UNIVERSIDADE DE SÃO PAULO
Instituto de Ciências Matemáticas e de Computação

Filtros de alta precisão e performance para mensagens de SMS fraudulentas, com base em redes neurais

Cristiano Mendes Franco

Monografia - MBA em Inteligência Artificial e Big Data

ICMC USP
SÃO CARLOS

Cristiano Mendes Franco

Filtros de alta precisão e performance para mensagens de SMS fraudulentas, com base em redes neurais

Monografia apresentada ao Departamento de Ciências de Computação do Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo - ICMC/USP, como parte dos requisitos para obtenção do título de Especialista em Inteligência Artificial e Big Data.

Área de concentração: Inteligência Artificial

Orientador: Prof. Diego Furtado Silva

Versão original

São Carlos

2024

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi, ICMC/USP, com os dados fornecidos pelo(a) autor(a)

S856m	Franco, Cristiano Filtros de alta precisão e performance para mensagens de SMS fraudulentas, com base em redes neurais / Cristiano Mendes Franco ; orientador Diego Furtado Silva. – São Carlos, 2024. 74 p. : il. (algumas color.) ; 30 cm. Monografia (MBA em Inteligência Artificial e Big Data) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, 2024. 1. LaTeX. 2. abnTeX. 3. Classe USPSC. 4. Editoração de texto. 5. Normalização da documentação. 6. Tese. 7. Dissertação. 8. Documentos (elaboração). 9. Documentos eletrônicos. I. Silva, Diego Furtado, orient. II. Título.

Cristiano Mendes Franco

High precision and performance filters for fraudulent SMS messages based on Neural Networks

Monograph presented to the Departamento de Ciências de Computação do Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo - ICMC/USP, as part of the requirements for obtaining the title of Specialist in Artificial Intelligence and Big Data.

Concentration area: Artificial Intelligence

Advisor: Prof. Diego Furtado Silva

Original version

São Carlos

2024

Dedico este trabalho à minha esposa, Fabiola Villela Mamede,
cujo incentivo e parceria de vida sempre foram fundamentais não apenas para a realização deste projeto, mas em toda minha jornada profissional.
Sua força e apoio, sempre nos momentos mais difíceis, são indispensáveis.

AGRADECIMENTOS

Agradeço meus pais, Rômulo e Elisa, que desde minha infância me ensinaram o valor inestimável do estudo e do esforço como pilares para conhecimento e o crescimento pessoal. Sempre foram grandes estimuladores da minha curiosidade e nunca mediram esforços para me oferecerem oportunidade de avançar nos estudos, inclusive fora da minha cidade natal. Sem suas orientações e exemplo, não chegaria onde estou hoje profissionalmente.

Obrigado também a empresa Zenvia que me acolheu desde 2021, a Cassio Bobsin, um grande questionador e empreendedor desde adolescência, e a todos profissionais do time da área de infra-estrutura e segurança da informação da Engenharia, que disponibilizaram além de conselhos, incentivo e a base de dados anonimizada, crucial para análise de mensagens de fraude, objeto deste TCC.

Ao Professor Diego Furtado Silva expresso meu agradecimento especial, que além de ser um ótimo orientador, foi na verdade um mentor ao longo do desenvolvimento do meu trabalho. Sua mentoria não me trouxe tão somente o conhecimento técnico, mas também contribuiu para que pensasse estrategicamente, permitindo que eu superasse o desafio de tempo escasso para a conclusão do curso de MBA.

Gostaria de agradecer ao Instituto de Ciências Matemáticas e de Computação de São Carlos da USP, em nome da Professora e Coordenadora do curso, Solange Rezende, não somente pela oportunidade de participar da turma do MBA, mas por despertar em mim o interesse em aprofundar meus estudos sobre Inteligência Artificial, em especial aos grandes modelos de linguagem (LLMs). A instituição me incentivou a refletir sobre como aplicar esses conhecimentos de forma prática e empreendedora, visando gerar um impacto na produtividade das empresas, indo além do trabalho aqui apresentado.

"Eu falhei muitas vezes na vida,
e é por isso que eu consegui."
Michael Jordan

RESUMO

FRANCO, C. **Filtros de alta precisão e performance para mensagens de SMS fraudulentas, com base em redes neurais**. 2024. 74 p. Monografia (MBA em Inteligência Artificial e Big Data) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2024.

Este trabalho visa o desenvolvimento de filtros de alta precisão e alta performance para mensagens SMS fraudulentas, baseados em redes neurais e aprendizado de máquina. Diante do aumento de fraudes conhecidas como smishing, o objetivo é identificar mensagens fraudulentas ou spam com elevada precisão e baixo custo computacional. Foi utilizada uma base de dados com quase um milhão de mensagens previamente classificadas, e abordagens supervisionadas e semi-supervisionadas, à partir da extração das características das URLs e processamento de texto para treinamento dos modelos de classificação. Foram testados e comparados os seguintes modelos: Random Forest, XGBoost, LightGBM, K-Nearest Neighbors (KNN), MLP Classifier (um tipo de rede neural simples com camada densa ampla), LSTM e Stacking (uma técnica que combina múltiplos modelos para melhorar a performance geral das métricas à partir dos modelos individuais). As métricas do experimento demonstraram resultados que atestam a eficácia dos modelos avaliados na detecção de mensagens fraudulentas, com significativa redução de falsos positivos e contribuição para melhorar a segurança nas comunicações móveis.

Palavras-chave: SMS. Redes Neurais Profundas. Aprendizado Semi-supervisionado. Detecção de Span. Smishing. Modelo de detecção de fraude. Filtro de Mensagens. Trabalho de conclusão de curso (TCC).

ABSTRACT

FRANCO, C. **High precision and performance filters for fraudulent SMS messages, based on Neural Networks**. 2024. 74 p. Monograph (MBA in Artificial Intelligence and Big Data) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2024.

This work aims to develop high precision and high performance filters for fraudulent SMS messages, based on neural networks and machine learning. Given the increase in fraud known as smishing, the goal is to identify fraudulent or spam messages with high accuracy and low computational cost. A dataset with almost one million previously classified messages was used, and supervised and semi-supervised approaches were applied, starting with URL feature extraction and text processing for training classification models. The following models were tested and compared: Random Forest, XGBoost, LightGBM, K-Nearest Neighbors (KNN), MLP Classifier (a simple neural network type with a wide dense layer), LSTM, and Stacking (a technique that combines multiple models to improve overall metric performance from individual models). The experiment's metrics demonstrated results that attest to the effectiveness of the evaluated models in detecting fraudulent messages, with a significant reduction in false positives and contribution to improving security in mobile communications.

Keywords: SMS, Deep Neural Networks, Semi-supervised Learning, Spam Detection, Smishing, Fraud Detection Model, Message filter, Course Completion Work.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

2FA	Two Factor Authentication ou Fator duplo de autenticação
A2P	Application to Person communication
ABNT	Associação Brasileira de Normas Técnicas
ANATEL	Agência Nacional de Telecomunicações
BoW	Bag of Words (Saco de Palavras)
CAGR	Compound Annual Growth Rate
CPaaS	Communication Platform as a Service
CSV	Comma-Separated Values (Arquivos de dados "normalmente"separados por vírgulas
EUA	Estados Unidos da América
GSM	Global System for Mobile Communications
IA	Inteligência Artificial
LA	Logical Address
LLM	Large Language Models
LSTM	Long Short-Term Memory
ML	Machine Learning
MLP	Multilayer Perceptron
OC-SVM	One Class Support Vector Machine
P2P	Person to Person communication
PCA	Principal Component Analisys
RCS	Rich Communications Service ou Serviço de Comunicações Ricas
SDCCH	Stand-alone Dedicated Control Channel
SMS	Short Message Service ou Serviço de Mensagens Curtas
SMSC	Short Message Service Center ou Centro de Serviços de Mensagens Curtas

SVM	Support Vector Machine
URL	Uniform Resource Locator
USP	Universidade de São Paulo
USPSC	Campus USP de São Carlos

SUMÁRIO

1 CONTEXTUALIZAÇÃO

Com o avanço da comunicação móvel, o SMS (*Short Message Service* ou Serviço de Mensagens Curtas) tornou-se uma das formas mais populares de comunicação rápida e conveniente para diversas aplicações empresariais, como cobranças, transações, campanhas promocionais, lembretes e envio de códigos de segurança (ZENVIA, 2023). Apesar do surgimento de novos aplicativos de mensagens instantâneas, como o WhatsApp, Messenger, RCS, Telegram, dentre outros, estima-se que a tecnologia irá gerar mais de cinquenta bilhões de dólares no mundo (DIGITAL, 2022) e no Brasil mais de 5 bilhões de mensagens SMS ainda são trocadas mensalmente entre grandes empresas de diversos setores, como financeiro, varejo e telecomunicações, e seus clientes, devido ao seu baixo custo e independência de conexão à internet (Abayomi-Alli *et al.*, 2019).

Tamanha popularidade trouxe consigo desafios inéditos, principalmente o aumento de mensagens fraudulentas, conhecidas como *phishing* ou *'smishing'*. As fraudes buscam enganar os destinatários para obter informações confidenciais ou induzi-los a ações prejudiciais, e representam uma ameaça crescente à segurança e privacidade. Com um impacto financeiro superior a seiscentos milhões de dólares anualmente e um aumento de 61% nos ataques desde 2021 (SINCH, 2023), o *'smishing'* tornou-se uma preocupação global, exigindo a atenção de órgãos reguladores no Brasil. A Agência Nacional de Telecomunicações (ANATEL) tem buscado operadoras, brokers e outras entidades prestadoras de serviço de telecomunicações para regulamentar a utilização de campanhas que utilizam SMS e telefones fixos, como 0800, até que existam outras medidas que promovam a detecção eficaz das fraudes e que protejam os usuários de possíveis violações de segurança (TECNOBLOG, 2024).

Apesar do crescimento da aplicação de tecnologias de Inteligência Artificial (IA) para combate a fraudes, ainda existem lacunas de estudo importantes no treinamento e aplicação de filtros de mensagens de *smishing*, seja por meio de sistema de detecção nos aparelhos celulares (*client-side*), ou seja na rede das operadoras e prestadores de serviços (*server-side*). A revisão das técnicas de classificação de spam em SMS (Abayomi-Alli *et al.*, 2019) destacam a necessidade de pesquisas adicionais, especialmente na utilização de conjuntos de dados desatualizados e desbalanceados, assim como na exploração do uso do aprendizado profundo em combinação com algoritmos de otimização, visando minimizar a taxa de falsos positivos e reduzir a complexidade computacional: consumo de memória e velocidade na detecção. A inexistência de filtros bem implementados nas operadoras de telecomunicações e nos aplicativos de SMS são fatores que contribuem para o crescimento do spam (Cossa *et al.*, 2021).

A Zenvia, maior empresa de CPaaS (Communications Platform as a Service) do

Brasil, e onde trabalho atualmente, desempenha um papel fundamental no ecossistema de SMS, oferecendo soluções de comunicação que conectam empresas e clientes em larga escala há mais de 20 anos. Com bilhões de mensagens trafegadas mensalmente, é fundamental garantir a segurança das comunicações na plataforma da Zenvia, especialmente em relação ao combate a mensagens fraudulentas e que possam representar alto risco a operação dos clientes.

Este trabalho explora as estratégias e os desafios enfrentados na luta contra o smishing, tipos de fraudes mais comuns, suas características e a implementação de medidas eficazes para detectar SMS fraudulentos. O objetivo é criar um filtro de SMS de alta precisão e baixo custo computacional, explorando algoritmos de aprendizado profundo e considerando base de dados com e sem mensagens spams, gerando maior acuracidade no bloqueio de disparos massivos que podem prejudicar usuários e melhorando a performance de entrega de campanhas para clientes idôneos.

Trabalhos anteriores mostram que o uso de aprendizado supervisionado é a tendência mais popular na detecção de spam no contexto de SMS (Yerima; Bashar, 2022). Diferentemente de estudos anteriores, a proposta é utilizar uma abordagem de comparar filtros de mensagens utilizando de técnicas supervisionadas com dados reais do mercado brasileiro e semi-supervisionadas. Além disso, é utilizado sistemas de IA baseados em LLMs, como ChatGPT, durante o processo de pre-processamento do conjunto de dados, para extrair características das mensagens.

O treinamento semi-supervisionado é realizado apenas em uma classe de dados, ou seja, mensagens normais de SMS não spam são usadas para construir um modelo de classificação utilizando, por exemplo, One Class SVM (OC-SVM) (Yerima; Bashar, 2022). A vantagem deste último método é treinar o algoritmo com mensagens não spam, eliminando assim a necessidade de dados rotulados e problemas com conjuntos de dados desequilibrados durante a aprendizagem.

Para abordagem supervisionada, dispomos na Zenvia de uma base de dados histórica com bilhões de mensagens de SMS, incluindo substancial conjunto de casos identificados de fraudes. Este acervo se apresenta como uma oportunidade única para contribuir significativamente na criação e avaliação de filtros classificadores supervisionados de SMS. Assim, este trabalho de conclusão de curso contribui na prática para a melhoria contínua na segurança da comunicação via SMS.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Comunicação via SMS

2.1.1 Breve história do SMS como criador da era da mensageria

Neil Papworth (Papworth, 2012), um engenheiro britânico e arquiteto de software, enviou a primeira mensagem de texto em 3 de dezembro de 1992, de seu computador em um laboratório de pesquisa e desenvolvimento para o diretor da Vodafone, Richard Jarvis. Richard recebeu a mensagem em um celular Orbitel 901 que pesa mais de 2 quilos, durante a festa de fim de ano da empresa. A mensagem dizia “Feliz Natal” em inglês e representou o início do serviço de SMS. (BBC, 2022)

Na década de noventa, os celulares não possuíam teclado, e por este motivo o britânico digitou o texto pelo seu computador. A difusão do SMS ocorreu com a rede de segunda geração móvel padronizada pelo GSM (Global System for Mobile Communications), criada na Europa. Diferente das tecnologias anteriores, sua comunicação era digital, não mais analógica. Os primeiros dispositivos GSM que conseguiam enviar mensagens foram introduzidos pela Nokia a partir de 1993. A fabricante foi pioneira em uma linha de celulares capazes de mandar mensagens, introduzindo duas ou mais letras associadas aos números dos teclados.

No Brasil, o serviço surgiu em 1999, quando a BCP Telecomunicações lançou o produto "Digimemo"(Teleco, 2002). A BCP foi adquirida pelo grupo mexicano América Móvel e pertence atualmente à operadora Claro.

Quando o primeiro iphone da empresa Apple foi lançado em 2007, os americanos enviavam mais SMS do que realizavam ligações telefônicas (Gayomali, 2015), marcando uma transformação cultural que veio ao seu ápice com a chegada das redes sociais e dos novos canais digitais, como Messenger e Whatsapp. Pela sua abrangência de usuários, todos estes novos canais das grandes empresas de tecnologia ainda utilizam o SMS como ferramenta de autenticação de segurança (2FA). O SMS segue mais popular do que o Whatsapp nos EUA. (Digital, 2022)

Em 2010, os usuários de telefonia celular já trocavam dezenas de bilhões de mensagens todos os anos, e a palavra “texting” entrou para o dicionário. No Brasil, o serviço ficou mais conhecido como "Torpedo". A primeira mensagem enviada por Neil foi vendida na forma de NFT (Jornal, 2021) por 693 mil reais em 2021.

O crescimento do comércio eletrônico e a pandemia mundial em março de 2020 promoveram um aumento exponencialmente na necessidade de comunicação por mensagens instantâneas . Apesar de ter perdido popularidade entre os consumidores, o SMS continua

muito ativo nas grandes empresas de diversos segmentos, que passaram a incorporar estes canais em suas estratégias de vendas, atendimento e suporte.

A transformação tecnológica dos aparelhos celulares tornando-os indispensáveis na vida cotidiana, o crescimento da internet, e o surgimento da omnicanalidade trouxe aos consumidores uma infinidade de opções de comunicação para se relacionarem entre si e com as empresas. Com mais de 30 anos de existência, o SMS possui uma idade invejável já que atravessa essas transformações tecnológicas e culturais (Exame, 2022), sendo esta última a variável mais importante em que podemos afirmar que vivemos na "era conversacional", com bilhões de mensagens de negócios sendo trocadas diariamente, incluindo as interações entre seres-humanos e as máquinas (chatbots) que utilizam da Inteligência Artificial.

2.1.2 Característica do Serviço de Mensagens Curtas

A principal característica do SMS é sua limitação de tamanho de cada mensagem enviada, que é restrita a 160 caracteres alfanuméricos. Inicialmente, Friedhelm Hillebrand e Bernard Ghillebaert , pioneiros do GSM, criaram os conceitos do serviço em 1984 (Wikipedia, 2017). Os engenheiros que influenciaram na padronização dos serviços móveis, realizaram experimentos práticos para determinar um comprimento adequado para mensagens de texto, uma vez que os recursos e a disponibilidade de banda em telecomunicações são escassos. Avaliaram o comprimento médio das sentenças e perguntas em diferentes tipos de comunicação, e constataram que a maioria ficava abaixo de 160 caracteres, sugerindo que este seria um tamanho suficiente para comunicar a grande maioria das mensagens de forma concisa.

Esta limitação também decorre das condições originais de sua criação, na qual as mensagens são transmitidas nos sinais de controle, o que é conhecido como canais de sinalização como o SDCCH (Stand-alone Dedicated Control Channel) no GSM. Este canal não ocupa a mesma faixa de frequência dedicada às chamadas de voz, o que permite que as mensagens de texto sejam enviadas e recebidas durante chamadas telefônicas sem interferência, mantendo a integridade e a confiabilidade do serviço de mensagens.

O SMS suporta a codificação de mensagens usando formatos de compressão que permitem encaixar até 160 caracteres alfanuméricos em 140 bytes de dados. Isso é possível através do uso do padrão de codificação GSM 7-bit (Wikipedia, 2009). Quando uma mensagem não pode ser codificada em GSM-7, o padrão UCS-2 (ou mais recentemente UTF-16) (Wikipedia, 2002) que possui um comprimento fixo de 16 bits (2 bytes) é usado como alternativa, especialmente para línguas que requerem mais de 128 caracteres para serem representadas, como é o caso do Mandarim.

As plataformas tecnológicas de disparo de mensagens, como no caso da Zenvia, buscam enviar automaticamente as mensagens no sistema de codificação mais compacto possível. Quando o conteúdo de uma mensagem inclui caracteres que não podem ser repre-

sentados no GSM-7, o sistema automaticamente converte para codificação em UCS-2, o que acaba por limitar o tamanho do conteúdo da mensagem para 70 caracteres cada (70 bytes), uma vez que o sistema utiliza 2 bytes para codificação de cada caractere. Nas aplicações do Brasil, isso acontece bastante quando os clientes utilizam de acentuações características do português, tais como: cedilha (“ç”) e acentuações (“ã”, “á”,”ó”). Consequentemente é muito comum recebermos mensagens de SMS sem acentuações para evitar a quebra envio de mensagens, em duas ou mais mensagens para compor um mesmo texto, dobrando o custo de uma campanha. Este processo de envio de mensagens acima de 160 caracteres, ou com codificação especial, é chamado de concatenação de SMS. Portanto, para fazer uso efetivo desse canal, é preciso atentar para alguns pontos como: o tipo ideal de texto para SMS, os possíveis casos de uso, e boas práticas para construção das mensagens.(Zenvia, 2023)

Outra característica importante do SMS é a sua capacidade de interoperação entre redes de diferentes operadoras. Seu design foi concebido para operar em diversas plataformas de rede, incluindo, mas não se limitando a, GSM, CDMA e TDMA. Isso não apenas aumentou sua ubiquidade, tornando-o um serviço universal em diferentes dispositivos e redes ao redor do mundo, mas solidificou sua posição como ferramenta essencial de comunicação em massa.

Adicionalmente, o SMS possui uma confiabilidade alta na entrega de mensagens. As operadoras de telecomunicações utilizam de mecanismos de armazenamento e encaminhamento por meio do SMSC (Centro de Serviços de Mensagens Curtas). Similar a aplicações como o e-mail, o SMSC é um servidor que verifica o número de telefone do destinatário e determina a rede apropriada para entregar a mensagem. Esse processo permite persistência no envio que é crucial para garantir a entrega de mensagens mesmo quando o destinatário está fora de área de cobertura ou com o dispositivo desligado. Atualmente as taxas de entregas em campanhas estão entre 85% a 90%, dependendo da qualidade da base de números utilizadas pelas empresas.

2.1.3 Importância e aplicações empresariais do SMS

Apesar da queda da popularidade do canal pelos usuários, que hoje preferem ferramentas multimídia e de interações mais ricas tais como o Whatsapp, Telegram, Facebook messenger, Instagram Messenger e mais recentemente o Google Messages com RCS, o serviço de Mensagens Curtas desempenha um papel crucial no mundo empresarial.

As aplicações em ambientes empresariais são vastas e variadas, sendo que as grandes empresas do segmento Financeiro, Varejo, Educação, Saúde e Serviços continuam sendo os maiores usuários do canal, dado a sua capacidade de atingir rapidamente uma ampla audiência, confiabilidade, entregabilidade e principalmente custo. Quando comparado com os canais digitais mais recentes, como o Whatsapp, o SMS pode custar para uma empresa

dez vezes menos.

O mercado global de SMS A2P (Serviço de Mensagens Curtas de Aplicativo para Pessoa) e CPaaS (Communication Platform as a Service) foi de US$ 29 bilhões em 2021 de acordo com (Insights, 2024), o mercado deverá atingir US$ 56 bilhões até 2028, exibindo um CAGR de 9,7%. O surto de Covid-19 teve um impacto relevante no crescimento do mercado. À medida que as interações físicas foram limitadas, as empresas recorreram a canais digitais para divulgação de informações críticas, orientações de saúde e atualizações de trabalho remoto.

As aplicações mais comuns encontradas no mercado do SMS A2P são:

1. **Marketing e campanhas promocionais:** empresas utilizam para enviar alertas de vendas, cupons de desconto, promoções especiais e atualizações de eventos diretamente para os celulares dos clientes. Esta estratégia é eficaz para impulsionar vendas imediatas e reforçar o engajamento do cliente com a marca. Exemplos muito comuns são os varejistas que utilizam em ações de "cash-back" para segunda compra.

2. **Autenticação de dois fatores (2FA):** aplicativos digitais precisam confirmar de forma segura se o usuário correto está solicitando alteração do seu acesso ou outra informação sensível. Grande parte dos aplicativos digitais atualmente oferecem múltiplos canais para segundo fator de autenticação.

3. **Alertas urgentes e notificações transacionais:** usado para enviar alertas críticos, como notificações de segurança, alertas de fraude ou lembretes de pagamento. Essa aplicação é crucial para manter os clientes informados e seguros. No mercado encontramos muito nas transações de cartão de crédito ou compras em máquinas de cartão, onde o usuário pode optar em receber o comprovante por SMS.

4. **Atendimento ao Cliente:** as empresas oferecem suporte ao cliente, permitindo um canal direto e eficiente para a resolução de dúvidas e problemas. Isso inclui o envio de lembretes de compromissos, atualizações de status de pedidos e respostas rápidas a perguntas de clientes, como por exemplo, a falta de energia em uma determinada região. Com um simples cadastro telefônico, restaurantes, bares e outras empresas de serviços costumam alertar sobre o status da posição de um cliente na fila de espera para atendimento.

5. **Pesquisas e coleta de feedback:** através de pesquisas rápidas, empresas podem coletar feedback importante dos clientes de maneira conveniente, como por exemplo pesquisas de NPS (Net Promoter Score). Isso permite que as empresas ajustem rapidamente seus produtos, serviços ou processos de acordo com as necessidades e preferências do cliente.

6. **Comunicações Internas:** é uma ferramenta valiosa para comunicações internas, especialmente para alertar funcionários sobre atualizações importantes ou mudanças de última hora. Devido à sua natureza imediata e à alta taxa de leitura, o SMS garante que as mensagens críticas sejam recebidas e lidas a tempo. Muito utilizado por equipes de TI remota no monitoramento de ambientes e infra-estrutura.

7. **Confirmações entregas ou pedidos:** para evitar chamados no atendimento das empresas de forma desnecessária, são enviadas mensagens para acompanhamento logístico de um pedido e confirmação da entrega. A maioria das empresas de e-commerce e o próprio Correios no Brasil encaminham mensagens com links de rastreamento e status.

2.2 Fraudes via SMS (Smishing): Um Desafio Crescente

2.2.1 Processo de ataque

As fraudes por SMS são conhecidas como "smishing"(SMS Phishing) e são um tipo de golpe cibernético direcionado aos usuários de smartphones em que os criminosos usam mensagens de texto para enganar as pessoas e obter informações pessoais, financeiras ou para induzi-las a realizar ações que possam causar prejuízos. O smishing difere de um ataque de phishing convencional em muitos aspectos, especialmente porque a quantidade de informação do SMS está limitada aos 160 caracteres. Portanto, a estratégia de ataque se concentra em encaminhar uma mensagem de texto fraudulenta disfarçada de uma mensagem genuína.

Conforme ilustrado na Figura 1, ao se passar por uma empresa, o fraudador promove por meio do texto de SMS uma ação de urgência no usuário (“call to action”) de forma que este acesse um site (URL) ou ligue para um número de telefone informado na mensagem, onde o fraudador consiga capturar informações relevantes deste usuário. Uma mensagem de fraude sem um “call to action” tem baixa probabilidade de dano ao usuário final, ainda que possa representar um Spam indesejado.

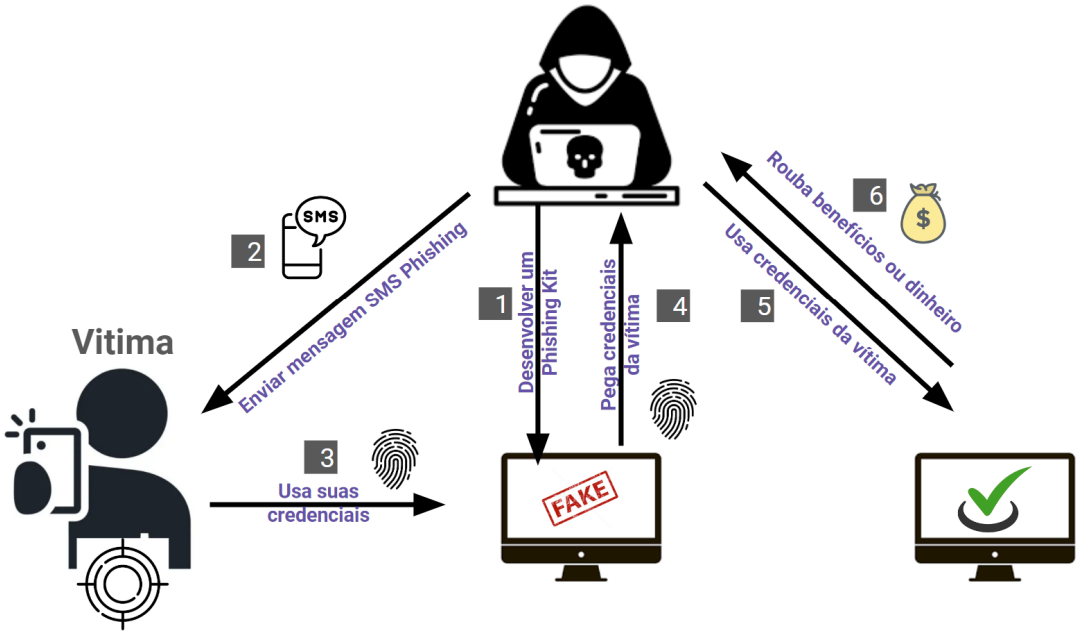
Os processos de ataque mais comuns incluem os seguintes tipos de mensagens:

1. **URL de phishing para capturar as credenciais de um usuário, tais como:** user-ids, senhas e detalhes financeiros.

2. **URL que baixa e instala uma aplicação (malware) no dispositivo da vítima** e consequentemente captura informações sensíveis do usuário.

3. **Telefones de contato do tipo 0800 ou 4004** solicitando que o usuário entre em contato com a empresa. Neste caso, na etapa 3 da figura 1, o usuário liga para uma central de atendimento, onde o fraudador se passa por um atendente da empresa, solicitando as credenciais ou outras informações sensíveis da vitima.

Figura 1 – Processo de ataque de smishing



Fonte: Adaptada de (Korkmaz; Sahingoz; Diri, 2020).

Importante destacar que os usuários de smartphones estão menos atentos aos riscos de segurança associados a este tipo de fraude, já que assumem que os dispositivos celulares são mais seguros que os computadores. Entretanto, os usuários possuem a maioria das aplicações bancárias e documentações digitais instaladas em seu aparelho. Outro fator de risco associado aos celulares é que as pessoas normalmente utilizam quando estão se locomovendo, o que faz com que leiam as mensagens rapidamente, levando o usuário a responder uma mensagem ou clicar em um link malicioso sem pensar. (Mishra; Soni, 2023)

Existem um outros tipos de fraudes, e que não são foco deste trabalho, a citar:

- **ERB (Estação Rádio Base) pirata:** o aparelho do usuário se autentica, sem consentimento do usuário em uma uma ERB pirata. O fraudador utiliza de equipamento que simula as estações móveis e conseguem encaminhar uma mensagem SMS em broadcast para todos os terminais que se conectam em seu equipamento. Normalmente, os fraudadores procuram por regiões com alta concentração de pessoas, como por exemplo a região da avenida Paulista em São Paulo. Para evitar esse tipo de fraude é essencial que algoritmos de detecção também sejam utilizados nos terminais dos usuários (client side) e não somente nas redes de telecomunicações ou nos gateways das plataformas das empresas de disparos.

- **Roubo ou vazamento de credenciais das plataformas de disparo:** neste caso a empresa (marca) que utiliza de sistemas de mensagens, como oferecido pela

Zenvia, tem suas credenciais roubadas, e o fraudador se passa pela empresa para fazer disparos massivos de campanhas, direcionando o usuário para sites falsos (fake). O combate de controle de credenciais é um processo continuo dos times de segurança da informação, principalmente dos fornecedores de tecnologia em nuvem, que acoplam múltiplos fatores de autenticação como ferramentas de proteção, assim como passam por processos de auditoria complexos como ISO27001.

2.2.2 Características das mensagens de Fraude

A Figura 2 ilustra uma mensagem de real e recente, onde o usuário é direcionado para um site para troca de pontos, e necessita ao longo da jornada digitar sua conta bancária, senha e inclusive tirar uma self pessoal. Todas credenciais necessárias para que o fraudador consiga logar no website real do banco.

Figura 2 – Mensagem de phishing real

Fonte: Elaborada pelo autor.

De uma maneira geral, as principais características de mensagens fraudulentas são descritas na Tabela 1.

2.2.3 Impacto Financeiro e Social das Fraudes

O impacto financeiro dos ataques de smishing é crescente. Não há dados mais recentes e específicos de smishing para o mercado brasileiro, que deve seguir as tendências internacionais. Entretanto, segundo levantamento da Kaspersky em 2020 (Coclin, 2021) sobre práticas de phishing, o Brasil é líder mundial em tentativas de roubo de dados pessoais e financeiros na internet ao longo 2020.

Nos Estados Unidos, as perdas financeiras atribuídas a vários golpes online, incluindo smishing, alcançaram impressionantes $10.3 bilhões em 2023. (Splash, 2023) O setor financeiro é o mais afetado com aproximadamente 30% de todas as fraudes do ano passado, seguidos por saúde, varejo e manufatura. Segundo o relatório, as técnicas mais utilizadas são: Falsas promessas de premiação (40%), pedidos de verificação de conta (30%) e ofertas de emprego (20%).

Tabela 1 – Características das mensagens de fraude

Característica	Propósito
Aparência Legítima	• Mensagens disfarçadas de fontes confiáveis e que possuem alta abrangência de consumidores finais, como: bancos, grandes varejistas, agências governamentais ou empresas com marcas conhecidas. • Utilizam os nomes das empresas, logos e formatos semelhantes aos dessas entidades para criar mensagens convincentes.
Pedidos de Ação Imediata	• A urgência é um componente crítico, pressionando as vítimas a agir rapidamente. Essa pressão pode fazer com que a vitima reaja sem verificar a autenticidade da mensagem. • As mensagens podem incluir um pedido para resolver um problema aparentemente sério, como conta bloqueada ou um problema com pagamento. No caso do exemplo da figura 2, o usuário irá perder um volume grande de pontos de um programa de fidelidade.
Links Maliciosos ou Telefones dos fraudadores	• As mensagens de SMS contêm links que direcionam para sites falsos ou telefones do fraudador. • Objetivo é persuadir as vítimas a fornecerem dados pessoais, como números de cartões de crédito, senhas ou informações de contas bancárias.

Socialmente, os ataques de smishing corroem a confiança no ecossistema de mensagens e impõem um custo emocional às vítimas, aumentando a ansiedade e o estresse. Isso é agravado pelo fato de que muitos usuários ainda não estão totalmente cientes dos riscos associados, o que os torna particularmente vulneráveis a esses golpes. Em (Splash, 2023), afirma que 63.5% dos usuários de smartphone no Japão, que possui um alto nível educacional, não estão familiarizados com o termo 'smishing'.

Além das perdas imediatas, as vítimas de smishing frequentemente enfrentam custos significativos relacionados à recuperação de sua identidade financeira e segurança de contas. Isso pode incluir taxas legais, custos bancários para contestar transações fraudulentas, e serviços de monitoramento de crédito.

O aumento dos ataques de smishing pode levar a uma desconfiança generalizada em mensagens de texto e outras formas de comunicação digital, impactando a forma como indivíduos e empresas se comunicam. Ao longo do tempo isso vem ocorrendo com o SMS,

o que ajudou na criação de novas oportunidades de negócio como é o caso do canal RCS do Google Messages que exige a criação e aprovação de um agente junto as operadoras de telecomunicações para que uma empresa possa realizar campanhas de mensagens.

O smishing muitas vezes explora informações pessoais obtidas ilegalmente, o que levanta preocupações significativas sobre privacidade. A percepção de que as informações pessoais não são seguras pode alterar o comportamento online das pessoas, tornando-as mais relutantes em compartilhar informações ou utilizar serviços digitais. As empresas também são obrigadas a investir cada vez mais em mecanismos de bloqueio e exposição de dados, muitas vezes retrocedendo processos anteriormente automatizados para outros manuais e burocráticos.

2.2.4 Ações Regulatórias contra o Smishing

A ANATEL tem promovido no Brasil um debate próximo às operadoras e brokers de serviços de mensageria no mercado para o combate à fraude. Todas as ações tem o objetivo primário de melhorar o entendimento sobre o papel e responsabilidade dos envolvidos na cadeia de mensageria, além de tentar criar uma estrutura de compartilhamento de informações, principalmente referente uma base de dados centralizada para consulta de números de 0800, URLs e palavras chaves fraudulentas. (TECNOBLOG, 2024)

As medidas encabeçadas pelo órgão regulador visam a proteção do usuário final de aparelhos móveis e incluem:

- Processo de contratação e gestão de números 0800
- Combate a formas alternativas de envio de SMS, como é o caso de ERBs fakes, ou empresas que realizam disparos utilizando de chips de celular P2P.
- Trabalho educativo para conscientização sobre os golpes.
- Central de reporte de smishing
- Canal específico de denúncias para instituições financeiras
- Orientação e boas práticas para utilização de números 0800

Além dessas medidas, há um processo em discussão com as operadoras de telecomunicações sobre a evolução na atribuição dos “short-codes” ou LAs (Logical Address). Os LAs são números de 5 dígitos utilizados para identificar o remetente ou o destino das mensagens dentro da rede, facilitando o correto roteamento e entrega das mensagens. Isso é especialmente relevante em sistemas que utilizam plataformas de SMS, onde múltiplos agentes ou serviços estão envolvidos na transmissão de mensagens curtas. A questão é que atualmente, os “short-codes” não são de caráter exclusivo das empresas remetentes

na maioria dos casos. Portanto, um usuário pode receber por meio de um mesmo LAs mensagens de diversas empresas, inclusive de fraudadores.

Uma das propostas do órgão regulador é ampliar o espectro de código disponíveis de 5 para 7 dígitos, para que a maioria das empresas que utilizam o SMS não possam mais compartilhar LAs. Faixas de numeração também serão dedicadas de acordo com o tipo de tráfego: Compartilhado, Internacional, Autenticação, Setor Público, Setor Corporativo, Setor Financeiro, Pequenas e Médias Empresas, etc. Isso permitirá um cadastro centralizado e a possibilidade de identificação com confiabilidade da empresa remetente em uma comunicação A2P. Entretanto, esta ação requer a criação de uma entidade para coordenar e gerir de forma centralizada os short-codes, o que pode ser um entrave burocrático para implantação.

As ações em curso são paliativas uma vez que as medidas atuais não estão sendo suficientes, e os cyber-ataques são dinâmicos, mudando constantemente as técnicas e os canais de comunicação utilizados. Estudos com trabalhos educativos e conscientização para ajudar pessoas a identificar mensagens de SMS fraudulentas não surtiram efeitos, segundo (Kubilay *et al.*, 2023). Neste sentido, uma abordagem multifacetada é urgente, para complementar ações regulatórias com o desenvolvimento e aplicação de novas tecnologias que utilizam de IA na detecção de smishing.

A maioria das empresas que oferecem plataforma de disparos de SMS, assim como as operadoras, aplicam filtros de Spam por meio de "gray-list" de termos, números e links de URLs; ou "whitelists" de clientes verificados. A detecção de tráfego suspeito ainda é feita de forma manual acompanhado caso-a-caso pelos times de cibersegurança das empresas. Com a evolução das técnicas de ataque, novas palavras chaves são incorporadas nas listas. e de acordo com as regras de contratação de cada empresa.

2.3 Tecnologias de Inteligência Artificial na Detecção de Fraudes

2.3.1 Introdução à Inteligência Artificial e Aprendizado de Máquina

A Inteligência Artificial (IA) e o Aprendizado de Máquina (AM) são campos da ciência da computação que visam criar sistemas capazes de realizar tarefas que, até então, requerem intervenção humana (Russell; Norvig, 2016). A ideia é permitir que os computadores aprendam com a experiência do passado (dados) e compreendam o mundo em termos de uma hierarquia de conceitos.

Essa abordagem evita a necessidade de um operador humano (programador) formalmente especifique todo o conhecimento necessário para que um programa de computador (algoritmo) resolva um problema, porém mantém o desafio chave de transcrever o conhecimento informal e intuitivo humano para uma máquina.

O **Aprendizado de Máquina** é uma subárea da IA focada no desenvolvimento

de técnicas que permitem que as máquinas aprendam a partir de dados. Essencialmente, AM envolve alimentar um modelo computacional com grandes quantidades de dados e permitir que o modelo ajuste seus parâmetros internos para melhorar seu desempenho em uma tarefa específica, como a previsão de mensagens fraudulentas, objeto deste trabalho. A performance dos algoritmos de AM dependem fortemente da representação dos dados que são apresentados. Cada pedaço de informação incluída na representação dos dados é conhecida como **característica, atributo, ou "feature"** no inglês.

Muitas atividades de IA podem ser resolvidas pelo design correto da extração de atributos de um conjunto de dados, e posteriormente fornecendo esses atributos para um algoritmo simples de AM (Goodfellow; Bengio; Courville, 2016). Entretanto, para muitas atividades do cotidiano é difícil saber qual atributo deve ser extraído, como o exemplo de descrever imagem de um "gato"ou "cão"para um computador. Esse problema foi resolvido por meio de **aprendizado de representação**, que utiliza algoritmos como auto-encoder, para descobrir um bom conjunto de features dos próprios dados. Definir as características manualmente para uma tarefa complexa requer um grande quantidade de esforço e tempo.

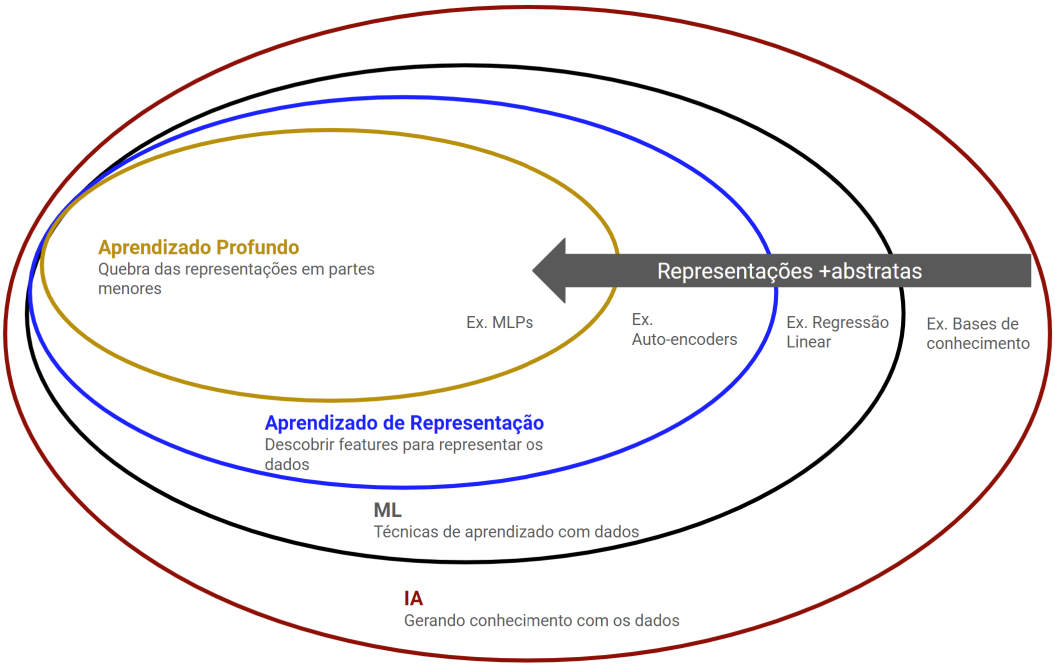
O desafio em fazer o design correto das características seja pelos dados diretamente, ou por meio de algorítmos que aprendem as características, está em separar os fatores de variação que explicam os dados observados. Em muitas aplicações de IA, é difícil separar os diferentes fatores que afetam os dados que observamos, para focar apenas nos que são relevantes para nossa aplicação.

O **aprendizado profundo (Deep Learning)** resolve esse problema criando representações complexas a partir de representações mais simples. Ele permite que o computador construa conceitos complexos (como a imagem de uma pessoa) combinando conceitos mais simples (como cantos e contornos). A rede neural profunda é a representação usada para o aprendizado profundo. O aprendizado em si é o processo de ajustar a rede, mapeando valores de entrada para valores de saída, usando funções matemáticas que calculam o erro das entradas e saídas, ajustando os parâmetros de pesos e bias para cada elemento da rede.

De acordo com Goodfellow, Bengio e Courville (2016), o AM é a única abordagem viável para construir sistemas de IA que podem operar no ambiente complexo do mundo real. O Deep Learning é um tipo particular de AM que alcança grande poder e flexibilidade ao aprender a representar o mundo como uma hierarquia aninhada de conceitos, com cada conceito definido em relação a conceitos mais simples, e representações mais abstratas calculadas em termos de representações menos abstratas. A figura 3 ilustra a relação entre essas diferentes disciplinas e apresenta um esquemático alto nível como cada disciplina de IA trabalha a questão das representações das características dos dados.

Neste trabalho utilizamos o AM tanto com técnicas mais simples de representação das características das mensagens de SMS, quanto o aprendizado profundo para comparação

Figura 3 – Diagrama de Venn mostrando como o Aprendizado profundo é um tipo de Aprendizado de Representação, que é um tipo de AM, que é usado em muitas abordagens do campo de estudo da IA.



Fonte: Adaptada de (Goodfellow; Bengio; Courville, 2016).

da performance no processo de criação do filtro de SMS.

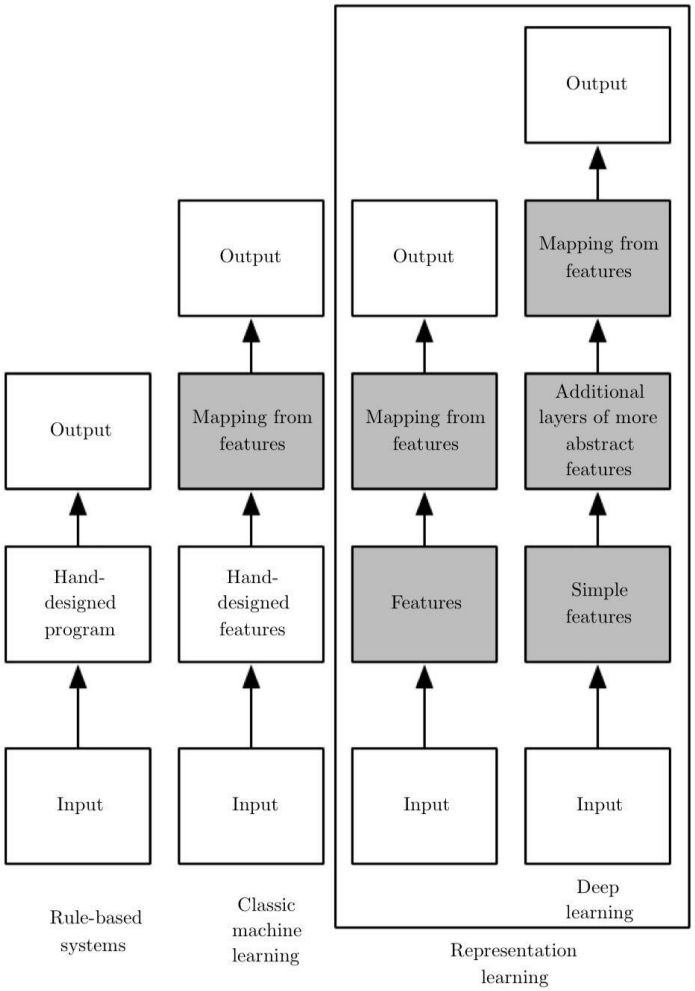
2.3.2 Algoritmos de aprendizado de máquina

Os algoritmos de aprendizado podem ser classificados em 4 categorias: supervisionado, semi-supervisionado, não supervisionado e por reforço, cada um com suas próprias abordagens e aplicações específicas.

Tabela 2 – Tipos de algoritmos de aprendizado de máquina

Tipo de Aprendizado	**Abordagem**	**Aplicações Específicas**	**Exemplos Práticos**
Supervisionado	Modelos treinados com dados rotulados	Classificação, regressão	Previsão de preços de casas, detecção de spam
Semi-Supervisionado	Combinam dados rotulados e não rotulados	Melhoria da aprendizagem com dados limitados	Classificação de texto, análise de documentos
Não Supervisionado	Modelos que identificam padrões em dados não rotulados	Agrupamento, redução de dimensionalidade	Segmentação de clientes, detecção de anomalias
Por Reforço	Aprendem a tomar decisões através de recompensas	Jogos, robótica autônoma, otimização de processos	Aprendizado de estratégias de jogo, robôs de navegação

Figura 4 – Esquemático mostrando como as diferentes partes de sistemas de IA relacionam uma com a outra dentro das disciplinas das diferentes tecnologias. Os componentes sombreados representam elementos capazes de aprender com os dados



Fonte: Elaborada por (Goodfellow; Bengio; Courville, 2016).

Observa-se que para o problema de detecção de fraude, o modelo de aprendizado supervisionado é mais adequado para classificação, porém o desbalanceamento de dados rotulados como Spams ou fraudes, para o propósito deste trabalho é tipicamente muito elevado. Portanto, buscamos uma abordagem de comparar os dois tipos de modelos: supervisionado e semi-supervisionado.

Além disso, para identificação de padrões nas mensagens de texto, o tipo de aprendizado mais adequado é o não supervisionado, com o propósito de mapeamento de features das mensagens fraudulentas.

2.3.3 Desafios para detecção do Smishing

Os principais desafios no processo de detecção de fraudes seguem com as mesmas problemáticas:

1. Os fraudadores frequentemente alteram suas abordagens, utilizando técnicas como o uso de linguagem natural em suas mensagens para simular comunicação legítima e a rápida mudança de links maliciosos.

2. Mensagens de texto abreviadas tornam difícil a análise de classificação e limita o número de features ou variáveis de classificação que podem ser extraídas da mensagem.

3. A troca de palavras ou caracteres, idiomas e palavras mal escritas são usadas na mensagens de SMS, o que dificulta a identificação de palavras-chave de smishing.

4. Mensagens de Spam contêm um conjunto semelhante de características em comparação com as mensagens de smishing. Portanto, diferenciar entre mensagens de spam e mensagens de smishing é uma tarefa árdua.

5. A escassez de conjuntos de dados públicos sobre smishing torna um desafio avaliar a detecção de smishing por algoritmos de aprendizado de AM.

6. Detecção em tempo real, necessário para impedir fraudes antes que elas afetem as vítimas, o que exige sistemas altamente responsivos e atualizados continuamente.

7. Equilibrar a precisão da detecção com a minimização dos falsos positivos, para não interferir na experiência do usuário com bloqueios indevidos.

2.3.4 Técnicas para detecção do Smishing

Pesquisas anteriores que trabalharam no problema do smishing identificaram 3 categorias principais com métodos para detectar mensagens de texto maliciosas. Os autores categorizam as técnicas conforme tabela 3 (Cossa *et al.*, 2021).

A revisão e pesquisa feita por Abayomi-Alli *et al.* (2019) em mais de 1198 artigos, considera, naquele momento, a taxonomia do estado da arte dos principais métodos utilizados para detecção de Spam de SMS, que são ilustrados na Figura 5. As técnicas de aprendizado de máquina SVM e Bayesianas são as mais utilizadas nas pesquisas, mas segundo Abayomi-Alli *et al.* (2019), muitos métodos ainda não foram profundamente explorados, como o caso de Aprendizado profundo (Deep Learning).

Além das técnicas, a pesquisa detalha outras características das abordagens de diferentes artigos:

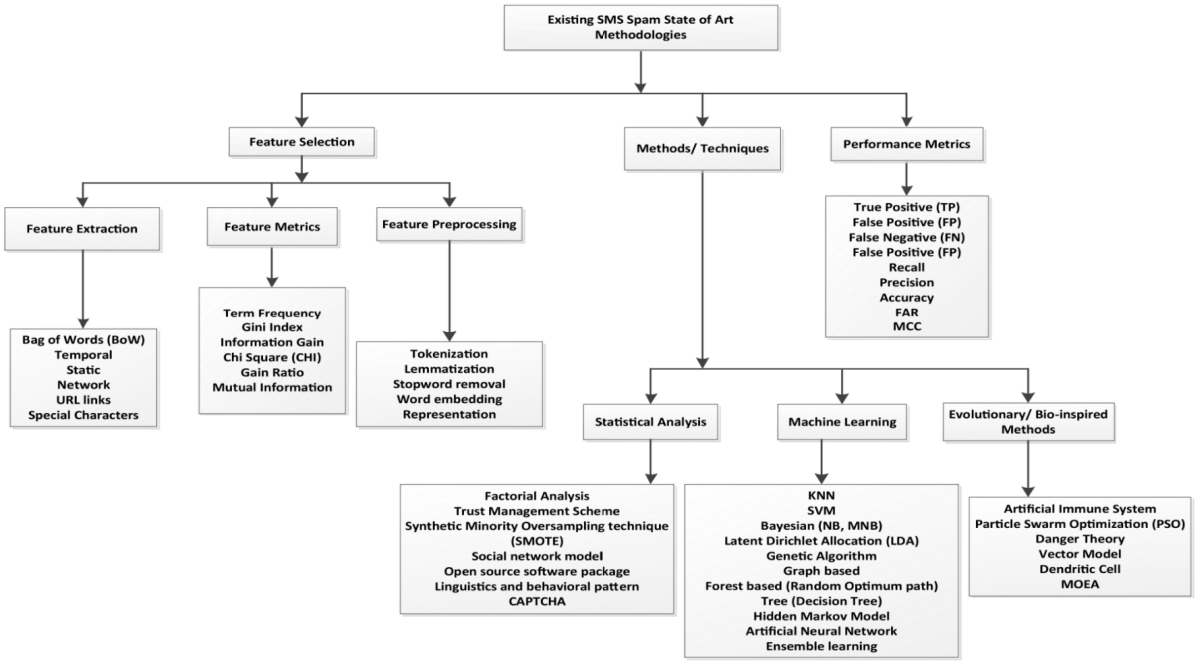
Tabela 3 – Categorias de Técnicas AntiSpan identificadas.

Categoria	**Característica**	**Métodos mais utilizados**
Machine Learning	Amplamente utilizados na classificação de texto e envolvem o uso de paradigmas de aprendizagem e validação experimental com foco na geração de expressões de classificação, fornecendo assim, uma visão de processos de decisão.	• SVM • Naive Bayes • Decision Table • Nearest Neighbour (KNN)
Análises Estatísticas (Statistical Analysis)	Baseados em formulação de modelos matemáticos e probabilísticos que não requerem intervenção humana e o funcionamento não requer intervenção humana.	• Análise de comportamento
Algoritmos Evolutivos (Evolutionary / Bio-inspired Methods)	São caracterizados pelo uso de técnicas evolutivas ou bioinspiradas.	• Sistema Imunológico Artificial (AIS) • Algoritmo de Células Dendríticas (DCA), • SLAVE

1. **Seleção ou Extração de Features (ou características da mensagem):** é importante selecionar características altamente relevantes para a classificação eficaz de mensagens de textos, minimizando a redundância. A escolha adequada das features pode reduzir significativamente a dimensionalidade do espaço analisado, aumentando o desempenho dos modelos de classificação. A eliminação de características redundantes pode melhorar a precisão do classificador. O desequilíbrio nos conjuntos de dados de SMS é um problema comum que reduz a performance dos modelos. Logo, há um comprometimento significativo dos pesquisadores para aprimorar os algoritmos de seleção de features, consequentemente, melhorar a performance dos filtros. Diversos métodos foram propostos para a categorização de textos, como Frequência de Documento, Informação Mútua, métricas de Qui-quadrado, entre outros. Segundo (Uysal *et al.*, 2013), o impacto de várias metodologias de extração e seleção de características na filtragem de spam em SMS, foi examinado e indicou que a combinação de características estruturais e BoW (Bag of Words), em vez das características BoW isoladas, oferece um desempenho de classificação superior na maioria das vezes. A inspeção de novas características estruturais, a avaliação de outros métodos de seleção de características e de classificação no problema de filtragem de spam em SMS permanecem como trabalhos futuros interessantes.

2. **Abordagens:** os estudos identificam três abordagens: baseada em conteúdo, não baseada em conteúdo e híbrida. A abordagem baseada em conteúdo usa a frequência de palavras ou caracteres para representar documentos, enquanto a abordagem não baseada em conteúdo se vale de características específicas das mensagens para

Figura 5 – Taxonomia do estado da arte das metodologias anti-spam para SMS.



Fonte: Elaborada por (Abayomi-Alli *et al.*, 2019).

detectar anomalias. A abordagem híbrida combina características das duas anteriores para fins de classificação.

3. **Arquitetura:** filtros antispam de SMS se baseiam em três camadas principais: cliente (solução no dispositivo móvel), no servidor (soluções no lado do provedor de rede) e colaborativa (soluções que envolvem tanto o cliente quanto o servidor). Muitos dos estudos selecionados adotaram soluções baseadas no cliente. Este trabalho foca na arquitetura baseada em rede (Server-based), que ainda possuem poucos estudos executados, e uma vez que os filtros devem evitar a entrega das mensagens na origem do disparo. apesar de existirem arquiteturas para implantação nos devices dos usuários, e arquiteturas colaborativas, que são mais difíceis na prática, pois requerem a criação de comunidades para compartilhamento de palavras de spam.

As técnicas aplicadas atualmente para detecção de fraudes de smishing são baseadas em listas do tipo "grey"ou "whitelists". Estes métodos não são eficientes, pois estas listas são atualizadas de forma manual e com baixa recorrência. Este trabalho buscou avaliar a aplicação de métodos de redes neurais profundas considerando endereçar os aspectos cruciais de identificação das fraudes.

Neste sentido, observamos que os estudos apresentados focam nos métodos, extração de features, abordagem e arquitetura, em diferentes combinações. Entretanto independente

destes métodos, para aprimorar os filtros de mensagens de SMS fraudulentas, o trabalho focou nos aspectos cruciais das fraudes:

- Inspecção do link da URL presente na mensagem SMS e o seu redirecionamento (Mishra; Soni, 2023). Alguns estudos como (Korkmaz; Sahingoz; Diri, 2020) utilizaram de técnicas de extração de mais de 58 features de URLs e avaliou 8 diferentes algoritimos de AM para classificar as URLs utilizando de datasets coletados de "Phishtank.com". Algoritimos como Randon Forest (RF) ofereceram melhor acuracidade e performance para classificação de novas URLs. Neste estudo abordaremos a utilização de APIs externas e LLMs para classificação das URLs.
- Inspeção do número telefônico, caso presente na mensagem (que não será abordado neste estudo).
- Inspeção do conteúdo da mensagem, se possui características de SPAM ou falsas promessas.
- Base de dados de treinamento e teste relevante para ser usada em algoritmos de Deep Learning.

3 METODOLOGIA

3.1 Design da Pesquisa

Neste trabalho foi realizado um estudo experimental com o objetivo de desenvolver e avaliar filtros de alta precisão e performance para mensagens de SMS fraudulentas utilizando aprendizado de máquina clássico e aprendizado profundo com redes neurais.

O design da pesquisa envolveu a análise comparativa de diferentes abordagens de extração de características das mensagens de SMS, com foco em algoritmos supervisionados e semi-supervisionados para detecção de fraudes.

A pesquisa comparou a performance de diferentes algoritmos de AM para reduzir a taxa de falsos positivos em mensagens legítimas e avaliando o impacto do pré-processamento de dados na precisão dos filtros.

As hipóteses foram formuladas com base na revisão da literatura, que sugere novos experimentos no processo de determinação das características das mensagens (aumento performance computacional dos filtros) e utilização de um conjunto de dados maiores para melhoria na precisão da identificação de mensagens fraudulentas. Utilizamos também de forma inovadora, a ferramenta de LLM da OpenAI (ChatGPT) para ajudar no processo de captura de features das URLs presentes nos dados textuais.

Apesar do volume considerável de mensagens, cabe ressaltar que o conjunto de dados foi extraído de forma aleatória pelo time de infra-estrutura da Zenvia, com muitas mensagens de conteúdo repetido. Essa limitação pode afetar a generalização dos resultados para outros contextos ou bases de dados diferentes.

3.2 Pipeline do processo

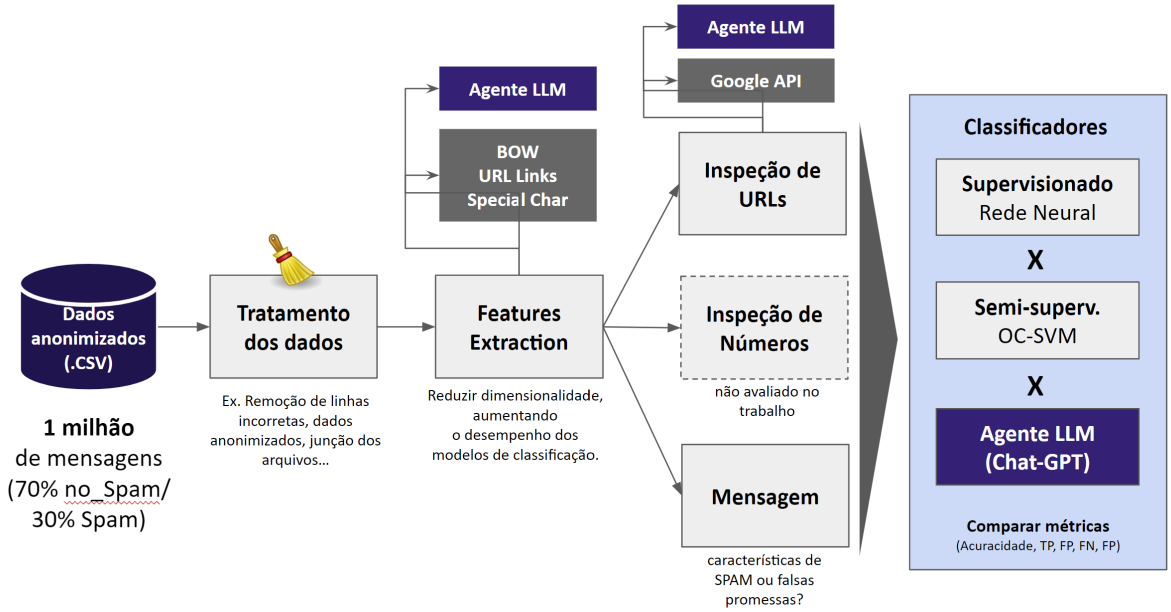
O pipeline apresentado na Figura 6 detalha o fluxo de processamento para o desenvolvimento e avaliação dos filtros de alta precisão para mensagens SMS fraudulentas utilizando redes neurais e outros métodos de aprendizado de máquina.

O diagrama pode ser dividido em 3 fases macros: Coleta dos dados, Pré-processamento e Classificação, sendo que neste último as métricas são comparadas. As etapas de tratamento, extração de features e inspeções, representam o detalhamento do pré-processamento dos dados.

A seguir, cada etapa do pipeline, desde a coleta de dados até a comparação de métricas dos modelos é detalhada.

Coleta de Dados: o ponto de partida do processo é a coleta de um grande conjunto de dados anonimizados, consistindo de mensagens SMS "não spam"e mensagens

Figura 6 – Metodologia e pipeline do processo de classificação de SMS.



Fonte: Elaborada pelo Autor.

SMS "fraudulentas"ou "spam". A coleta foi feita por extração diretamente no banco de dados SQL em arquivos de dados separados para o formato CSV.

Tratamento dos Dados: envolve a leitura dos arquivos e o tratamento dos dados. Isso inclui a leitura correta dos arquivos, remoção de linhas incorretas e inconsistentes. Após a leitura e processamento, os arquivos são combinados em um único arquivo em formato consolidado para facilitar o processamento. Além disso, como muitas das mensagens possuem texto semelhantes, e estão ordenados pela extração, foi feito um embaralhamento das tuplas. Durante esta etapa, diversas análises exploratórias do conjunto de dados foram executadas para melhor entendimento do dataset e visando garantir a qualidade e integridade dos dados.

Extração de Características: após o tratamento inicial, a próxima etapa é a extração de características (features) dos dados. Técnicas como remoção de "StopWords"(conjunções, preposições, pronomes, etc.), lematização, contagem da quantidade de caracteres especiais, palavras, frases, Bag of Words (BoW), Embeddings, análise de URLs são utilizadas de forma combinada para extrair informações relevantes que ajudam na etapa de classificação. Este processo também é fundamental para ajudar a reduzir a dimensionalidade dos dados, melhorando o desempenho dos modelos classificadores. Para efeitos comparativos foi utilizado um Agente de LLM para executar a extração das features das mensagens.

Inspeção de URLs: utilizando APIs do Google e agentes LLM (ChatGPT), os links presentes nas mensagens são inspecionados para detectar padrões suspeitos ou maliciosos. Esta análise é crucial para identificar fraudes que utilizam URLs como vetor

de ataque e é executada antes da classificação da mensagem.

Inspeção de números: números telefônicos como 0800 e 4004 colocados nas mensagens podem ser de fraudadores. É possível consultar bases da ANATEL para avaliação de números bloqueados pela órgão regulamentador. Neste trabalho esse processo de avaliação foi desconsiderado devido a complexidade da consulta e o volume de fraudes desse tipo de abordagem ser menor quando comparado com ataques utilizando-se de URLs.

Inspeção da mensagem: separação e pré-processamento do corpo textual por meio de suas características para avaliação pelos classificadores.

Classificação das Mensagens: três abordagens principais são utilizadas para classificar as mensagens:

- Modelo Supervisionado (Rede Neural): algoritmos de redes neurais são treinados com os dados rotulados para identificar padrões complexos e realizar a classificação.
- Modelo Semi-supervisionado (OC-SVM): modelos como One-Class SVM são utilizados para detecção de anomalias em cenários onde os dados rotulados são limitados. Durante a etapa de pré-processamento, os dados de mensagens "ham"são considerados.
- Agente LLM (Chat-GPT): implementação de agentes LLM como o Chat-GPT para auxiliar na classificação e detecção de fraudes.

Comparação de Métricas: as diferentes abordagens de classificação são comparadas utilizando métricas de performance como acurácia e matriz de confusão. Esta comparação permite identificar a abordagem mais eficaz para a detecção de SMS fraudulentos. Foi medido também o tempo de classificação das diferentes abordagens para comparar a performance, fundamental para garantir a performance do filtro em altos volumes de mensagens enviadas.

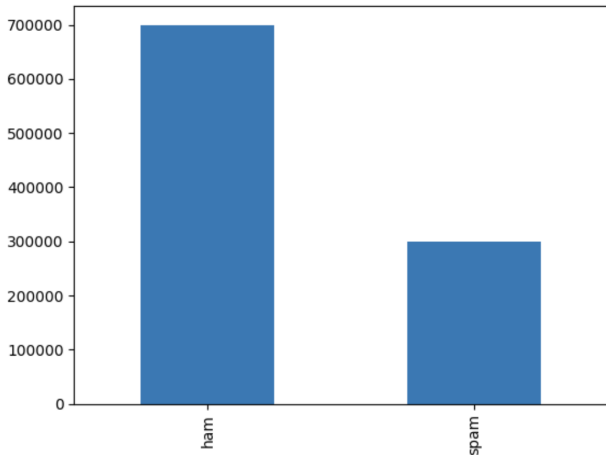
3.3 Coleta de Dados

Foi utilizada uma base de dados histórica fornecida pela Zenvia contendo o conteúdo da mensagem SMS enviada e sua classificação. O arquivo foi extraído diretamente do banco de dados e entregue em formato "CSV"com separador de dados por quebra de linha.

A coleta de dados foi realizada garantindo a anonimização e segurança das informações, conforme as diretrizes legais vigentes.

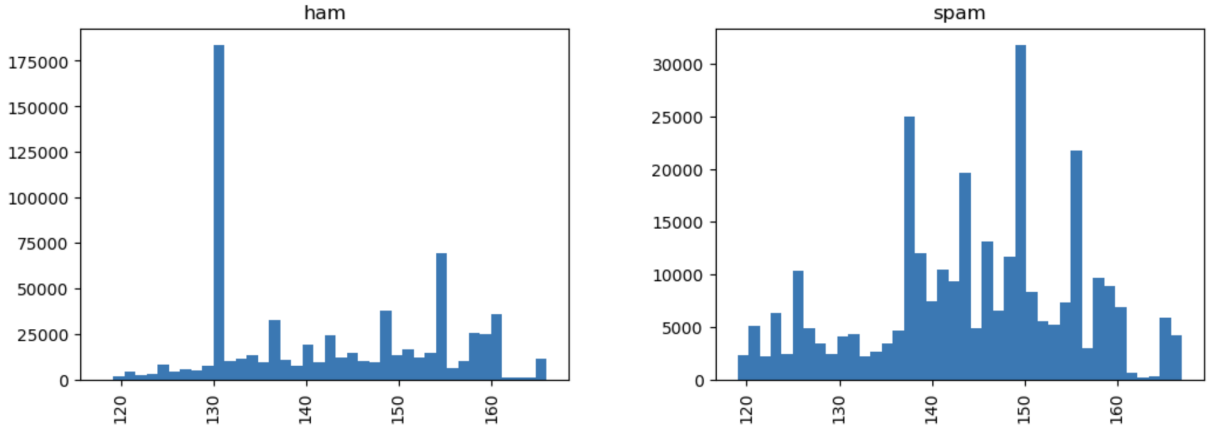
Conforme figura 7, foi executada uma análise exploratória a base de dados, que contém 1 milhão de mensagens SMS, sendo 300 mil mensagens rotuladas como casos identificados de fraudes (spam) e 699 mil mensagens normais (ham). Importante destacar

Figura 7 – Base de dados histórica de SMS da Zenvia.



Fonte: Elaborada pelo Autor.

Figura 8 – Histograma da quantidade de caracteres das mensagens SMS do conjunto de dados.



Fonte: Elaborada pelo Autor.

que um dado foi perdido, pois se tratava de informação interna do processo de extração do banco de dados.

As mensagens desse conjunto de dados possuem entre 120 a 170 caracteres como apresentado no histograma da 8

A análise exploratória dos dados é apresentada na figura 9. A tabela mostra a descrição do dataset com quantidade de dados, o número de caracteres, quantidade de palavras e quantidade de sentenças das mensagens de cada tupla.

3.4 Pré-processamento dos dados

O pré-processamento de dados é uma etapa crucial no desenvolvimento de modelos de AM, especialmente quando se trata de dados textuais, como mensagens SMS. A seguir

Figura 9 – Conjunto de dados após tratamento exploratório inicial

	count	mean	std	min	25%	50%	75%	max
spam	999999.0	0.299999	0.458257	0.0	0.0	0.0	1.0	1.0
char	999999.0	143.041401	11.633513	118.0	130.0	143.0	154.0	167.0
words	999999.0	27.35894	4.618953	1.0	24.0	28.0	32.0	51.0
sentence	999999.0	2.637798	0.86075	1.0	2.0	3.0	3.0	8.0

Fonte: Elaborada pelo Autor.

são apresentados passos essenciais para preparar os dados anonimizados, visando garantir a qualidade e a integridade necessárias para a aplicação dos algoritmos de classificação.

Nem todos os passos aqui descritos foram utilizados na preparação dos dados de estudo e aplicação nos algorítimos de classificação, mas serviram como um guia geral para construção da etapa de pré-processamento.

3.4.1 Limpeza dos Dados

Conforme apresentado em 3.3, a limpeza de dados envolveu a remoção de dados inconsistentes, duplicados ou irrelevantes que possam afetar a performance do modelo, incluindo a remoção de linhas Incorretas, tais como informações incompletas, caracteres especiais inválidos ou formatações incorretas.

Mensagens irrelevantes que não contribuíam para a detecção de spam, como mensagens muito curtas ou de serviços automatizados, foram excluídas. Este processo é feito por meio do cálculo dos quartis de acordo com a quantidade de caracteres, eliminando assim os outliers.

Problemas de codificação de caracteres, como caracteres especiais mal representados, foram corrigidos. Um exemplo de mensagem com caracteres especiais devido a processo de anonimização na extração dos dados do banco:

```
"{nome_cliente}, vc ganhou um BONUS de {valor_bonus} no site da REGINA
RIOS (na compra de R$ 194,35 ou +)! Clique no Link: bit.ly/40iNPRH
e gere seu cupom."
```

3.4.2 Anonimização dos Dados

Para proteger a privacidade dos usuários, todas as informações de identificação pessoal foram removidas ou mascaradas. Informações como números de telefone, nomes e en-

dereços foram eliminadas dos dados. Quando necessário, identificadores foram substituídos por pseudônimos para manter a estrutura dos dados sem comprometer a privacidade.

3.4.3 Normalização dos Dados

A normalização é necessária para garantir que os dados estejam em um formato consistente, facilitando a extração de características e a aplicação dos modelos de AM.

- Conversão para minúsculas: todas as mensagens foram convertidas para letras minúsculas para evitar duplicidade de palavras devido à diferenciação de maiúsculas e minúsculas.
- Remoção de pontuação e caracteres especiais: pontuações e caracteres especiais que não contribuem para a análise foram removidos, mantendo apenas caracteres alfanuméricos essenciais.
- Tokenização: as mensagens foram divididas em tokens (palavras individuais), permitindo uma análise mais detalhada e a extração de características textuais.

3.4.4 Extração de Características (Features)

- Bag of Words (BoW): as mensagens foram transformadas em vetores de frequência de palavras utilizando a técnica Bag of Words. Esta técnica ajuda a quantificar a presença de palavras em cada mensagem.
- Análise de Links (URLs): URLs presentes nas mensagens foram extraídas e analisadas para detectar padrões suspeitos ou maliciosos.
- Caracteres especiais e estruturais: características como o uso de caracteres especiais, a estrutura da mensagem e padrões específicos foram extraídas para melhorar a classificação.

3.4.5 Redução de Dimensionalidade

Para melhorar o desempenho dos modelos, técnicas de redução de dimensionalidade foram aplicadas minimizando a complexidade dos dados.

- Seleção de Características: técnicas de seleção de características foram utilizadas para identificar e manter apenas as características mais relevantes, descartando as que não contribuem significativamente para a detecção de spam.
- Redução por Componentes Principais (PCA): quando necessário, a técnica de Análise de Componentes Principais foi aplicada para reduzir a dimensionalidade dos dados, preservando a maior parte da variância dos dados originais.

3.5 Análise dos resultados

Por se tratar de um processo de classificação binário de mensagens, como métricas comparativas de performance foram escolhidas:

1. Matriz de confusão: TP (Verdadeiro Positivo), TN (Verdadeiro Negativo), FP (Falso Positivo), FN (Falso Negativo)
2. Acurácia: Proporção de previsões corretas em relação ao total de previsões.
3. Precisão (Precision): Proporção de verdadeiros positivos em relação ao total de positivos preditos (TP / (TP + FP)).
4. Revocação (Recall) ou Sensibilidade (Sensitivity): Proporção de verdadeiros positivos em relação ao total de positivos reais (TP / (TP + FN)).
5. F1-Score: Média harmônica entre precisão e revocação, útil quando há desequilíbrio de classes (2 * (Precision * Recall) / (Precision + Recall)).
6. AUC-ROC (Area Under the Receiver Operating Characteristic Curve): Mede a capacidade do modelo de distinguir entre classes (gráfico de taxa de verdadeiros positivos contra taxa de falsos positivos).

A análise visou demonstrar a eficácia do modelo desenvolvido e identificar possíveis áreas de melhoria para futuras pesquisas.

4 AVALIAÇÃO EXPERIMENTAL

4.1 Introdução

Este capítulo fornece uma visão do processo de avaliação experimental do algoritmo desenvolvido para o filtro antispam de mensagens SMS, desde a configuração inicial do ambiente até a interpretação dos resultados, assegurando que todas as etapas sejam devidamente documentadas e analisadas para garantir a robustez e a aplicabilidade do algoritmo proposto.

A avaliação experimental foi desenvolvida na linguagem **Python** e executado através do ambiente **Colab** gratuito do Google e inclui uma série de etapas que envolvem o pré-processamento dos dados, a aplicação de diferentes modelos de aprendizado de máquina e a comparação dos resultados obtidos a partir de métricas de desempenho.

A avaliação experimental validou a eficácia dos diferentes modelos de aprendizado de máquina, garantindo que o de melhor desempenho possa ser posteriormente aplicado em um cenário real para avaliação da precisão e confiabilidade na tarefa de classificação de mensagens SMS.

O foco principal do experimento foi na extração, análise e classificação das URLs das mensagens. Essa decisão deve-se ao fato de que as URLs são chaves no processo de captura de dados sensíveis das vítimas, e portanto representam o maior risco para os usuários. Uma mensagem de SMS sem URL ou número telefônico para que a vitima interaja com o fraudador, apesar do incomodo de um Spam, não gera impactos financeiros para o usuário final.

Para escolha e extração das principais features de URL, antes do treinamento dos algoritmos, foi utilizada a referência da tabela II do artigo de (Korkmaz; Sahingoz; Diri, 2020) que identificou a analisou mais de 58 características de URL. Além disso, foi utilizada também a LLM do ChatGPT para avaliar e incluir novas características para complementar o modelo a ser testado no experimento.

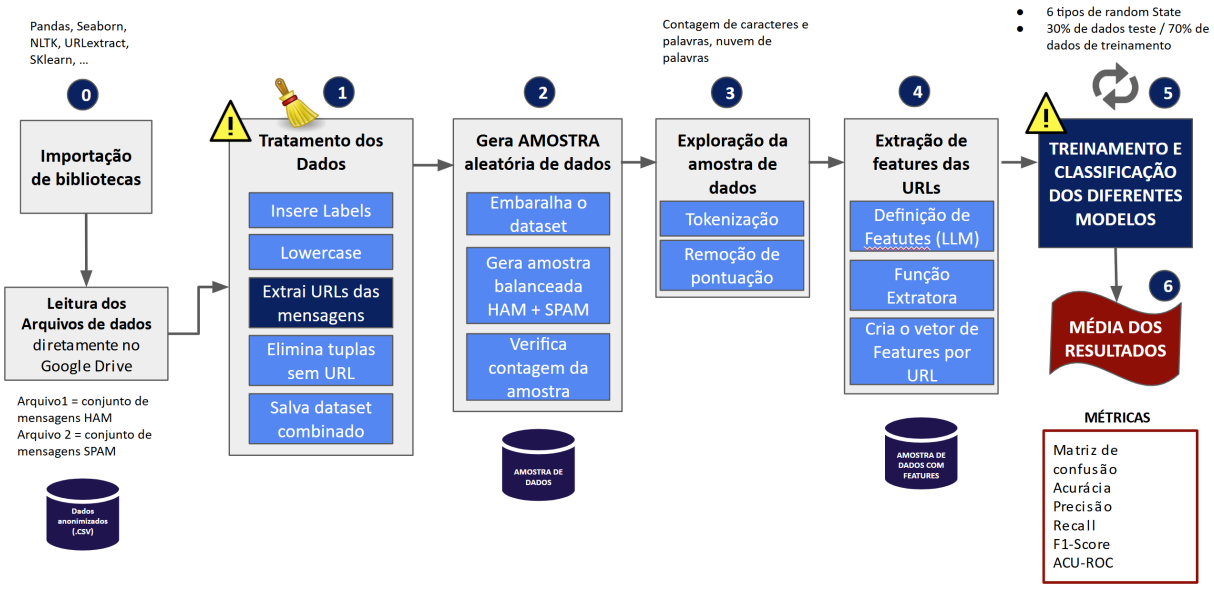
4.2 Estrutura do experimento

A estrutura deste capítulo é organizada conforme o algoritmo desenvolvido e disponibilizado em (Franco, 2024):

1. **Configuração do Ambiente:** são detalhadas as ferramentas e bibliotecas utilizadas para a implementação e execução do experimento.

2. **Leitura e tratamento dos dados:** apresenta descrição do processo de leitura e

Figura 10 – Pipeline do Código Python desenvolvido em Colab



Fonte: Elaborada pelo Autor.

tratamentos do conjunto de dados utilizado, incluindo as etapas de pré-processamento aplicadas e a geração de amostras balanceadas para a análise.

3. **Exploração da amostra do Dataset:** um visão geral do processo para avaliação da amostra do dataset gerada em uma execução específica do algoritmo.

4. **Extração das features:** descreve as características que foram extraídas das URLs.

5. **Treinamento de diferentes modelos de classificação das URLs:** mostra como os dados foram divididos entre os conjuntos de treinamento e teste, os modelos de aprendizado de máquina testados e o processo de treinamento.

6. **Resultados do experimento:** apresenta o processo de validação dos modelos de aprendizado, os resultados obtidos, incluindo a comparação do desempenho, acompanhados por gráficos e figuras que ilustram a eficácia de cada abordagem. Discute os achados do experimento, interpretando os resultados à luz dos objetivos do trabalho, identificando limitações e sugerindo melhorias no algoritmo.

4.3 Configuração do Ambiente

A linguagem Python oferece atualmente um conjunto abrangente de ferramentas prontas, o que facilita e acelera muito a manipulação de dados, tokenização, vetorização de textos e treinamento de modelos de aprendizado de máquina, que são atividades cruciais para construção e execução de experimentos com conjunto de dados.

As bibliotecas essenciais utilizadas no experimento foram:

- **pandas**: leitura e salvamento de arquivos de dados.
- **spacy**: permite utilização de GPUs e acelerar no processamento de linguagem natural (biblioteca nltk).
- **nltk**: toolkit para processamento de linguagem natural, suportando na remoção de stopwords, lematização e tokenização das palavras das mensagens SMS.
- **urlextract**: possui função pronta de extrator de URL para um texto fornecido. Importante destacar que inicialmente utilizou-se da biblioteca de **regex (re)** com apontamento dos caracteres para reconhecimento de uma URL da mensagem. Após várias tentativas e execuções, constatou-se que URLs de diversas tuplas de mensagens não eram devidamente reconhecidas, principalmente por mensagens específicas misturarem inicio e fim de URLs com conjunções de pontuações e outros caracteres especiais, dificultando a manipulação.
- **re**: fornece suporte para expressões regulares (regex), uma poderosa ferramenta para a manipulação e análise de strings. Foi utilizada para avaliar caracteres especiais das URLs extraídas.
- **wordcloud**: permite análise das amostras das mensagens SMS em formato de "nuvem de palavras".
- **scikit-learn**: oferece os principais algoritmos de aprendizado de máquina para classificação de mensagens.
- **matplotlib**: permitiu plotar os gráficos dos resultados, incluindo as matrizes de confusão.

4.4 Leitura e tratamento dos dados:

4.4.1 Leitura dos arquivos de dados

Os dados do experimento entregues eram compostos por 2 arquivos distintos de dados no formato CSV. Os dados dos arquivos estavam separados por quebra de linhas ("\n"), sendo um arquivo continha somente mensagens do tipo Spam e outro as mensagens “não Spam” ou “ham”.

Para leitura dos arquivos fornecidos, foi elaborada uma função personalizada de forma a garantir que os dados fossem carregados corretamente e estivessem prontos para processamento. O corpus foi criado separando as mensagens classificadas como “spam” e “ham”, armazenadas em dois DataFrames distintos.

Após leitura dos arquivos, cada Dataframe possui em suas tuplas uma única coluna com as mensagens de texto SMS.

4.4.2 Tratamento do arquivo de Dados

Nesta etapa foi inserido um label de texto ‘spam” e “ham” para cada Dataframe específico, e na sequência, estes foram unificados com a função "concat"do pandas em um único Dataframe.

Todas as mensagens de texto SMS foram colocadas em caixa-baixa ("lower-case") e o dataset combinado salvo em um novo arquivo CSV unificado agora contendo as mensagens e labels para cada tipo de mensagem SMS.

Foi criada uma função de extração de todas as URLs identificadas nas mensagens. A quantidade de URLs encontradas para cada mensagem foi contada e colocada em uma coluna específica do dataset conforme ilustrado na tabela 4:

Tabela 4 – Características das mensagens de fraude

label	**num_urls = 0**	**num_urls = 1**	**num_urls = 2**	**num_urls = 3**
ham	2007	661608	36339	46
spam	26999	257832	15164	4

Considerando o foco do experimento em classificar as URLs fraudulentas, as tuplas sem URL foram desconsideradas do dataset e salvas em um novo arquivo de dados. De acordo com a tabela 4, um pouco mais de 27 mil mensagens foram descartadas.

Visando evitar algum tipo de viés no treinamento dos algoritmos de aprendizado de maquina, uma vez que as mensagens se repetem no dataset, assim como garantir a criação de um conjunto balanceados de dados com mensagens "ham"e "spam", foi criada uma função que gera uma amostra de dados. Inicialmente o dataset é embaralhado com o método "sample"do dataframe e na sequencia a função gera uma amostra balanceada de acordo com o volume desejado de dados.

Como vimos na figura 7, o dataset possui uma quantidade de 300 mil mensagens identificadas como fraudes (spam). Isso permite gerar uma amostra de no máximo 600 mil mensagens de amostra balanceadas.

4.5 Exploração da amostra do Dataset

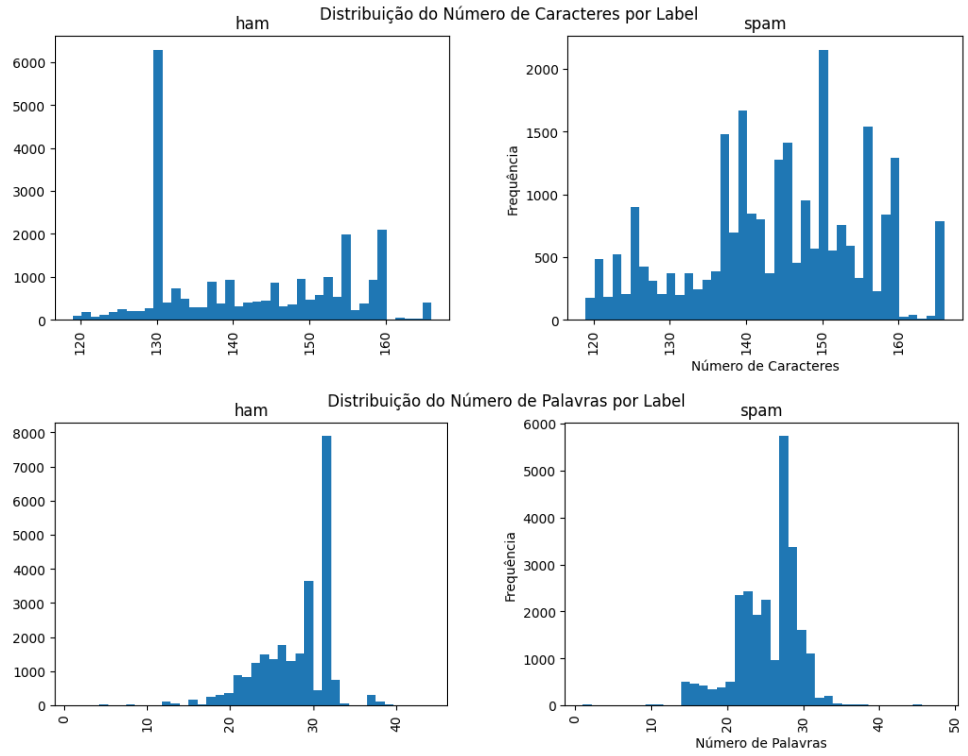
EM função da capacidade de processamento local e tempo para desenvolvimento do experiemento, foi gerado uma amostra de 50 mil mensagens de SMS balanceada. Maioria dos estudos avaliados neste trabalho considerava amostra de dados com 5 mil mensagens de SMS de dataset abertos.

Como apresentado na metologia, foram feitas explorações na amostra tokenizando as mensagens, permitindo assim a contagem das palavras e sentenças da amostra, conforme ilustrado na figura 11.

Uma nuvem de palavras foi gerada da amostra com objetivo de visualizar as palavras mais frequentes para cada tipo de label e está apresentada na figura 12

Observando os gráficos gerados, pode-se constatar uma frequência maior de mensagens do tipo "ham"com 130 caracteres, enquanto uma distribuição mais normal para as mensagens "spam". Por outro lado uma alta concentração de mais de 30 palavras para mensagens de SMS normais e também uma distribuição mais normal para mensagens Spam. Isso pode ser devido ao dataset conter uma quantidade maior de mensagens "ham"o que leva a um processo de amostragem gerar uma volume maior de amostras de mensagens SMS "ham"que possuem texto repetidos ou muito próximos.

Figura 11 – Distribuição do número de caracteres e palavras por label da amostra de dados.



Fonte: Elaborada pelo Autor.

Quando observamos a nuvem de palavras, chama atenção a forte presença de URLS com "https"nas mensagens do tipo "ham", assim como a presença de verbos de ação, como "envie"nas mensagens de "spam".

Neste último caso, os fraudadores buscam palavras que geram uma "call to action"para garantir que suas vitimas tomem uma ação de acesso de forma imediatista, por exemplo acessando um link fraudulento.

Figura 12 – Nuvem de palavras da amostra de dados.

Fonte: Elaborada pelo Autor.

4.6 Extração das Features da URL

A seguir, detalhamos as principais features extraídas de URLs, explicando seu significado e importância. As características de uma URL são extraídas tanto por meio da biblioteca do Python "urllib.parse"que quebra uma URL em componentes, quanto por meio da utilização de "regex"para encontrar caracteres especiais das URLs.

A definição das características foram feitas com base no artigo (Korkmaz; Sahingoz; Diri, 2020), e incrementadas por meio de interação via prompt junto a LLM do ChatGPT para avaliar as features propostas e sugestão de novas features para serem incluídas no modelo. A análise da relevância das características pode ser feita utilizando a correlação de Pearson ou Spearman para verificar a relação entre cada feature e a variável alvo. Foi utilizado um processo, não exaustivo, e a posteriori para avaliar a importância das features utilizando modelos de árvore de decisão Random Forest.

As 50 características extraídas estão apresentadas na tabela 5 e representam um conjunto robusto de informações que foram usadas para avaliar a legitimidade da URL.

Tabela 5 – Características extraídas das URLs

Classificação	#	Nome da Feature e Descrição
Básica	1	**url_length**: Comprimento total do URL.
Básica	2	**num_words**: Número de palavras no URL (delimitadas por caracteres alfanuméricos).
Básica	3	**num_digits**: Número de dígitos no URL.
Básica	4	**num_ampersands**: Número de caracteres "&"no URL.
Básica	5	**num_sensitive_words**: Contagem de palavras sensíveis como 'login', 'signup', 'admin' no URL.
Básica	6	**num_at_symbol**: Número de símbolos '@' no URL.
Básica	7	**num_special_chars**: Número de caracteres especiais no URL.
Básica	8	**num_punctuation**: Número de caracteres de pontuação como ':', ',', ';', ':' no URL.
Básica	9	**num_dots_subdomain**: Número de pontos no subdomínio.
Básica	10	**num_tld_paths**: Número de TLDs seguidos por uma barra ('/') no URL.
Básica	11	**num_subdomains**: Número de subdomínios no URL.
Básica	12	**num_digits_host**: Número de dígitos no hostname.
Básica	13	**num_dots**: Número total de pontos no URL.
Básica	14	**num_words_host**: Número de palavras no hostname.
Básica	15	**num_hyphens_path**: Número de hífens no caminho (path) do URL.
Básica	16	**num_double_hyphens**: Número de duplos hífens ('–') no URL.
Básica	17	**num_underscore**: Número de underscores ('_') no URL.
Básica	18	**num_dots_host**: Número de pontos no hostname.
Básica	19	**num_dots_path**: Número de pontos no caminho do URL.
Básica	20	**num_hyphens_host**: Número de hífens no hostname.
Básica	21	**num_uppercase**: Número de caracteres maiúsculos no URL.
Básica	22	**num_repeated_chars**: Número de caracteres repetidos consecutivamente três ou mais vezes no URL.
Básica	23	**num_letter_number_sequences**: Número de sequências de letras seguidas de números ou vice-versa no URL.
Booleana	24	**has_https_protocol**: Indica se o URL usa protocolo HTTPS.
Booleana	25	**has_ip_address**: Verifica se o hostname contém um endereço IP.
Booleana	26	**has_www**: Verifica se o hostname contém 'www'.
Booleana	27	**has_query**: Indica se há uma parte de query no URL.
Booleana	28	**has_redirect**: Verifica se o URL contém uma redireção ('//') após a parte do protocolo.
Booleana	29	**has_suffix**: Verifica se o URL termina com um TLD válido.
Booleana	30	**has_sensitive_words**: Indica a presença de palavras sensíveis como 'login', 'signup', 'admin'.
Booleana	31	**has_at_symbol**: Verifica se o URL contém um '@'.
Booleana	32	**has_tld**: Verifica se o hostname contém um TLD.
Booleana	33	**has_hyphen_url**: Indica se o URL contém hífen.
Comprimento	34	**length_path**: Comprimento do caminho (path) do URL.
Comprimento	35	**length_subdomain**: Comprimento do subdomínio.
Comprimento	36	**length_url_path**: Comprimento do caminho completo do URL.
Comprimento	37	**length_domain_name**: Comprimento do nome do domínio.
Comprimento	38	**length_longest_word**: Comprimento da palavra mais longa no URL.
Comprimento	39	**length_shortest_word**: Comprimento da palavra mais curta no URL.
Comprimento	40	**length_longest_word_hostname**: Comprimento da palavra mais longa no hostname.
Comprimento	41	**length_avg_word**: Comprimento médio das palavras no URL.
Comprimento	42	**length_ratio_vowel_consonant**: Razão entre o número de vogais e consoantes no URL.
Comprimento	43	**length_ratio_digit_letter**: Razão entre o número de dígitos e letras no URL.
Comprimento	44	**length_ratio_longest_shortest**: Razão entre o comprimento da palavra mais longa e a mais curta no URL.
Comprimento	45	**length_std_words**: Desvio padrão do comprimento das palavras no URL.
Razão	46	**ratio_url_path**: Razão entre o comprimento do URL e do caminho.
Razão	47	**ratio_host_path**: Razão entre o comprimento do hostname e do caminho.
Razão	48	**ratio_host_query**: Razão entre o comprimento do hostname e da query.
Adicional	49	**length_parameters**: Comprimento dos parâmetros na query do URL.
Adicional	50	**port_number**: Número da porta especificada no URL (ou -1 se não houver).

Cada característica foi colocada em uma coluna específica da URL para cada tupla, e contribui para a capacidade do modelo de identificar padrões que são comuns em URLs

fraudulentas das mensagens SMS.

A tabela 6 apresenta um exemplo do cabeçalho para as 6 primeiras tuplas do dataset, após executar a extração das features para uma amostra de 50 mil mensagens.

Tabela 6 – Amostra das Features Extraídas para seis Mensagens

Feature	**Msg 1**	**Msg 2**	**Msg 3**	**Msg 4**	**Msg 5**	**Msg 6**
label	spam	spam	spam	ham	ham	ham
url_length	24	25	20	25	23	32
num_words	4	5	4	4	4	4
num_digits	7	2	1	3	1	5
num_ampersands	0	0	0	0	0	0
num_sensitive_words	0	0	0	0	0	0
num_at_symbol	0	0	0	0	0	0
num_special_chars	2	2	2	2	2	3
num_punctuation	2	2	2	2	2	3
num_dots_subdomain	0	0	-1	0	0	0
num_tld_paths	1	1	1	0	1	1
num_subdomains	1	1	0	1	1	1
num_digits_host	0	0	0	0	0	0
num_dots	1	1	2	1	1	2
num_words_host	2	2	0	2	2	2
num_hyphens_path	0	0	0	0	0	0
num_double_hyphens	0	0	0	0	0	0
num_underscore	0	0	0	0	0	0
num_dots_host	1	1	0	1	1	1
num_dots_path	0	0	2	0	0	1
num_hyphens_host	0	0	0	0	0	0
num_uppercase	0	0	0	0	0	0
num_repeated_chars	0	0	0	0	0	1
num_letter_number_sequences	1	2	1	1	1	3
has_https_protocol	true	true	false	true	false	true
has_ip_address	false	false	false	false	false	false
has_www	false	false	false	false	false	false
has_query	false	false	false	false	false	false
has_redirect	false	false	false	false	false	false
has_suffix	false	false	false	false	false	false
has_sensitive_words	false	false	false	false	false	false
has_at_symbol	false	false	false	false	false	false
has_tld	true	true	false	false	true	true
has_hyphen_url	false	false	false	false	false	false
length_path	9	10	20	8	9	17
length_subdomain	4	4	0	4	4	4
length_url_path	9	10	20	8	9	17
length_domain_name	2	2	0	4	2	2
length_longest_word	8	7	7	7	8	15
length_shortest_word	2	1	1	4	2	2
length_longest_word_hostname	4	4	0	4	4	4
length_avg_word	4.75	3.8	4.25	5	4.5	6.5
length_ratio_vowel_consonant	0.33	0.21	0.45	0.31	0.42	0.17
length_ratio_digit_letter	0.58	0.12	0.06	0.18	0.06	0.24
length_ratio_longest_shortest	4	7	7	1.75	4	7.50
length_std_words	2.17	2.14	2.38	1.22	2.18	5.02
ratio_url_path	2.67	2.50	1	3.13	2.56	1.88
ratio_host_path	0.78	0.70	0	1.13	0.78	0.41
ratio_host_query	7	7	0	9	7	7
length_parameters	0	0	0	0	0	0
port_number	-1	-1	-1	-1	-1	-1

4.7 Modelos de classificação das URLs utilizados

Para execução do experimento inicial foi definida uma função de avaliação de cada modelo com parâmetros dos dados de entrada e saída de testes. As características e rótulos foram separados utilizando o método "train_test_split"considerando 30% das amostras como dados de teste. O algoritmo foi rodado diversas vezes para diferentes "random_state", a citar: 1, 10, 12, 42 , 20 e 5. Ao final calculou-se uma média dos indicadores de acurácia, precisão, recall, F1-Score, AUC-Score e Matriz de confusão para cada experimento.

Cada modelo possui características específicas que podem proporcionar vantagens distintas dependendo do contexto. Esses diversos modelos de aprendizado de máquina supervisionados e semi-supervisionados (ex. OC-SVM) foram avaliados para determinar os melhores indicadores na detecção de spam por meio das características extraídas das URLs.

A seguir, são descritos os modelos utilizados e suas principais vantagens comparativas. Para os modelos testados, quando não explicitado, foram mantidos os parâmetros de entrada padrões de cada método.

4.7.1 Random Forest

O *Random Forest* é um método de aprendizado de conjunto que constrói múltiplas árvores de decisão durante o treinamento e utiliza a média das predições dessas árvores para melhorar a precisão e controlar o overfitting. O Random Forest é robusto a outliers e ruídos, e é particularmente eficaz em conjuntos de dados grandes e com muitas características. Ele também oferece uma boa capacidade de generalização devido ao processo de agregação (bagging).

Nenhum parâmetro específico foi passado, utilizando os valores padrões do modelo.

4.7.2 Support Vector Machine (SVM)

A *Support Vector Machine (SVM)* é um classificador baseado em encontrar um hiperplano que melhor separa as classes no espaço de características. O modelo utilizado permite estimar probabilidades , o que é útil em tarefas onde a confiança da predição é importante. A SVM é muito eficaz em espaços de alta dimensionalidade e é particularmente útil em problemas onde as classes são claramente separáveis.

O parâmetro `(probability=True)` foi utilizado. Este parâmetro habilita a obtenção de probabilidades de predição para calcular métricas como AUC-ROC.

4.7.3 Naive Bayes

O *Naive Bayes* é um classificador probabilístico baseado no teorema de Bayes, assumindo independência entre as características. O Naive Bayes costuma ter um bom

desempenho em tarefas de classificação de texto, onde a independência entre características (como palavras) é uma suposição razoável.

Nenhum parâmetro específico foi passado para o modelo, então os valores padrões são usados.

4.7.4 K-Nearest Neighbors (KNN)

O *K-Nearest Neighbors (KNN)* é um modelo baseado em instâncias que classifica um dado ponto em função das classes dos seus vizinhos mais próximos. Além de simples de entender e implementar, pode capturar complexas relações não lineares nos dados sem a necessidade de uma suposição de distribuição subjacente.

Nenhum parâmetro específico foi passado para o modelo, então os valores padrões são usados.

4.7.5 Regressão Logística

O modelo de *Regressão Logística* é um modelo estatístico que utiliza uma função logística para modelar a probabilidade de um determinado ponto pertencer a uma classe.

O parâmetro `(max_iter=1000)` foi aumentado para garantir a convergência do modelo durante o treinamento.

4.7.6 Árvore de Decisão

Árvore de decisão é um modelo que utiliza uma estrutura em forma de árvore para tomar decisões baseadas nas características dos dados. Cada nó interno representa uma condição em uma característica, e cada folha representa uma decisão final. As árvores de decisão são fáceis de interpretar e visualizar. Elas também não exigem a normalização das características e são capazes de lidar tanto com variáveis categóricas quanto contínuas.

Nenhum parâmetro específico foi passado para o modelo, então os valores padrões são usados.

4.7.7 Gradient Boosting

O *Gradient Boosting* é um modelo de conjunto que constrói sequencialmente uma série de árvores de decisão, onde cada árvore tenta corrigir os erros das anteriores. Pode criar modelos altamente precisos, em cenários onde a acurácia é crucial.

Nenhum parâmetro específico foi passado para o modelo, então os valores padrões são usados.

4.7.8 AdaBoost

O *AdaBoost* é outro método de aprendizado de conjunto que combina vários classificadores fracos para formar um classificador forte. Ele ajusta adaptativamente os pesos dos exemplos, focando mais nos casos difíceis. AdaBoost melhora a performance de classificadores simples, como árvores de decisão de um único nó (stumps).

Nenhum parâmetro específico foi passado para o modelo, então os valores padrões são usados.

4.7.9 XGBoost

O *XGBoost* é uma implementação otimizada de Gradient Boosting, que se destaca por sua velocidade e performance. O modelo é extremamente eficiente e pode lidar com datasets grandes e complexos, oferecendo controles finos para overfitting e uma ampla gama de parâmetros ajustáveis.

Os seguintes parâmetros foram configurados:

- `(use_label_encoder=False)`: desativa o uso do label encoder (aviso do XGBoost).
- `(eval_metric='logloss')`: define a métrica de avaliação para o log loss.

4.7.10 LightGBM

O *LightGBM* é uma outra variação de Gradient Boosting que é otimizada para velocidade e uso de memória. Ele é particularmente eficaz em grandes datasets com muitas características. É conhecido por ser rápido e eficiente em termos de memória, sendo capaz de lidar com grandes volumes de dados e oferecendo uma alta acurácia.

Nenhum parâmetro específico foi passado para o modelo, então os valores padrões são usados.

4.7.11 Extra Trees

Semelhante ao Random Forest, mas com uma diferença que ele usa uma maior aleatoriedade na escolha dos pontos de corte para as árvores. Pode ser mais eficiente que o Random Forest, pois reduz o viés sem aumentar significativamente a variância, especialmente em grandes conjuntos de dados.

Nenhum parâmetro específico foi passado para o modelo, então os valores padrões são usados.

4.7.12 One-Class SVM

O *One-Class SVM* é uma versão da SVM que é utilizada para detecção de anomalias, treinada apenas em um conjunto de dados de uma única classe. É útil em situações onde

se tem exemplos de apenas uma classe (como não-spam) e se deseja identificar casos que diferem significativamente dessa classe (possível spam). Na maioria dos datasets de SMS abertos encontrados esse é um problema recorrente. Não se aplica muito ao caso deste trabalho, uma vez que retiramos uma amostra grande de dados de forma balanceada.

Os seguintes parâmetros foram configurados:

- `(kernel='rbf')`: define o kernel para ser Radial Basis Function (RBF).
- `(gamma='auto')`: define o parâmetro gamma como auto.

4.7.13 Rede Neural MLP

Adicionalmente aos modelos testados de aprendizado de máquina, foi utilizada neste trabalho uma **Multi-layer perceptron (MLP)**, modelo de aprendizado supervisionado disponível como padrão na biblioteca *scikit-learn*. A MLP é uma arquitetura de rede neural que consiste em uma camada de entrada, múltiplas camadas ocultas e uma camada de saída, onde cada neurônio aplica uma função de ativação para processar os sinais de entrada e gerar uma saída. Essa estrutura permite que a rede aprenda relações complexas entre as variáveis de entrada e a saída desejada.

Em comparação com modelos mais simples, como Regressão Logística ou Naive Bayes, a MLP pode ser capaz de aprender relações não lineares entre as características das URLs, o que é fundamental em problemas de classificação onde os dados não possuem uma separação linear.

No experimento, a MLP foi configurada com 200 neurônios em uma única camada oculta, ou seja a rede possui: (a) entrada com as 50 característica extraídas da URL, (b) 200 neurônios conectados na camada oculta e (c) 1 neurônio conectado na saída como classificador. A função de ativação utilizada foi a *ReLU* (Rectified Linear Unit), uma das mais populares devido à sua simplicidade computacional e à capacidade de mitigar o problema de gradientes. A função de ativação no neurônio de saída foi a função *sigmoid*, apropriada para classificação binária das URLs dos SMS para distinguir entre mensagens de spam e não-spam (ham), pois produz uma probabilidade entre 0 e 1.

Durante o processo de treinamento, foi utilizado o algoritmo de retropropagação, com o objetivo de ajustar os pesos das conexões entre os neurônios e minimizar o erro entre as previsões da rede e os valores reais. A métrica de erro utilizada foi a perda por entropia cruzada, que é amplamente empregada como padrão em problemas de classificação desta função.

Foi empregada a regularização *L2* buscando evitar o problema de overfitting, o que é comum em redes neurais com muitos neurônios e poucas amostras de treinamento. O treinamento foi realizado com o otimizador *adam*, que combina a vantagem da adaptação

da taxa de aprendizado, buscando um treinamento mais rápido, em especial para uma rede profunda com 200 neurônios na camada oculta.

Os seguintes parâmetros foram configurados no modelo MLP:

- `(hidden_layer_sizes=200)`: define uma camada oculta com 200 neurônios.
- `(max_iter=1000)`: número máximo de iterações para garantir a convergência do modelo.

4.7.14 Rede Neural Rede Neural LSTM

Foi incluído no experimento para efeitos comparativos uma Rede Neural do tipo Long Short-Term Memory (LSTM). As redes LSTM são um tipo especial de Redes Neurais Recorrentes (RNNs), projetadas para lidar com dependências de longo prazo em dados sequenciais. A principal vantagem das LSTMs é a sua capacidade de armazenar informações relevantes ao longo de sequências, o que as torna adequadas para aplicações em Processamento de Linguagem Natural (NLP), como classificação de textos.

A arquitetura do modelo foi implementada utilizando a biblioteca *Keras* da plataforma *TensorFlow*, que oferece uma interface de alto nível para construção de redes neurais. A seguir, cada camada da rede é descrita detalhadamente:

1. Camada de entrada: a primeira camada da rede é uma camada LSTM com 100 unidades. Esta camada é responsável por capturar as relações temporais presentes nas sequências de entrada. O uso de 100 unidades permite que a rede tenha uma boa capacidade de armazenamento de informação relevante ao longo da sequência, neste caso pelo menos 50 features processadas das URLs.

2. Camadas Dropout: foi adicionada uma camada *Dropout* com uma taxa de 30% (*rate=0.3*). O Dropout é utilizada visando a regularização que desativa aleatoriamente uma fração dos neurônios durante o treinamento, evitando o overfitting e melhor generalização dos dados.

3. Camadas Densas (Fully Connected): a rede possui quatro camadas densas (*Dense*), com 200 e 128 neurônios, respectivamente. Cada camada utiliza a função de ativação *ReLU*. Essas camadas densas têm como objetivo refinar as características extraídas pela camada LSTM, capturando relações entre as features das URLs das mensagens de SMS.

4. Camada de Saída: a última camada é uma camada densa com um único neurônio e uma função de ativação *sigmoid*. Esta camada, como feito para rede MLP, gera uma probabilidade de a mensagem ser spam ou não-spam por meio da função *sigmoid*.

Para o treinamento da rede, foi utilizado o otimizador *Adam* e a função de perda *binary_crossentropy*, que mede a diferença entre as previsões do modelo e os rótulos reais.

Os seguintes parâmetros foram configurados no modelo LSTM:

- `optimizer=Adam()`: define o otimizador como Adam.
- `loss='binary_crossentropy'`: define a função de perda como entropia cruzada binária.
- `metrics=['accuracy']`: define a métrica de avaliação como acurácia.

4.7.15 Modelo de Empilhamento (Stacking)

Um modelo de empilhamento ou *stacking*, foi utilizado no experimento como uma estratégia de combinar diferentes algoritmos de classificação e assim melhorar o desempenho geral do sistema preditivo.

O *stacking* é uma técnica de aprendizado ensemble, na qual dois ou mais modelos base são treinados separadamente e suas previsões são combinadas por um "meta-modelo"para produzir uma predição final. O objetivo desta abordagem é aproveitar as forças de cada modelo base, compensando as limitações de outros, de modo a melhorar a capacidade preditiva de acordo com as métricas avaliadas de cada modelo individualmente.

Para o modelo de empilhamento proposto, foram escolhidos os os algoritmos de classificação como estimadores base: **Random Forest, XGBoost e a rede neural MLP Classifier**.

As previsões dos estimadores base foram usadas como entrada para um meta-modelo, que neste caso é uma **regressão logística**, por se tratar de um modelo recomendado em experimentos de classificação binária. O papel do meta-modelo é aprender a combinar as previsões dos modelos base da maneira mais precisa possível, ponderando cada modelo de acordo com sua contribuição para a acurácia geral do sistema.

Os seguintes parâmetros dos estimadores foram configurados no modelo Stacking:

- **Base Estimators**:
  - **Random Forest**: utiliza os valores padrão do modelo.
  - **XGBoost**: `(use_label_encoder=False)`, `(eval_metric='logloss')` (métrica de avaliação).
  - **MLP Classifier**: `(hidden_layer_sizes=(200,))`, `(max_iter=1000)` (camada oculta e máximo de iterações).
- **Final Estimator**:

– **Logistic Regression**: Utiliza o modelo de regressão logística como estimador final com valores padrões do modelo.

4.8 Resultados do experimento

O experimento apresentou os resultados conforme tabela 7.

O algorítimo de aprendizado de máquina Randon Forest foi o que apresentou o melhor desempenho em termos de acurácia, enquanto KNN melhor precisão, o que pode ser uma grande vantagem devido ao baixo volume de falsos positivos na matriz de confusão, já que estamos em um contexto de melhor detecção de fraudes.

Por outro lado, apesar da alta sensibilidade do Naive Bayes, este não representa uma grande vantagem em uma processo de detecção de um baixo volume de falsos negativos para mensagens do tipo "ham".

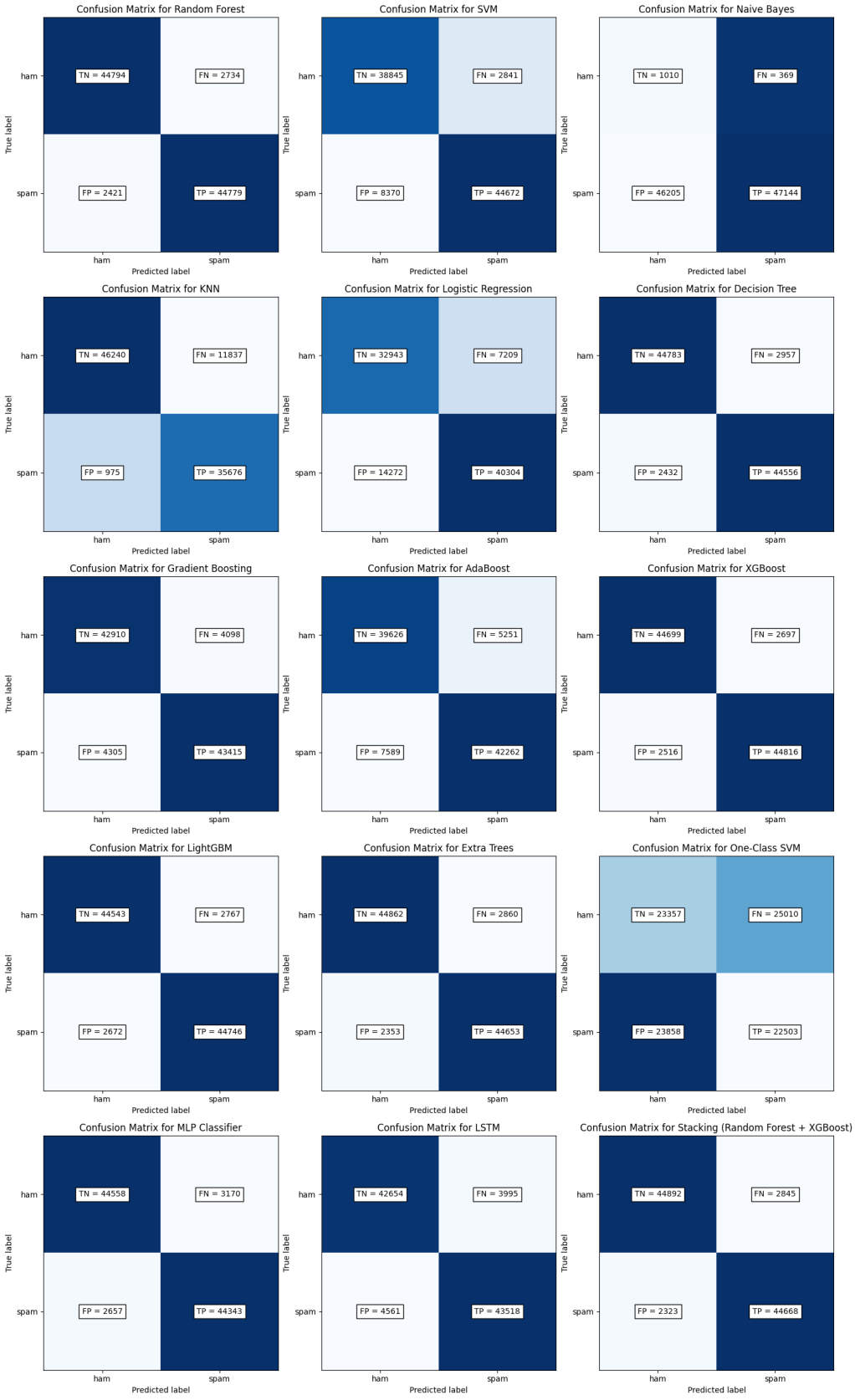
O melhor balanço entre precisão e recall (F1 Score) foi obtido pelo Randon Forest. XGBoost e a rede neural LSTM obtiveram o melhor desempenho na curva AUC-ROC, o que é excelente para avaliar a capacidade discriminativa do modelo, especialmente em cenários desbalanceados.

Tabela 7 – Média das métricas de desempenho para diferentes modelos de classificação

Modelo	**Accuracy**	**Precision**	**Recall**	**F1 Score**	**AUC-ROC**
Random Forest	**0.945581**	0.948731	0.942457	**0.945572**	0.988391
SVM	0.881651	0.842214	0.940192	0.888497	0.945033
Naive Bayes	0.508340	0.505031	**0.992235**	0.669364	0.540551
KNN	0.864750	**0.973370**	0.750821	0.847458	0.912702
Logistic Regression	0.773235	0.738532	0.848274	0.789589	0.831225
Decision Tree	0.943111	0.948272	0.937762	0.942973	0.975640
Gradient Boosting	0.911293	0.909797	0.913747	0.911759	0.973450
AdaBoost	0.864454	0.847865	0.889470	0.868129	0.943747
XGBoost	0.944969	0.946875	0.943235	0.945037	**0.988991**
LightGBM	0.942583	0.943655	0.941762	0.942703	0.987851
Extra Trees	0.944969	0.949961	0.939804	0.944845	0.985169
One-Class SVM	0.484123	0.485400	0.473624	0.479430	-
MLP Classifier	0.938487	0.943619	0.933290	0.938353	0.984310
LSTM	0.909678	0.905198	0.915902	0.910443	**0.990645**
Stacking	0.945444	0.950640	0.940125	0.945317	0.990067

Considerando acurácia e precisão dos modelos Random Forest e KNN, poderiam ser bons algorítimos para trabalharem em conjunto para otimizar a detecção de URLs fraudulentas. Por outro lado, caso a opção seja a utilização de um único algoritmo, XGBoost apresenta uma ótima opção de balanço devido a sua performance em todas as métricas.

Figura 13 – Matriz de Confusão para os diferentes modelos



Fonte: Elaborada pelo Autor.

Contrariando a hipótese inicial levantada durante a elaboração deste trabalho, o algoritmo de One-Class SVM teve uma performance muito abaixo dos demais algorítimos de machine learning para classificação de URLs Spam. Isso pode ser explicado já que o conjunto de dados utilizado é bastante expressivo e oferece amostras balanceadas suficientes para os modelos supervisionados.

Importante destacar que todos esses algoritmos possuem baixo custo computacional para treinamento comparada com Redes Neurais, em especial a rede LSTM.

O modelo MLP Classifier obteve um ótimo desempenho comparativo, dado a sua simplicidade de implementação e custo computacional de treinamento baixo.

Os resultados do modelo de empilhamento mostraram um desempenho, quando observadas as métricas combinadas, superior em relação aos modelos base isolados, destacando-se pela melhoria na métrica de F1 Score e AUC-ROC. Ao combinar as abordagens dos modelos isolados com o meta-modelo de regressão logística, o sistema pode ser capaz de gerar previsões mais precisas e robustas.

5 CONCLUSÕES

Os resultados apresentados demonstram que o modelo de Random Forest obteve o melhor desempenho em termos de acurácia (94,56%), sendo ligeiramente superior ao XGBoost e Extra Trees, que também mostraram excelente precisão, valores de F1 Score e AUC-ROC. Esses modelos se destacam não apenas pela alta performance, mas também por apresentarem um equilíbrio consistente entre recall e precisão, tornando-os potencialmente eficazes na prática de classificação de URLs como spam ou não-spam e capturando padrões complexos nas características extraídas das URLs. Além disso, os modelos Random Forest e Extra Trees se beneficiam de um custo computacional relativamente baixo em comparação com redes neurais mais profundas.

A rede neural LSTM apresentou a maior métrica de AUC-ROC (0,990645), indicando uma excelente capacidade discriminativa em distinguir entre mensagens fraudulentas e legítimas. O modelo XGBoost também se mostrou competitivo, obtendo alta acurácia e métricas AUC-ROC próximas, o que sugere que modelos de boosting são uma alternativa sólida, especialmente em contextos onde a performance precisa ser otimizada.

Os resultados do KNN, que apresentaram a maior precisão (97,34%), são valiosos em cenários onde a redução de falsos positivos é crítica, como no caso de evitar fraudes em SMS. No entanto, o recall mais baixo do KNN sugere que ele pode falhar na identificação de algumas mensagens fraudulentas, o que limita sua aplicabilidade isolada.

Modelos de redes neurais, como o MLP Classifier e o LSTM, mostraram boas capacidades de generalização e discriminação, especialmente o LSTM, que evidenciou sua eficácia em capturar padrões com alto AUC-ROC. No entanto, são modelos com maior custo computacional comparado aos demais.

Com base nos aprendizados deste trabalho e resultados obtidos, propõe-se como sugestões de tabalhos futuros:

- **Ensemble Methods Avançados**: Embora Random Forest e XGBoost já sejam exemplos de modelos ensemble, explorar outras combinações, como stacking ou blending, pode capturar a diversidade dos padrões nas URLs de forma mais robusta. Por exemplo, combinar Random Forest, XGBoost e LSTM utilizando um metamodelo para decidir a classificação final pode melhorar a acurácia e o F1 Score, aumentando a robustez contra fraudes emergentes.

- **Feature Engineering Avançada**: Investigar novas características, como informações de tráfego web, reputação de domínio, e análise semântica dos caminhos nas

URLs, pode fornecer mais contexto para os modelos, aumentando a capacidade de discriminação entre mensagens legítimas e fraudulentas.

- **Validação Prática de URLs**: Criar uma aplicação prática de validação de URLs exportando os filtros criados neste trabalho para uma API de consulta, possibilitando avaliar a eficácia do modelo no dia-a-dia da Zenvia. A aplicação poderia ser facilmente desenvolvida em um frontend como "Streamlit", oferecendo uma barra para que o usuário digite a URL, e a mesma devolveria se a URL é válida ou não.
- **Testar modelos com datasets abertos**: capturar datasets abertos utilizados em outros estudos para avaliar a eficácia do modelo treinado com o dataset da Zenvia.
- **Combinação com Análise de Texto das Mensagens**: Integrar o modelo de detecção com análise dos textos das mensagens pode ajudar a identificar padrões de spam, aumentando a capacidade de detectar mensagens fraudulentas mesmo quando os URLs não são os únicos indicadores, utilizando por exemplo stacking de filtros de URLs e mensagens para decisão.
- **Combinação com APIs de URLs Fraudulentas**: Combinar o modelo de detecção com consultas a APIs externas de URLs fraudulentas disponíveis (Exemplo: Google API Lookup - Web Risk) pode melhorar a capacidade de identificação, ampliando o contexto de detecção para incluir informações externas em tempo real.
- **Banco de dados para caching de URLs Fraudulentas**: Implementar um banco de caching para URLs fraudulentas, otimizando a consulta do filtro e acelerando o processo de verificação, contribuindo para a eficiência do sistema em ambientes de alto volume de requisições.
- **Comparar com agentes Autônomos baseados em modelos de LLM**: implementar agentes autônomos utilizando frameworks como a "Crew.ai"de forma que o agente que utiliza um ou mais modelos de linguagem, consulta a URL na internet e detecta automaticamente se a mensagem como todo é fraudulenta, comparando com a eficácia de detecção dos filtros.

Os resultados indicam que o uso de modelos robustos, como Random Forest, XGBoost, LSTM e Stacking podem ser eficazes para a detecção de URLs fraudulentas, que são grandes ofensores em fraudes em mensagens de SMS atualmente no Brasil. Evoluir nos estudos, incorporando as propostas aqui apresentadas em futuros trabalhos, permitirá desenvolver classificadores ainda mais precisos, com baixo custo computacional e resilientes frente às crescentes ameaças cibernéticas no ecossistema de comunicação móvel.

REFERÊNCIAS

ABAYOMI-ALLI, O. *et al.* A review of soft techniques for sms spam classification: Methods, approaches and applications. **Engineering Applications of Artificial Intelligence**, Elsevier, v. 86, p. 197–212, 2019.

BBC. **'Merry Christmas': 30 years of the text message**. 2022. Disponível em: https://www.bbc.com/news/technology-63825894. Acesso em: 13 abr 2024.

COCLIN, D. **Estudo revela que o Brasil é líder mundial em golpes de phishing**. 2021. Disponível em: https://securityleaders.com.br/estudo-revela-que-o-brasil-e-lider-mundial-em-golpes-de-phishing/. Acesso em: 14 abr 2024.

COSSA, O. F. *et al.* Revisão de fraudes bancárias por sms ou voz, a partir da análise de dados de telefones celulares: uma revisão sistemática de literatura. **Proceedings CISTI 2021**, IEEE, 2021.

DIGITAL, O. **INTERNET E REDES SOCIAIS: Qual recurso dos celulares faz 30 anos hoje e você ainda o usa todos os dias?** 2022. Disponível em: https://olhardigital.com.br/2022/12/03/internet-e-redes-sociais/qual-recurso-dos-celulares-faz-30-anos-hoje-e-voce-ainda-o-usa-todos-os-dias/. Acesso em: 21 jan 2024.

DIGITAL, O. **Tecnologias antigas: o que é e como funciona o SMS?** 2022. Disponível em: https://olhardigital.com.br/2022/03/09/tira-duvidas/tecnologias-antigas-o-que-e-como-funciona-o-sms/. Acesso em: 14 abr 2024.

EXAME. **SMS completa 30 anos: relembre a trajetória do serviço que (ainda) sobrevive**. 2022. Disponível em: https://exame.com/pop/sms-completa-30-anos-relembre-a-trajetoria-do-servico-que-ainda-sobrevive/. Acesso em: 10 abr 2024.

FRANCO, C. M. **Filtro AntiSpam SMS**. 2024. Accessed: August 18, 2024. Disponível em: https://github.com/cristianofranco76/DataScienceSpCourseNotes/blob/master/FIltro_AntiSpam_SMS.ipynb.

GAYOMALI, T. W. C. **The text message turns 20: A brief history of SMS**. 2015. Disponível em: https://theweek.com/articles/469869/text-message-turns-20-brief-history-sms. Acesso em: 11 mar 2024.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep learning**. [*S.l.: s.n.*]: MIT press, 2016.

INSIGHTS, B. R. **Relatório - Tamanho do mercado de SMS A2P e cPaaS**. 2024. Disponível em: https://www.businessresearchinsights.com/pt/market-reports/a2p-sms-cpaas-market-107756. Acesso em: 14 abr 2024.

JORNAL, N. **Primeiro SMS da história é vendido por R$ 693 mil**. 2021. Disponível em: https://www.nexojornal.com.br/extra/2021/12/22/Primeiro-SMS-da-hist%C3%B3ria-%C3%A9-vendido-por-R-693-mil. Acesso em: 13 mar 2024.

KORKMAZ, M.; SAHINGOZ, O. K.; DIRI, B. Detection of phishing websites by using machine learning-based url analysis. *In*: IEEE. **2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)**. [*S.l.: s.n.*], 2020. p. 1–7.

KUBILAY, E. *et al.* Can you spot a scam? measuring and improving scam identification ability. **Journal of Development Economics**, Elsevier, v. 165, p. 103147, 2023.

MISHRA, S.; SONI, D. Dsmishsms-a system to detect smishing sms. **Neural Computing and Applications**, Springer, v. 35, n. 7, p. 4975–4992, 2023.

PAPWORTH, N. **Sender of the world's first SMS.** 2012. Disponível em: https://neilpapworth.netfirms.com/. Acesso em: 10 mar 2024.

RUSSELL, S. J.; NORVIG, P. **Artificial intelligence: a modern approach**. [*S.l.: s.n.*]: Pearson, 2016.

SINCH. **FRAUD IN BUSINESS MESSAGING: É hora de acabar com a fraude de SMS.** 2023. Disponível em: https://www.sinch.com/pt-br/insights/fraud-in-business-messaging/. Acesso em: 5 jan 2024.

SPLASH, M. **Smishing Statistics: Phishing, SMS & Cybercrime**. 2023. Disponível em: https://marketsplash.com/smishing-statistics/. Acesso em: 15 abr 2024.

TECNOBLOG. **NOTICIAS: Fraudes por SMS terão resposta ainda em janeiro, diz Anatel.** 2024. Disponível em: https://tecnoblog.net/noticias/2024/01/02/anatel-preve-acoes-de-combate-a-fraudes-com-sms-ainda-em-janeiro/. Acesso em: 8 jan 2024.

TELECO. **O MERCADO DE WIRELESS MESSAGING: O Serviço de Mensagens Curtas - SMS**. 2002. Disponível em: https://www.teleco.com.br/tutoriais/tutorialmwm/pagina_3.asp. Acesso em: 13 mar 2024.

UYSAL, A. K. *et al.* The impact of feature extraction and selection on sms spam filtering. **Elektronika ir Elektrotechnika**, v. 19, n. 5, p. 67–72, 2013.

WIKIPEDIA. **UTF-16**. 2002. Disponível em: https://en.wikipedia.org/wiki/UTF-16. Acesso em: 14 abr 2024.

WIKIPEDIA. **GSM 03.38**. 2009. Disponível em: https://en.wikipedia.org/wiki/GSM_03.38#GSM_7-bit_default_alphabet_and_extension_table_of_3GPP_TS_23.038_/_GSM_03.38. Acesso em: 14 abr 2024.

WIKIPEDIA. **Friedhelm Hillebrand**. 2017. Disponível em: https://en.wikipedia.org/wiki/Friedhelm_Hillebrand#References. Acesso em: 14 abr 2024.

YERIMA, S. Y.; BASHAR, A. Semi-supervised novelty detection with one class svm for sms spam detection. *In*: IEEE. **2022 29th International Conference on Systems, Signals and Image Processing (IWSSIP)**. [*S.l.: s.n.*], 2022. p. 1–4.

ZENVIA. **ENVIAR SMS: Como enviar SMS para clientes de forma estratégica?** 2023. Disponível em: https://www.zenvia.com/blog/enviar-sms/. Acesso em: 21 jan 2024.

ZENVIA. **Texto para SMS: 6 dicas para escrever em até 160 caracteres**. 2023. Disponível em: https://www.zenvia.com/blog/texto-para-sms/. Acesso em: 12 fev 2024.