Utilizando Aprendizado de Máquina para Detecção Automática de URLs Maliciosas Brasileiras

Lucas Dantas Gama Ayres[1], **Italo Valcy S Brito**[1], **Rodrigo Rocha Gomes e Souza**[1]

[1]Departamento de Ciência da Computação
Universidade Federal da Bahia (UFBA) – Bahia – BA – Brasil

{lucas.ayres,italovalcy,rodrigorgs}@ufba.br

Abstract. *Phishing is an attack that uses social engineering and other techniques to steal personal or financial information from victims. Brazil leads the statistics of users attacked by phishing and more than 77% of these attacks are carried out through URLs. Despite the existence of bases and techniques for detecting malicious URLs, they are not effective when it comes to URLs directed to Brazilian users, which have different characteristics. This paper presents an effective method of detecting malicious URLs based on machine learning. We used more than 110 features (lexical, network, reputation and others) and different classifiers to evaluate the effectiveness of the proposed method. The evaluation was carried out with real data extracted from the fraud catalog of the Brazilian academic network and other sources. Results demonstrate high precision and accuracy above 96%.*

Resumo. *Phishing é um ataque que usa engenharia social e outras técnicas para roubar informações pessoais ou financeiras das vítimas. O Brasil lidera as estatísticas de usuários atacados por phishing e mais de 77% desses ataques são realizados por meio de URLs. Apesar da existência de bases e técnicas para detecção de URLs maliciosas, elas são ineficazes quando se trata de URLs direcionadas aos usuários brasileiros, que possuem características diferenciadas. Este trabalho apresenta um método eficaz de detecção de URLs maliciosas brasileiras com base em aprendizado de máquina. Foram utilizadas mais de 110 características (léxicas, de rede, de reputação e outras) e diferentes classificadores na avaliação do método proposto. A avaliação foi realizada com dados reais extraídos do catálogo de fraudes da rede acadêmica brasileira e outras fontes. Resultados demonstram altas taxas de precisão e acurácia acima de 96%.*

1. Introdução

Phishing é um ataque que usa engenharia social e outras técnicas para roubar informações pessoais ou financeiras das vítimas [Vazhayil et al. 2018]. Os atacantes disseminam *phishing* de várias maneiras: janela *pop-up* no navegador web, mensagens instantâneas, e-mails e redes sociais [Olivo et al. 2010, Vazhayil et al. 2018]. Grande parte destes ataques são realizados a partir de URLs maliciosas; segundo estatísticas da Kaspersky Lab, 77% dos ataques de *phishing* em 2016 foram a partir de URLs, o que equivale a um total de 261.774.932 URLs únicas [Garnaeva et al. 2016]. O número de vítimas deste ataque vem crescendo ao longo dos anos e o Brasil está no topo da lista com usuários mais

afetados desde 2015 [Gudkova et al. 2017]: em 2017, cerca de 29.02% dos brasileiros sofreram com ataques de *phishing*.

Sistemas eficazes para detecção dessas URLs maliciosas em tempo hábil podem ajudar muito no combate a essas ameaças à segurança cibernética. Consequentemente, pesquisadores e profissionais trabalharam para projetar soluções efetivas para detecção de URLs maliciosas. Várias abordagens têm sido utilizadas para enfrentar o problema de detecção de URLs maliciosas. No geral, essas abordagens podem ser agrupadas em duas categorias: (i) *blacklist*, e (ii) abordagens de aprendizado de máquina [Canali et al. 2011, Eshete et al. 2012].

O método mais comum para detectar URLs maliciosas implantados por muitos sistemas (e.g. antivírus) é *blacklist*. *Blacklists* são sistemas que mantêm listas de URLs que foram analisadas e classificadas como maliciosas. Apesar da simplicidade e rapidez na consulta, o grande desafio do método *blacklist* é manter a lista de URLs maliciosas atualizada, especialmente porque novas URLs são geradas todos os dias.

Para evadir as *blacklists*, atacantes utilizam técnicas cada vez mais avançadas de composição e ofuscação da URL maliciosa [Garera et al. 2007], tornando-a específica para as vítimas alvos de uma campanha de *phishing*. Em particular, estudos anteriores mostram que apenas 9% das URLs catalogadas no Catálogo de URLs Maliciosas da rede acadêmica brasileira (CaUMa[1]) são também identificadas em bases de *blacklist* internacionais [Brito et al. 2015]. Além disso, URLs direcionadas à comunidade brasileira possuem características específicas [Brito et al. 2016]: nomes de marcas de empresas brasileiras contidos na URL, tamanho das URLs entre 50 e 100 caracteres, tempo de vida das URLs de *phishing* igual ou superior a 5 dias, dentre outras.

Outra maneira de detectar URLs maliciosas é a aplicação de técnicas de aprendizado de máquina [Vazhayil et al. 2018, Patil and Patil 2015, Olivo et al. 2010]. Nestes casos, geralmente utiliza-se um conjunto de URLs como dados de treinamento e uma função de predição para classificar, com base em propriedades estatísticas, uma URL como maliciosa ou benigna. Dessa forma, pode-se generalizar a detecção de novas URLs, o que não ocorre em métodos de *blacklist*.

Este trabalho propõe um método para detecção de URLs maliciosas direcionadas aos usuários brasileiros com base em técnicas de aprendizagem de máquina. Foram utilizadas mais de 110 características das URLs para o processo de classificação, com base em informações léxicas, da rede, reputação da URL e outras. Diferentes classificadores foram adotados para medir a eficácia do método proposto. O modelo foi avaliado em conjuntos de dados reais com 3.950 URLs de phishing e 3.162 URLs benignas. As principais contribuições deste trabalho são (i) um conjunto de características utilizadas na detecção de URLs maliciosas; (ii) *dataset* para análise da proposta e trabalhos futuros; (iii) análise das características e dos modelos treinados com diferentes bases e dos classificadores.

Este artigo está organizado da seguinte forma: a Seção 2 apresenta os trabalhos relacionados; a Seção 3 descreve a metodologia proposta; a Seção 4 avalia o desempenho da proposta; e, por fim, a Seção 5 apresenta as conclusões e trabalhos futuros.

[1]Serviço associado ao Catálogo de Fraudes da RNP. Site: https://cauma.pop-ba.rnp.br

2. Trabalhos Relacionados

Como crime de empregar meios técnicos para roubar informações sensíveis de usuários, o *phishing* é atualmente uma ameaça crítica à Internet, e os danos causados por *phishing* estão crescendo constantemente [Yang et al. 2019]. Nesse cenário, diversas pesquisas vem sendo conduzidas para apoiar o processo de detecção e mitigação das fraudes, principalmente a detecção baseada em URLs, que configura-se como o método mais comum de disseminação.

Existem diversas técnicas para detecção de URLs de *phishing* baseadas em *blacklist* ou aprendizado de máquina. No entanto, a grande maioria das propostas não possuem foco nas URLs maliciosas direcionadas ao público brasileiro, resultando em baixa eficácia na detecção e mitigação das campanhas de *phishing* direcionadas.

Vazhayil et. (2018) conduzem um estudo comparativo de métodos de aprendizado de máquina e aprendizado profundo clássicos como arquiteturas para detecção de URLs maliciosas de *phishing*. Os autores utilizaram quatro conjunto de dados para validação dos modelos: Phishtank, OpenPhish, MalwareDomainlist e MalwareDomains. Os resultados mostram que a técnica de Redes Neurais Convolucionais (CNN) combinadas com uma rede recorrente *Long Short-Term Memory* (LSTM) apresentam melhores resultados, com até 98% de acurácia.

Bezzera e Feitosa (2015) conduzem uma análise crítica da eficácia de características, bases e formatos das URLs nos algoritmos de aprendizado de máquina para classificação de URLs. Os autores propõem agrupamento e simplificação das características de URLs para melhorar o desempenho do classificador. Resultados mostraram que o classificador J48 (árvore de decisão) obteve melhor desempenho, com 95,10% de precisão e 95,11% de taxa de detecção.

Outros trabalhos combinam características baseadas no HTML e na URL [Ludl et al. 2007], alguns analisam adicionalmente o código JavaScript [Eshete et al. 2012], outros aplicam técnicas de aprendizado de máquina nas informações do host que hospeda a URL [Ma et al. 2009] ou tentam combinar informações de *blacklist* com características da URL [Xiang et al. 2010].

Além da pesquisa acima, também estão disponíveis ferramentas *anti-phishing* baseadas em diferentes técnicas, muitas das quais exploram *blacklists*. Produtos bem conhecidos incluem o Microsoft Internet Explorer e o Google Safe Browsing.

3. Metodologia

Nesta seção apresentamos a metodologia proposta para criar um método eficaz de detecção automática de URLs maliciosas que são direcionadas à comunidade brasileira.

3.1. Características

A maioria das características extraídas das URLs foi retirada a partir da análise de trabalhos relacionados, acrescida de algumas características extras com base na experiência de grupos de segurança que atuam no tratamento de *phishing*, o que permite analisar seu grau de importância na detecção das URLs maliciosas. As características foram categorizadas em *léxicas*, *de reputação*, *de rede* e *outras*, totalizando 117 características. A seguir é possível ver mais detalhes sobre cada característica.

3.1.1. Características Léxicas

Características léxicas são recursos obtidos com base nas propriedades do nome da URL. Essas características são extraídas através de tokens (símbolos ou palavras-chave) da URL e então é feito algum tipo de contabilização. A seguir estão descritas algumas das principais características léxicas que foram utilizadas.

1. **Quantidade de Tokens na URL, Domínio, Diretório, Arquivo e Parâmetros**. Os tokens considerados foram: ".", "-", "_", "/", "?", "=", "@", "&", "!", " ", " ", ",", "+", "*", "#", "$" e "%". Uma quantidade incomum de tokens pode indicar a presença de uma URL maliciosa.
2. **Comprimento da URL, Domínio, Diretório, Arquivo e Parâmetros**. Essa característica refere-se à medição dos segmentos de uma URL. Existem URLs que possuem uma grande quantidade de caracteres, que diverge do número de caracteres de URLs benignas, podendo indicar a presença de uma URL maliciosa.
3. **Domínio da URL em formato de endereço IP**. Verifica se o domínio da URL está no formato de endereço IP. Alguns ataques de *phishing* utilizam máquinas sem nenhuma entrada DNS, referenciando-as através do endereço IP.
4. **Presença de TLD (*Top Level Domain*) nos Parâmetros da URL**. URLs que incluem outra URL como parâmetro podem ser utilizadas como uma forma de ataque, que visa enganar o usuário redirecionando-o para páginas falsas. Um exemplo de URL utilizada para enganar o usuário e redirecionar para uma página falsa: http://site.tld/index.php?return=http://phishing.tld/malware.exe

Além das citadas acima, outras características léxicas foram utilizadas: quantidade de TLD (*Top Level Domain*) na URL, quantidade de vogais no domínio, quantidade de parâmetros, e-mail presente na URL, extensão de arquivo na URL, entre outras.

3.1.2. Características baseadas em Reputação

Para as características baseadas em reputação, foram utilizados três provedores diferentes de serviços de reputação: Google Safe Browsing, Phishtank e WoT[2]. Para cada um desses serviços foi necessário realizar uma consulta para verificar se a URL, IP ou domínio estavam listados nessas bases de dados.

3.1.3. Características baseadas em rede

As características baseadas em rede são obtidas das propriedades do nome do host da URL analisada. Com elas, pode-se conhecer a localização dos hosts maliciosos, a identidade, o estilo de gerenciamento e as propriedades desses hosts. A seguir estão descritas algumas das principais características baseadas em rede que foram utilizadas:

1. **Presença do domínio em RBL (*Realtime Blackhole List*)**. Essa característica tem como objetivo identificar se um domínio está presente em uma RBL, que são listas da Internet na qual vários serviços de antispam realizam consultas.

[2]WoT (Web of Trust) é um plugin de navegadores web que permite verificar se um website é seguro antes de acessá-lo a partir de uma base colaborativa construída com o feedback dos usuários.

2. **Localização geográfica do IP**. Tal como acontece com as propriedades do endereço IP, os servidores de domínio com atividades maliciosas podem ser concentrados em regiões geográficas específicas.
3. **Tempo (em dias) de ativação do domínio**. Sites de phishing são criados apenas com o intuito de roubar informações dos usuários, e para isso são criados domínios de curta duração [Ma et al. 2009]. Essa característica visa obter o tempo (em dias) em que o domínio está ativo. Quanto mais próxima da data em que um nome de domínio foi registrado, maior a possibilidade de ser um site de *phishing*.
4. **Tempo (em dias) de expiração do domínio**. Com base no fato de um site de *phishing* viver por um curto período de tempo, tipicamente os domínios confiáveis são regularmente pagos por vários anos de antecedência, então se a data de expiração do domínio for muito curta, é provável que seja um site de *phishing*.

Além das citadas acima, outras características baseadas em rede foram utilizadas: tempo de resposta de domínio (*lookup*), registros SPF, número AS (ou ASN), se o domínio possui registro PTR, número de IPs resolvidos, número de servidores de nome resolvidos (*NameServers* – NS), número de servidores MX, valor do *time-to-live* (TTL) associado ao domínio, dentre outras.

3.1.4. Outras Características

Neste grupo estão as características que não se enquadram em nenhuma das outras categorias, a saber:

1. **HTTPS (*HTTP sobre TLS/SSL*)**. A existência de HTTPS é muito importante para dar a impressão de legitimidade do site, mas isso não é claramente suficiente. O ideal é verificar o certificado atribuído com HTTPS, para saber se é um certificado válido. O software utilizado para a extração das características verifica tanto a existência de HTTPS quanto se o certificado é válido.
2. **Quantidade de redirecionamentos**. Saber quantas vezes um site foi redirecionado ajuda a distinguir sites de *phishing* de benignos. As análise realizadas mostraram que sites benignos foram redirecionados geralmente uma vez, ao passo que alguns sites de *phishing* foram redirecionados pelo menos três vezes.
3. **URL e domínio está indexada no Google**. Essa característica verifica se um site ou domínio está indexado no Google. Quando um site é indexado pelo Google, ele é exibido nos resultados de pesquisa. Normalmente, os sites de *phishing* são acessíveis por um curto período de tempo e, como resultado, não são indexados pelo Google.

3.2. Classificadores

Nesta seção apresentamos os classificadores que foram selecionados para realizar os experimentos, as métricas de avaliação e os ajustes realizados em cada classificador.

Foi utilizado o ambiente de experimentação Weka[3], na versão 3.6.14. O Weka possui uma coleção de algoritmos de aprendizado de máquina para tarefas de mineração de dados.

[3]Disponível em `https://www.cs.waikato.ac.nz/ml/weka/`

3.2.1. Classificadores selecionados

Os classificadores selecionados para realizar os experimentos foram: Naive Bayes, KNN, SVM e Árvore de Decisão (J48). Essa escolha foi feita com base em trabalhos anteriores [Olivo et al. 2010, Bezzera and Feitosa 2015], que mostram que esses classificadores são os mais utilizados e que possuem um melhor resultado quando se trata de detecção de *phishing*. A maioria das pesquisas de detecção de fraudes se baseia na estratégia de aprendizado supervisionado, pois gera melhores resultados além de permitir a criação de um modelo preditivo para identificação de futuras fraudes.

3.2.2. Métricas de Avaliação

Para a medição dos resultados, foi utilizada a técnica de validação cruzada (*cross-validation*), através do método *k-fold*, com 10 partições. O método de validação cruzada consiste em dividir o conjunto de dados em dez partes iguais e testar dez vezes, onde em cada teste uma parte é usada no conjunto de teste e as outras nove são usadas no conjunto de treinamento. Após a execução dos testes, o resultado das métricas de desempenho é uma média entre o resultado de todas as execuções. Assim mantém-se a mesma proporção em todos os experimentos a fim de permitir a comparação dos resultados obtidos.

As métricas utilizadas para a análise de desempenho foram: (i) *acurácia*: representa a taxa de amostras que foram classificadas corretamente; (ii) *revocação*: mede a proporção de amostras de URLs maliciosas que foram corretamente preditas como maliciosas; (iii) *precisão*: mede o número de amostras preditas como maliciosas que de fato são maliciosas; e (iv) *F1 Score*: representa a média harmônica entre precisão e revocação.

3.2.3. Ajustes dos Classificadores

Foram realizados alguns ajustes nos valores dos principais parâmetros de cada classificador, para poder obter um melhor resultado sobre o conjunto de dados nacional. O único classificador para o qual não foi possível realizar ajustes foi o Naive Bayes, que não possui parâmetros ajustáveis no ambiente Weka.

1. **Classificador baseado em Árvore de Decisão (J48)**: No classificador J48 foram realizados ajustes no parâmetro de *fator de confiança*, com o objetivo de analisar a precisão das regras geradas. Esse fator estabelece a confiança na base de treinamento e na avaliação de erro. Quanto menor seu valor, maior é a probabilidade de o nó ser podado em função dos nós estáveis, ou seja, menor será o tamanho da árvore e a quantidade de nós que poderiam levar a erros de classificação. O valor padrão do Fator de Confiança no Weka para o J48 é 0,25.
2. **Classificador KNN**: No classificador KNN foram realizados ajustes também no fator de confiança. O valor padrão do fator de confiança no Weka para o classificador KNN é 1,0.
3. **Classificador SVM**: Para o classificador SVM, foi ajustado o parâmetro de regularização ou penalização, denominado parâmetro C, que determina a rigidez do modelo em relação à tolerância a erros. Ao aumentar o valor desse parâmetro,

o modelo se torna mais rígido e preciso, porém fica mais custoso na fase de treinamento. Por outro lado, ao diminuir esse valor, o modelo fica mais tolerante a erros e menos rígido. O valor padrão no Weka para o parâmetro C no classificador SVM é 1,0.

Diferentes valores foram analisados para cada parâmetro em questão. Os resultados destas análises são apresentados e discutidos na Seção 4.2.

3.3. Base de URLs

Para realizar a avaliação da metodologia ora proposta, foram utilizadas URLs extraídas de três bases:

- **CaUMa - Base de URLs maliciosas nacionais**. O CaUMa armazena apenas URLs maliciosas direcionadas para o público brasileiro. Todas as URLs são analisadas manualmente antes de serem inseridas, tornando-se portanto uma base confiável. Nessa base foram coletadas somente as URLs de *phishing* e que ainda estavam online.
- **UFBA - Base de URLs benignas nacionais**. Foi disponibilizado um conjunto de URLs acessadas pela comunidade de usuários da UFBA que foram classificadas como benignas pelo sistema de filtragem web e prevenção de intrusos. Por questões de privacidade e anonimidade dos dados, não é possível revelar as redes que foram coletadas. Além disso, também foi realizada uma análise manual em cada uma dessas URLs para poder garantir que realmente eram URLs benignas.
- **F-Securify - Base de URLs benignas e maliciosas internacionais**. As URLs internacionais, tanto maliciosas quanto benignas, foram extraídas do repositório `https://github.com/faizann24/Using-machine-learning-to-detect-malicious-URLs`, que disponibiliza um arquivo com 420.465 URLs, sendo 74.452 maliciosas e 346.013 benignas. Esse repositório foi disponibilizado em um artigo sobre detecção de URLs maliciosas utilizando aprendizado de máquina, que foi apresentado pela empresa F-Securify [Ahmad 2016].

Para realizar o treinamento e ajuste dos parâmetros dos classificadores, foram utilizadas 7.112 URLs das bases nacionais, sendo 3.950 oriundas da base CaUMa e 3.162 da base UFBA. Já da base internacional, foram utilizadas 4.000 URLs maliciosas e 3.600 URLs benignas. A base de URLs internacionais foi composta a partir de um subconjunto do *dataset* original (F-Securify), considerando aquelas que ainda estavam online. A quantidade de URLs utilizadas tomou como base o total de URLs online disponíveis no Catálogo da RNP, aproximando os demais conjuntos deste valor. É importante notar que, diferentemente das URLs maliciosas, as URLs benignas não foram extraídas de mensagens de e-mail, e sim de dados de navegação, o que pode introduzir vieses nos resultados.

3.4. Construção do dataset

Para construir o *dataset*, foi desenvolvido um software em Python que possui um conjunto de bibliotecas que facilitam a extração das características. O funcionamento desse software consiste nas seguintes etapas:

1. O software recebe como entrada dois argumentos: arquivo de entrada com lista de URLs e arquivo de saída.

2. Cria-se o arquivo de saída e então preenche-se a primeira linha com 118 colunas que são separadas por vírgula. Essas colunas contêm o nome de cada característica e a última coluna indica se a URL é maliciosa ou não.
3. É iniciado um *loop* para obter cada URL do arquivo de entrada, extrair as 117 características e escrever no arquivo de saída nas colunas correspondentes. Existe uma função para extrair cada característica; essas características podem depender de conexão com a Internet ou não. É importante ressaltar que todos os vetores de características foram preenchidos com interrogação (?) quando informações não puderam ser extraídas da URL, o que ocorre somente para características que dependem de conexão e outros serviços.
4. Após o fim do *loop*, o *dataset* é gerado, já estruturado e no formato CSV (*comma-separated values*).

O software juntamente com o dataset estão públicos e podem ser encontrados em `https://github.com/lucasayres/url-feature-extractor`.

Após analisar o dataset, foi possível perceber que algumas características possuíam valores faltantes (*missings*); isso acontece devido a algumas características realmente não existirem para todas as URLs, como os parâmetros de URL, extensão do arquivo na URL, dados do WHOIS, etc. O tratamento destes casos é necessário para que os resultados sejam confiáveis. Para tratar os valores faltantes, foi necessário utilizar o método de imputação pela média ou moda, aplicando os valores da média para os atributos numéricos e valores da moda para os atributos do tipo nominal.

4. Resultados

Nesta seção, são apresentados os resultados obtidos através da execução dos classificadores conforme metodologia descrita na Seção 3.2.

4.1. Verificando o funcionamento de modelos treinados com bases internacionais para classificação de dados nacionais

Antes de iniciar os experimentos com as bases nacionais, foi realizado um experimento para verificar se os modelos treinados com bases internacionais funcionam bem quando testados com conjuntos de URLs nacionais. Dessa forma, é possível analisar se os métodos de detecção de URLs maliciosas que existem atualmente, focados em URLs internacionais, são eficazes quando usados com URLs nacionais.

Para isso foi utilizado o software Weka com os mesmos algoritmos e configurações supracitados. A fase de treinamento foi realizada com o conjunto de URLs internacionais, já a fase de classificação fez uso do *dataset* nacional. A Figura 1 mostra os resultados obtidos no teste em questão.

Ao analisar os resultados, percebe-se que modelos treinados com bases internacionais não funcionam bem quando são testados com bases nacionais. O desempenho dos modelos foi ruim, chegando a ter uma taxa de acurácia de no máximo 45,26%, utilizando o classificador J48, que foi o classificador que apresentou o melhor resultado.

4.2. Ajustes dos classificadores

Nesta seção são exibidas as análises e os resultados obtidos ao realizar os ajustes nos classificadores.

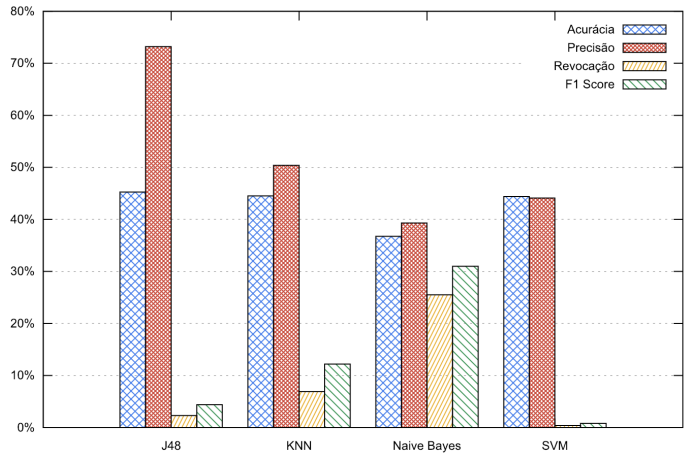

Figura 1. Resultado dos classificadores ao utilizar o modelo treinado na base internacional para testar com a base nacional

4.2.1. Classificador baseado em Árvore de Decisão (J48)

A Tabela 1 mostra o resultado do ajuste do fator de confiança que foi aplicado no classificador J48.

Tabela 1. Ajuste do fator de confiança para o classificador J48

Fator de Confiança	Acurácia	Precisão	Recall	F1 Score
0,001	94,37%	94,50%	95,50%	95,00%
0,01	94,83%	94,90%	95,90%	95,40%
0,1	95,19%	95,60%	95,70%	95,70%
0,25	95,47%	95,90%	95,90%	95,90%
0,5	95,55%	96,10%	95,90%	96,00%
1,0	95,55%	96,40%	95,60%	96,00%

Analisando a Tabela 1, podemos perceber que o melhor resultado foi o que teve o ajuste no Fator de Confiança de valor 1,0, que apresenta a acurácia (95,55%) igual a outro índice, mas possui a melhor precisão (96,40%).

4.2.2. Classificador KNN

A Tabela 2 mostra o resultado do ajuste do fator de confiança que foi aplicado no classificador KNN.

Analisando a Tabela 2 podemos dizer que o ajuste com melhor resultado foi o fator de confiança de valor 1, que conseguiu ficar com porcentagem maior em todas as métricas.

Tabela 2. Ajuste do fator de confiança para o classificador KNN

Fator de Confiança	Acurácia	Precisão	Recall	F1 Score
1	96,02%	96,30%	96,50%	96,40%
3	95,26%	95,50%	96,00%	95,70%
5	95,00%	95,10%	95,90%	95,50%
7	94,61%	94,50%	95,90%	95,20%

4.2.3. Classificador SVM

A Tabela 3 mostra o resultado do ajuste do parâmetro de regularização ou penalização que foi aplicado no classificador SVM.

Tabela 3. Ajuste do Parâmetro de regularização ou penalização para o Classificador SVM

Fator de Confiança	Acurácia	Precisão	Recall	F1 Score
0,001	81,62%	77,60%	94,10%	85,00%
0,01	90,86%	90,20%	93,80%	91,90%
0,5	93,84%	94,20%	94,80%	94,50%
1,0	94,17%	94,60%	94,90%	94,80%
3,0	94,36%	94,90%	94,90%	94,90%
5,0	94,55%	95,00%	95,20%	95,10%
10,0	94,55%	95,10%	95,10%	95,10%
50,0	94,61%	95,00%	95,30%	95,20%

Percebe-se na Tabela 3 que, com o aumento do parâmetro C, aumenta a porcentagem da acurácia e da precisão. O melhor resultado foi obtido ao utilizar o valor 50,0 para o parâmetro C.

4.3. Escolha do classificador

Após aplicados os ajustes dos parâmetros, foi possível realizar uma comparação entre os resultados obtidos pelos classificadores, para determinar qual classificador possui um melhor desempenho. A Tabela 4 mostra a comparação entre os classificadores J48, KNN, Naive Bayes e SVM:

Tabela 4. Comparação entre os classificadores J48, KNN, Naive Bayes e SVM

	J48	KNN	Naive Bayes	SVM
Acurácia	95,55%	96,02%	79,17%	94,61%
Precisão	96,40%	96,30%	74,40%	95,00%
Recall	95,60%	96,50%	95,40%	95,30%
F1 Score	96,00%	96,40%	83,60%	95,20%

Analisando o resultado da Tabela 4 foi possível constatar que o classificador Naive Bayes teve um desempenho inferior em relação aos demais, tendo uma taxa de acurácia

de 79,17% e taxa de precisão de 74,40%, mostrando que esse classificador possui uma capacidade baixa de aprendizado para esse conjunto de dados. O restante dos classificadores tiveram um desempenho bom e com valores próximos, mas o que se saiu melhor foi o classificador KNN, obtendo uma taxa de acurácia de 96,02% e 96,30% de taxa de precisão.

4.4. Análise das características

Nesta seção encontram-se os resultados obtidos a partir da análise feita em cada uma das características de forma separada, para poder enxergar o grau de importância de cada uma delas.

4.4.1. Características com maior poder preditivo

Para avaliar quais são as características mais relevantes, isto é, com maior poder preditivo, foi utilizada a métrica de avaliação de atributos *ReliefF* juntamente como o método de busca do tipo *Ranker*, ambos implementados pelo Weka. O *ReliefF* avalia o atributo individualmente de acordo com o seu valor para a classe majoritária entre múltiplas instâncias mais próximas do conjunto de dados fornecidos. Já a busca do tipo *Ranker* organiza os atributos em ordem decrescente de acordo com a relevância atribuída pelo *ReliefF*.

Após executar esses métodos no conjunto de dados, foi gerado um resultado trazendo as 10 características com maior poder preditivo, em ordem decrescente de importância, como é mostrado na Tabela 5.

Tabela 5. Características avaliadas individualmente de acordo com o critério *Relief* e classificadas pelo *Ranker*

Peso	Característica
0.38	tempo de ativação do domínio
0.355	valor ttl associado
0.277	tempo de resposta
0.264	comprimento da url
0.239	tempo de expiração do domínio
0.238	comprimento do arquivo
0.226	quantidade de redirecionamentos
0.214	extensão do arquivo
0.211	comprimento dos parâmetros
0.21	quantidade de parâmetros

A Tabela 5 mostra que a característica “tempo de ativação do domínio” (característica numérica, que possui os valores representados em dias) foi a que conseguiu se sair melhor no conjunto de dados.

As características que se saíram melhor são as que foram obtidas das propriedades do nome do host da URL, como tempo de ativação do domínio, valor TTL associado e tempo de resposta, mostrando a importância dessa categoria na detecção das URLs maliciosas.

4.4.2. Distribuição geográfica de *phishings*

Para poder saber quais os países que mais hospedam as páginas de *phishing*, foi necessário carregar o dataset no WEKA para melhor visualizar os valores e tipos de cada característica de forma separada. Na Tabela 6 são exibidos os 10 países que mais hospedam páginas de *phishing*. Essa Tabela mostra que algumas áreas são altamente associadas à atividade de *phishing*. Enquanto os Estados Unidos abrigam o maior número de páginas de *phishing*, o Brasil vem em segundo lugar e em seguida os países do centro e sul europeu.

Tabela 6. Países que hospedam a maioria das páginas de *phishing*

País	URLs de Phishing	URLs Benignas	% Phishing
Estados Unidos	2518	1611	63,74%
Brasil	358	1288	9,06%
Canadá	189	65	4,78%
Itália	88	2	2,22%
Alemanha	82	13	2,07%
Holanda	73	6	1,84%
França	61	22	1,54%
Rússia	48	5	1,21%
Reino Unido	46	24	1,16%
Polônia	36	2	0,91%

4.5. Discussão

Através desses experimentos, a análise dos resultados demonstra que o método proposto é capaz de apoiar o processo de detecção automatizada de URLs maliciosas direcionadas à comunidade brasileira. De acordo com a Tabela 4, o classificador KNN mantém uma alta acurácia e precisão, com o conjunto de dados utilizado. Comparado com outros trabalhos bem sucedidos [Bezzera and Feitosa 2015, Basnet et al. 2014], o método mostra um desempenho com taxas de precisão e acurácia similares ou superiores à maioria desses trabalhos. Destaca-se que essa comparação está sendo feita com trabalhos que utilizam bases internacionais, pois não foi encontrado nenhum trabalho que utilize algum tipo de base de URLs nacionais.

O resultado da tabela 4 sugere que é possível gerar um modelo eficaz para o cenário nacional, usando um conjunto de dados limitado. Nos experimentos realizados, os conjuntos de dados para as avaliações são subamostrados aleatoriamente para simular as diferentes características. Como esse conjunto de dados foi coletado a partir dos dados reais de *phishing*, e contendo diferentes características presentes nas URLs, os resultados podem refletir um cenário *anti-phishing* mais próximo da realidade nacional.

Nos resultados obtidos, algumas URLs de *phishing* foram classificadas como benignas. Isso ocorreu pois algumas estavam bem camufladas, estavam online por um longo período de tempo, hospedadas gratuitamente em serviços de hospedagem legítimos, sem possuir palavras-chave relacionadas a *phishing* e em alguns casos, essas URLs estavam

em resultados de pesquisa do Google. Acredita-se que, olhando e analisando o conteúdo da página, alguns desses falso-negativos podem ser eliminados.

É necessário que as características utilizadas no treinamento sejam sempre revistas e atualizadas, pois os atacantes estão criando páginas de *phishing* cada vez mais difíceis de serem detectadas, utilizando novos recursos de camuflagem, como encurtamento de URLs, domínios com maior tempo de vida, aplicando SEO (*search engine optimization*) nas páginas para poder ser melhor rankeado nas buscas do Google, entre outros métodos.

Como o foco desse trabalho é na URL, esse modelo pode ser aplicado em qualquer lugar em que uma URL possa ser incorporada, como no e-mail, páginas da web, chat, etc. Pode ser desenvolvido por exemplo, um software de detecção de URLs maliciosas, aplicando as regras geradas pela árvore de decisão do algoritmo J48.

5. Conclusão e Trabalhos Futuros

Para alcançar o objetivo proposto neste trabalho, o modelo foi avaliado em conjuntos de dados nacionais reais, comparando os resultados de desempenho dos classificadores J48, KNN, Naive Bayes e SVM. Os resultados obtidos nos experimentos mostraram que a solução *anti-phishing* proposta foi capaz de detectar URLs de *phishing* com uma acurácia e precisão de mais de 96%.

Em trabalhos futuros pretende-se: (i) Realizar estudos para identificar novas características que contribuam para a detecção de *phishing*; (ii) Uso de novas bases de dados, em particular considerando URLs benignas presentes no corpo de e-mails, inclusive e-mails fraudulentos, tendo em vista que os atacantes podem misturar URLs benignas e malignas em um phishing para subverter as filtragens; (iii) Uso de outros classificadores para obtenção de novos resultados; (iv) Analisar desempenho de cada classificador; (v) Realizar análises complementares considerando processamento, tempo de execução e outros fatores em uma implantação de produção; (vi) Disparar automaticamente e-mail para a instituição relacionada com a fraude, alertando-a.

Agradecimentos

Os autores gostariam de agradecer o apoio financeiro do MCTIC e da UFBA, por meio do edital PROPCI/PROPG – UFBA 004/2016.

Referências

Ahmad, F. (2016). Using machine learning to detect malicious urls. Acesso em 28 de nov. 2018.

Basnet, R. B., Sung, A. H., and Liu, Q. (2014). Learning to detect phishing URLs. In *International Journal of Research in Engineering and Technology, IJRET*, volume 3, pages 11–24.

Bezzera, M. and Feitosa, E. (2015). Investigando o uso de Características na Detecção de URLs Maliciosas. In *XV Simpósio em Segurança da Informação e de Sistemas Computacionais, SBSeg 2015*, pages 100–113, Florianópolis, SC.

Brito, I., Borges, J. L., Ayres, L., Tavares, P., Bastos, R., Lima, E., and Solha, L. V. (2015). Catálogo de fraudes da rnp: 7 anos de experiência no tratamento de fraudes eletrônicas brasileiras. In *2015 Conferência Integrada ICCyber ICMedia*, pages 1–5, Brasília, DF.

Brito, I., Borges, J. L., Tavares, P., Bastos, R., Lima, E., and Solha, L. V. (2016). Catálogo de fraudes e catálogo de urls maliciosas: Identificação e combate a fraudes eletrônicas na rede acadêmica brasileira. In *Sexta Conferencia de Directores de Tecnologia de Información, TICAL 2016*, pages 1–16, Buenos Aires, AR.

Canali, D., Cova, M., Vigna, G., and Kruegel, C. (2011). Prophiler: a fast filter for the large-scale detection of malicious web pages. In *20th international conference on World wide web*, page 197–206, India.

Eshete, B., Villafiorita, A., and Weldemariam, K. (2012). Binspect: Holistic analysis and detection of malicious web pages. In *International Conference on Security and Privacy in Communication Systems*, pages 149–166, Springer, Berlin, Heidelberg.

Garera, S., Provos, N., Chew, M., and Rubin, A. D. (2007). A framework for detection and measurement of phishing attacks. In *Proceedings of the 2007 ACM workshop on Recurring malcode*, pages 1–8.

Garnaeva, M., Sinitsyn, F., Namestnikov, Y., Makrushin, D., and Liskin, A. (2016). Kaspersky Security Bulletin: OVERALL STATISTICS FOR 2016. URL: https://goo.gl/sJvhGG (último acesso 23/12/2018).

Gudkova, D., Vergelis, M., Shcherbakova, T., and Demidova, N. (2017). Kaspersky Lab: Spam and phishing report in 2017. URL: https://securelist.com/spam-and-phishing-in-2017/83833/ (último acesso 24/10/2018).

Ludl, C., Mcallister, S., Kirda, E., and Kruegel, C. (2007). On the effectiveness os techniques to detect phishing sites. In *International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*, pages 20–39, Springer, Berlin, Heidelberg.

Ma, J., Saul, L. K., Savage, S., and Voelker, G. M. (2009). Beyond blacklists: learning to detect malicious web sites from suspicious urls. In *15th ACM SIGKDD international conference on Knowledge discovery and data mining*, page 1245–1254.

Olivo, C. K., Santin, A. O., and Oliveira, L. (2010). Avaliação de Características para Detecção de Phishing de E-mail. In *Pontifícia Universidade Católica do Paraná*, pages 1–2, Curitiba, PR.

Patil, D. R. and Patil, J. (2015). Survey on malicious web pages detection techniques. In *International Journal of u-and e-Service, Science and Technology*, pages 195–206.

Vazhayil, A., Vinayakumar, R., and Soman, K. (2018). Comparative study of the detection of malicious urls using shallow and deep networks. In *2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, pages 1–6. IEEE.

Xiang, G., Pendleton, B. A., Hong, J. I., and Rose, C. P. (2010). A hierarchical adaptive probabilistic approach for zero hour phish detection. In *15th European Symposium on Research in Computer Security*, page 268–285.

Yang, P., Zhao, G., and Zeng, P. (2019). Phishing website detection based on multidimensional features driven by deep learning. In *IEEE*, pages 1–14.