IV Workshop de Matemática, Estatística e Computação Aplicadas à Indústria

Detecção de URLs de *Phishing* com Aprendizado de Máquina: Abordagem Prática e Avaliação de Modelos

Murilo Couto de Oliveira[1], Eduardo Poleze[2], Roseli Aparecida Francelin Romero[3]
Instituto de Ciências Matemáticas e de Computação – ICMC, Universidade de São Paulo (USP)

1 Introdução

Ataques de *phishing* são uma das principais ameaças à segurança digital, explorando engenharia social e disfarce técnico para obter informações sensíveis. O aumento de fraudes baseadas em URLs falsas exige métodos automáticos para identificar e bloquear esses endereços em tempo real. Este trabalho apresenta uma abordagem baseada em aprendizado de máquina para detecção automática de URLs de *phishing*, com foco em desempenho e aplicabilidade prática. São comparados dez algoritmos supervisionados quanto à acurácia, precisão, *recall* e tempo de execução [1, 3, 10].

2 Base de Dados e Atributos

O estudo utiliza a base *PhiUSIIL Phishing URL* [7], contendo 235.795 registros e 54 atributos (31 quantitativos e 23 categóricos), incluindo:

- Da URL: número de subdomínios e caracteres ofuscados;
- Do HTML: número de scripts, imagens e referências externas;
- Derivados: índice de similaridade com domínios legítimos [6].

Foi verificada a presença de desbalanceamento entre classes legítimas e maliciosas, mas não foi necessário o uso de técnicas como *oversampling* ou *SMOTE*, visto que a distribuição estava próxima de 52/48, permitindo avaliação equilibrada dos modelos.

[1] murilo.couto-oliveira@usp.br
[2] eduardo.poleze@usp.br
[3] rafrance@icmc.usp.br

3 Metodologia

3.1 Pré-processamento

Foram eliminados atributos redundantes e observações extremas. Atributos categóricos foram convertidos para inteiros via codificação ordinal simples.

3.2 Seleção de Atributos

Os critérios aplicados foram: (i) correlação com a variável-alvo; (ii) importância dos atributos via *Random Forest* [2]; (iii) custo computacional no ambiente real. A análise demonstrou que variáveis relacionadas ao comportamento dinâmico da página, como chamadas externas e inserção de scripts, possuem maior poder discriminativo que características mais estáticas, como tamanho da URL. Tal resultado reforça a predominância de estratégias modernas de *phishing*, baseadas em manipulação visual e estrutura de carregamento, em vez de simples variações textuais.

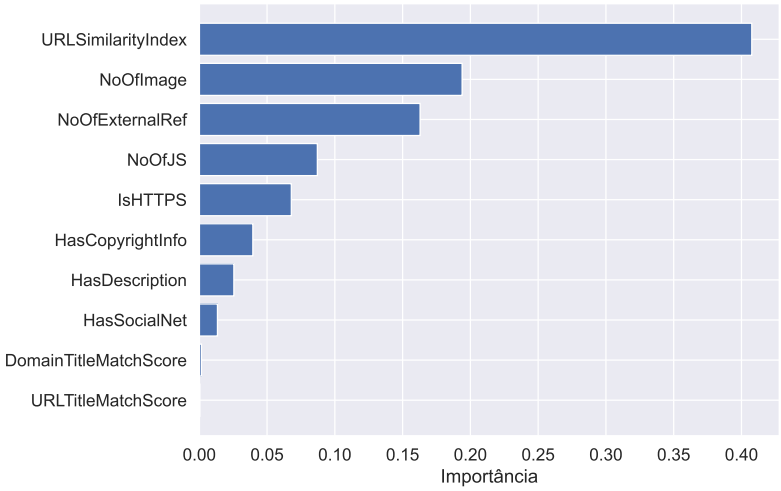

Figura 1: Importância dos atributos calculada por floresta aleatória.

3.3 Modelagem e Avaliação

Foram testados dez classificadores: Naive Bayes, KNN, Regressão Logística, Decision Tree (DT), Random Forest (RF), AdaBoost, SVM, LightGBM, XGBoost e MLP [4, 5]. A validação cruzada (k=10) foi utilizada no treino, e o *F1-score* foi a métrica principal por equilibrar precisão e *recall* [9]. Também foi considerado o tempo de inferência como métrica prática adicional, uma vez que modelos aplicados a filtros de tráfego devem operar em milissegundos, evitando impacto perceptível na navegação do usuário.

4 Resultados e Discussão

KNN, RF, XGBoost e DT obtiveram *F1-score* acima de 0,98, destacando-se DT pelo menor tempo de processamento. A Tabela 1 resume os resultados. O XGBoost apresentou o melhor desempenho

geral (F1=0,9847).

Tabela 1: Desempenho no conjunto de teste.

Modelo	Acurácia	Precisão	Recall	F1
DT	0.983	0.9879	0.976	0.9819
RF	0.984	0.9849	0.9826	0.9837
XGBoost	0.986	0.9831	0.9863	0.9847

Figura 2: Curva ROC e AUC para o modelo *XGBoost*.

Um teste t indicou diferença estatística significativa entre DT e XGBoost ($p < 0.05$). Com *Optuna* [8], a DT obteve pequena melhoria (F1=0,9831), reforçando que a qualidade dos atributos impacta mais que a otimização de hiperparâmetros. Observou-se ainda que o treinamento do XGBoost demandou 8,4 vezes mais tempo que a DT, evidenciando que o ganho marginal em desempenho deve ser considerado frente ao custo operacional em ambientes de grande escala.

5 Conclusões

Modelos clássicos de aprendizado de máquina demonstraram elevada precisão na detecção de URLs maliciosas com baixo custo computacional. A Decision Tree apresentou excelente equilíbrio entre desempenho e velocidade, permitindo inferência rápida e interpretabilidade direta. Embora existam modelos mais complexos, a eficácia de algoritmos simples pode dispensar soluções mais sofisticadas com maior tempo de processamento, reforçando a importância de considerar escalabilidade, consumo de recursos e explicabilidade em sistemas reais de segurança.

Referências

[1] ALMOWAIAD, A. Detecting phishing websites using ML. *Applied Sciences*, 2024.

[2] BREIMAN, L. Random forests. *Machine Learning*, v. 45, n. 1, p. 5–32, 2001.

[3] JAMES, G.; WITTEN, D.; HASTIE, T.; TIBSHIRANI, R. *An Introduction to Statistical Learning*. Springer, 2013.

[4] MOHAMMAD, R.; McCLUSKEY, L.; THABTAH, F. Predicting phishing websites. In: *International Conference on Innovations in Information Technology (ICIT)*, 2015.

[5] NUR, N. A systematic review on phishing detection. *Security and Communication Networks*, 2021.

[6] PRASAD, A.; CHANDRA, S. PhiUSIIL: a diverse security profile empowered phishing URL detection framework. *Computers & Security*, 2023.

[7] PRASAD, A.; CHANDRA, S. PhiUSIIL phishing URL dataset. *UCI Machine Learning Repository*, 2024.

[8] SNOEK, J.; LAROCHELLE, H.; ADAMS, R. Practical Bayesian optimization. In: *Advances in Neural Information Processing Systems (NeurIPS)*, 2012.

[9] SOKOLOVA, M.; LAPALME, G. Performance measures for classification tasks. *Information Processing & Management*, v. 45, n. 4, p. 427–437, 2009.

[10] VERMA, R.; DYER, J. On the characterization of phishing URLs. In: *eCrime Researchers Summit*, 2017.

[11] ZHANG, W. et al. Deep learning based phishing detection. *IEEE Access*, 2020.

[12] ABROSHAN, M. et al. Phishing detection via machine learning and deep learning. *Expert Systems with Applications*, v. 184, p. 115–152, 2021.

[13] BRITT, P.; CASTILLO, C. Detection of malicious URLs using deep learning techniques. *IEEE Security & Privacy*, 2019.

[14] MARCHAL, S.; ASOKAN, N. PhishStorm: Detecting phishing with streaming analytics. *IEEE Transactions on Network and Service Management*, v. 14, n. 3, p. 1011–1025, 2017.