a aprovação ou negação de crédito de um cliente. Esta informação é valiosa para stakeholders externos, como reguladores e auditores, que precisam entender as decisões de crédito sem a necessidade de conhecimentos técnicos aprofundados em machine learning. Por exemplo, vamos analisar o cliente com id: '2'. Pelos gráficos de Force Plot SHAP na Figura 4, é possível entender quais foram os fatores que mais contribuíram para ele ser considerado um bom pagador pelo modelo de

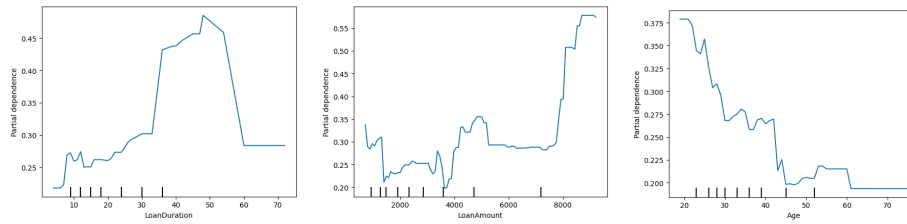


Figura 1: Efeito da Duração do Empréstimo nas decisões de crédito

Figura 2: Impacto do Valor do Empréstimo na decisão de crédito

Figura 3: Influência da Idade na aprovação de crédito

pontuação de crédito. Notavelmente, a "Idade" foi a variável que mais contribuiu para esta classificação. Embora a idade seja uma variável sensível [7] e não devesse influenciar significativamente a decisão de crédito, observamos que, aos 45 anos, o cliente tem uma menor probabilidade de inadimplência, conforme mostrado no gráfico da Figura 3.

A segunda variável que mais influenciou na classificação positiva foi o "Valor do Empréstimo" de 3,049. O gráfico de PDP dessa variável (Figura 2) confirma que este valor está associado à aprovação do empréstimo. Uma variável que contribuiu negativamente para a pontuação do cliente, mas não o suficiente para classificá-lo como um mau pagador, foi "Anos no Emprego Atual", com apenas 1 ano. O gráfico na Figura 5 resume essas observações, apresentando de forma simplificada as variáveis mais importantes para a pontuação do cliente com id 2.

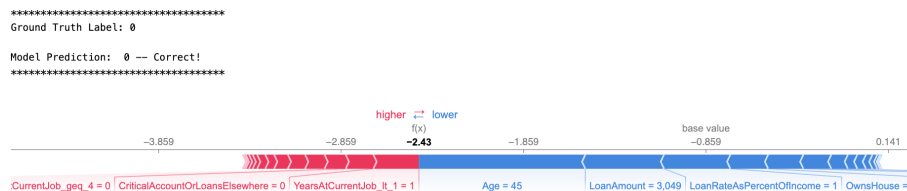


Figura 4: Resultado do Force Plot SHAP para o Cliente 2

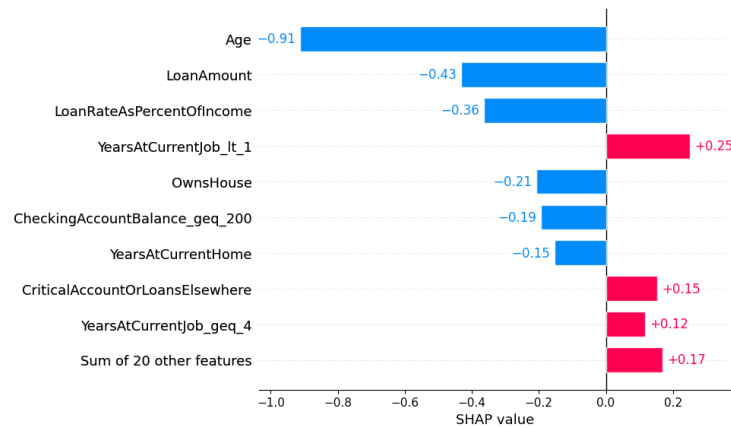


Figura 5: Features mais importantes para o Cliente 2

5 Resultados

Inicialmente, empregaram-se os modelos Falcon-7B e Falcon-40B como LLMs (Grandes Modelos De Linguagem) para orientar a pesquisa. Optou-se por esses modelos devido à sua reputação como um dos melhores modelos open source disponíveis no Hugging Face [4]. No estágio inicial, utilizamos o Falcon com um modelo de pergunta simples do tipo "Questão + Exemplo". No entanto, observamos que as respostas obtidas, que podem ser conferidas nas Figuras 6 e 7, foram generalistas, baseadas no conhecimento prévio do Falcon.

Model Input

Instruction: Now evaluating how the age influences the target variable, if we have <array of values> as the age array and <array of averages> as an array of averages for the target variable. What could we infer about the relationship between the target variable and the age? Remember to consider all values in the array before generating a final answer.

Model Output

Based on the array, one could observe that the target variable (decoding array) is generally higher for individuals under age 50 (0.2) and starts to decline around age 55 to age 60 (0.18). However, as individuals get older, the target variable tends to increase slightly (0.18). This means that individuals between 50 and 60 are more likely to be good payers, while those 60 and over may have lower payment reliability in general.

Figura 6: Model Input

Figura 7: Model Output

Diante disso, introduziu-se técnicas de Engenharia de Prompt para aprimorar o desempenho do modelo na resolução de problemas complexos de forma simplificada, utilizando o modelo existente [15]. Entretanto, não se observou melhoria na qualidade das respostas, pois a técnica *Chain of Thought Prompting* só melhora o desempenho quando utilizada com modelos que possuem cerca de 100 bilhões de parâmetros [13].

Modelos menores tendem a produzir cadeias de pensamentos ilógicas, resultando em uma redução na precisão quando comparados ao prompt padrão. Normalmente, as melhorias obtidas nos prompts usando a técnica de CoT são proporcionais ao tamanho do modelo [13]. Contudo, o uso de um modelo com mais de 100 bilhões de parâmetros, como o Falcon-180B, exige significativamente mais recursos computacionais, levando-nos a considerar outras alternativas para gerar respostas com maior qualidade, detalhamento e estabilidade. Nesse sentido, optou-se por utilizar o Llama 3 da Meta AI, amplamente reconhecido como um dos mais avançados modelos open-source disponíveis no mercado [4]. Este modelo é capaz de suportar uma vasta gama de casos de uso e oferece recursos sofisticados, incluindo capacidades de raciocínio avançado [11]. Além disso, o Llama 3 se beneficia de uma comunidade online robusta e ativa, proporcionando um suporte significativo em comparação com o Falcon.

Dessa forma, Foram realizados testes utilizando as técnicas de engenharia de prompt: *Generated Knowledge*, *One Zero Shot* e *Chain of Thought*, focando na variável *LoanDuration* do caso de estudo 1. A técnica *Generated Knowledge* envolve estruturar o modelo para primeiro gerar fatos relevantes necessários para o modelo completar o prompt, resultando frequentemente em maior qualidade de conclusão, uma vez que o modelo é condicionado a fatos relevantes[15]. A técnica *One Zero Shot* consiste em fornecer ao modelo apenas uma única instância como exemplo, junto com a tarefa a ser realizada, para avaliar como o modelo generaliza a partir de informações limitadas [15]. E a técnica *Chain of Thought* guia o modelo através de uma cadeia de raciocínios passo a passo para chegar a uma conclusão lógica, proporcionando uma explicação mais detalhada e robusta [15]. Os resultados dessas técnicas podem ser observados nas figuras 8, 9, e 10.

No entanto, os resultados dessas técnicas foram insatisfatórios devido à falta de completude e detalhamento das respostas, o que nos levou a procurar outra técnica. Diante disso, adotamos a técnica *Function Calling* [3]. No contexto do problema em questão, identificou-se a necessidade de integrar o LLM com uma API. Isso se deve à exigência de aplicar métodos de interpretabilidade aos dados previstos pelo modelo XGBoost. Para tanto, é crucial acessar o conjunto de dados, realizar o treinamento e teste utilizando o XGBoost, e posteriormente gerar os resultados do PDP (Partial Dependence Plot) e do SHAP (SHapley Additive exPlanations). Para atender a essa demanda, o método *Function Calling Prompt* surgiu como uma solução ideal, permitindo a criação de uma conexão robusta entre o LLM e múltiplas APIs externas. Uma dessas APIs é responsável pelo cálculo dos resultados do PDP, enquanto outra calcula os resultados do SHAP.

Uma visualização detalhada do sistema construído pode ser conferida na Figura 11, que ilustra o fluxo completo do processo, desde a entrada dos dados até a geração das explicações interpretáveis.

Ao empregar o método *Function Calling Prompt* em conjunto com o Llama 3-70B, foram obtidos resultados satisfatórios, inclusive em português, fornecendo insights valiosos para os clientes sobre como variáveis específicas, como o valor do empréstimo e o propósito do empréstimo, influenciam diretamente o perfil de crédito. Esta resolução foi construída pensando no caso de estudo 1. Essas informações auxi-

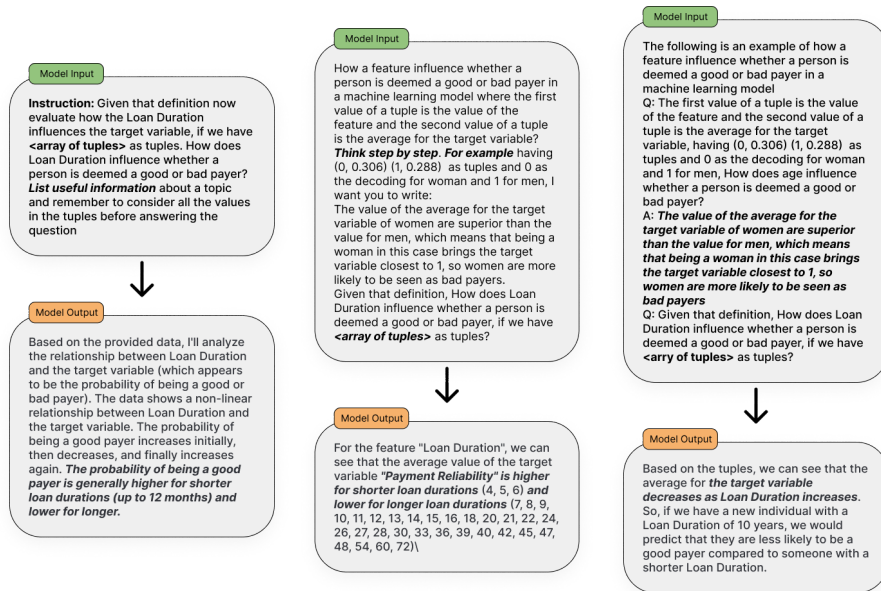


Figura 8: Generated Knowledge para LoanDuration

Figura 9: One Zero Shot para LoanDuration

Figura 10: Chain of Thought para LoanDuration

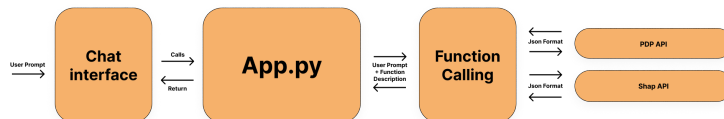


Figura 11: Fluxograma do Sistema

liam o modelo na determinação da probabilidade de um cliente ser considerado um bom ou mau pagador. A análise das respostas geradas pelo modelo para essas variáveis pode ser visualizada nas figuras 12 e 13.

Pensando no caso de estudo 2, que envolve a análise de como a variação em uma variável pode afetar a decisão final de um cliente ser considerado um bom ou mau pagador, também foi possível obter resultados satisfatórios. Isto pode ser observado na Figura 14, que mostra que o modelo foi capaz de detectar a particularidade do gráfico PDP na Figura 1, contradizendo o senso comum de que conforme a duração do empréstimo aumenta, a probabilidade de o cliente ser considerado um mau pagador também aumenta. De fato, de 40 a 60 meses, ocorre o contrário: a probabilidade de inadimplência cai de 0,407 para cerca de 0,232.

Finalmente, no que diz respeito ao caso de estudo 3, que aborda a identificação das variáveis mais importantes para a aprovação ou negação de crédito de um cliente,

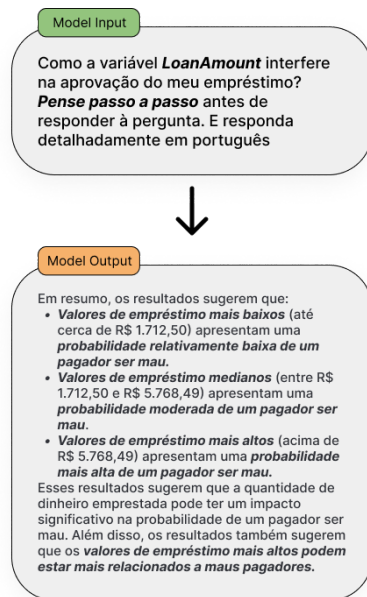


Figura 12: Partial Dependence Plot para o Valor do Empréstimo

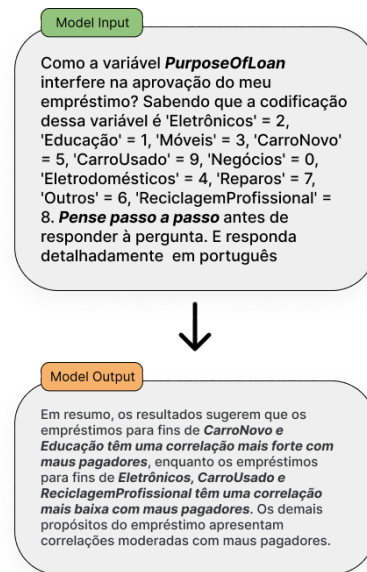


Figura 13: Partial Dependence Plot para o Propósito do Empréstimo

a análise foi realizada para um cliente específico com id igual a 2. A resposta gerada pelo modelo pode ser visualizada na Figura 15, que ilustra as variáveis mais relevantes para a decisão de crédito desse cliente.

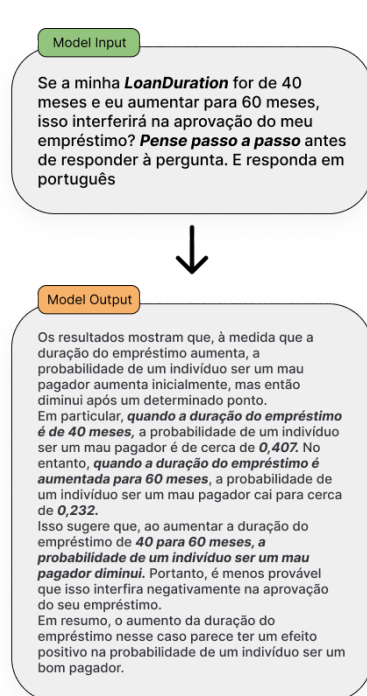


Figura 14: Como a variação da Loan-Duration pode afetar a variável target

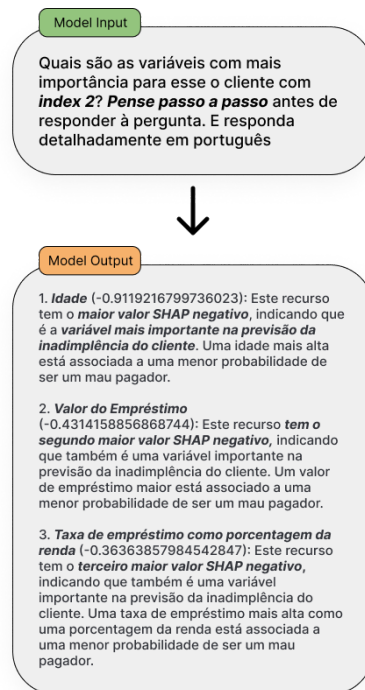


Figura 15: Variáveis mais importantes para a decisão de crédito do cliente com id 2

5.1 Interface UI

Para facilitar o uso da aplicação construída e torná-la mais acessível, foi desenvolvida uma interface de usuário (UI) utilizando o Gradio [6]. Esta interface foi projetada para orientar o usuário nas escolhas do prompt, limitando as opções e, assim, simplificando a interação com o sistema. A interface oferece duas principais funcionalidades:

- 1. Análise de Features de um Cliente Específico:** O usuário pode optar por analisar as características de um cliente específico, selecionando o identificador (ID) correspondente.
- 2. Análise da Relação entre Features e a Variável Alvo:** O usuário pode escolher analisar como uma determinada feature se relaciona com a variável alvo. As features disponíveis para seleção são: *LoanDuration*, *LoanAmount*, *PurposeOfLoan*, *YearsAtCurrentHome*, *LoanRateAsPercentOfIncome*, *Age*, e *NumberOfOtherLoansAtBank*.

Esta interface foi concebida para operar em conjunto com o sistema desenvolvido, permitindo ao usuário realizar análises detalhadas e obter insights específicos sobre as variáveis que influenciam o perfil de crédito. A funcionalidade e a usabilidade da interface podem ser observadas na Figura 16, que ilustra a integração das opções disponíveis com o backend do sistema.

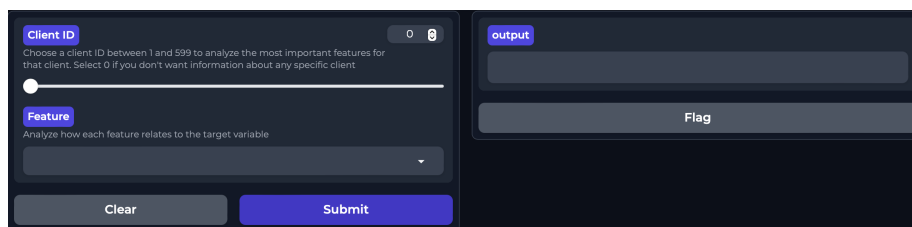
The image shows a web application interface with a dark theme. On the left, there is a 'Client ID' section with a dropdown menu and a 'Feature' section with a text input field. Below these are 'Clear' and 'Submit' buttons. On the right, there is an 'output' section with a large text area and a 'Flag' button.

Figura 16: Interface de Usuário desenvolvida com Gradio

5.2 Limitações do Sistema

Apesar dos avanços e da utilidade do sistema desenvolvido, é importante reconhecer algumas limitações inerentes à sua implementação e funcionamento. Primeiramente, a eficácia do sistema depende da qualidade e precisão dos modelos LLM (Large Language Models) utilizados, como o Llama 3-70B. Caso esses modelos apresentem vieses ou limitações em seus dados de treinamento, isso pode impactar negativamente os resultados gerados pelo sistema [5].

Outra limitação relevante é a escalabilidade. O uso de modelos grandes, como o Llama 3-70B, requer considerável poder computacional. Isso pode limitar a escalabilidade do sistema, especialmente em cenários onde se requer análise em tempo real ou em grande volume.

Os sistemas de machine learning, incluindo os que utilizam técnicas de interpretabilidade, podem ser vulneráveis a ataques adversariais. Atacantes podem manipular as entradas para produzir saídas favoráveis ou prejudiciais, comprometendo a integridade das decisões de crédito. Ademais, a precisão do sistema está diretamente relacionada à qualidade dos dados de treinamento do modelo XGBoost. Dados de baixa qualidade, desatualizados ou enviesados podem levar a previsões incorretas e interpretações errôneas das variáveis de crédito [16].

A implementação do *Function Calling Prompt* com múltiplas APIs externas para cálculos de PDP e SHAP adiciona uma camada de complexidade à manutenção do sistema. Qualquer alteração ou falha nas APIs pode interromper o funcionamento correto do sistema. Embora o sistema forneça insights valiosos sobre o perfil de crédito de clientes específicos, a capacidade de generalizar esses insights para outros domínios ou contextos pode ser limitada. O sistema é altamente especializado e pode não ser aplicável fora do escopo definido.

Finalmente, a utilização de dados sensíveis, como idade e histórico de crédito,

levanta questões legais e éticas. É crucial garantir a conformidade com regulamentos de proteção de dados e práticas éticas no desenvolvimento e implementação do sistema [8]. Reconhecer essas limitações é fundamental para continuar aprimorando o sistema, garantir sua robustez e confiabilidade, e explorar novas abordagens que possam mitigar essas restrições.

6 Conclusão

Este relatório apresentou uma abordagem detalhada sobre o uso de Grandes Modelos de Linguagem (LLMs) para a geração de explicações em tomadas de decisão no contexto de risco de crédito. Utilizando os modelos Falcon-7B e Falcon-40B [10], e posteriormente o Llama 3-70B [11], demonstrou-se como técnicas avançadas de engenharia de prompt podem melhorar a interpretabilidade e a precisão das respostas geradas pelos modelos, proporcionando insights valiosos para a análise de crédito [15].

Os casos de estudo abordados ilustraram como variáveis específicas, como o valor do empréstimo e o propósito do empréstimo, influenciam diretamente o perfil de crédito de um cliente. Através da integração com APIs externas para cálculos de PDP e SHAP, foi possível fornecer explicações detalhadas e contextuais sobre as decisões de crédito, facilitando a compreensão para usuários finais, reguladores e auditores.

A implementação de uma interface de usuário intuitiva, construída com o Gradio [6], permitiu simplificar a interação do cliente com o sistema, limitando as escolhas do prompt do usuário e oferecendo opções claras para a análise de features específicas. Esta interface demonstrou ser uma ferramenta eficaz para tornar as decisões de crédito mais acessíveis, mesmo para aqueles sem familiaridade com conceitos de machine learning.

No entanto, foram identificadas várias limitações no sistema, incluindo dependência de modelos LLM de alta qualidade, desafios de escalabilidade, vulnerabilidade a ataques adversariais, e questões éticas e legais relacionadas ao uso de dados sensíveis [8]. Estas limitações destacam a necessidade contínua de aprimoramento e a consideração de novas abordagens para mitigar esses desafios.

Em suma, o uso de LLMs e técnicas de interpretabilidade representa um avanço significativo na transparência e explicabilidade das decisões de crédito. A aplicação de métodos como o *Function Calling Prompt* [3] e a integração com APIs externas mostram-se promissoras para futuras pesquisas e desenvolvimento de sistemas mais robustos e confiáveis. Continuar a explorar e desenvolver essas tecnologias é crucial para melhorar a tomada de decisão automatizada e assegurar práticas justas e transparentes no setor financeiro.

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.1515
- [4] Clémentine Fourrier, Nathan Habib, Alina Lozovskaya, Konrad Szafer, and Thomas Wolf. Open llm leaderboard v2. https://huggingface.co/spaces/open-llm-leaderboard/open_llm_leaderboard, 2024. page.99, page.1010
- [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.1414
- [6] Gradio Inc. Gradio: Build and deploy machine learning models in python, 2024. Accessed: 2024-07-12. page.1313, page.1515
- [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.88
- [8] Ministério da Defesa. Proteção de dados - lgpd, 2020. <https://www.gov.br/defesa/pt-br/aceso-a-informacao/lei-geral-de-protecao-de-dados-pessoais-lgpd>, Last accessed on 2022-09-26. page.11, page.1515
- [9] Christoph Molnar. *Interpretable Machine Learning*. Lulu, 2 edition, 2022. page.11, page.33, page.44
- [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.55, page.1515
- [11] Hugo Touvron, Louis Martin, et al. Llama 3: Open and efficient foundation language models, 2023. *arXiv preprint arXiv:2307.01423*. page.55, page.1010, page.1515
- [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.99, page.1010
- [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.66, page.99, page.1010, page.1515
- [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.1414
- [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.55