

Detecção de Indisponibilidades em Aplicações Web

Daniel Schulz¹, Marcelo Marotta¹, Lucas Bondan^{1,2}, Geraldo P. Rocha Filho^{1,3},
Marcos Caetano¹, Aleteia Araujo¹

¹Departamento de Ciência da Computação
Universidade de Brasília – UnB – Brasília – DF – Brasil

²Rede Nacional de Ensino e Pesquisa – RNP

³Departamento de Ciências Exatas e Tecnológicas
Universidade Estadual do Sudoeste da Bahia – UESB – Vitória da Conquista – BA – Brasil

daniel.schulz@aluno.unb.br, marcelo.marotta@unb.br, lucas.bondan@rnp.br,
geraldo.rocha@uesb.br, mfcaetano@unb.br, aleteia@unb.br

Abstract. The internet's growth transformed business services but also brought challenges with failures in Information Systems. Stable operations require continuous monitoring aligned with ITIL. This study suggests a novel method, applying data mining for rapid failure detection. It evaluates real-time web analytics data to swiftly identify critical web app failures. Preprocessing and hyperparameters were optimized using hyperopt for improve accuracy. Seven models were made; LSTM excelled, validating its effectiveness in differentiating failures from successes.

1. Introdução

A expansão da Internet e o uso generalizado de smartphones revolucionaram a prestação de serviços financeiros pelos bancos [Malaquias, 2019]. No Brasil, o número de smartphones superou o de computadores e habitantes em 2022 [Meirelles, 2022]. Nesse cenário, o mobile banking, tornou-se o principal canal de interação entre usuários e instituições financeiras, desempenhando um papel essencial na satisfação do cliente.

No entanto, o aumento dos acessos digitais trouxe consigo desafios de interrupções nos Sistemas de Informação (SI) no setor bancário, resultando em prejuízos e impactos negativos nos negócios. Isso afeta a reputação das empresas e a satisfação dos clientes. Portanto, identificar falhas rapidamente é crucial para que as equipes de suporte possam corrigi-las e restabelecer os serviços afetados.

Esta pesquisa apresenta uma abordagem de detecção de falhas com base nas interações dos usuários com um aplicativo móvel bancário. A proposta é classificar em tempo real as interações dos usuários para identificar falhas na infraestrutura. Foi criado um conjunto de dados rotulados com as interações dos usuários e a disponibilidade do aplicativo móvel. Em seguida, um processo de otimização foi realizado, explorando diferentes técnicas de engenharia de atributos e ajuste de hiperparâmetros em diversos algoritmos de classificação. Os algoritmos foram avaliados quanto à sua capacidade de classificar as interações dos usuários de acordo com a disponibilidade do serviço.

2. Trabalhos Relacionados

Com o objetivo de melhorar a detecção de falhas e agir de forma proativa, diversos estudos têm aplicado técnicas de aprendizado supervisionado em Sistemas de Informação. Du et al. [Du, 2017] propuseram um sistema de detecção de anomalias em aplicações distribuídas, como *Hadoop Distributed File System* (HDFS) e *Open Stack*, utilizando registros de sistema (*logs*). Eles aplicaram redes LSTM e processamento textual para identificar padrões normais de execução e detectar anomalias quando os *logs* eram incomuns, utilizando aprendizado supervisionado.

Zhang et al. [Zhang, 2019] propuseram um modelo de detecção de anomalias que lida com a variação dos logs ao longo do tempo. Eles utilizaram redes neurais Bi-LSTMs com mecanismo de atenção para capturar informações contextuais e aprender a importância dos diferentes eventos de logs. Além disso, Zhang et al. [Zhang, 2016] propuseram uma abordagem para reduzir o espaço de características utilizando a técnica TF-IDF, combinada com redes neurais profundas LSTMs, resultando em uma otimização no processo de detecção de anomalias.

Apesar dos avanços na detecção e previsão de falhas em aplicações específicas, nenhum desses estudos abordou a perspectiva do usuário ou adotou uma estratégia que contemplasse uma análise abrangente de sistemas compostos por múltiplos subsistemas, permitindo a avaliação do serviço final entregue. Diante desse cenário, este trabalho propõe uma abordagem de detecção de falhas, que leva em consideração a perspectiva do usuário e uma análise abrangente de sistemas e subsistemas.

3. Referencial Teórico

3.1 Web Analytics

Os clientes desempenham um papel crucial no sucesso dos serviços online. Para compreender sua interação e extrair métricas relevantes, surgiram as ferramentas de Web Analytics. Elas vão além do simples registro de visualizações de páginas, fornecendo informações valiosas sobre o comportamento do usuário, como tempo entre cliques, localização do cursor e mensagens recebidas. Essas ferramentas rastreiam toda a jornada do cliente, desde o início até a saída da aplicação.

Essas métricas detalhadas permitem uma compreensão profunda do envolvimento do cliente, auxiliando as empresas a tomar decisões informadas e otimizar suas estratégias online. A análise desses dados revela padrões, áreas de melhoria e possibilita melhorias na experiência do usuário e no desempenho do serviço na Internet.

3.2 Definição de Falha

Neste trabalho, a falha refere-se a problemas que causam interrupções ou disfunções no sistema, resultando na indisponibilidade ou mau funcionamento do serviço. Essas falhas podem ser causadas por diversos fatores, como problemas técnicos, erros de configuração, sobrecarga, falhas de energia e erros humanos.

A maioria das falhas está relacionada a problemas técnicos e sobrecarga, que podem ocorrer isoladamente ou em conjunto. As falhas abordadas neste trabalho afetam uma parte significativa dos clientes e não se limitam a funcionalidades específicas,

comprometendo o funcionamento geral do sistema e resultando em indisponibilidade e mau desempenho do serviço.

4. Metodologia

Este trabalho adota o modelo CRISP-DM (*Cross Industry Standard Process for Data Mining*) como referência para conduzir o processo de mineração de dados [Chapman et al., 2000]. O CRISP-DM é um modelo cíclico e iterativo que inclui 6 fases recomendadas para garantir o sucesso do projeto de mineração de dados.

4.1 Entendimento do Negócio e Entendimento dos Dados

Devido à importância crescente do Mobile Banking, é essencial para as instituições financeiras monitorar continuamente a disponibilidade dos serviços oferecidos aos clientes. Assim, propõe-se a criação de um classificador binário com base nas interações dos usuários com a aplicação. Esse classificador terá a capacidade de identificar e sinalizar automaticamente falhas no sistema que prejudicam o uso dos serviços pelos clientes. Essa abordagem permitirá que as instituições ajam de forma proativa na resolução de problemas, garantindo a disponibilidade e qualidade dos serviços no Mobile Banking.

Os dados foram obtidos através do projeto Matomo, uma ferramenta de análise web amplamente adotada e de código aberto [Gamalielsson et al., 2021]. Essa plataforma coleta informações anonimizadas sobre as interações dos usuários com um aplicativo online. Neste estudo, foram utilizados dados provenientes de um banco brasileiro, abrangendo registros como páginas visitadas, mensagens de sistema enviadas aos usuários, tempos de resposta de requisições, mensagens de erro e atividades relacionadas a operações financeiras, entre outros aspectos.

O conjunto de dados foi explorado para facilitar o entendimento e gerar percepções. A matriz de correlação de Pearson revelou que muitos atributos possuem baixa correlação com o rótulo, indicando uma oportunidade de seleção de atributos. Assim, a seleção de atributos foi aplicada pelo método de filtragem, utilizando como função de pontuação o p -valor da técnica de variância estatística ANOVA.

As métricas deste estudo representam as interações entre os usuários e o aplicativo. Essas métricas apresentam uma distribuição assimétrica com uma longa calda à direita, sugerindo a possibilidade de ajuste de escala devido às diferentes escalas dos atributos. Tal ajuste será implementado com a técnica de padronização.

Outro aspecto analisado foi a proporção desequilibrada das classes na variável de rótulo. Dos 800 mil registros, apenas 3106 correspondem a indisponibilidades, representando uma proporção de apenas 0,388%. Esse desequilíbrio pode impactar o desempenho dos algoritmos, tornando necessário considerar uma estratégia de balanceamento e a escolha adequada das métricas de avaliação, levando em conta as características da variável de interesse. Dessa forma, foi aplicada a técnica de geração de amostras sintéticas, o SMOTE (*Synthetic Minority Over-Sampling Technique*).

4.2 Modelagem

Foram utilizados diversos algoritmos de aprendizado de máquina para abordar diferentes estratégias. O Naive Bayes (NB) e o K-vizinhos mais próximos (KNN) foram selecionados como representantes probabilístico e não paramétrico, respectivamente. A

Árvore de Decisão, a Máquina de Vetores de Suporte (SVM), a Floresta Aleatória (RF) e o Adaboost foram escolhidos como algoritmos convencionais e compostos. Além disso, as redes LSTM foram incluídas devido à sua capacidade de reconhecimento de eventos raros e como uma representação de redes neurais.

Durante o estudo, foram exploradas diferentes abordagens para separação dos dados visando encontrar a estratégia mais adequada. Inicialmente, o embaralhamento foi utilizado para obter uma distribuição equilibrada das classes nos conjuntos, mas resultou em sobreajuste dos modelos. Por isso, optou-se por utilizar a ordem cronológica dos dados, considerando a evolução temporal e facilitando a generalização para diferentes cenários. Os dados foram divididos em 25% para teste e 75% para treinamento. No conjunto de treinamento, houve uma divisão adicional de 25% para validação e 75% para treinamento, permitindo a exploração de diferentes combinações de hiperparâmetros sem viés nos resultados de validação.

Na etapa de otimização dos hiperparâmetros, foi realizado um processo iterativo para encontrar a combinação ideal de hiperparâmetros para os modelos de classificação no conjunto de dados avaliado. Primeiramente, foi criado um espaço de busca, incluindo os hiperparâmetros e etapas de pré-processamento. Em seguida, a métrica de desempenho selecionada foi o F1-Score [Hosin et al, 2015]. Por fim, todas as iterações foram registradas utilizando a ferramenta MLFlow para extrair as métricas do experimento.

5. Avaliação

Foram registrados os melhores resultados, hiperparâmetros e pré-processamento utilizados para cada algoritmo. Os modelos foram treinados com o conjunto completo de dados de treinamento e avaliados no conjunto de testes. Cada algoritmo foi avaliado em 10 amostras diferentes contendo 50% dos dados do conjunto de teste, garantindo a replicabilidade das amostras com a mesma semente. Os resultados obtidos são apresentados na Tabela 1.

Tabela 1. Resultados Obtidos

Modelo	F1-Score	ROC-AUC
Adaboost	0.840 ± 0.011	0.926 ± 0.007
Árvore de Decisão	0.638 ± 0.020	0.757 ± 0.011
Floresta Aleatória	0.783 ± 0.018	0.831 ± 0.012
KNN	0.780 ± 0.017	0.843 ± 0.012
LSTM	0.826 ± 0.020	0.961 ± 0.025
Naïve Bayes	0.033 ± 0.001	0.866 ± 0.001
SVM	0.852 ± 0.012	0.924 ± 0.009

No geral, os três primeiros algoritmos (SVM, Adaboost e LSTM) tiveram resultados semelhantes, indicando que não há evidências estatisticamente significativas para provar a superioridade de um sobre o outro. Da mesma forma, os algoritmos Random Forest e KNN, que obtiveram o quarto e quinto melhores resultados, apresentaram resultados comparáveis em termos de desempenho, conforme a métrica F1-Score.

Devido à semelhança dos resultados dos melhores algoritmos, é relevante considerar outras métricas para a avaliação. Em relação à métrica ROC-AUC [Hosin et

al, 2015], o Adaboost e o SVM apresentam resultados próximos, ocupando a segunda posição. Em primeiro lugar na área sob a curva ROC está a rede LSTM, que se destaca pelo alto recall, indicando sua capacidade de diferenciar as classes e identificar a maioria das indisponibilidades. No entanto, é importante mencionar que o modelo LSTM gera um número ligeiramente maior de falsos positivos.

6. Conclusão

Este trabalho apresentou uma abordagem para detectar indisponibilidades em aplicações web, focando na experiência do usuário e visando reduzir o tempo de inatividade e aumentar a satisfação dos clientes. Diversos modelos de classificação foram desenvolvidos e avaliados, obtendo resultados promissores, com F1-Score próximo a 0,81 e ROC-AUC superior a 0,92. O modelo LSTM se destacou com um recall de 0,922, evidenciando a eficácia na identificação das indisponibilidades. Embora a abordagem não tenha realizado previsões de falhas ou identificado a causa raiz, a incorporação de dados da infraestrutura pode ajudar a abordar essas lacunas. O estudo oferece perspectivas importantes para a pesquisa em monitoramento e gerenciamento de aplicações web, melhorando a detecção e mitigação de indisponibilidades, resultando em uma experiência aprimorada para os clientes e benefícios para as empresas.

6. Referências

- R. F. Malaquias and Y. Hwang, Mobile banking use: A comparative study with Brazilian and US participants, *International Journal of Information Management*, vol. 44, pp. 132–140, 2019, Elsevier.
- Meirelles, F. S. Pesquisa anual do uso de TI. Fundação Getúlio Vargas, 2022.
- Afshan, S. and Sharif, A. Acceptance of mobile banking framework in Pakistan. *Telematics and Informatics*, vol. 33, no. 2, pp. 370–387, 2016. Elsevier.
- Du, M., Li, F., Zheng, G., Srikumar, V. Deeplog: Anomaly detection and diagnosis from system logs through deep learning. *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1285–1298, 2017.
- Zhang, X., Xu, Y., Lin, Q., Qiao, B., Zhang, H., Dang, Y., Xie, C., Yang, X., Cheng, Q., Li, Z., et al. Robust log-based anomaly detection on unstable log data. *Proceedings of the 2019 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, pp. 807–817, 2019.
- K. Zhang, J. Xu, M. R. Min, G. Jiang, K. Pelechris, and H. Zhang, “Automated IT system failure prediction: A deep learning approach,” in *2016 IEEE International Conference on Big Data (Big Data)*, IEEE, 2016, pp. 1291–1300.
- Gamalielsson, Jonas, Bjorn Lundell, Simon Butler, Christoffer Brax, Tomas Persson, Anders Mattsson, Tomas Gustavsson, Jonas Feist e Erik Lonroth: Towards open government through open source software for web analytics: The case of matomo. *JeDEM-eJournal of eDemocracy and Open Government*, 13(2):133–153, 2021.
- Hossin, Mohammad, and Md Nasir Sulaiman. "A review on evaluation metrics for data classification evaluations." *International journal of data mining & knowledge management process* 5.2 (2015): 1.