Desenvolvimento de um sistema preditivo para classificação de e-mails como spam ou não spam utilizando Máquinas de Vetores de Suporte (SVM)

DIEGO NUCCI GONÇALVES
GUSTAVO ANTONIO DA COSTA
VINICIUS GODOY MARQUES

Resumo

Com o crescimento acelerado da comunicação digital, o recebimento de mensagens indesejadas conhecidas como spam, se tornou um problema para era da informação. Esse problema afeta diversos setores, desde usuários individuais até grandes corporações, que enfrentam a necessidade de proteger suas redes e dados contra esse tipo de ameaça. Atualmente foram desenvolvidas algumas soluções para identificar os padrões nas mensagens consideradas spam, mas, no entanto, não funcionou perfeitamente, essas soluções apresentaram limitações e não detectava todos os spams. Para solucionar esse problema, nos propõe-se um sistema preditivo, que utiliza técnicas de Machine Learnig (ML). O objetivo é de classificar as mensagens com alta precisão, identificando a probabilidade de ser realmente spam, alertando os usuários sobre a chance de a mensagem ser spam. Acredita-se que assim podemos ajudar a combater esses problemas que muitas pessoas vêm enfrentando.

Palavras-chave*:* Comunicação Digital; Spam; Machine Learning; Classificação de Mensagens; Sistemas Preditivos; Detecção de Padrões; Segurança Digital; Proteção de Dados.

Development of a Predictive System for Classifying Emails as Spam or Not Spam Using Support Vector Machines (SVM)

Abstract

With the rapid growth of digital communication, the receipt of unwanted messages known as spam has become a significant problem in the information age. This issue affects various sectors, from individual users to large corporations, which face the need to protect their networks and data against this type of threat. Currently, some solutions have been developed to identify patterns in messages considered spam; however, they haven't worked properly as these solutions have limitations and have not detect all spam effectively. To solve this problem, we propose a predictive system that utilizes Machine Learning (ML) techniques. Our goal is to classify messages with high precision, identifying the likelihood that a message is indeed spam, thereby alerting users about the chances of it being spam. We believe that this can help combat the issues that many people are facing.

Keywords: *Digital Communication; Spam; Machine Learning; Message Classification; Predictive Systems; Pattern Detection; Digital Security; Data Protection.*

1 INTRODUÇÃO

A evolução da comunicação digital trouxe não apenas avanços significativos, mas também desafios consideráveis. Atualmente, muitas pessoas expressam preocupação em relação às mensagens indesejadas que recebem, especialmente em virtude de golpes e fraudes (Das; Nayak, 2013). Um problema recorrente é o fato de mensagens importantes, como *feedbacks* de empresas, frequentemente acabarem na caixa de *spam*, resultando na não visualização dessas comunicações cruciais (Siponen; Stucke, 2006). Apesar dos avanços nas tecnologias de filtragem, um grande número de mensagens indesejadas ainda consegue ultrapassar as barreiras de proteção, enquanto comunicações legítimas são equivocadamente classificadas como spam. Esse fenômeno não apenas expõe os usuários a riscos de fraudes, mas

também compromete a confiabilidade das plataformas de comunicação, dificultando a recepção de mensagens importantes (Vijayakumar; Thomas, 2024).

Com o crescimento da comunicação digital, a questão central que se impõe é: como aprimorar a precisão dos sistemas de detecção de spam para identificar mensagens indesejadas de forma mais eficiente, sem comprometer a entrega de comunicações relevantes? As ferramentas de filtragem de spam precisam se adaptar continuamente para manter sua eficácia. A tecnologia avança rapidamente e, por isso, tanto usuários quanto empresas devem se ajustar a essas mudanças (Werner W.; Werner I., 2020). A colaboração entre tecnologia e educação é essencial para garantir que essas ferramentas permaneçam eficazes e que os usuários saibam como utilizá-las corretamente, permitindo que usufruam plenamente das vantagens proporcionadas pela tecnologia (Maranholi; Santos, 2024).

Este estudo propõe o desenvolvimento de um modelo de detecção de *spam* baseado em *machine learning* e processamento de linguagem natural (PLN), com o objetivo de melhorar a precisão na filtragem de mensagens, reduzindo a ocorrência de falsos positivos e aumentando a segurança e a eficácia das comunicações digitais. A combinação de inteligência artificial com técnicas avançadas de PLN pode aprimorar significativamente a precisão dos sistemas de filtragem de spam. Algoritmos de IA podem ser treinados para reconhecer padrões complexos e características específicas de mensagens indesejadas, como palavras ou frases ambíguas e links suspeitos (Soares, 2024). Essa abordagem busca diferenciar de maneira mais eficaz o spam de comunicações legítimas, reduzindo assim o volume de mensagens fraudulentas que chegam aos usuários e promovendo um ambiente digital mais seguro e eficiente (Henke, *et al.*, 2011).

A finalidade principal desse modelo é aprimorar a segurança e a privacidade dos usuários, filtrando ameaças antes que elas cheguem à caixa de entrada e utilizando métodos de aprendizado de máquina e PLN para realizar uma análise precisa das mensagens. Com essa solução, busca-se proporcionar uma experiência digital mais segura e sem interrupções causadas por spam, promovendo a confiança dos usuários nas plataformas de comunicação eletrônica. A abordagem proposta integra dados contextuais e históricos, permitindo a detecção de padrões e características específicas de ameaças, o que aumenta a precisão na identificação de conteúdos suspeitos e minimiza a ocorrência de falsos positivos ou negativos (Murakami, 2020).

Para atingir esses objetivos, serão utilizados modelos de aprendizado de máquina, como SVM, em conjunto com técnicas como TF-IDF, que permitem a extração de características e a vetorização de texto. Dessa maneira, a IA identificará padrões associados a práticas maliciosas, incluindo links suspeitos, termos frequentemente utilizados em fraudes financeiras e conteúdos prejudiciais. A relevância deste estudo reside na necessidade premente de aprimorar os sistemas de detecção de *spam* em um contexto digital em constante evolução. O aumento das comunicações online e a proliferação de fraudes exigem soluções tecnológicas eficazes e adaptáveis que protejam os usuários e garantam a integridade das informações (Guedes; Moreira, 2023).

Acredita-se que a combinação de técnicas de aprendizado de máquina e processamento de linguagem natural pode levar a uma melhoria significativa na precisão da detecção de *spam*, minimizando o impacto de falsos positivos e aumentando a segurança das comunicações digitais (Pontes, 2024). Como aprimorar a precisão dos sistemas de detecção de spam para identificar mensagens indesejadas de forma mais eficiente, sem comprometer a entrega de comunicações relevantes?

Desse modo, o objetivo geral deste estudo é desenvolver um modelo de detecção de *spam* baseado em *machine learning* e processamento de linguagem natural, visando melhorar a precisão na filtragem de mensagens. Entre os objetivos específicos, destaca-se a análise das técnicas existentes de filtragem de spam e suas limitações, a implementação de algoritmos de

aprendizado de máquina, como SVM e Naive Bayes, para classificar mensagens, a utilização de técnicas de processamento de linguagem natural para extrair características relevantes das mensagens e a avaliação da eficácia do modelo proposto em reduzir falsos positivos e aumentar a segurança das comunicações digitais.

Este trabalho está estruturado em capítulos que contemplam a introdução apresentada, o referencial bibliográfico no capítulo seguinte, a metodologia no terceiro capítulo, o desenvolvimento no quarto, os resultados no quinto, seguidos das considerações finais e referências nos capítulos subsequentes.

2 REFERÊNCIAL BIBLIOGRÁFICO

Spam é uma prática de comunicação em que mensagens não solicitadas são enviadas em grande quantidade para vários destinatários ao mesmo tempo, geralmente com o objetivo de promover produtos, serviços ou até mesmo golpes. Ele pode aparecer em várias formas, como e-mails, mensagens de texto, comentários em redes sociais e até ligações telefônicas automatizadas. As mensagens de spam são, em sua maioria, genéricas e não segmentadas, o que significa que são enviadas para públicos amplos, sem consideração por seus interesses ou consentimento (Fabre, 2005). O *spam* geralmente é gerado por softwares automatizados que enviam as mesmas mensagens para listas extensas de contatos. Muitas vezes, essas listas são adquiridas de forma ilegal ou sem o conhecimento dos destinatários. Empresas e indivíduos que praticam o *spam* visam obter lucro com a promoção de produtos, geração de tráfego para sites ou coleta de informações pessoais. Em casos mais graves, o *spam* também pode ser utilizado para espalhar *links* maliciosos ou vírus, comprometendo a segurança do destinatário (Maia, 2021).

Esse modelo tem um papel crucial na segurança digital, já que muitos *spams* contêm mensagens indesejadas. Ao filtrar essas mensagens, o sistema reduz consideravelmente os riscos de ataques, protegendo os dispositivos e os dados pessoais dos usuários. Além disso, ao eliminar distrações causadas pelo excesso de *spam*, o modelo contribui diretamente para o aumento da produtividade, tanto em ambientes de trabalho quanto para o usuário comum (Brusiquese, 2016). Mensagens irrelevantes, que anteriormente demandariam tempo para serem deletadas ou verificadas, são automaticamente filtradas, permitindo que os usuários se concentrem em atividades importantes e economizem tempo. Por fim, a detecção eficaz de spam melhora significativamente a experiência do usuário. Com uma caixa de entrada organizada e sem a presença de mensagens indesejadas, a navegação entre e-mails importantes se torna mais simples e agradável, promovendo uma comunicação digital mais fluida e segura (Guimarães *et al*., 2023). A implementação de um modelo de detecção de *spam* é essencial para garantir a segurança, aumentar a produtividade e proporcionar uma experiência de uso mais satisfatória (Gomes, 2012).

A Processamento de Linguagem Natural (PLN) é uma área da inteligência artificial que se dedica ao estudo e desenvolvimento de técnicas e ferramentas para permitir que computadores compreendam, interpretem e gerem a linguagem humana de maneira significativa (Fiuza Bueno; Fonseca Santos, 2024). O objetivo principal do PLN é possibilitar que as máquinas consigam interagir com os seres humanos de forma natural, utilizando a linguagem verbal ou escrita de maneira fluente e eficiente. A tecnologia por trás do PLN envolve a combinação de linguística, ciência da computação e estatísticas, buscando criar algoritmos que possam lidar com a complexidade e a riqueza da linguagem humana (Guimarães, 2022). A linguagem é dinâmica, com novos termos e expressões surgindo o tempo todo. Para lidar com essas questões, as técnicas de PLN utilizam modelos computacionais que analisam grandes volumes de dados linguísticos, identificando padrões e relações semânticas.

Isso é feito por meio de métodos como análise sintática, análise semântica, reconhecimento de entidades, tradução automática e geração de texto (Ladeira, 2010).

A aplicação do PLN é ampla e se estende a diversos campos, como assistentes virtuais (como Siri e Alexa), tradutores automáticos, *chatbots*, sistemas de recomendação, análise de sentimentos, busca inteligente e muitas outras tecnologias que dependem de interações baseadas em linguagem natural (Caseli; Nunes, 2024). A capacidade de entender o contexto de uma conversa, reconhecer intenções, gerar respostas coerentes e adaptar-se a novas informações são algumas das características que tornam o PLN uma das áreas mais inovadoras e promissoras da inteligência artificial (Duque-Pereira; Moura, 2023). Com o avanço das técnicas de aprendizado de máquina, especialmente os modelos baseados em redes neurais profundas, os sistemas de PLN têm alcançado resultados cada vez mais sofisticados (Rosseto, 2024). Esses avanços estão transformando a maneira como interagimos com a tecnologia e têm o potencial de alterar diversos setores da sociedade, desde a educação até os negócios, passando pela medicina e as ciências sociais (Rocha, 2023).

Inteligência Artificial (IA) é um ramo da ciência da computação que busca desenvolver sistemas capazes de realizar tarefas que normalmente exigiriam inteligência humana, como aprender, raciocinar, resolver problemas, perceber o ambiente e tomar decisões (Silva *et al.*, 2024). O objetivo da Inteligência Artificial é criar máquinas que possam simular aspectos do comportamento humano, permitindo que elas executem funções complexas de maneira autônoma ou assistida, com ou sem a intervenção direta de um ser humano (Melo; Guerra; Silva, 2024). Entre as principais técnicas utilizadas na Inteligência Artificial estão o aprendizado de máquina (*machine learning*), onde algoritmos são treinados a partir de grandes volumes de dados para identificar padrões e tomar decisões (Queiroz; Disconzi, 2024). Redes neurais artificiais, inspiradas no funcionamento do cérebro humano, que são aplicadas em tarefas como reconhecimento de voz e imagem e o aprendizado profundo (*deep learning*), uma subárea do aprendizado de máquina que utiliza redes neurais complexas para resolver problemas de alta complexidade, como a tradução automática e a geração de texto (Rover, 2024).

As aplicações da Inteligência Artificial estão presentes em diversas áreas, como saúde, onde pode ajudar no diagnóstico de doenças e na personalização de tratamentos, na indústria, com a automação de processos e otimização de produção, no setor financeiro com análise preditiva e prevenção de fraudes e no cotidiano, por meio de assistentes virtuais e sistemas de recomendação (Nabeto, 2020). A Inteligência Artificial tem o potencial de transformar vários setores, oferecendo soluções inovadoras, mais rápidas e precisas (Soares, 2024). *Machine Learning* (Aprendizado de Máquina) é uma área da Inteligência Artificial (IA) que torna possível fazer com que os computadores aprendam a realizar tarefas de uma forma autônoma, sem serem programados de forma específica para cada uma delas, o que exigiria muito tempo (Soori; Arezoo; Dastres, 2023). O *Machine Learning* pode ser classificado em diferentes tipos, dentre eles estão o Aprendizado Supervisionado, Aprendizado Não Supervisionado e o Aprendizado por Reforço, esses são alguns dos mais utilizados e cada um funciona de uma forma diferente (Morales; Escalante, 2022). No Aprendizado Supervisionado, o algoritmo é treinado com um conjunto de dados rotulados, isso quer dizer que cada exemplo de treinamento que for inserido terá uma entrada e uma saída esperada. O objetivo desse método é que o algoritmo aprenda a associar as entradas com as saídas de uma forma correta (Sen, Hajra e Ghosh, 2020).

Máquinas de Vetores de Suporte, ou SVM, é um algoritmo de Aprendizado de Máquina Supervisionado, utilizado tanto para classificação quanto para regressão. Ele é muito preciso e uma das ideias principais do SVM é encontrar o melhor hiperplano que separe os dados em diferentes classes (Sen, Hajra e Ghosh, 2020). O Hiperplano é uma fronteira que faz as separações das classes. Quando houver a existência de um espaço bidimensional, o hiperplano

será uma linha, em um espaço tridimensional, o hiperplano será plano, e em espaços de dimensões superiores, ele será um hiperplano. Esse é um dos conceitos chaves de SVM (Bharadwaj, 2021). Usar uma combinação de SVM e PLN é uma maneira perfeita para lidar com dados textuais e realizar classificação de forma eficiente. O SVM, que possui uma excelente capacidade de lidar com altas dimensionalidades, tem sido amplamente utilizado em conjunto com o Processamento de Linguagem Natural (PLN) para organizar e transformar dados, sendo aplicado em problemas como classificação de documentos, análise de sentimentos e, claro, na detecção de *spam* em e-mails. O SVM é muito escolhido como modelo de classificação em PLN, principalmente por ser robusto e eficaz, especialmente quando é necessário lidar com grandes volumes de dados e textos (Dogra *et al*., 2022).

As métricas de avaliação de modelos de *Machine Learning* são de extrema importância para entendermos como o modelo se comporta visando seu objetivo e analisarmos se ele está performando de forma assertiva. Existem várias métricas, mas as principais para classificação são: *Accuracy; Precision; Recall; F1-Score*; AUC-ROC; Matriz de Confusão (Brito *et al.*, 2022). *Accuracy* (Acurácia) é uma métrica utilizada quando as classes estão balanceadas, ela mede a proporção de previsões corretas realizadas pelo modelo em relação ao total de previsões feitas (Aldania *et al.*, 2023). *Precision* (Precisão) é importante quando falsos positivos têm um custo alto, por exemplo, classificar algo como positivo quando na verdade ele não é. O Precision mede a proporção de previsões positivas corretas em relação ao total de predições positivas feitas pelo modelo (CS. LG, 2020). *Recall* (Taxa de Verdadeiros Positivos) mede a proporção de verdadeiros positivos, sendo importante quando falsos negativos têm um custo alto, ou seja, quando perder uma instância positiva pode ter consequências graves (Forman, 2008). *F1-Score* é uma medida harmônica entre o *Precision* e o *Recall*. Ele é uma métrica equilibrada, que leva em consideração os falsos positivos e os falsos negativos. Ele também é utilizado quando existe a necessidade de balancear o *Precision* e o *Recall*, principalmente em classes desbalanceadas (Chicco; Jurman, 2020).

A AUC-ROC (Área sob a Curva ROC) é uma métrica que mede a capacidade do modelo de distinguir entre as classes. Essa curva ROC é um gráfico capaz de mostrar a taxa de verdadeiros positivos, contra a taxa de falsos positivos. Essa métrica é útil quando for necessária uma visão geral da capacidade do modelo em lidar com diferentes limiares de decisão (Acevedo *et al.*, 2012). A Matriz de Confusão é uma tabela que descreve o desempenho do modelo de classificação, comparando as classes reais com as classes previstas. A matriz contém 4 valores principais: TP (*True Positives*); FP (*False Positives*); TN (*True Negatives*); FN (*False Negatives*). Ela é utilizada para entender os tipos de erros cometidos pelo modelo e pode ajudar a identificar se o modelo está mais propenso a cometer falsos positivos ou falsos negativos (Kuhn; Johnson, 2013).

2.1 Trabalhos Relacionados

A classificação de e-mails como *spam* ou não *spam* tem sido amplamente estudada na literatura, com diversas abordagens baseadas em aprendizado de máquina sendo propostas para melhorar a precisão dos filtros. Entre essas técnicas, as Máquinas de Vetores de Suporte (SVM) têm se destacado devido à sua eficácia em problemas de classificação binária, como é o caso da detecção de *spam*.

Olatunji (2019) propôs um modelo aprimorado de detecção de spam baseado em SVM, alcançando uma precisão de 95,87% no conjunto de treinamento e 94,06% no conjunto de teste, superando modelos anteriores em 3,11%. O autor destacou a importância da seleção adequada dos parâmetros do SVM para otimizar o desempenho do classificador. Em um trabalho anterior, Olatunji (2017) comparou o desempenho de SVM com Máquinas de Aprendizado Extremo

(ELM), concluindo que, embora o SVM tenha obtido maior precisão na classificação, o ELM foi significativamente mais rápido em termos de tempo de processamento.

Amayri e Bouguila (2010) investigaram diferentes *kernels* baseados em distância para SVM, demonstrando que *kernels* de *string* são particularmente eficazes na filtragem de spam. Além disso, os autores propuseram um *framework* ativo online para classificação em tempo real, alcançando altas taxas de precisão e *recall*. Singh, Pamula e Shekhar (2018) avaliaram o desempenho de SVM com diferentes funções de *kernel*, comparando os *kernels* linear e gaussiano no *dataset SpamAssassin*. Os resultados indicaram que a escolha do *kernel* influencia diretamente a eficácia do classificador, com o *kernel* gaussiano apresentando melhor desempenho em determinados cenários. Roy *et al*. (2018) exploraram o uso de Deep SVM, SVM tradicional e Redes Neurais Artificiais (RNA) para classificação de *spam*. O estudo mostrou que o Deep SVM superou os outros modelos em termos de precisão, reforçando a viabilidade de abordagens híbridas para detecção de spam.

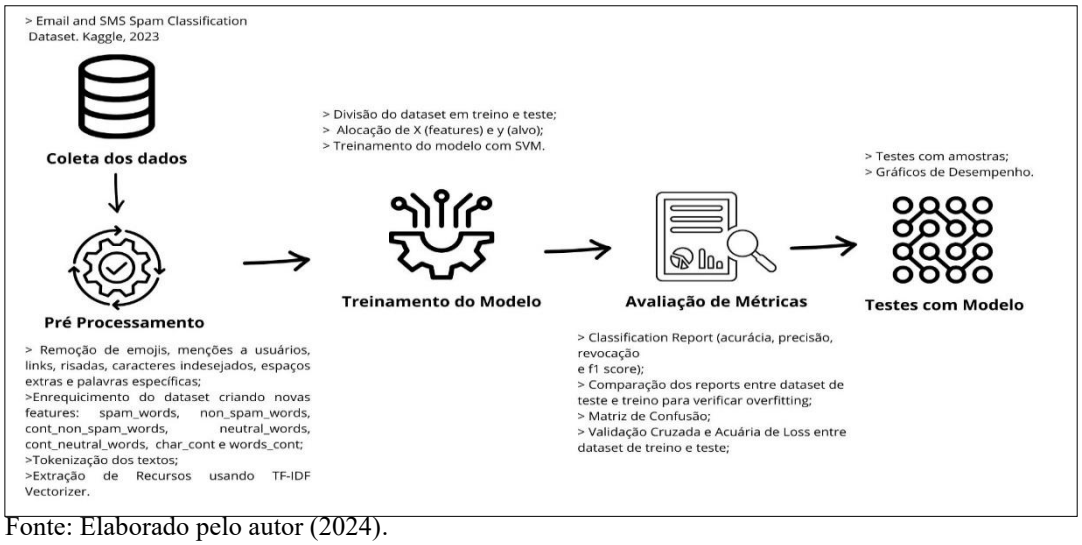
Esses trabalhos demonstram que o SVM é uma técnica robusta para classificação de spam, mas sua eficácia depende da seleção adequada de *kernels*, parâmetros e estratégias de pré-processamento de dados.

3 MÉTODOS E MATERIAIS

O trabalho foi desenvolvido em um ambiente de programação interativo, utilizando ferramentas essenciais para a manipulação e análise de dados. O Google Colab foi escolhido como plataforma principal para o desenvolvimento do projeto, pois oferece um ambiente de *notebooks* que facilita a execução de código em Python, além de permitir o uso de infraestrutura em nuvem, essencial para o processamento de modelos complexos sem a necessidade de recursos locais robustos. A linguagem Python foi utilizada para a implementação dos algoritmos e manipulação de dados, uma vez que dispõe de uma ampla coleção de bibliotecas dedicadas à ciência de dados e *machine learning*, abrangendo desde o pré-processamento até a avaliação dos modelos.

As bibliotecas Pandas e Numpy foram fundamentais para a manipulação dos dados. A biblioteca Pandas permitiu a leitura, limpeza e transformação do *dataset*, enquanto o Numpy viabilizou operações matemáticas e a manipulação eficiente de *arrays*. Além disso, a biblioteca Scikit-Learn, ou Sklearn, foi empregada para a construção e avaliação do modelo de machine learning. Com ela, foi possível implementar o modelo de classificação por *SVM (Support Vector Machine)*, realizar a divisão dos dados em conjuntos de treino e teste, e aplicar métricas de avaliação, como matriz de confusão.

Figura 1 – Resumo dos métodos e materiais utilizados no desenvolvimento do trabalho.



Fonte: Elaborado pelo autor (2024).

A metodologia seguiu os seguintes passos. Primeiramente, o *dataset "Email and SMS Spam Classification"*, disponível no Kaggle em 2023, foi utilizado como base para a criação do modelo de classificação de spam. Esse *dataset* contém informações que auxiliam na identificação de mensagens classificadas como spam ou não spam. Na etapa de pré-processamento dos dados, foram removidos elementos irrelevantes, como emojis, menções, links, risadas e caracteres indesejados, para garantir que apenas informações relevantes fossem utilizadas. O *dataset* foi enriquecido com novas variáveis que representam contagens de palavras e caracteres específicas para cada classe, com o intuito de aumentar a precisão do modelo. Os textos foram, então, *tokenizados*, e os recursos foram extraídos por meio do vetorizador TF-IDF, que converte palavras em representações numéricas.

Após o pré-processamento, os dados foram divididos em conjuntos de treino e teste. O modelo de SVM foi então treinado utilizando as variáveis selecionadas como preditores e o target definido como a classificação de spam ou não spam. Em seguida, o desempenho do modelo foi avaliado com o uso de várias métricas. O *Classification Report* incluiu métricas como acurácia, precisão, revocação e f1 score, permitindo uma avaliação detalhada do modelo. A matriz de confusão foi utilizada para identificar erros de classificação entre as classes, enquanto a validação cruzada foi aplicada para avaliar a estabilidade do modelo entre os *datasets* de treino e teste.

Por fim, a fase de testes envolveu a realização de avaliações adicionais com amostras específicas para validar o desempenho do modelo em condições variadas. Além disso, gráficos de desempenho foram criados para visualizar melhor os resultados das métricas em diferentes cenários. Essa metodologia permitiu a construção de um modelo preditivo robusto para a classificação de mensagens spam, utilizando ferramentas amplamente aplicadas na área de machine learning. O uso do Google Colab e das bibliotecas Python garantiu um fluxo de trabalho eficiente e acessível, possibilitando o desenvolvimento e a avaliação completa do modelo de classificação.

4 DESENVOLVIMENTO

A coleta de dados para o presente trabalho foi realizada a partir do *e-mail and SMS Spam Classification Dataset*, disponível na plataforma *Kaggle*. O conjunto de dados contém mensagens de e-mails e SMS categorizadas como spam ou não spam, e foi desenvolvido com

o objetivo de facilitar a construção de modelos preditivos robustos voltados à detecção de spam. Esse *dataset* inclui uma variedade de mensagens realistas, permitindo que modelos de aprendizado de máquina sejam treinados e avaliados de forma eficaz para identificar conteúdo indesejado (Patil, 2023).

Na fase de pré-processamento do *dataset* de e-mails e SMS, foi implementada uma função com o objetivo de limpar e padronizar o conteúdo das mensagens, preparando-as para a classificação como *spam* ou não *spam*. A função realiza várias etapas de limpeza para remover elementos irrelevantes que podem interferir na análise e no treinamento dos modelos de *machine learning*. Primeiramente, a função remove *emojis*, utilizando uma conversão de caracteres que ignora elementos não-ASCII, garantindo que apenas texto legível seja mantido. Em seguida, menções a usuários (geralmente representadas por "@usuário" em e-mails ou SMS) são eliminadas, pois essas referências são desnecessárias para o processo de classificação. *Links* também são removidos do texto, já que URLs não contribuem diretamente para o conteúdo textual significativo que será avaliado, especialmente no contexto de detecção de *spam*. A função, além disso, elimina espaços extras, substituindo qualquer sequência de espaços em branco por um único espaço, garantindo que o texto seja limpo e padronizado. Outra etapa importante é a remoção de palavras específicas, como "user" e "https", que geralmente são encontradas em mensagens automatizadas ou formatadas e não fornecem valor para o modelo de classificação.

Após essa série de transformações, a função é aplicada ao conteúdo das mensagens no *dataset*, resultando em uma versão limpa e padronizada de cada e-mail ou SMS, pronta para análise. Foi encontrado a necessidade de enriquecimento do *dataset* tratado com novas *features* para o algoritmo de machine learning treinar o modelo. Abaixo a Figura 2 demonstra o *dataset* pós tratamento inicial.

Figura 2 – *Dataset* após o tratamento.

	message_content	is_spam
0	Hello Lonnie, Just wanted to touch base regard...	0
1	Congratulations, you've won a prize! Call us n...	1
2	You have been pre-approved for a credit card w...	1
3	Limited time offer, act now! Only a few spots ...	1
4	Your loan has been approved! Transfer funds to...	1
5	You have been selected to receive a special of...	1

Fonte: Elaborado pelo autor (2024).

O processo de enriquecimento do *dataset* envolveu várias etapas, onde foram criadas novas *features* com base na análise do conteúdo textual de e-mails. Inicialmente, foram estabelecidas duas listas principais de palavras: uma contendo termos considerados *spam* e outra contendo termos não *spam*, com base na análise dos e-mails classificados previamente. Essas listas permitiram a construção de funções que verificam a presença de palavras dessas categorias nos textos dos e-mails. A primeira etapa foi a criação de uma função que verifica se um e-mail contém alguma palavra da lista de *spam*. Essa função serviu como base para gerar a nova *feature* chamada *spam_words*, que indica se o conteúdo de um e-mail contém pelo menos uma palavra identificada como *spam*. Em seguida, foi desenvolvida uma função para contar quantas palavras de *spam* estão presentes em cada e-mail, resultando na *feature cont_spam_words*, que armazena essa contagem para cada mensagem. De maneira análoga, foi criada uma função para verificar se uma mensagem contém palavras da lista de não *spam*, gerando a *feature non_spam_words*, que atua de forma semelhante à de *spam*, mas focada nas

palavras não associadas a mensagens de *spam*. Também foi criada uma função para contar quantas dessas palavras de não *spam* estão presentes em cada e-mail, resultando na *feature cont_non_spam_words*.

Adicionalmente, observou-se que algumas palavras poderiam ser neutras, ou seja, aparecerem tanto em e-mails de *spam* quanto de não *spam*. Assim, foi desenvolvida uma função para identificar essas palavras neutras, gerando a feature *neutral_words*. Para complementar essa análise, uma função foi criada para contar quantas dessas palavras neutras estão presentes em cada e-mail, culminando na feature *cont_neutral_words*. Para enriquecer ainda mais a análise do conteúdo textual, foram incluídas duas novas features quantitativas. A primeira, *char_cont*, registra a contagem de caracteres presentes em cada e-mail. A segunda, *words_cont*, calcula o número de palavras contidas em cada mensagem. Essas últimas *features* ajudam a compreender melhor a estrutura das mensagens em termos de tamanho e conteúdo, fornecendo mais informações ao modelo de classificação. Na Figura 3 pode-se verificar como o *dataset* ficou após o enriquecimento.

Figura 3 – Dataset com novas features após o enriquecimento.

	message_content	is_spam	spam_words	cont_spam_words	non_spam_words	cont_non_spam_words	neutral_words	cont_neutral_words	char_cont	words_cont
0	Hello Lonnie, Just wanted to touch base regard...	0	1	15	1	60	1	15	387	60
1	Congratulations, you've won a prize! Call us n...	1	1	36	1	21	1	21	202	36
2	You have been pre-approved for a credit card w...	1	1	38	1	27	1	27	213	38
3	Limited time offer, act now! Only a few spots ...	1	1	30	1	18	1	18	187	30
4	Your loan has been approved! Transfer funds to...	1	1	32	1	18	1	18	196	32
5	You have been selected to receive a special of...	1	1	35	1	23	1	23	194	35

Fonte: Elaborado pelo autor (2024).

Após o enriquecimento do *dataset* foi feito a *tokenização* dos textos para dividir o conteúdo das mensagens em unidades menores, chamadas *tokens*, que geralmente correspondem a palavras individuais. Para isso, aplicou-se a função de *tokenização* aos textos presentes na coluna *message_content*. Essa técnica é essencial para transformar o texto bruto em uma forma que pode ser mais facilmente processada por algoritmos de *machine learning*. Ao aplicar a função *word_tokenize*, cada mensagem foi dividida em uma lista de palavras, permitindo uma análise mais detalhada de seu conteúdo. O resultado dessa *tokenização* foi armazenado na nova coluna *tokens*, onde cada mensagem agora é representada por uma sequência de *tokens*, facilitando a manipulação e exploração dos dados textuais.

Figura 4 – Novas *feature tokens* que contém a separação de cada mensagem.

```
                                                tokens
[Hello, Lonnie, ,, Just, wanted, to, touch, ba...
[Congratulations, ,, you, 've, won, a, prize, ...
[You, have, been, pre-approved, for, a, credit...
[Limited, time, offer, ,, act, now, !, Only, a...
[Your, loan, has, been, approved, !, Transfer,...
                                                ...
[Hello, Virginia, ,, It, was, great, to, catch...
[Final, notice, :, Claim, your, inheritance, f...
[Hot, singles, in, your, area, want, to, chat,...
[Your, loan, has, been, approved, !, Transfer,...
[Dear, Brian, ,, I, wanted, to, let, you, know...
```

Fonte: Elaborado pelo autor (2024).

Após a *tokenização* foi feita a extração de recursos utilizando a técnica *TF-IDF (Term Frequency-Inverse Document Frequency)*, empregando o *TfidfVectorizer*. Essa abordagem converte documentos de texto em uma matriz numérica, refletindo a importância de cada palavra em relação ao conjunto total de documentos. O *TF* mede a frequência de uma palavra em um documento, enquanto o *IDF* diminui o peso de palavras comuns que aparecem em muitos documentos, destacando aquelas mais exclusivas. O resultado é uma representação

vetorial que permite a aplicação de algoritmos de classificação, facilitando o aprendizado a partir dos dados textuais.

Após a fase de pré-processamento de dados foi feito um teste de algoritmos de classificação que envolveu a instância de diversos algoritmos, incluindo *Random Forest*, Regressão Logística, Máquina de Vetores de Suporte, *K-Nearest Neighbors*, Árvore de Decisão e *Naive Bayes*. Cada modelo foi treinado com os dados de treinamento (`X_train`, `y_train`) e, em seguida, avaliado com os conjuntos de treinamento e teste. Para cada modelo, foram calculadas as acurácias de treinamento e teste, além de relatórios de classificação que resumem o desempenho em métricas como precisão e recall. Os resultados foram armazenados em uma lista, que foi posteriormente convertida em um DataFrame para facilitar a análise. O DataFrame foi ordenado pela acurácia no teste e formatado condicionalmente, destacando os três melhores modelos em cores distintas. Por fim, o DataFrame formatado foi exibido, permitindo uma comparação clara do desempenho dos diferentes modelos de classificação.

Figura 5 – Os três melhores algoritmos testados.

	Modelo	Acurácia Treino	Acurácia Teste	precision	recall	f1-score	support
0	Random Forest	1.000000	1.000000	1.000000	1.000000	1.000000	200.000000
1	Logistic Regression	1.000000	1.000000	1.000000	1.000000	1.000000	200.000000
2	Support Vector Machine	1.000000	1.000000	1.000000	1.000000	1.000000	200.000000

Fonte: Elaborado pelo autor (2024).

A escolha da Máquina de Vetores de Suporte (SVM) após a Regressão Logística e o *Random Forest* terem resultados semelhantes se justifica por várias razões. O SVM se destaca pela sua eficácia em lidar com dados de alta dimensionalidade, uma característica comum em conjuntos textuais. Essa capacidade torna o SVM uma opção atraente para tarefas de classificação, onde as relações entre as características podem ser complexas e não lineares. Além disso, o SVM maximiza a margem de separação entre as classes, o que contribui para uma maior robustez e potencializa a generalização do modelo. Isso significa que ele tende a performar melhor em dados não vistos, pois busca encontrar o limite de decisão que se distancia o máximo possível das classes de treinamento.

Após a seleção do algoritmo de Máquina de Vetores de Suporte (SVM) como o mais adequado para o problema, foi realizado o treinamento do modelo. Um modelo SVM foi criado utilizando uma abordagem linear, permitindo que o algoritmo aprendesse a classificar os dados entre as categorias de spam e não-*spam*. Uma vez que o modelo foi treinado com os dados, ele foi testado em um conjunto separado, conhecido como conjunto de teste, os resultados serão apresentados no próximo capítulo. Esse teste foi crucial para avaliar a capacidade do modelo de fazer previsões em dados que não foram utilizados durante o treinamento. O modelo também fez previsões no conjunto de treinamento, o que permitiu verificar sua performance em dados que já conhecia. Esses processos garantiram que o modelo não apenas aprendesse os padrões dos dados, mas também conseguisse aplicar esse aprendizado a novos exemplos de forma eficaz.

5 RESULTADOS

Os resultados obtidos com o modelo de classificação SVM foram analisados por meio de métricas de desempenho, incluindo precisão, revocação, f1-*score* e acurácia, tanto para o conjunto de teste quanto para o conjunto de treino. A imagem de métricas do conjunto de teste apresenta resultados com valores perfeitos para todas as classes: a classe "0" (não *spam*) e a classe "1" (*spam*) atingiram precisão, revocação e f1-*score* de 1.00, com acurácia total de 100%

no conjunto de dados de teste. Esses valores demonstram que o modelo foi capaz de identificar corretamente todas as mensagens, sem falsos positivos ou falsos negativos, o que sugere uma alta eficiência na classificação de novas instâncias. A média macro e a média ponderada das métricas também atingiram valores perfeitos de 1.00, o que confirma a consistência dos resultados em ambas as classes.

Figura 6 – *Relatório de classificação do conjunto de dados* de Testes.

```
Classification Report (Test Set):
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        99
           1       1.00      1.00      1.00       101

    accuracy                           1.00       200
   macro avg       1.00      1.00      1.00       200
weighted avg       1.00      1.00      1.00       200
```

Fonte: Elaborado pelo autor (2024).

Resultados semelhantes foram obtidos no conjunto de treino, onde o modelo também apresentou valores perfeitos para precisão, revocação e f1-*score* nas duas classes, resultando novamente em uma acurácia de 100%. Esses resultados indicam que o modelo não apenas se ajustou adequadamente aos dados de treino, mas também generalizou eficazmente ao conjunto de teste, o que minimiza preocupações de sobreajuste.

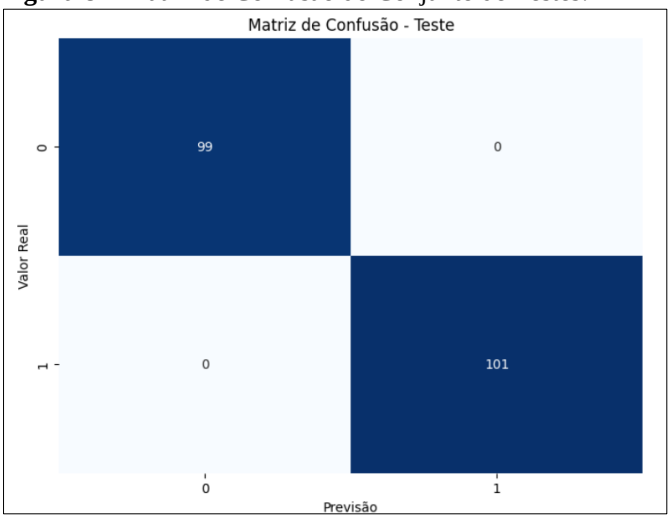
Figura 7 – Relatório de classificação do conjunto de dados de Treino.

```
Classification Report (Training Set):
              precision    recall  f1-score   support

           0       1.00      1.00      1.00       401
           1       1.00      1.00      1.00       399

    accuracy                           1.00       800
   macro avg       1.00      1.00      1.00       800
weighted avg       1.00      1.00      1.00       800
```

Fonte: Elaborado pelo autor (2024).

A precisão do modelo SVM, tanto nos dados de treino quanto nos de teste, reforça a sua capacidade de distinguir eficazmente entre mensagens de *spam* e não *spam*, o que pode ser um indicativo da adequação desse algoritmo para a tarefa de classificação de mensagens em contextos semelhantes. A matriz de confusão para o conjunto de teste fornece uma visão detalhada do desempenho do modelo na tarefa de classificação entre as classes "*spam*" e "não *spam*". A matriz mostra que o modelo classificou corretamente todas as instâncias de teste, sem cometer erros. Na diagonal principal, observa-se que 99 instâncias da classe "0" (não *spam*) foram corretamente identificadas como "não *spam*", enquanto 101 instâncias da classe "1" (*spam*) foram corretamente classificadas como "*spam*". Não houve falsos positivos (mensagens classificadas incorretamente como *spam*) nem falsos negativos (mensagens de *spam* classificadas como não *spam*). Esses resultados reforçam o alto desempenho do modelo, evidenciando que ele conseguiu distinguir com precisão entre as mensagens das duas classes. A matriz de confusão demonstra a capacidade do modelo SVM em realizar uma classificação confiável e sem erros, o que se alinha com as métricas perfeitas de precisão, revocação e f1-*score* observadas no relatório de classificação.

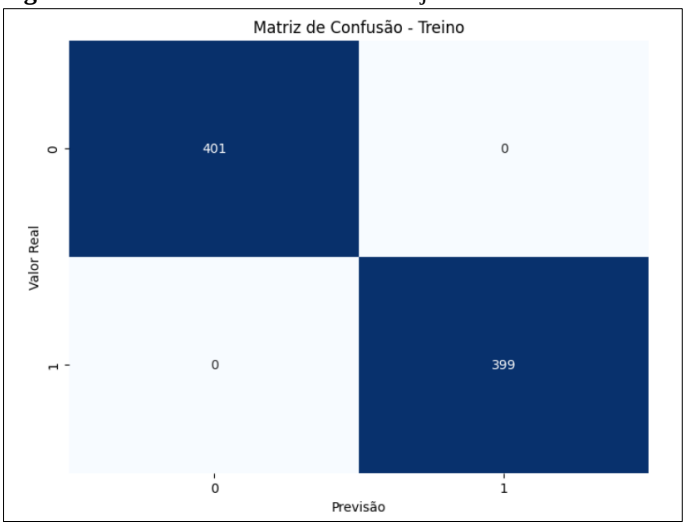
Figura 8 – Matriz de Confusão do Conjunto de Testes.



Fonte: Elaborado pelo autor (2024).

Na matriz de confusão do conjunto de treino, os resultados refletem uma performance perfeita do modelo, sem erros de classificação. Na diagonal principal, observa-se que todas as 401 instâncias da classe "0" (não *spam*) foram corretamente classificadas como "0", assim como todas as 399 instâncias da classe "1" (*spam*) foram corretamente identificadas como "1".

Figura 9 – Matriz de Confusão do Conjunto de Treino.



Fonte: Elaborado pelo autor (2024).

Esse resultado evidencia a eficácia do modelo durante o treinamento, mostrando sua capacidade de distinguir precisamente entre mensagens de *spam* e não *spam*. No entanto, a precisão perfeita no conjunto de treino pode sugerir a possibilidade de *overfitting*, ou seja, o modelo pode ter memorizado as características específicas desse conjunto em vez de generalizar para novos dados. Após obter boas métricas no modelo, foi implementado um selecionador randômico para testar a classificação de e-mails. Esse recurso escolhe aleatoriamente um e-mail do conjunto de dados e exibe seu conteúdo junto com a classificação feita pelo modelo (*Spam* ou Não *Spam*) permitindo que o modelo fosse testado posteriormente. Na Figura 10 pode-se ver o funcionamento do selecionador randômico. Em seguida, o e-mail é convertido em uma representação vetorial (usando o vetorizador treinado) e enriquecido com algumas características adicionais, como contagem de palavras específicas e caracteres. Essas informações são então combinadas e passadas para o modelo de classificação, que retorna a previsão: "*Spam*" ou "Não *Spam*".

Figura 10 – Selecionador Randômico de e-mails para teste do modelo.

```
Registro aleatório escolhido para teste de classificação: 477

E-mail (Registro 477):
Hello Ashley, It was great to catch up with you earlier. Lets discuss the next steps

Classificação: Não Spam
```

Fonte: Elaborado pelo autor (2024).

Foi desenvolvido um algoritmo que permite ao modelo classificar novos e-mails como *Spam* ou Não *Spam*. Esse processo envolve a entrada do texto do e-mail, no teste abaixo foi inserido o e-mail que foi selecionado anteriormente pelo algoritmo de seleção randômica, que passa por uma etapa de pré-processamento para remover *emojis*, *links*, menções e palavras indesejadas, além de limpar espaços em excesso.

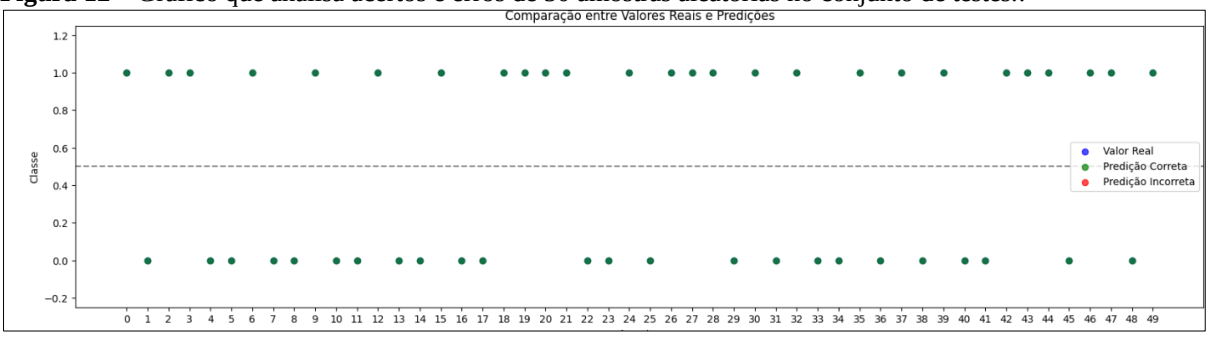
Figura 11 – Algortimo para testar os e-mails selecionados randômicamente.

```
Digite o e-mail a ser classificado: Hello Ashley, It was great to c

Digite o e-mail a ser classificado: Hello Ashley, It was great to catch up with you earlier. Lets discuss the next steps
O e-mail é: Não Spam
```

Fonte: Elaborado pelo autor (2024).

Foram selecionadas 50 amostras aleatórias do conjunto de teste para análise gráfica. De acordo com os resultados exibidos, o modelo de SVM acertou todas as predições. Os pontos verdes, que representam as predições corretas, estão presentes em todas as amostras, o que indica que o modelo conseguiu prever a classe correta para cada uma delas. Não há pontos vermelhos no gráfico, o que significa que não houve erros de predição nas amostras analisadas. Isso demonstra que o modelo teve um desempenho excelente nesse conjunto de amostras, sem falhas nas classificações.

Figura 12 – Gráfico que analisa acertos e erros de 50 amostras aleatórias no conjunto de testes..



Fonte: Elaborado pelo autor (2024).

Essa visualização permite observar de forma clara o desempenho do modelo em relação às classificações reais, destacando tanto os acertos quanto os erros nas 50 amostras aleatórias selecionadas e o resultado corrobora com o *Classification Report* apresentado anteriormente.

6 CONSIDERAÇÕES FINAIS

Neste trabalho, desenvolvemos e avaliamos um modelo preditivo utilizando *machine learning* mais precisamente máquina de vetores de suporte (SVM) juntamente com técnicas de processamento de linguagem natural (PLN) para a classificação de mensagens como *spam* ou não *spam*. Os resultados obtidos tanto no conjunto de treino quanto no conjunto de teste foram extremamente satisfatórios, com métricas de desempenho (precisão, revocação, f1-*score* e

acurácia) atingindo valores perfeitos de 100%. Isso demonstra que o modelo foi altamente eficaz em identificar corretamente as mensagens, sem apresentar falsos positivos ou falsos negativos. A análise visual das amostras aleatórias reforça ainda mais a confiabilidade do modelo, já que ele acertou todas as predições nas 50 amostras selecionadas do conjunto de teste, sem erros de classificação. A matriz de confusão, assim como o Classification Report, evidenciou a consistência do desempenho, confirmando que o modelo foi capaz de generalizar bem o aprendizado adquirido nos dados de treino. Não houve indícios de *overfitting*, uma vez que o modelo apresentou a mesma acurácia no conjunto de teste e no de treino, o que sugere que ele não apenas memorizou os dados, mas também generalizou eficazmente para novas amostras.

Embora o desempenho do modelo tenha sido excelente nos dados utilizados, trabalhos futuros podem explorar abordagens alternativas para garantir sua robustez em cenários mais amplos e variados. Uma sugestão seria testar o modelo com conjuntos de dados maiores ou mais desbalanceados, a fim de avaliar sua eficácia em situações mais próximas da realidade cotidiana. Outra linha de investigação interessante seria a implementação de técnicas de detecção de spam em tempo real, integrando o modelo a sistemas de filtragem de mensagens de serviços de e-mail. Além disso, seria valioso explorar outros algoritmos de classificação, como redes neurais e modelos baseados em *deep learning*, para comparar os resultados e verificar a possibilidade de alcançar uma precisão ainda maior ou uma solução mais eficiente do ponto de vista computacional. Adicionalmente, o modelo desenvolvido pode ser aplicado em diferentes contextos, como a detecção de fraudes financeiras, a identificação de notícias falsas ou qualquer cenário que envolva a classificação de texto. Isso abre um vasto campo de estudos e oportunidades para aprimoramento e novas implementações no futuro.

REFERÊNCIAS

ACEVEDO, P.; JIMÉNEZ-VALVERDE, A.; LOBO, J.M.; REAL, R. Delimitando o contexto geográfico na modelagem de distribuição de espécies. **Journal of Biogeography**, v. 39, p. 1383-1390, 2012. DOI: 10.1111/j.1365-2699.2012.02713.x. Disponível em: https://doi.org/10.1111/j.1365-2699.2012.02713.x. Acesso em: 5 nov. 2024.

ALDANIA, Ana; SOLEH, A.M.; NOTODIPUTRO, K.A. Um estudo comparativo de CatBoost e Double Random Forest para classificação multiclasse. **Journal RESTI (Sistem Rekayasa dan Informasi Teknologi)**, v. 7, n. 1, p. 129-137, fev. 2023. DOI: 10.29207/resti.v7i1.4766. Disponível em: https://doi.org/10.29207/resti.v7i1.4766. Acesso em: 5 nov. 2024.

BHARADWAJ, P.; PRAKASH, K.B.; KANAGACHIDAMBARESAN, G.R. **Pattern Recognition and Machine Learning.** In: Programming with TensorFlow. 2021. DOI: 10.1007/978-3-030-57077-4_11. Disponível em: https://doi.org/10.1007/978-3-030-57077-4_11 . Acesso em: 5 nov. 2024.

BRITO, Lucas C.; SUSTO, Gian Antonio; BRITO, Jorge N.; DUARTE, Marcus A.V. **An explainable artificial intelligence approach for unsupervised fault detection and diagnosis in rotating machinery**. Mechanical Systems and Signal Processing, 2021. DOI: 10.1016/j.ymssp.2021.108105. Disponível em: https://doi.org/10.1016/j.ymssp.2021.108105. Acesso em: 5 nov. 2024.

BRUSIQUESE, Romildo Garcia. **A influência de fatores organizacionais na qualidade de vida no trabalho em atividades com interação humano-sistema**. 2016. Disponível em: http://icts.unb.br/jspui/bitstream/10482/22210/1/2016_RomildoGarciaBrusiquese.pdf. Acesso em: 5 nov. 2024.

CHICCO, D.; JURMAN, G. As vantagens do coeficiente de correlação de Matthews (MCC) sobre a pontuação F1 e a precisão na avaliação da classificação binária. **BMC Genomics**, v. 21, n. 6, 2020. DOI: 10.1186/s12864-019-6413-7. Disponível em: https://doi.org/10.1186/s12864-019-6413-7. Acesso em: 5 nov. 2024.

DAS S., NAYAK T. Impact of Cyber Crime: Issues and Challenges. **International Journal of Engineering** Sciences & Emerging Technologies, October 2013; 6(2): 142-153, ISSN: 22316604. Disponível em: https://www.ijeset.com/media/0002/2N12-IJESET0602134A-v6-iss2-142-153.pdf. Acesso em: 5 nov. 2024.

DOGRA, Varun; VERMA, Sahil; KAVITA; CHATTERJEE, Pushpita; SHAFI, Jana; CHOI, Jaeyoung; IJAZ, Muhammad Fazal. **Um processo completo de sistema de classificação de texto usando modelos de PNL de última geração. Inteligência Computacional e Neurociência**, 2022, p. 1883698. DOI: 10.1155/2022/1883698. Disponível em: https://doi.org/10.1155/2022/1883698. Acesso em: 5 nov. 2024.

DUQUE-PEREIRA, I. da S.; MOURA, S. A. de. **Compreendendo a inteligência artificial generativa na perspectiva da língua**. SciELO Preprints, 2023. DOI: 10.1590/SciELOPreprints.7077. Disponível em: https://preprints.scielo.org/index.php/scielo/preprint/view/7077. Acesso em: 5 nov. 2024.

FABRE, Recimero Cesar. **Metodos avançados para controle de Spam**. 2005. 81f. Dissertação (mestrado profissional) - Universidade Estadual de Campinas, Instituto de Computação, Campinas, SP. Disponível em: https://hdl.handle.net/20.500.12733/1600282. Acesso em: 5 nov. 2024.

FIUZA BUENO, E.; FONSECA SANTOS, M. Inteligência artificial: desafios para regulação jurídica. **Revista Eletrônica Direito & TI**, [S. l.], v. 1, n. 18, p. 112–139, 2024. Disponível em: https://www.direitoeti.com.br/direitoeti/article/view/175. Acesso em: 5 nov. 2024.

FORMAN, G. Quantificação de contagens e custos via classificação. **Data Mining and Knowledge Discovery**, v. 17, p. 164-206, 2008. DOI: 10.1007/s10618-008-0097-y. Disponível em: https://doi.org/10.1007/s10618-008-0097-y. Acesso em: 5 nov. 2024.

GOMES, Hélio Márcio. **Segurança em correio eletrônico baseado em sistemas Microsoft**. 2012. Disponível em: https://repositorio.uniceub.br/jspui/bitstream/235/8140/1/51106157.pdf. Acesso em: 5 nov. 2024.

GUEDES R. de P., MOREIRA J., **Uma análise das tecnologias de detecção e mitigação na identificação de páginas de phishing**. 2023. Disponível em: http://200.131.116.17/index.php/enpe/article/view/339/260. Acesso em: 2 nov. 2024.

GUIMARÃES L. J. B. L. S**., Chatbot em contexto: Design de experiência do usuário aplicado à recuperação da informação no catálogo de teses e dissertações da CAPES**,

Universidade Federal de Minas Gerais., 2022. Disponível em: https://repositorio.ufmg.br/handle/1843/50840. Acesso em: 5 nov. 2024.

GUIMARÃES, Wesley Silvério et al. **SPAM-K: uma aplicação SDN para a definição de padrões visando a redução de falsos na detecção de SPAMs em serviços de correio eletrônico.** 2023. Disponivel em: https://repositorio.ufu.br/bitstream/123456789/39252/1/Spam-kUmaAplica%c3%a7%c3%a3o.pdf. Acesso em: 5 nov. 2024.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. **Overview of Supervised Learning**. In: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2. ed. Springer, 2009. DOI: 10.1007/978-0-387-84858-7_14. Disponível em: https://doi.org/10.1007/978-0-387-84858-7_14. Acesso em: 5 nov. 2024.

HENKE M., SANTOS C., NUNAN E., FEITOSA E., SANTOS E. dos, SOUTO E., **Aprendizagem de máquina para segurança em redes de computadores: métodos e aplicações.** 2011. Disponível em: https://www.researchgate.net/profile/Eulanda-Santos/publication/228447003_Aprendizagem_de_Maquina_para_Seguranca_em_Redes_de_Computadores_Metodos_e_Aplicacoes/links/0fcfd510a7af45b479000000/Aprendizagem-de-Maquina-para-Seguranca-em-Redes-de-Computadores-Metodos-e-Aplicacoes.pdf. Acesso em: 5 nov. 2024.

KUHN, M.; JOHNSON, K. **Medindo o desempenho em modelos de classificação. In: Modelagem preditiva aplicada**. Springer, 2013. DOI: 10.1007/978-1-4614-6849-3_11. Disponível em: https://doi.org/10.1007/978-1-4614-6849-3_11. Acesso em: 5 nov. 2024.

MARANHOLI H. N. G., SANTOS F. M. de M. Gamificação e jogos educacionais, compreender e planejar o ambiente urbano sustentável, através de jogos educacionais. **Geografia: Ambiente, Educação e Sociedades**. 2024; v. 2 n. 6. Recuperado de https://periodicos.unemat.br/index.php/geoambes/article/view/12640. Acesso em: 2 nov. 2024.

MELO, N. J. G. de; GUERRA, A. de L. e R.; SILVA, R. A. da. Tecnologias na educação e os desafios do uso da inteligência artificial: ética e perspectivas. **Revista Acadêmica da Lusofonia**, [S. l.], v. 1, n. 2, p. 1–14, 2024. Disponível em: https://revistaacademicadalusofonia.com/index.php/lusofonia/article/view/3. Acesso em: 5 nov. 2024.

LADEIRA A. P., **Processamento de linguagem natural: caracterização da produção científica dos pesquisadores brasileiros**, Universidade Federal de Minas Gerais., 2010. Disponível em: http://hdl.handle.net/1843/ECID-8B3Q6C. Acesso em: 5 nov. 2024.

MAIA, Andressa Moreira. **Aplicação da Lei Geral de Proteção de Dados (LGPD) em face da privacidade do consumidor no e-commerce**. 2021. Tese de Doutorado. Disponível em: https://repositorio.apps.uern.br/xmlui/bitstream/handle/123456789/325/TCC%20-%20Andressa%20Moreira%20Maia.pdf?sequence=1&isAllowed=y. Acesso em: 2 nov. 2024.

MORALES, Eduardo F.; ESCALANTE, Hugo Jair. **A brief introduction to supervised, unsupervised, and reinforcement learning**. In: Deep Learning Applications: From Theory

to Practice. Elsevier, 2023. DOI: 10.1016/B978-0-12-820125-1.00017-8. Disponível em: https://doi.org/10.1016/B978-0-12-820125-1.00017-8. Acesso em: 5 nov. 2024.

MURAKAMI B. G., **Performatividades hackers**. 2020. Dissertação (Mestrado em Poéticas Visuais) - Escola de Comunicações e Artes, University of São Paulo, São Paulo, 2020; doi:10.11606/D.27.2020.tde-03032021-164159. Disponível em: https://www.teses.usp.br/teses/disponiveis/27/27159/tde-03032021-164159/publico/BeatrizGarciaMurakami.pdf. Acesso em: 5 nov. 2024.

NABETO, Ana Maria**. A Transformação Digital no Sector da Saúde. Instituto Superior de Gestão, Departamento Mestrado em Estratégia de Investimento e Internacionalização**, 2020. Disponível em: https://comum.rcaap.pt/bitstream/10400.26/33074/1/Tese%20Mestrado%20Ana%20Nabeto%2030Junho%202020.pdf. Acesso em: 5 nov. 2024.

PATIL, R. **Email and SMS Spam Classification Dataset**. Kaggle, 2023. Disponível em: https://www.kaggle.com/datasets/devildyno/email-spam-or-not-classification. Acesso em: 4 nov. 2024.

PONTES F. J. da S., **Avaliação do desempenho de técnicas de aprendizado de máquina na detecção de malware em tráfego de Redes IoT**. 2024. 59 f. TCC (Graduação em Engenharia de Computação) – Campus de Sobral, Universidade Federal do Ceará, Sobral, 2024. Disponível em: http://repositorio.ufc.br/handle/riufc/77679. Acesso em: 5 nov. 2024.

POWERS, David M. W. **Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation**. 2020. arXiv:2010.16061v1 [cs.LG]. DOI: 10.48550/arXiv.2010.16061. Disponível em: https://doi.org/10.48550/arXiv.2010.16061. Acesso em: 5 nov. 2024.

QUEIROZ, Gabriel Noll; DISCONZI, Verônica Silva do Prado. O impacto da inteligência artificial no direito: questões éticas e legais. **Revista Ibero-Americana de Humanidades**, Ciências e Educação, [S. l.], v. 10, n. 4, p. 1388–1406, 2024. DOI: 10.51891/rease.v10i4.13550. Disponível em: https://periodicorease.pro.br/rease/article/view/13550. Acesso em: 5 nov. 2024.

ROCHA, W. F. da. **Revisão bibliográfica do uso de inteligência artificial na indústria nacional brasileira e regional do espírito santo**. Instituto Federal do Espírito Santo, 2023. Disponível em: https://repositorio.ifes.edu.br/bitstream/handle/123456789/3734/TCC_Revis%C3%A3o_Bibliogr%C3%A1fica_Uso_IA_Ind%C3%BAstria.pdf?sequence=1&isAllowed=y. Acesso em: 5 nov. 2024.

ROSSETO, Lourdes Maria Porto. **A Ciência Política e as Ciências Sociais Computacionais: uma revisão sistemática**. Universidade Federal de São Carlos (UFSCar). 2024. Disponível em: https://repositorio.ufscar.br/bitstream/handle/ufscar/20709/Monografia%20-%20ROSSETO%2c%20Lourdes.pdf?sequence=1&isAllowed=y. Acesso em: 5 nov. 2024.

ROVER, Vinicius**. Implementação de uma interface gráfica para uso de algoritmos de aprendizado de máquina**. Universidade Federal de Santa Catarina, Campus Araranguá,

2024. Disponível em: https://repositorio.ufsc.br/bitstream/handle/123456789/255864/TCC.pdf?sequence=1&isAllowed=y. Acesso em: 5 nov. 2024.

SEN, P.C.; HAJRA, M.; GHOSH, M. **Supervised Classification Algorithms in Machine Learning: A Survey and Review**. In: Emerging Technology in Modelling and Graphics. 2020. DOI: 10.1007/978-981-13-7403-6_11. Disponível em: https://doi.org/10.1007/978-981-13-7403-6_11. Acesso em: 5 nov. 2024.

SILVA, C. A.; ROCHA, G. S. F.; SILVA, A. L. da. **Além do hype: investigando o impacto real da inteligência artificial na fidelidade do cliente na era digital**. FATEC SEBRAE e FATEC Ipiranga, 2024. Disponível em: https://revista.fatecsebrae.edu.br/index.php/em-debate/article/view/265/295. Acesso em: 5 nov. 2024.

SIPONEN M., STUCKE C., **Effective Anti-Spam Strategies in Companies: An International Study,** Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS'06), Kauai, HI, USA, 2006; pp. 127c-127c, doi: 10.1109/HICSS.2006.140. 2006.

SOARES, Marta. O poder da inteligência artificial no mundo empresarial. **The Trends Hub**, Porto, n. 4, 2024. DOI: 10.34630/tth.vi4.5663. Disponível em: https://parc.ipp.pt/index.php/trendshub/article/view/5663. Acesso em: 5 nov. 2024.

SOARES F. J. L. dos R. **O estado da arte da aplicação da Inteligência Artificial (IA) e Aprendizagem Automática (AA) ao cálculo de estrutura**. 2024. Disponível em: https://repositorio-aberto.up.pt/bitstream/10216/162129/2/693183.pdf. Acesso em: 3 nov. 2024

SOORI, Mohsen; AREZOO, Behrooz; DASTRES, Roza. Artificial intelligence, machine learning and deep learning in advanced robotics, a review. **Cognitive Robotics**, 2023. DOI: 10.1016/j.cogr.2023.04.001. Disponível em: https://doi.org/10.1016/j.cogr.2023.04.001. Acesso em: 5 nov. 2024.

VIJAYAKUMAR B., THOMAS C., **The ethics of envisioning spam free email inboxes**, AI Ethics, 2024. https://doi.org/10.1007/s43681-024-00526-2. Acesso em: 5 nov. 2024.
WERNER W., WERNER I., Gestão do conhecimento: ferramentas tecnológicas e portais do conhecimento para empresas desenvolvedoras de tecnologias de médio e pequeno portes, **Revista Terra & Cultura: Cadernos de Ensino e Pesquisa**. 2020;20(38):183–241. Disponível em: http://publicacoes.unifil.br/index.php/Revistateste/article/view/1337/1227. Acesso em: 3 nov. 2024.

ZOU, L.; XIA, L.; DING, Z.; SONG, J.; LIU, W.; YIN, D. **Reinforcement Learning to Optimize Long-term User Engagement in Recommender Systems.** 2019. DOI: 10.1145/3292500.3330668. Disponível em: https://doi.org/10.1145/3292500.3330668. Acesso em: 5 nov. 2024.