Insper

DETECÇÃO DE BUGS EM MODELOS DE MACHINE LEARNING

Giovanni Cardoso Pertence dos Santos (giovannicps@al.insper.edu.br)
João Pedro Gianfaldoni de Andrade (joaopga1@al.insper.edu.br)
William Augusto Reis da Silva (williamars@al.insper.edu.br)

Trabalho de Conclusão de Curso

Relatório
Versão Preliminar
do Projeto Final de Engenharia

São Paulo-SP
SETEMBRO 2022

Giovanni Cardoso Pertence dos Santos
João Pedro Gianfaldoni dos Santos
William Augusto Reis da Silva

DETECÇÃO DE BUGS EM MODELOS DE MACHINE LEARNING

Relatório Preliminar do Projeto Final de Engenharia

Relatório apresentado ao curso de Engenharia, como requisito para o Trabalho de Conclusão de Curso.

Professor Orientador: Prof. Fabricio Jailson Barth

Mentor na Empresa: Thiago Cardoso

Coordenador TCC/PFE: Prof. Dr. Luciano Pereira Soares

São Paulo - SP
SETEMBRO 2022

Sumário

RESUMO **3**

ABSTRACT **4**

1 INTRODUÇÃO **7**

1.1 ESCOPO DO PROJETO 8
1.2 RECURSOS 8
1.3 CRONOGRAMA 8
1.4 MAPEAMENTO DOS STAKEHOLDERS 10

RESUMO

Modelos de Machine Learning em produção muitas vezes apresentam comportamento inesperado, que não foi detectado durante as etapas de treino, teste e validação, ao serem expostos a dados do mundo real. Esses comportamentos inesperados, ou bugs do modelo, possuem diversos motivos, por exemplo o não cumprimento de regras de negócio, generalizações errôneas, ou dados de treino rotulados incorretamente.

O projeto tem como objetivo desenvolver uma ferramenta de identificação desses bugs em modelos de Machine Learning de classificação, na forma de "caixa-preta", que já estão em produção. Os dados de entrada dos modelos são estruturados, ou seja são tabulares e não texto ou imagens.

Essa ferramenta, desenvolvida na linguagem de programação Python, com o auxílio de frameworks como Pandas, Scikit-Learn e SHAP, realiza análises tanto do modelo em si, quanto dos dados utilizados para sua construção, tendo como objetivo final analisar e encontrar bugs nos modelos de Machine Learning do cliente IFOOD.

Palavras-chave: Machine Learning; Identificação de bugs; Modelos em produção.

ABSTRACT

Machine Learning models in production phases sometimes present unexpected behavior, which have not been detected during training, testing orvalidation phases, when faced with data from the real world. Those unexpected behaviors, or model bugs, have different explanations, such as not following business rules, wrong generalizations, or mislabeled training data.

The goal of this project is to develop a tool that identifies those bugs in black-box classification Machine Learning models that are in production.

This tool developed with Python programming language and frameworks such as Pandas, Scikit-Learn and SHAP, performs analysis of the model and the data, and it's main goal is to analyse and fing bugs in IFOOD's Machine Learning models.

Keywords: Machine Learning; bug identification; models in production

1. Introdução

A empresa Ifood.com Agência de Restaurantes Online S.A, conhecida como IFOOD, é uma empresa brasileira que aproxima clientes, restaurantes e entregadores, oferecendo serviços de delivery de restaurantes e mercados, vale-refeição, cartão presente, entre outros. Fundada em 2011, a empresa cresceu exponencialmente no país, possuindo atualmente 80% de participação no mercado de entrega de alimentos.

Nessas diversas áreas de atuação, o IFOOD utiliza modelos de Machine Learning para as mais diversas tarefas, como em prevenção de fraudes de pagamentos, recomendações de restaurantes ou categorias e monitoramento de textos e catálogos. Esses modelos, apesar de essenciais para o funcionamento da empresa, não são perfeitos, e apresentam comportamentos inesperados, conhecidos popularmente como bugs, quando já estão em produção, apesar de terem passado por todas as etapas de treinamento, teste e validação.

Um exemplo de um bug ou comportamento inesperado seria num modelo de prevenção de fraude que deveria identificar, por exemplo, que quanto maior o número de cartões registrados em uma conta, maior a probabilidade de uma transação realizada por essa conta ser uma fraude. Nas etapas de treinamento e teste esse comportamento ocorreu como previsto, porém ao ser colocado em produção o modelo passou a identificar alguns casos com muitos cartões registrados como transações não fraudulentas.

Esse tipo de comportamento é mais comum em situações adversariais, ou seja, situações em que existe um agente malicioso tentando burlar o sistema, porém também pode ocorrer em diversas outras situações, como até em clientes que estão tentando fazer uma compra normal, porém o modelo apresenta como uma fraude e acaba cancelando, o que prejudica a empresa por impedir uma venda franca. As razões para a existência desses bugs também são as mais diversas, como o não cumprimento de regra de negócios, generalizações erradas, ou dados de treinamento com classificação incorreta.

Este projeto tem como objetivo desenvolver uma ferramenta que seja capaz de identificar esses bugs em modelos de Machine Learning de classificação, permitindo assim que sejam corrigidos e evitando prejuízos tanto para a empresa quanto para os seus clientes. A ideia parte desses princípios citados anteriormente das razões da existência de bugs, tendo assim a ideia principal de analisar se há o cumprimento total de regras de negócios no banco de dados,

para posteriormente analisar o impacto de *features* para o resultado, que define se é uma fraude ou não.

O projeto se demonstra relevante para a empresa cliente tendo em vista que possibilitará a identificação de anomalias tanto no banco de dados quanto na aplicação de seus modelos em produção, permitindo que a empresa corrija essas anomalias, evitando assim casos prejudiciais como fraudes ou negação de serviços para clientes legítimos.

Por fim, por se tratar de uma ferramenta construída com linguagem de programação, bibliotecas e frameworks *open source*, não haverá problemas regulatórios ou financeiros, podendo a empresa utilizá-la como protótipo ou como produto.

1.1 Escopo do projeto

O projeto consiste no desenvolvimento de uma ferramenta que recebe como entrada um *dataset* ou um modelo de Machine Learning e regras de negócio visando à identificação de quebras dessas regras dentro do modelo ou do banco de dados. Posteriormente, a realização de análise ou geração de ataques adversariais para a identificação de problemáticas na previsão ou nos próprios dados que já se tem.

1.2 Recursos

Por se tratar de uma ferramenta simples de aplicação em quesito de bibliotecas necessárias e pela falta de necessidade de uma plataforma complexa para aplicação, os recursos imprescindíveis para a execução do projeto são apenas computadores com capacidade de carregar a base de dados de treinamento dos modelos de aprendizado de máquina. Para tal, foi importante se ter instalado o Anaconda, maior plataforma de distribuição de Python, com o objetivo de se utilizar do Jupyter Notebook, que é onde se faz todos os carregamentos dos *datasets* além das análises, e também a instalação de bibliotecas e frameworks, a exemplo de Pandas, Scikit-Learn e Numpy.

1.3 Cronograma

O cronograma do projeto, visando à melhor organização e entendimento do que deve ser feito até o final, foi dividido em etapas, as quais têm determinadas em qual mês é realizada. Posteriormente, com o objetivo de se ter uma noção um pouco mais profunda, também se tem uma análise quinzenal das atividades a serem feitas.

As etapas citadas anteriormente são:

- Etapa 1: Compreensão do problema e busca de literatura relevante ao tema

- Etapa 2: Escolha do *dataset* e treinamento de modelos base

- Etapa 3: Construção do algoritmo inicial capaz de identificar violações de regras de negócio tanto nos dados quanto nos modelos

- Etapa 4: Início da construção da ferramenta que será utilizada pela empresa

- Etapa 5: Inclusão de análises SHAP e ataques adversariais

- Etapa 6: Finalização da ferramenta, aprimoramento e generalização para modelos e datasets

Tabela 1 - Mapeamento mensal das etapas do projeto

	Agosto	***Setembro***	***Outubro***	***Novembro***	***Dezembro***
Etapa 1					
Etapa 2					
Etapa 3					
Etapa 4					
Etapa 5					
Etapa 6					

Tabela 2 - Análise quinzenal de atividades do projeto

Quinzena	**Início**	**Atividades**
Primeira	8/ago	Entendimento do problema; Busca por literatura para compreensão da área;
Segunda	22/ago	Produção do Relatório Preliminar; Escolha do *dataset*; Treinamento de modelos base;
Terceira	5/set	Construção do algoritmo de violação de regras de negócio; Início da construção da ferramenta;
Quarta	19/set	Produção do Relatório Intermediário;
Quinta	3/out	Bancas Intermediárias e Apresentação para Empresa;
Sexta	17/out	Inclusão do SHAP;

		Inclusão de ataques adversariais com ferramentas como ToolBox;
Sétima	31/out	Finalização da ferramenta; Validação com cliente e melhorias finais;
Oitava	14/nov	Produção do Relatório Final;
Nona	28/nov	Bancas Finais;
Décima	12/dez	Apresentação para empresa;

1.4 Mapeamento dos stakeholders

Os stakeholders são as pessoas envolvidas e impactadas com o projeto de certa forma e, por conta disso, o grupo inclui pessoas do Insper, da empresa IFOOD e o próprio grupo desenvolvedor do projeto. O mapeamento encontra-se na Tabela 3.

Tabela 3 - Mapeamento dos stakeholders do projeto

Stakeholder	Posição	Papel no Projeto	Expectativas
Thiago Cardoso	Diretor de Data Science do iFood	Instrução, acompanhamento e avaliação	
Guilherme Righetto	Cientista de Dados do iFood	Instrução, acompanhamento e avaliação	
Prof. Fabricio Barth	Professor do Insper	Supervisão, acompanhamento e orientação	
Equipe desenvolvedora do projeto	Estudantes do Insper	Desenvolvimento da ferramenta	