UNIVERSIDADE FEDERAL DE SANTA CATARINA

DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA

CURSO DE SISTEMAS DE INFORMAÇÃO

Migueh: uma ferramenta para detecção de social bots no X/Twitter

Indiara Camillo Menegat

Florianópolis - SC

2024

UNIVERSIDADE FEDERAL DE SANTA CATARINA

DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA

CURSO DE SISTEMAS DE INFORMAÇÃO

Migueh: uma ferramenta para detecção de social bots no X/Twitter

Indiara Camillo Menegat

Trabalho de conclusão de curso apresentado como parte dos requisitos para obtenção do grau de Bacharel em Sistemas de Informação.

Orientador: José Eduardo de Lucca

Florianópolis - SC

2024

Menegat, Indiara
Migueh: uma ferramenta para detecção de social bots no X/Twitter / Indiara Menegat ; orientador, José De Lucca, 2024.
51 p.

Trabalho de Conclusão de Curso (graduação) - Universidade Federal de Santa Catarina, Centro Tecnológico, Graduação em Sistemas de Informação, Florianópolis, 2024.

Inclui referências.

1. Sistemas de Informação. 2. Bots. 3. Fake News. 4. Redes Sociais. 5. Estatística. I. De Lucca, José. II. Universidade Federal de Santa Catarina. Graduação em Sistemas de Informação. III. Título.

Dedico este trabalho a Rosane e Anévio, meus pais, a quem sempre carregarei em mim como uma parte fundamental do meu ser.

RESUMO

Este trabalho desenvolveu uma ferramenta para a detecção de bots no X/Twitter, utilizando análise comportamental e metadados para impedir a propagação de informações falsas. A ferramenta aplica regras heurísticas baseadas em características identificadas em estudos anteriores, como proporção de seguidos para seguidores, uso de retweets e menções, e atividade temporal das contas. Os resultados indicam a necessidade de ajustar os pesos das regras e incorporar mais dados para aumentar a precisão. Futuras melhorias incluem a integração de regras adicionais e a utilização de aprendizado de máquina para lidar com o volume de dados. A ferramenta, desenvolvida de forma desacoplada, pode ser adaptada para outras redes sociais, contribuindo para a redução de desinformação na internet.
Palavras-chave: Fake News. Bots. Redes Sociais.

ABSTRACT

This study developed a tool for detecting bots on X/Twitter, leveraging behavioral analysis and metadata to prevent the spread of false information. The tool applies heuristic rules based on features identified in previous studies, such as the ratio of followers to following, the use of retweets and mentions, and the temporal activity of accounts. The results highlight the need to adjust rule weights and incorporate additional data to improve accuracy. Future enhancements include integrating more rules and employing machine learning to handle large data volumes. The tool, developed in a decoupled manner, can be adapted for other social networks, contributing to reducing misinformation on the internet.

Keywords: Fake News. Bots. Social Networks.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE SIGLAS E ABREVIATURAS

API - Application Programming Interface (Interface de Programação de Aplicações)
AUC - Area Under the Curve (Área Sob a Curva)
AWS - Amazon Web Services (Serviços Web da Amazon)
BFF - Backend for Frontend (Backend para Frontend)
CSV - Comma-Separated Values (Valores Separados por Vírgulas)
FPR - False Positive Rate (Taxa de Falsos Positivos)
JSON - JavaScript Object Notation (Notação de Objetos JavaScript)
MVP - Minimum Viable Product (Produto Mínimo Viável)
REST - Representational State Transfer (Transferência de Estado Representacional)
ROC - Receiver Operating Characteristic (Curva Característica de Operação do Receptor)
TPR - True Positive Rate (Taxa de Verdadeiros Positivos)

SUMÁRIO

1. INTRODUÇÃO

Desinformação e disseminação de informações falsas não são estranhas à humanidade, é possível traçar sua atuação e impactos desde o antigo Império Romano, sendo utilizada como uma arma contra inimigos e também como uma forma de convencimento do próprio povo (POSETTI; MATTHEWS, 2018), o que não surpreende que as novas formas de comunicação que foram introduzidas no século XXI por causa da Internet, como as redes sociais, fossem potencializadoras dessa prática.

Segundo o Instituto Brasileiro de Geografia e Estatística (IBGE), em 2018, 74,7% dos brasileiros utilizavam a internet frequentemente, sendo 95,7% desses com a finalidade de enviar ou receber mensagens de texto, voz ou imagens por aplicativos diferentes de e-mail (IBGE, 2018). Dentro desses aplicativos de trocas de mensagens, ocorre a circulação livre de informações, muitas vezes pouco rastreável e com uma capacidade disseminadora que vai além dos discursos em praça pública realizados em Roma. Hoje, eventos ocorridos no Brasil e no mundo, como o PizzaGate durante as eleições americanas de 2016 (WU; MORSTATTER; CARLEY; LIU, 2019), eleições brasileiras de 2018 (NOBRE; ALMEIDA; FERREIRA, 2019), o julgamento do ex-presidente Luiz Inácio “Lula” da Silva (RECUERO; GRUZD, 2019), mostram-se como grandes exemplos de como a disseminação de informações falsas nas redes sociais pode afetar a consciência e o julgamento de diversos grupos específicos de usuários.

Com o início da pandemia de COVID-19 no fim de 2019 surgiu uma necessidade imediata de informar a população mundial sobre a gravidade da situação. Devido à falta de conhecimento até o momento sobre como atuava o vírus, era necessário repassar ao mundo as medidas básicas de prevenção que poderiam evitar o contágio, mas essa falta gerou uma brecha muito ampla para o surgimento de informações falsas.

Hoje, a própria Organização Mundial de Saúde (OMS) caracteriza que o mundo vive duas epidemias: uma de um vírus mortal e outra informacional. Tedros Adhanom Ghebreyesus, Diretor Geral da OMS, afirmou ainda em fevereiro de 2020 que “Não estamos apenas lutando contra uma epidemia; estamos lutando contra uma infodemia” (ZAROCOSTAS, 2020), e desde lá, a quantidade de informações tendenciosas sobre o vírus aumentou consideravelmente. A capacidade da

transmissibilidade de informação através das redes sociais é tão grande quanto a capacidade do próprio vírus infectar pessoas, assim, existe a necessidade constante de ser mais rápido que a onda de desinformação, para garantir que toda a população seja informada da maneira correta (ZAROCOSTAS, 2020).

Paralelamente, o Twitter, objeto de estudo deste trabalho, que agora é denominado X, enfrentou desafios legais no Brasil relacionados à disseminação de desinformação e discursos de ódio. Em agosto de 2024, o Supremo Tribunal Federal, por meio do ministro Alexandre de Moraes, determinou o bloqueio da plataforma no país após o descumprimento de ordens judiciais que exigiam a remoção de conteúdos considerados ilícitos. Essa medida foi uma resposta à instrumentalização da rede para a divulgação em massa de discursos de ódio e informações falsas, especialmente no contexto das eleições municipais [34].

Dadas as circunstâncias atuais, considera-se extremamente necessário que se voltem esforços para o estudo da detecção de notícias e informações falsas na busca de construir metodologias e ferramentas que possam reduzir os danos colaterais dessa prática. Na luz dos acontecimentos citados, muitos estudos surgiram nos últimos anos, apontando propostas como mineração de dados, aprendizagem de máquina, detecção de bots, análise de sentimentos e demais alternativas como uma forma de complementar a já difícil tarefa que se propõe, ressaltando a importância de políticas claras e colaborativas para o combate à desinformação e a promoção de um ambiente digital saudável.

Este trabalho, então, propõe-se a estudar estratégias para identificação de informações falsas em redes sociais, com o objetivo final de construir uma ferramenta que possa ser disponibilizada aos usuários destas redes, catalogando ativamente contas maliciosas automatizadas, conhecidas como bots, e informações falsas sobre que, futuramente, possa se tornar um projeto aberto alimentado por diversos colaboradores e que seja capaz de expandir sua análise para outros assuntos que são frequentemente alvos dessa estratégia.

1.1. OBJETIVOS

1.1.1. Objetivo geral

Desenvolver a ferramenta Migueh para detectar bots no X/Twitter, prevenindo a disseminação de informações falsas por meio de análise de comportamento e metadados.

1.1.2. Objetivos específicos

- Realizar o levantamento e a preparação dos dados necessários para o desenvolvimento da ferramenta, o que inclui o desenvolvimento de scripts para coleta de dados do Twitter utilizando APIs, a limpeza, normalização e enriquecimento dos dados coletados, e a criação de uma base de dados balanceada com exemplos de bots e humanos.
- Definir regras de detecção para identificar métricas relevantes que distinguem comportamentos de bots e humanos.
- Implementar a ferramenta, desenvolvendo a interface de usuário para exibir os resultados das classificações, integrando os algoritmos de detecção com a base de dados e a API do Twitter, e implementando funções para visualização de dados e relatórios.
- Documentar o desenvolvimento e as decisões técnicas do projeto, além de preparar a ferramenta para futuras expansões, considerando possíveis novas métricas e integrações com APIs.
- Construir uma ferramenta livre e de código aberto que possa continuar a ser ampliada por uma comunidade mesmo após o fim do desenvolvimento deste trabalho.

2. REFERENCIAL TEÓRICO

2.1. INFORMAÇÕES FALSAS EM REDES SOCIAIS

Antes de buscar qualquer abordagem tecnológica para detecção de informações falsas, é importante ter uma definição do que seriam informações falsas em uma rede social, para assim, entender melhor o problema e quais as melhores formas de abordá-lo.

Para Wu, Morstatter, Carley e Liu (2019), por definição, informação falsa é algo definitivamente falso ou definido com imprecisão, que deliberadamente é criado e, intencionalmente ou não, é propagado. Para estes autores, o termo *informações falsas* serve como um termo guarda-chuva para demais temas menores, dentre eles rumores e *fake news*, além de separar o que seria uma informação falsa intencionalmente propagada de uma não intencionalmente propagada.

Para Kumar e Shah (2018), a categorização de informação falsa baseia-se na intenção e no conhecimento. Pela intenção, uma informação falsa pode ser categorizada como uma “má informação”, que foi criada sem a intenção de

confundir, enquanto desinformação, é um pedaço de informação falsa criada para confundir e manipular o interlocutor. Pelo conhecimento, a informação falsa pode ser baseada em opinião, o que representa a opinião individual do usuário que cria, intencionalmente ou não, uma informação baseada no que mesmo acha sobre o assunto, e também a informação falsa pode ser categorizada baseada em fatos, que envolve informações que contradizem, fabricam ou combinam uma informação verdadeira, fazendo com que o usuário tenha dificuldade em discernir a informação verdadeira da falsa, dentro dessa categoria entram as fake news, os rumores e as farsas.

Dentro do escopo deste trabalho, serão usados como base para a proposta final conceitos de duas áreas abaixo do guarda-chuva informações falsas: notícias falsas e rumores.

2.1.1. Notícias falsas

Notícia falsa, popularmente conhecida pelo termo em inglês *fake news,* é um termo recente dentro do meio jornalístico, utilizado para categorizar textos contendo informações falsas escritas em forma de notícia com o intuito de desvirtuar e confundir o leitor.

O principal objetivo da utilização de notícias falsas é a obtenção de vantagens, sejam comerciais ou políticas, influenciar pessoas e também como uma ferramenta da disseminação de ódio (QUEIROZ; FRANCÊS; COSTA; ANDRADE; HARB, 2019). A terminologia, que é antiga, passou a ser frequentemente utilizada pela imprensa durante as eleições presidenciais dos EUA em 2016.

Dentro do contexto de redes sociais, uma notícia falsa atinge novos patamares ao se moldar à forma que as publicações funcionam dentro da rede, nisso também inclui a forma que são compartilhadas. Notícias falsas são constantemente disseminadas por contas maliciosas, em grandes cargas, com o objetivo de gerar amplo alcance dessas informações. Esse amplo alcance busca gerar uma falsa percepção de consenso, construindo ambientes conhecidos como *câmaras de eco*, onde, pela repetição constante da informação e ao impedir que posições contrárias adentrem o espaço, os indivíduos ali presentes passam a acreditar que aquilo é verídico (RECUERO; GRUZD, 2019).

Não somente a estrutura textual e de conteúdo devem ser consideradas para compreensão das fake news, a efetividade da tática também se baseia em princípios

psicológicos e sociais, ao explorar vulnerabilidades dos usuários, como a tendência do indivíduo de acreditar e aproximar-se de ideias que reforçam suas próprias crenças pessoais, sejam elas verídicas ou não, e a necessidade de aceitação e afirmação social perante à um grupo, o que faz com que o indivíduo siga as normas comunicativas deste, por mais que o que elas compartilhem seja falso (SHU; ZHOU; WANG; ZAFARANI; LIU, 2019).

2.1.2. Rumores

Pelo dicionário Oxford, a definição de rumor é "*uma história ou relato que circula, no presente, com verdades incertas e duvidosas.*"[1].

Zubiaga, Aker, Bontcheva, Liakata e Procter (2018) definem rumor como "um item de informação circulante cujo o status de veracidade ainda não foi verificado no momento da publicação", isso pois, não necessariamente, rumores referem-se à informações não verdadeiras, mas sim a informações que ainda precisam ser verificadas que podem, inclusive, se tornarem verdadeiras. Um rumor é definido então como uma peça de informação não verificada caso não existam evidências ou fontes oficiais que garantam sua confirmação.

Rumores podem surgir a partir de notícias da atualidade, o que faz com que seu tempo de vida seja breve mas que também exista pouco material para sua detecção, ou podem ser rumores de longa data, discutidos há muito tempo e sem uma origem definida, que por algum motivo não tiveram sua veracidade definida ou que suas comprovações de que são falsos não convenceram a totalidade dos que acreditam nele (ZUBIAGA; AKER; BONTCHEVA; LIAKATA; PROCTER, 2018).

2.2. DETECÇÃO DE INFORMAÇÕES FALSAS

2.2.1. Características gerais

Existem características chaves que são comuns em relação aos tipos de informações falsas em redes sociais e são usadas como base para construção de métodos de detecção.

Publicações e notícias contendo informações falsas tendem a ter um tipo específico de escrita, sendo mais simples e direta para fácil compreensão dos leitores, com a utilização de poucos termos mais complexos para trazer

[1] https://www.lexico.com/definition/Rumour

confiabilidade ao conteúdo (SAKURAI, 2019). Também carregam títulos sensacionalistas que atraem os olhares e *click baits*[2].

Além da forma de escrita, também se nota, principalmente em notícias falsas, que não existe uma imparcialidade da parte dos autores. As publicações carregam, misturadas ao conteúdo, a opinião tendenciosa do autor, que usa isso para tentar atingir grupos de usuários que já são propensos a acreditarem por terem visões parecidas, criando assim bolhas chamadas *câmaras de eco* (RECUERO; GRUZD, 2019).

Outra característica é a forma como ocorre a propagação da informação, sendo essa feita principalmente através de grandes redes de contas maliciosas, muitas vezes automatizadas. Shu, Silva, Wang, Tang e Liu (2017) dividem essas contas maliciosas em três categorias: *bots*, *trolls* e *cyborgs*. *Bots* são contas de redes sociais automatizadas para realizar certas tarefas; nesse contexto, *bots* se tornam entidades maliciosas desenvolvidas especificamente com o propósito de manipular e espalhar informações falsas. *Trolls* são usuários humanos que buscam provocar e perturbar comunidades online para gerar respostas emocionais ao repetidamente espalhar o mesmo assunto através das redes. E finalmente, *cyborgs* referem-se a contas que misturam atividade automatizada com atividade humana, pois são criadas por um humano mas realizam atividades automatizadas para amplificar as visões do usuário que as controla.

Contas maliciosas também tendem a passar por longos períodos de hibernação e então apresentar atividade intensa durante os períodos em que a informação falsa está sendo propagada (WU; MORSTATTER; CARLEY; LIU, 2019). Para Kumar e Shah (2018), o papel desses tipos de contas propagadoras sugere que repetição e perseverança são importantes peças da disseminação de informações falsas, já que assim cria-se uma ilusão de consenso sobre o assunto.

2.2.2. Métodos

Múltiplas propostas de métodos e algoritmos de detecção automática de informações falsas surgiram nos últimos anos, sendo que estas seguem principalmente estratégias no campo de mineração de dados e processamento de linguagem natural. Na maioria das vezes, os métodos para detecção de informações falsas são focados especialmente no tipo de informação que desejam detectar,

[2] Do inglês “isca de cliques", o clickbait é uma estratégia de divulgação online que usa títulos sensacionalistas para chamar a atenção de usuários e gerar mais acessos ao conteúdo.

talvez pela dificuldade de criar um modelo geral, que sirva para todos os casos. Múltiplos autores categorizam esses métodos, mas pouco consenso existe entre essas categorias na literatura, mesmo que muitas tenham definições semelhantes.

Kumar e Shah (2018) separam os algoritmos e técnicas já existentes em três categorias: *feature-based*, *graph-based* e *propagation-modeling based*.

- **Feature-based:** É a principal e única abordagem da maioria dos algoritmos e refere-se às características únicas da informação. A característica central considerada é o próprio texto, este que pode ser categorizado a partir de uma perspectiva estilométrica[3], psicolinguística[4] ou pela complexidade da escrita.

  Demais características, como usuário, rede, tempo e metadados são utilizadas como um apoio aos métodos de análise textual.

  Esses algoritmos buscam definir características que, sozinhas ou em conjunto, possam distinguir entre verdade e mentira de forma eficiente. Alguns desses também utilizam características dos emissores para detectar tipos de contas e atividades maliciosas, que, como já dito, tem um papel importante na manutenção da informação falsa.

- **Graph-based:** Aborda grandes blocos e subgrafos de usuários e informações numa grande rede.

  Baseia-se na percepção de que sub-redes compostas por grafos densos (com mais usuários, publicações e horários semelhantes) são produzidas por ações coordenadas e estas estão associadas a ações fraudulentas, ou seja, focam-se em identificar grandes grupos de usuários com comportamentos repetitivos e pouco orgânicos para determinar a veracidade da informação. A semelhança entre os textos e os horários das publicações são importantes características para montar os grafos de usuários.

  São normalmente usados para identificar opiniões falsas, como reviews enganosas em sites de e-commerce.

---

[3] Estudo do estilo linguístico de uma linguagem escrita. Disciplina linguística que avalia o estilo de um autor por meio da aplicação da análise estatística sobre corpo de texto de sua obra.

[4] Estudo da correlação entre fatores linguísticos e aspectos psicológicos. Foca principalmente em mecanismos pelos quais a linguagem é processada e apresentada dentro da mente e do cérebro, considerando fatores psicológicos e neurobiológicos que permitem que humanos utilizem a linguagem.

- **Propagation-modeling based:** Refere-se a técnicas e algoritmos que tentam construir modelos de propagação de informações verdadeiras em redes para então identificar as falsas. O intuito dessa abordagem vem da concepção de que, já que a maioria das informações divulgadas são verdadeiras, seus métodos de propagação devem ser similares, então, ao simular os caminhos pelos quais as informações verdadeiras circulam, estes algoritmos conseguem identificar as falsas através de anomalias encontradas nos seus modelos de propagação.

  São frequentemente utilizados para analisar a veracidade de informações em redes sociais, e são importantes para essa tarefa específica, pois muitos estudos já demonstraram que a manutenção da prática de disseminação de informações falsas nas mídias sociais somente é possível devido a alta taxa de propagação que as publicações têm.

Kumar e Shah (2018) apontam que, naturalmente, a efetividade de todos os algoritmos destas técnicas depende da tarefa e dos dados utilizados, mas, muitos atingem altos níveis de desempenho, mostrando efetividade em grandes *datasets* de *reviews* falsas, notícias falsas, rumores e farsas.

Já Wu, Morstatter, Carley e Liu (2019) encaixaram as abordagens em outras três categorias, similares às de Kumar e Shah (2018), mas mais pontuais e com nomenclaturas diferentes. As três categorias de métodos de detecção para os autores são: *content-based, context-based e propagation-based.*

- **Content-based**: Trata da detecção de informações falsas atentando diretamente ao conteúdo da informação, sendo este textos, imagens e vídeos.

  Algumas destas técnicas buscam coletar posts que contém pedaços já conhecidos de informações falsas para ajudar na classificação, passando os textos coletados para diversos classificadores de texto para identificar padrões. Nestes casos, os algoritmos podem encontrar dificuldades em identificar a veracidade da publicação contendo a informação falsa caso ela tenha sido reescrita intencionalmente.

  Essa proposta é utilizada baseando-se na concepção de que as informações falsas perpetuadas na rede consistem de múltiplas palavras chaves, então, uma única publicação com termos suspeitos o suficiente pode ser classificada como falsa.

Em adição ao conteúdo, consideram-se também *clusters* de mensagens para identificação, que levam em conta também o período temporal das publicações e os autores.

São métodos que ajudam na detecção da informação após estágios avançados da propagação e principalmente quando trata-se de informações falsas que são bastante populares, que geraram múltiplas publicações, logo, existe uma quantidade maior de material para ser analisado e classificado.

- **Context-based**: Trata da detecção através das informações contextuais da rede, não referentes diretamente ao conteúdo da publicação, mas o que a cerca, como dados de localização e tempo. São métodos utilizados em apoio a outros métodos de classificação para facilitar a detecção, principalmente em casos onde ocorrem “rajadas” de publicações, ou seja, múltiplas publicações parecidas em um mesmo espaço de tempo.

Essa abordagem baseia-se na suposição de que informações falsas são intencionalmente divulgadas por grupos específicos de usuários e com diferentes padrões de publicação do que seriam informações verídicas.

- **Propagation-based**: Trata da detecção de informações falsas através dos padrões de circulação dela entre os usuários. Foca principalmente nos usuários que postam e repassam as informações para tentar predizer a influência da mensagem.

A motivação para desenvolver estratégias de detecção através da propagação vem da ideia de que o caminho que a informação percorre dentro da rede social revela características pessoais e comunitárias. Assim como nas abordagens baseadas em contexto, as abordagens baseadas em propagação também utilizam a premissa de grandes rajadas de informação sendo disparadas, neste caso, por indivíduos específicos e comunidades de usuários e como essas rajadas são repassadas de um ponto a outro na rede.

Atualmente, já existem propostas de frameworks, como o *TraceMiner*, que classifica o caminho da informação baseada na rede dos usuários que a compartilham (WU; LIU, 2018). Testes iniciais indicam que esse método em específico atingiu graus de acurácia maiores do que abordagens baseadas em conteúdo.

2.3. *BOTS* EM REDES SOCIAIS

Em geral, um *bot* no contexto de redes sociais é uma conta automatizada por algum serviço computacional para publicar conteúdos automaticamente e interagir com outras contas, sejam elas outros *bots* ou não. O baixo custo para criar contas em redes sociais estimula o surgimento de contas maliciosas, que são criadas especificamente com o propósito de manipular e disseminar informações falsas nas redes sociais, e geralmente, a maioria dessas contas maliciosas são automatizadas. Um exemplo disso foram as eleições estadunidenses de 2016, onde foram reportados que mais de 19 milhões de contas automatizadas, no Twitter, realizaram publicações em apoio a Donald Trump ou Hillary Clinton na semana prévia às eleições e que tiveram impacto razoável na resolução do pleito (SHU; SILVA; WANG; TANG; LIU, 2017).

Para Abokhodair, Yoo e McDonald (2015) *bots* podem ser considerados como atores sociais automatizados, que são desenvolvidos para agir de forma similar a um humano dentro daquele espaço social virtual. Mesmo que por definição um bot não seja malicioso, ele pode ser utilizado para causar perturbação no ambiente social, se tornando um grande perigo para a rede e os usuários, principalmente quando são responsáveis por gerar uma intensa propaganda ideológica que acaba criando noções de falso consenso.

Bots muitas vezes tem características específicas que podem ser identificadas até por usuários comuns. Num estudo realizado sobre bots dentro do Twitter, durante as eleições presidenciais do Brasil em 2018, Nobre, Almeida e Ferreira (2019) analisaram mais de 100 mil contas, onde identificaram pouco mais de 2 mil bots através de uma ferramenta de detecção e então analisaram 18 métricas destes usuários automatizados para comparação com usuários comuns. Dentro das métricas utilizadas, identificaram que bots tendem a possuir contas mais novas, retweetar menos usuários distintos, mencionar menos outros usuários, possuir menos seguidores, publicar menos tweets, ter proporções parecidas de retweets aos demais bots, etc. Além disso, hoje já existem ferramentas públicas que classificam bots a partir de suas métricas específicas, das quais serão apresentadas algumas destas na próxima seção.

Para atingir o efeito desejado de propagação de informações falsas, um bot sozinho não é o suficiente, então, a partir de múltiplas contas automatizadas formam-se as botnets que trabalham em conjunto nessa tarefa. Abokhodair, Yoo e

McDonald (2015) também estudaram profundamente a formação de uma botnet no Twitter focada na Guerra Civil da Síria e conseguiram extrair algumas características delas, que acreditam poder ser utilizadas como referência de estudo para as demais botnets existentes. Por exemplo, a rede analisada é composta por 130 *bots* e três tipos de atores, segundo os autores: *generator bots* (37,6% da rede), *core bots* (50,7% da rede) e *peripheral bots* (11,5% da rede) . Generator *bots* são responsáveis pela parte da criação dos *tweets* na botnet e *retweetam* poucos *tweets* de outras contas, enquanto core *bots* *tweetam* menos e *retweetam* mais, principalmente os *tweets* gerados pelos generator *bots*. Peripheral *bots* são contas entre generator *bots* e core *bots*, que *tweetam* e *retweetam* em menores e similares proporções, podem ser usuários comuns que estão engajando com a botnet ou até *bots* mais sofisticados e mais capazes de imitar o comportamento comum de um usuário da rede.

3. TRABALHOS RELACIONADOS

Esta seção irá apresentar algumas ferramentas já existentes utilizadas para detecção de *bots*, além de descrever um pouco das métricas que são utilizadas para categorizar os usuários automatizados. Todas as ferramentas abaixo foram construídas se baseando no Twitter e na forma que a rede social monta suas redes de usuários. Existem outros métodos além dos mencionados, porém, foram considerados somente os que foram disponibilizados ao público através do acesso por APIs e plataformas próprias. Dentre todos os citados, nenhum assume que é capaz de categorizar uma conta como um *bot* com total acurácia, todas demonstram pontuações de análise entre **mais provável** e **menos provável** de ser uma conta automatizada.

- *BotSentinel*[5] é uma ferramenta gratuita desenvolvida para classificar e rastrear contas automatizadas. Seu classificador é baseado em algoritmos de aprendizagem de máquina e inteligência artificial, porém, os criadores, até o presente momento, não revelaram quais características específicas são consideradas pelos algoritmos. No geral, as contas são classificadas numa pontuação de 0% a 100%, sendo que, quanto mais alta a pontuação, mais provável que a conta seja maliciosa. A análise é feita em cima das publicações das contas.

[5] https://botsentinel.com/

Atualmente, o acesso ao *BotSentinel* só pode ser realizado através da plataforma da ferramenta, pois, segundo os desenvolvedores, a API e sua documentação ainda estão em processo de construção. Dentre as ferramentas citadas, é a única a ser disponibilizada através de extensões de navegador para ser utilizada.

Apesar de não ter código aberto, o *BotSentinel* mantém pública sua base de dados sobre quais contas já foram analisadas por usuários. Isso auxilia no monitoramento destas contas, onde pode-se identificar a atividade da conta e quando a mesma é suspensa ou removida da plataforma.

- *Botometer*, conhecido inicialmente como *BotOrNot* (DAVIS; VAROL; FERRARA; FLAMMINI; MENCZER, 2016), é uma ferramenta de detecção de *bots* que surgiu em 2016 e já conta com quatro versões, sendo a última lançada em setembro de 2020. Foi desenvolvida pelo Observatório das Mídias Sociais, da Universidade da Indiana e conta com uma API que permite acesso ao seu sistema de classificação. O sistema de classificação utiliza um algoritmo de aprendizagem de máquina e se baseia em mais de mil características que são classificadas em 6 classes:
  - **Rede**: captura as dimensões da propagação de informação daquele usuário, como *retweets*, menções, *hashtags*, etc.
  - **Usuário**: dados sobre o usuário como a língua, sua localização e data e hora de criação da conta.
  - **Amigos**: relacionado aos contatos próximos dos usuários analisados, dos quais observa-se número de seguidores, publicações, etc.
  - **Temporal**: trata dos períodos temporais onde acontecem as postagens e o consumo de informação por parte do usuário, tanto quanto a relação de quantas publicações por hora, etc.
  - **Conteúdo**: refere-se aos dados linguísticos das publicações do usuário, que passam por um processamento de linguagem natural por dentro do algoritmo.
  - **Sentimento**: considera a análise de sentimento sobre as publicações do usuário, incluindo principalmente a relação excitação-dominância-valência.

Após passar pelo processamento do algoritmo, cada uma das 6 categorias recebe seus pesos e valores e então, retorna um relatório com a pontuação do usuário analisado em cada uma delas, além de uma pontuação geral de 0 à 5, sendo 0 a menor chance de ser um *bot* e 5 a maior chance de ser um *bot*. O *Botometer* ainda permite que se analise uma parte dos seguidores e dos amigos daquele usuário, porém, está funcionalidade só está disponível pelo portal da ferramenta e não pela API.

- *PEGABOT*[6] é uma ferramenta de detecção de *bots* brasileira, desenvolvida em 2018 pelo Instituto de Tecnologia e Sociedade do Rio de Janeiro e do Instituto Tecnologia & Equidade. A ferramenta não tem uma API disponível, mas pode ser utilizada através de um portal público. Seu algoritmo de detecção aborda 3 critérios principais:
  - **Perfil de Usuário**: gera variáveis de análise em cima do nome do perfil do usuário, quantos caracteres ele possui, a quantidade de perfis que o usuário segue e que o seguem, o texto da descrição do perfil, número de publicações e curtidas.
  - **Rede**: gera variáveis de análise em cima de uma amostra da linha do tempo do usuário, identificando hashtags e menções. O objetivo é entender características da distribuição de informação na conta analisada.
  - **Análise de sentimentos**: analisa as últimas 100 publicações do usuário para entender os níveis de neutralidade daquela conta. A pontuação dos sentimentos vai de -5 para mais negativo e 5 para mais positivo. Partem do princípio que quanto mais forte é o sentimento geral das publicações (seja positivo ou negativo) mais chances existe da conta ser automatizada.

    Ao final da análise do classificador, a ferramenta exibe uma pontuação de 0% a 100%, onde 0% indica que a conta tem baixo índice de comportamento similar à um *bot*, enquanto 100% indica que a conta tem alto índice de comportamento similar à um *bot*.

[6] https://pegabot.com.br/

4. DESENVOLVIMENTO

O desenvolvimento da ferramenta foi quebrado em várias etapas: definição da ferramenta, coleta e preparação de dados, desenvolvimento das regras de detecção, categorização de contas, desenvolvimento da interface do usuário, testes e validação. Cada uma dessas etapas será detalhada ao longo da seção.

4.1. DEFINIÇÃO DAS REGRAS DE DETECÇÃO

Antes de iniciar o desenvolvimento técnico da ferramenta, a primeira etapa foi definir uma lista de regras de detecção que guiaram as próximas fases, como a análise exploratória dos dados e o desenvolvimento das funções heurísticas de classificação de contas.

Com base em estudos como os de Ferrara et al. (2017), Satardakar e Chaudhari (2020), e Zheng et al. (2015), que identificaram características de usuários com alta probabilidade de serem contas automatizadas, foram definidas nove características importantes para categorizar uma conta entre humano e bot. Essas características serão tratadas como regras, cada uma norteando o desenvolvimento de uma função heurística de classificação, seguida de uma hipótese que será validada ao fim do trabalho. Abaixo, o detalhamento das regras e suas hipóteses:

- **Proporção de seguidos para seguidores**: Bots tendem a seguir muitos perfis e ter poucos seguidores. Eles buscam se propagar e ser seguidos de volta, ao contrário de humanos que geralmente têm uma proporção mais equilibrada, seguindo e sendo seguidos por pessoas reais.
- **Proporção de retweets entre os tweets da timeline**: Bots tendem a retweetar mais do que postar conteúdo original, funcionando para disseminar informações. Usuários reais costumam ter uma maior quantidade de postagens próprias, refletindo suas opiniões e atividades pessoais.
- **Menções únicas**: Bots frequentemente mencionam repetidamente as mesmas contas para engajar ou assediar. Usuários reais têm menções mais espaçadas e diversificadas, interagindo de maneira mais natural e variada.
- **Média do tamanho dos tweets**: Bots geralmente fazem tweets mais curtos com menos informações para maximizar o engajamento rápido. Em

contraste, usuários verdadeiros escrevem tweets mais longos, detalhando suas ideias e experiências.

- **Idade da conta em dias**: Bots tendem a ser mais novos, já que são frequentemente criados e descartados. Embora a idade da conta não deva ser o único critério, quando combinada com outras métricas, pode ajudar a identificar contas suspeitas e automatizadas.
- **Utilização de hashtags únicas**: Semelhante às menções, bots podem usar excessivamente uma mesma hashtag para promover tópicos específicos. Usuários reais geralmente usam uma variedade maior de hashtags para diferentes contextos e conversas.
- **Quantidade de tweets em relação à idade da conta**: Contas novas com muitos tweets são frequentemente bots que postam excessivamente. Já contas novas com poucos tweets são mais provavelmente de novos usuários humanos.
- **Média de tempo de postagem entre um tweet e outro**: Bots tendem a postar com intervalos muito curtos entre os tweets, enquanto humanos têm tempos mais espaçados, refletindo uma atividade mais orgânica.
- **Similaridade entre o nome do usuário e o username**: Bots são frequentemente criados em lote com usernames aleatórios e não relacionados ao nome exibido, muitas vezes contendo números, indicando um padrão genérico de criação automatizada.

4.2. DEFINIÇÃO E OBJETIVOS DA FERRAMENTA

Este trabalho desenvolve uma ferramenta web para detectar bots no Twitter, visando impedir a disseminação de informações falsas em redes sociais. Baseada em literatura explicitada no referencial teórico, a abordagem concentra-se na identificação de padrões de bots, que são elementos chave na propagação de desinformação online. O Twitter foi escolhido por suas características únicas e relevância acadêmica, porém mantendo a ferramenta adaptável a outras plataformas no futuro.

A metodologia envolve coletar e analisar dados de usuários do Twitter, focando em contas que disseminam informações falsas. Testes com métodos selecionados da literatura serão realizados para identificar as técnicas mais eficientes na detecção de bots e na análise de sua influência na disseminação de

desinformação. O objetivo final é a construção de uma ou mais funções heurísticas para serem utilizadas na categorização das contas.

4.2.1. Requisitos funcionais e não funcionais

4.2.1.1. Requisitos funcionais

- **Coleta de Dados**: A ferramenta precisa ser capaz de coletar dados de contas de usuários do Twitter, além de informações sobre postagens realizadas ou republicadas por esses usuários.
- **Análise de Dados**: Os dados coletados necessitam ser analisados e transformados em dados quantitativos e qualitativos para serem utilizados como entrada para funções heurísticas.
- **Construção de Funções Heurísticas**: As funções heurísticas serão construídas com base em informações fornecidas por datasets já anotados e estudados na literatura.
- **Identificação de Comportamentos:** O conjunto de funções heurísticas deverá receber informações de usuários e postagens e gerar uma análise sobre a probabilidade da conta pertencer a um humano comum ou a uma conta automatizada.

4.2.1.2. Requisitos não funcionais

- **Desempenho:** A ferramenta deve ser capaz de processar grandes volumes de dados de forma eficiente.
- **Usabilidade:** A interface deve ser intuitiva e fácil de usar.
- **Segurança:** Os dados coletados devem ser armazenados e manipulados de forma segura.
- **Disponibilidade:** A integração com o Twitter através de sua API deve estar disponível em tempo integral para a utilização da ferramenta.

4.2.2. Tecnologias

A ferramenta utiliza uma combinação de tecnologias modernas para garantir eficiência, escalabilidade e segurança. Node.js[7] e TypeScript[8] foram escolhidos por sua capacidade de lidar com operações assíncronas e grande volume de dados, além de proporcionar maior segurança no desenvolvimento com tipagem estática. A

[7] https://nodejs.org/pt/about
[8] https://www.typescriptlang.org/pt/

interface do usuário foi desenvolvida utilizando HTML, CSS e JavaScript devido a sua simplicidade. Para o armazenamento de dados, foi escolhido o MongoDB[9], um banco de dados NoSQL[10] que oferece flexibilidade e capacidade de armazenar dados em formato de documento, facilitando a manipulação de dados não estruturados.

4.2.3. Amazon Web Services

Amazon Web Services (AWS) é uma plataforma de serviços de computação em nuvem oferecida pela Amazon[11]. Ela fornece uma ampla gama de serviços, incluindo armazenamento, processamento, banco de dados, inteligência artificial, análise de dados e muito mais. A AWS é amplamente utilizada por empresas de todos os tamanhos devido à sua escalabilidade, confiabilidade e flexibilidade. Com data centers distribuídos globalmente, a plataforma permite que usuários implementem soluções robustas e seguras sem a necessidade de gerenciar infraestrutura física.

AWS Lambda[12] é um serviço de computação sem servidor oferecido pela AWS que permite executar código em resposta a eventos, sem a necessidade de provisionar e gerenciar servidores. Com o Lambda, os desenvolvedores podem criar funções que são ativadas automaticamente por eventos, como alterações em bancos de dados, uploads em buckets S3 ou chamadas a APIs. Essas funções são altamente escaláveis e cobram apenas pelo tempo de execução, tornando-as uma solução eficiente e econômica para aplicações baseadas em eventos.

4.2.4. MongoDB Atlas

MongoDB Atlas é uma plataforma de banco de dados como serviço baseada em nuvem, projetada para simplificar o uso do MongoDB[13]. Disponível em provedores de nuvem como AWS, Google Cloud e Azure, o MongoDB Atlas permite criar, escalar e gerenciar clusters de banco de dados com alta disponibilidade e segurança, sem a necessidade de gerenciar infraestrutura física ou virtual. A plataforma inclui ferramentas integradas para backup, monitoramento e otimização de desempenho, além de suporte para recursos como agregações, consultas em

---

[9] https://www.ibm.com/br-pt/topics/mongodb
[10] Termo que se refere a tipos não relacionais de bancos de dados
[11] https://aws.amazon.com/pt/what-is-aws/
[12] https://aws.amazon.com/pt/lambda/
[13] https://www.mongodb.com/pt-br/atlas

tempo real e integração com pipelines de dados. Por ser altamente escalável e flexível, o MongoDB Atlas é ideal para aplicações modernas que exigem alto desempenho e armazenamento dinâmico de dados não estruturados.

4.2.5. GitHub Pages

GitHub Pages é um recurso do GitHub que permite hospedar páginas da web diretamente de um repositório de código, tornando-o uma solução simples e gratuita para criar sites estáticos[14]. Ideal para documentações, portfólios, blogs e projetos, o GitHub Pages suporta HTML, CSS e JavaScript. O serviço é integrado ao repositório, o que facilita o versionamento e a publicação de alterações no site. Os sites são hospedados em domínios do GitHub ou em domínios personalizados configurados pelo usuário. Com facilidade de uso e integração contínua, o GitHub Pages é amplamente utilizado por desenvolvedores para compartilhar informações e projetos de forma prática e eficiente.

4.2.6. Twitter

Twitter é uma rede social que permite aos usuários publicarem e interagirem com mensagens curtas, conhecidas como tweets, que podem conter até 280 caracteres. Fundada em 2006, a plataforma ganhou popularidade rapidamente devido à sua capacidade de disseminar informações em tempo real[15]. Usuários do Twitter podem seguir outros usuários, compartilhar (retweetar) postagens, curtir tweets e participar de discussões em tempo real. O Twitter é amplamente utilizado por indivíduos, organizações e figuras públicas para comunicação instantânea, marketing, divulgação de notícias e interação com o público. Sua estrutura de comunicação rápida e concisa faz com que seja uma plataforma ideal para a disseminação de informações, mas também a torna suscetível à propagação de informações falsas e atividades de bots.

A API do Twitter permite que desenvolvedores acessem dados públicos do Twitter e interajam programaticamente com a plataforma. A API é baseada em REST e oferece diversos endpoints que permitem a coleta de dados como tweets, informações de perfis, listas de seguidores e seguidos, e tendências. Existem diferentes níveis de acesso à API, incluindo o acesso padrão, que é gratuito e

[14] https://pages.github.com/
[15] https://www.britannica.com/money/Twitter

oferece funcionalidades básicas, e os níveis premium e enterprise, que oferecem funcionalidades mais avançadas e maior volume de acesso.

A API do Twitter é utilizada para várias finalidades, incluindo análise de dados, monitoramento de redes sociais, desenvolvimento de bots e integração com outras aplicações. Na ferramenta desenvolvida neste projeto, a API do Twitter é utilizada para coletar dados de contas de usuários e informações sobre postagens realizadas ou republicadas.

Nos últimos anos, o Twitter passou por mudanças significativas que afetaram tanto sua estrutura quanto o ecossistema de ferramentas associadas. Em outubro de 2022, a plataforma foi adquirida por um novo proprietário, resultando em diversas alterações na política de acesso à sua API. Em fevereiro de 2023, foi anunciado que o acesso gratuito à API seria descontinuado, afetando desenvolvedores que utilizavam essas interfaces [34]. Essas mudanças tiveram um impacto direto em ferramentas de detecção de bots, como o Botometer e o Pegabot exibidos neste trabalharam, que ficaram inativas devido às restrições impostas. A necessidade de pagamento para acesso à API inviabilizou a continuidade de projetos acadêmicos e de monitoramento que dependiam desses dados para identificar comportamentos automatizados na plataforma.

4.2.7. Arquitetura

4.2.7.1. Visão geral da arquitetura

A arquitetura escolhida para a ferramenta é baseada em microsserviços serverless, utilizando funções AWS Lambda para implementar os principais componentes e garantindo baixo acoplamento e alta escalabilidade. A comunicação entre os serviços é mediada por API Gateway e ocorre de forma eficiente, permitindo a orquestração dos sistemas com foco na modularidade e independência. Os principais componentes são: coleta de dados, processamento e análise, e interface com o usuário.

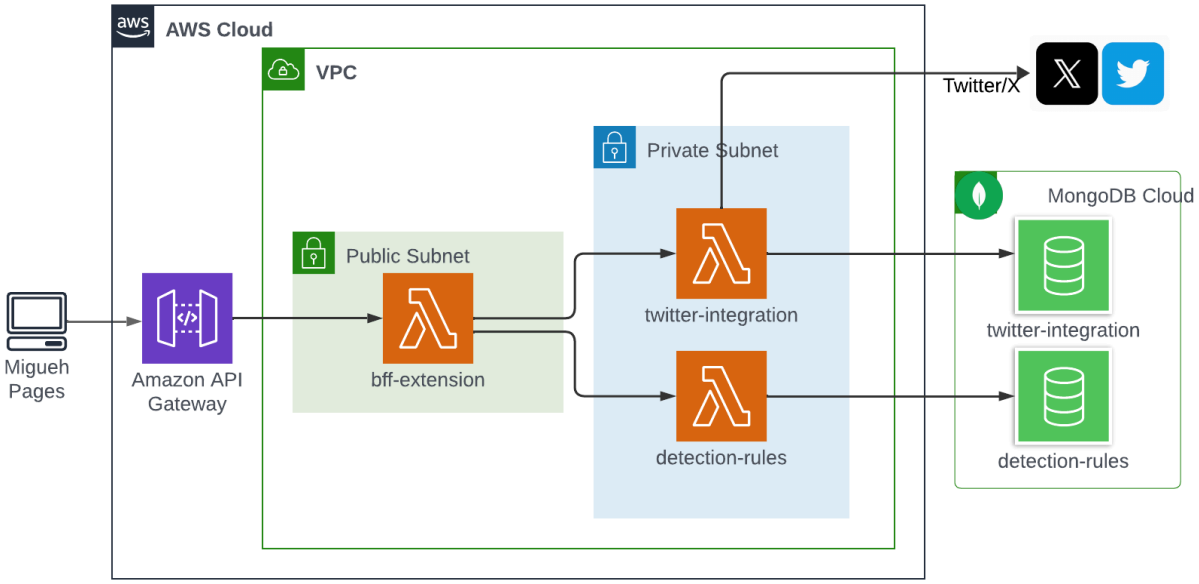


Figura 1. Desenho da arquitetura do sistema da ferramenta de detecção de bots

A escolha dessa arquitetura serverless proporciona escalabilidade automática, redução de custos operacionais e simplificação no gerenciamento da infraestrutura. A separação entre sub-redes públicas e privadas aumenta a segurança, garantindo que serviços internos não sejam expostos diretamente à internet. Além disso, a integração com o MongoDB Cloud e a utilização do GitHub Pages garantem alta disponibilidade e desempenho tanto no backend quanto no frontend.

- **Coleta de Dados (twitter-integration)**: Esse serviço, implementado como uma função Lambda, conecta-se à API do Twitter para coletar dados de contas de usuários e suas postagens. Ele opera dentro de uma sub-rede privada para aumentar a segurança, garantindo que os dados sejam processados de forma protegida. O serviço envia os dados coletados para armazenamento e análise.
- **Processamento e Análise (detection-rules)**: Este componente também é uma função Lambda, responsável por aplicar as regras heurísticas para identificar comportamentos que possam indicar a presença de bots ou contas humanas. Operando dentro da mesma sub-rede privada que o serviço de integração com o Twitter, ele processa os dados coletados e retorna os resultados de classificação. Esse serviço utiliza os dados armazenados no MongoDB Cloud para suportar suas análises.

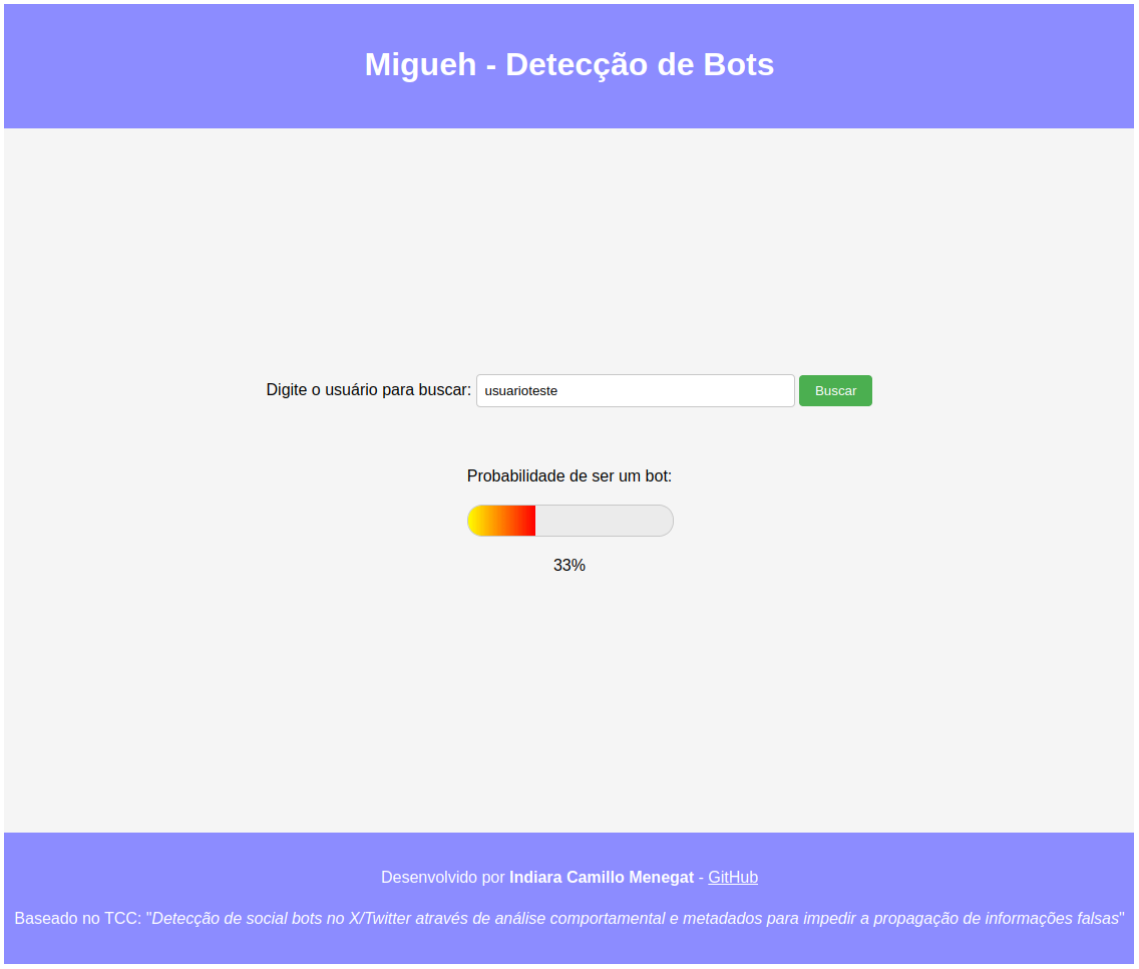


Figura 2. Design da tela da ferramenta

- **Coordenação entre serviços (bff-extension):** A função Lambda que atua como Backend for Frontend (BFF) é a única responsável por expor a API pública através do Amazon API Gateway. Esse serviço faz a ponte entre o frontend e os microsserviços internos, como o twitter-integration e o detection-rules. Ele opera em uma sub-rede pública, permitindo acesso controlado e seguro às APIs. Essa estratégia centraliza a lógica de coordenação sem expor diretamente os outros serviços.
- **Interface com o Usuário**: A interface com o usuário foi desenvolvida com HTML e CSS e está hospedada no GitHub Pages. O design dela pode ser observado na Figura 2. Ela fornece um ambiente responsivo e acessível onde os usuários podem interagir com a ferramenta, visualizar os resultados das análises e fornecer inputs. Ao inserir o username do usuário desejado, a interface gráfica conecta-se ao BFF para acessar os serviços de coleta e análise e retorna para o usuário final uma barra de calor que vai de 0% à 100% contendo a probabilidade da conta inserida

ser um bot, considerando que quanto mais próximo de 100% mais provável. Para acessar a página online consulte o Apêndice B.

- **Armazenamento de Dados (MongoDB Cloud)**: Os dados coletados e analisados são armazenados em formato de documento no MongoDB Cloud. Essa solução foi escolhida por sua escalabilidade e flexibilidade na manipulação de dados, permitindo a fácil recuperação e suporte ao crescimento da base de dados.

4.3.CONSTRUÇÃO DA FERRAMENTA

Nessa seção serão detalhadas as etapas da construção do serviço responsável por categorizar um usuário, numa escala de 0 à 100, onde próximo de 0 refere-se a uma probabilidade baixa da conta ser um bot e com um valor próximo de 100 a probabilidade alta de ser. Para esta abordagem, foi escolhida uma metodologia de análise de contas através de um conjunto de regras, que servem como parâmetros comportamentais de uma conta em rede social, que pode categorizar a conta como mais propensa ou menos propensa a ser uma conta automatizada.

4.3.1. Desenvolvimento inicial

O desenvolvimento inicial envolveu a configuração do ambiente de desenvolvimento, incluindo a instalação de dependências e a configuração dos repositórios de código. Foi desenvolvido um template com uma estrutura de código básica na qual os projetos dos serviços twitter-integration e bot-detection se baseiam.

Assim, foi construído um MVP que incluía funcionalidades básicas. No serviço twitter-integration, foram implementadas as funcionalidades de buscar dados de usuários e buscar postagens, transformando os dados retornados da API do Twitter em informações mais simples. Isso inclui detalhes sobre as postagens, como se eram retweets, postagens próprias ou respostas a outros usuários, além da categorização do uso de hashtags. Para as contas de usuários, foram salvas informações básicas, como o número de seguidores, o número de seguidos, e a data de criação da conta. Já o serviço bot-detection ficou responsável em transformar os dados qualitativos oriundos do pré-tratamento realizado pelo twitter-integration em dados quantitativos utilizados para a análise exploratória nas próximas etapas.

Tabela 1. Relação dos dados de usuários e postagens tratados pelo serviço twitter-integration

Tipo de Dados	Descrição
Dados de usuário	
id	Identificador único da conta do usuário
username	Nome de usuário
name	Nome completo do usuário
description	Biografia do usuário
location	Localização do usuário
verified	Indica se a conta é verificada
accountCreatedAt	Data de criação da conta
accountDeletedAt	Data de exclusão da conta, se aplicável
nFollowers	Quantidade de seguidores do usuário
nFollowing	Quantidade de contas que o usuário segue
nTweets	Quantidade de tweets postados pelo usuário
Dados de postagens	
id	Identificador único do tweet
text	Conteúdo textual do tweet
authorId	Identificador do autor do tweet
nRetweet	Quantidade de retweets recebidos pelo tweet
nReply	Quantidade de respostas recebidas pelo tweet
nLike	Quantidade de curtidas recebidas pelo tweet
nQuote	Quantidade de citações recebidas pelo tweet
mentions	Lista de menções a outros usuários no tweet
isReply	Indica se o tweet é uma resposta
isRetweet	Indica se o tweet é de outra conta que não do autor
geolocation	Localização geográfica do tweet
tweetCreatedAt	Data de criação do tweet

Tabela 2. Relação dos dados gerados para análise de usuarios e postagens pelo serviço bot-detection

Tipo de Dados	Descrição
nTweet	Quantidade total de tweets postados pelo usuário
nFollower	Quantidade total de seguidores do usuário
nFollowing	Quantidade total de contas seguidas pelo usuário
location	Localização do usuário
hasLocation	Indica se o usuário possui localização
username	Username do usuário
usernameSize	Comprimento do nome de usuário
nNumberUsername	Quantidade de números no nome de usuário

nLettersUsername	Quantidade de letras no nome de usuário
name	Nome exibido pelo usuário
nameSize	Comprimento do nome completo
descriptionSize	Comprimento da biografia do usuário
accountAgeInDays	Quantidade de dias desde a criação da conta
timelineSampleFullSize	Quantidade total de tweets na amostra
timelineSampleReplySize	Quantidade de tweets na amostra que são respostas
timelineSampleRetweetSize	Quantidade de tweets na amostra que são retweets
timelineSampleUserTweetSize	Quantidade de tweets na amostra que são postagens próprias
timelineSampleUserTweetTextSizeAvg	Tamanho médio dos tweets criados pelo usuário na amostra
timelineSampleHashtagCount	Quantidade de hashtags usadas na amostra
timelineSampleMentionCount	Quantidade de menções feitas na amostra
timelineSamplePostCreatedAtDates	Datas de criação dos tweets na amostra
mentions	Mapa de menções e suas contagens na amostra
hashtags	Mapa de hashtags e suas contagens na amostra
retweets	Mapa de retweets e suas contagens na amostra

4.3.2. Análise Exploratória

A análise exploratória de dados foi uma etapa crucial no desenvolvimento da ferramenta, pois permitiu a compreensão preliminar dos dados e a identificação de padrões e comportamentos característicos de bots e contas humanas no Twitter. Nesta seção, descreve-se o processo de coleta, preparação e análise inicial dos dados.

4.3.2.1. Coleta e Preparação do Dataset

Para testar as hipóteses definidas nas regras, foi utilizado o dataset TwiBot-22 [29]. Até o momento, TwiBot-22 é o mais amplo e completo benchmark para a detecção de bots no Twitter. Ele foi desenvolvido para superar os desafios de escala limitada de conjuntos de dados, estrutura de grafos incompleta e baixa qualidade de anotações em datasets anteriores.

O TwiBot-22 foi construído utilizando um processo de coleta de dados em duas etapas. A primeira etapa emprega o método de busca em largura para a coleta de usuários, iniciando a partir de "usuários-semente" e expandindo com base nas relações de seguidores. Esse método permite alcançar uma representação mais abrangente da rede social. Para a anotação dos dados, foi empregada uma estratégia de aprendizado por supervisão fraca, que permite classificar contas como

bots ou humanos com base em inferências, sem depender exclusivamente de rótulos perfeitos. Além disso, 35 modelos representativos de detecção de bots foram aplicados para comparação de desempenho em diferentes datasets, incluindo o próprio TwiBot-22, promovendo uma análise abrangente do progresso da pesquisa. O TwiBot-22 também incorpora métodos baseados em grafos, representando contas e suas conexões como nós e arestas, explorando as relações sociais e comportamentais entre os usuários. Por fim, todas essas etapas foram consolidadas em um framework de avaliação, permitindo que pesquisadores reproduzam os experimentos realizados e testem o benchmark em novos datasets e modelos.

Tabela 3. Estrutura do dataset TwiBot-22

Arquivo	Descrição
tweet_i.json	Informações dos tweets. Idêntico ao que pode ser recuperado com a API do Twitter. São 8 arquivos de 10GB cada.
user.json	Informações dos usuários. Idêntico ao que pode ser recuperado com a API do Twitter.
list.json	Informações das listas. Idêntico ao que pode ser recuperado com a API do Twitter.
hashtag.json	Informações das hashtags. Idêntico ao que pode ser recuperado com a API do Twitter.
split.csv	Divisão dos dados, onde a primeira coluna (id) é o id do usuário e a segunda coluna (split) é a divisão correspondente (train, valid ou test).
label.csv	Rótulos verdadeiros, onde a primeira coluna (id) é o id do usuário e a segunda coluna (label) é o rótulo correspondente (humano ou bot).
edge.csv	Relações das entidades que aparecem em todos os outros arquivos. Cada uma das entradas contém source_id, target_id e tipo de relação.

Fonte: Repositório do projeto Twibot-22[16]

Para este trabalho, foram utilizados os oito arquivos JSON contendo tweets, o arquivo JSON contendo usuários além do arquivo CSV com os rótulos sobre a natureza das contas, sendo um bot ou um humano.

Para facilitar a análise e tornar os dados mais gerenciáveis, foi gerada uma amostra de usuários representativa de 200 mil usuários, sendo 100 mil categorizados como humanos e 100 mil categorizados como bots. A amostra foi selecionada de forma aleatória.

Após a seleção e preparação dos dados dos usuários, foi necessário coletar e preparar os tweets associados a essas contas. Cada um dos oito arquivo de tweets foi processado para filtrar apenas os tweets relacionados aos usuários da amostra.

Todo o tratamento dos dados e geração da amostra foram feitos através da plataforma Google Colab. O Google Colab oferece um ambiente interativo baseado

[16] https://github.com/LuoUndergradXJTU/TwiBot-22

em Jupyter Notebooks, sendo ideal para lidar com grandes volumes de dados e realizar operações computacionais intensivas. Para estes processos, foram utilizadas inúmeras bibliotecas Python, especialmente o Pandas, uma biblioteca focada em ciência de dados.

Tabela 4. Comparação entre os dados do TwiBot-22 e os dados da amostra

	TwiBot-22	**Amostra**
Usuários	1.000.000	200.000
Humanos	860.057	100.000
Bots	139.943	100.000
Tweets	86.764.167	13.299.135

Após a seleção da amostra e a coleta dos tweets a partir dos arquivos, os dados crus dos usuários foram adicionados através de um script na base de dados do serviço twitter-integration para serem utilizados futuramente pelo serviço bot-detection.

4.3.2.2. Transformação dos dados

Para realizar a análise, foi necessário transformar os dados da amostra em um formato adequado para a aplicação das regras de detecção de bots. Esta transformação foi realizada diretamente no serviço bot-detection, onde foram extraídas e calculadas as métricas específicas utilizadas para identificar comportamentos típicos de bots e contas humanas no Twitter. As funções responsáveis pela transformação já haviam sido implementadas no serviço previamente no desenvolvimento inicial da ferramenta.

Para criar os dados transformados da análise, foi executado um script em python que, para cada uma das contas da amostra, gerava a análise fazendo chamadas de API ao bot-detection com informações de id e username. Após adicionados os dados, foi feita uma busca de usuários por usernames e comparado os ids destes usuários com o ids contidos no arquivo de anotações sobre a conta sendo bot ou humana. As anotações adicionadas as análises no banco de dados do serviço bot-detection.

Os dados transformados foram salvos no banco de dados do serviço bot-detection, depois exportados para CSV e abertos dentro do Google Colab para realizar a análise exploratória. A transformação logarítmica foi aplicada aos dados para reduzir a escala de variação e facilitar a visualização dos dados.

Tabela 5. Resumo das métricas calculadas.

Métrica	Descrição
followingToFollowerRatioScore	Proporção de seguidos para seguidores
retweetToTweetRatioScore	Proporção de retweets entre os tweets da timeline
mentionsPerUserScore	Proporção de menções únicas feitas
tweetSizeAvgScore	Média do tamanho dos tweets
accountAgeScore	Idade da conta em dias
hashtagUsageScore	Proporção de hashtags únicas utilizadas
tweetCountToAccountAgeScore	Quantidade de tweets postados em relação à idade da conta
similarityBetweenNameAndUsernameScore	Similaridade entre o nome do usuário e o username
avgTimeBetweenPostsScore	Média de tempo de postagem entre um tweet e outro

4.3.2.3. Avaliação individual das regras de detecção

Para avaliar a eficácia de cada regra na distinção entre bots e humanos, foram utilizados diversos métodos estatísticos e gráficos. Histogramas permitiram observar as distribuições gerais das métricas, enquanto box plots identificaram medidas centrais, dispersões e outliers. Violin plots foram empregados para detalhar as densidades dos dados, e estatísticas descritivas, como médias e medianas, complementam a análise. Essas abordagens possibilitaram uma visão abrangente do comportamento de bots e humanos em relação a cada métrica.

1. **Proporção de Seguidos para Seguidores:** Essa métrica revelou padrões marcantes que diferenciam bots de humanos. Os bots tendem a apresentar proporções significativamente altas, concentrando-se próximos ao limite superior arbitrariamente definido como 1.3882. Isso reflete o comportamento automatizado de seguir muitas contas para ampliar sua exposição, sem receber o mesmo número de seguidores em retorno. Em contraste, humanos exibem uma distribuição mais equilibrada, com predominância de valores baixos, sugerindo uma interação mais orgânica. Os histogramas e violin plots confirmaram essa diferença, enquanto os box plots mostraram a forte concentração de bots no limite superior. Essa métrica demonstrou alta relevância, sendo essencial para a classificação.

2. **Proporção de Retweets entre os Tweets da Timeline:** Bots frequentemente replicam conteúdo de forma intensa, resultando em proporções de retweets consideravelmente altas, enquanto humanos equilibram a criação de conteúdo original e o compartilhamento. Essa tendência foi destacada pelos histogramas, que mostraram maior densidade de bots em valores elevados, e pelos box plots, que indicaram medianas mais altas para bots. Apesar de a diferença entre os grupos não ser tão acentuada quanto em outras métricas, a regra ainda contribui moderadamente para a função de classificação.
3. **Menções Únicas:** A proporção de menções únicas evidenciou comportamentos similares entre bots e humanos, com ambos os grupos mostrando altas concentrações em valores baixos, refletindo a tendência de mencionar contatos recorrentes. No entanto, humanos demonstraram leve diversificação, com uma densidade marginalmente maior em valores mais altos. Os box plots destacaram a presença de outliers significativos, indicando que algumas contas, em ambos os grupos, mencionam uma grande variedade de usuários. Essa métrica, embora útil, tem impacto limitado devido à variabilidade e à sobreposição observada.

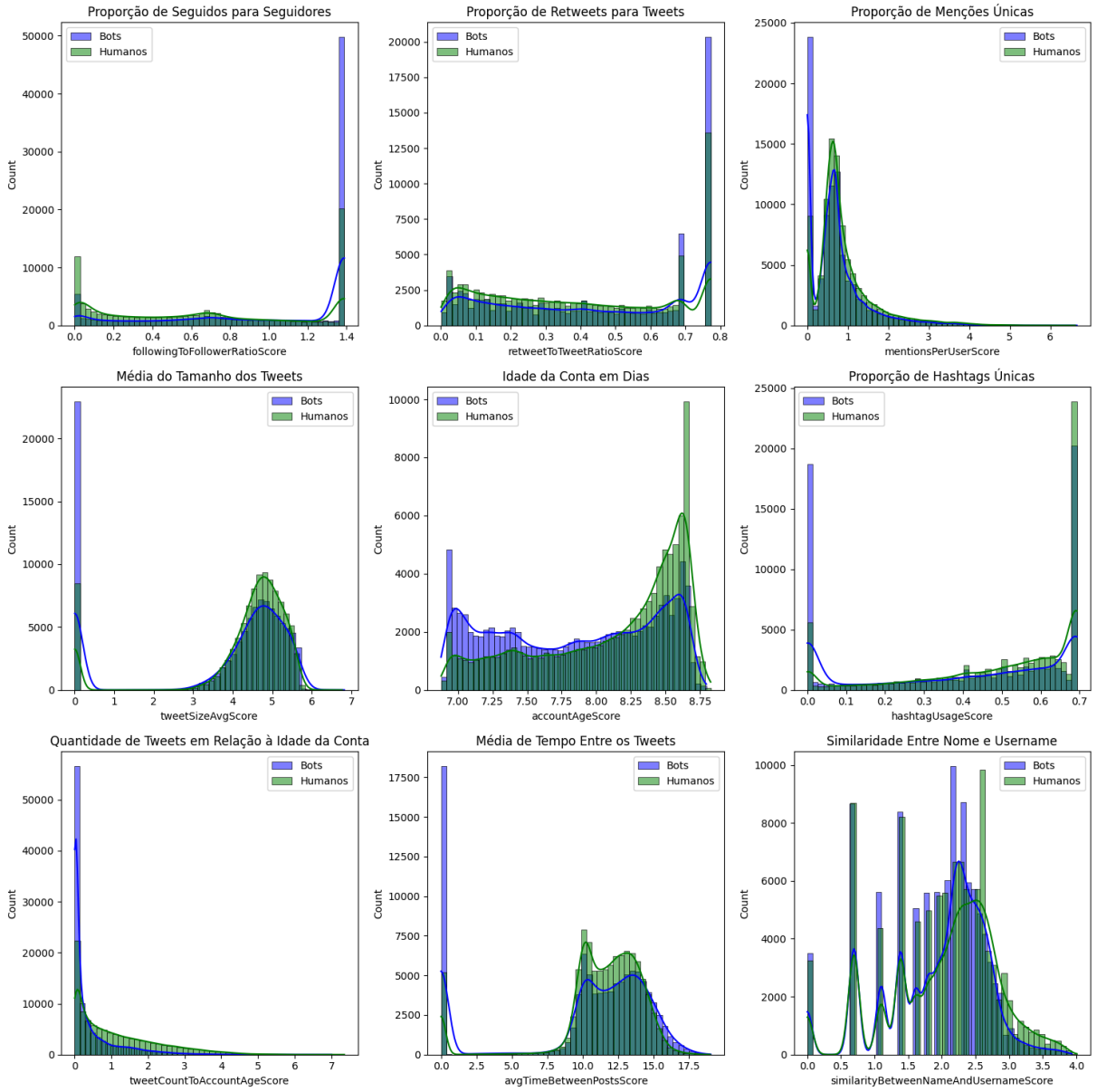


Figura 3. Histogramas das nove regras definidas separados entre bots e humanos

4. **Média do tamanho dos Tweets:** Bots tendem a postar mensagens curtas e padronizadas, enquanto humanos exibem maior diversidade no tamanho médio dos tweets. Histogramas mostram uma densidade maior para bots em valores entre 3 e 5 palavras, enquanto humanos apresentaram uma distribuição mais ampla. Box plots corroboraram essas diferenças, com medianas ligeiramente superiores para humanos. Apesar de a diferença ser sutil, a métrica fornece insights consistentes sobre padrões de comportamento e contribui moderadamente para a classificação.

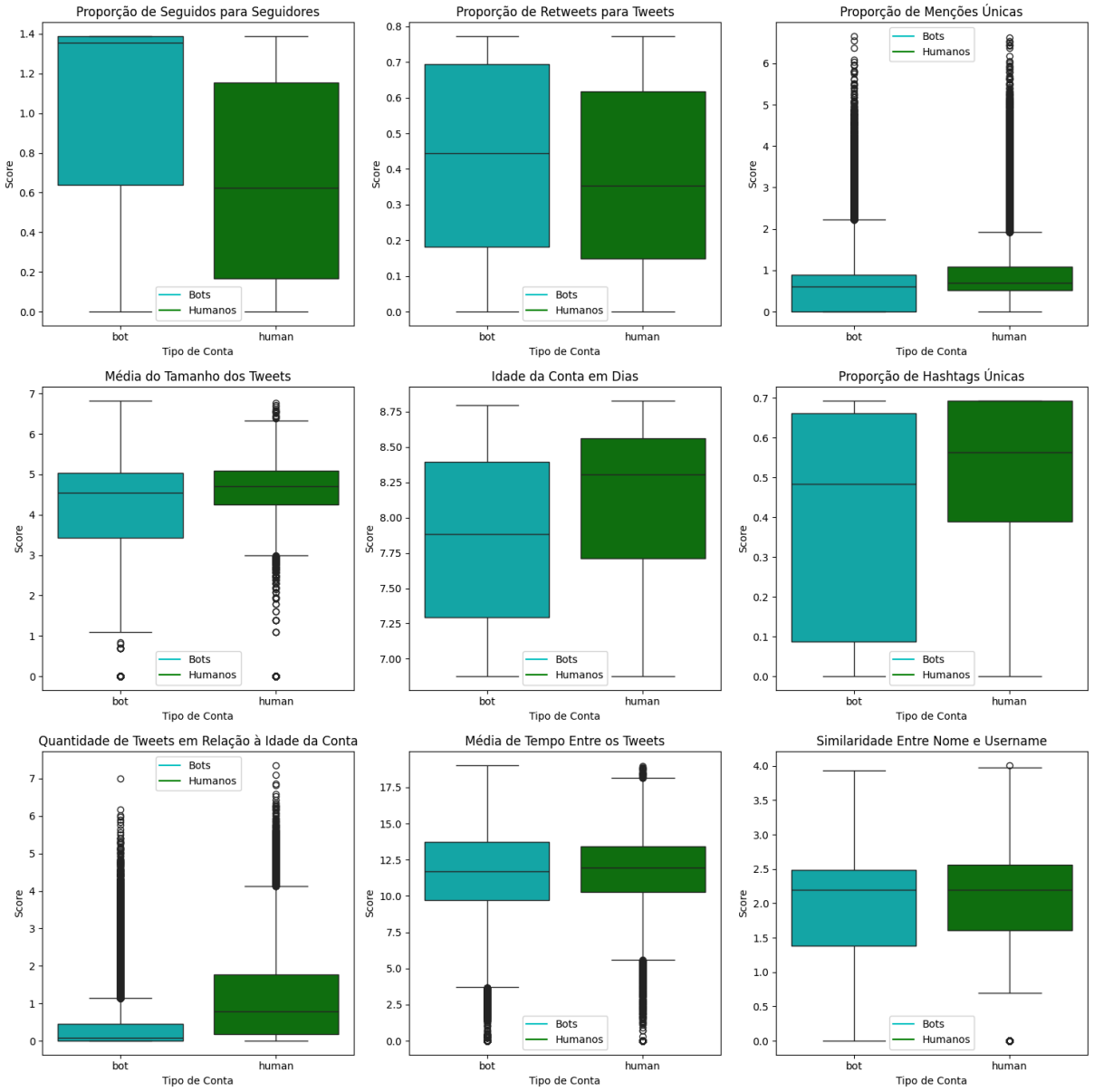


Figura 4. Box plots das nove regras definidas separados entre bots e humanos

5. **Idade da Conta em Dias:** A idade da conta mostrou-se uma métrica poderosa para diferenciar bots e humanos. Contas humanas geralmente são mais antigas, refletindo um uso contínuo ao longo do tempo, enquanto bots apresentam maior dispersão, sendo frequentemente criados para atividades temporárias. Histogramas mostraram concentrações claras de humanos em valores elevados, enquanto os bots se distribuíram de forma mais uniforme. Box plots reforçaram essas observações, indicando uma faixa interquartil superior para humanos. Essa regra destacou-se como uma das mais relevantes na detecção de bots.

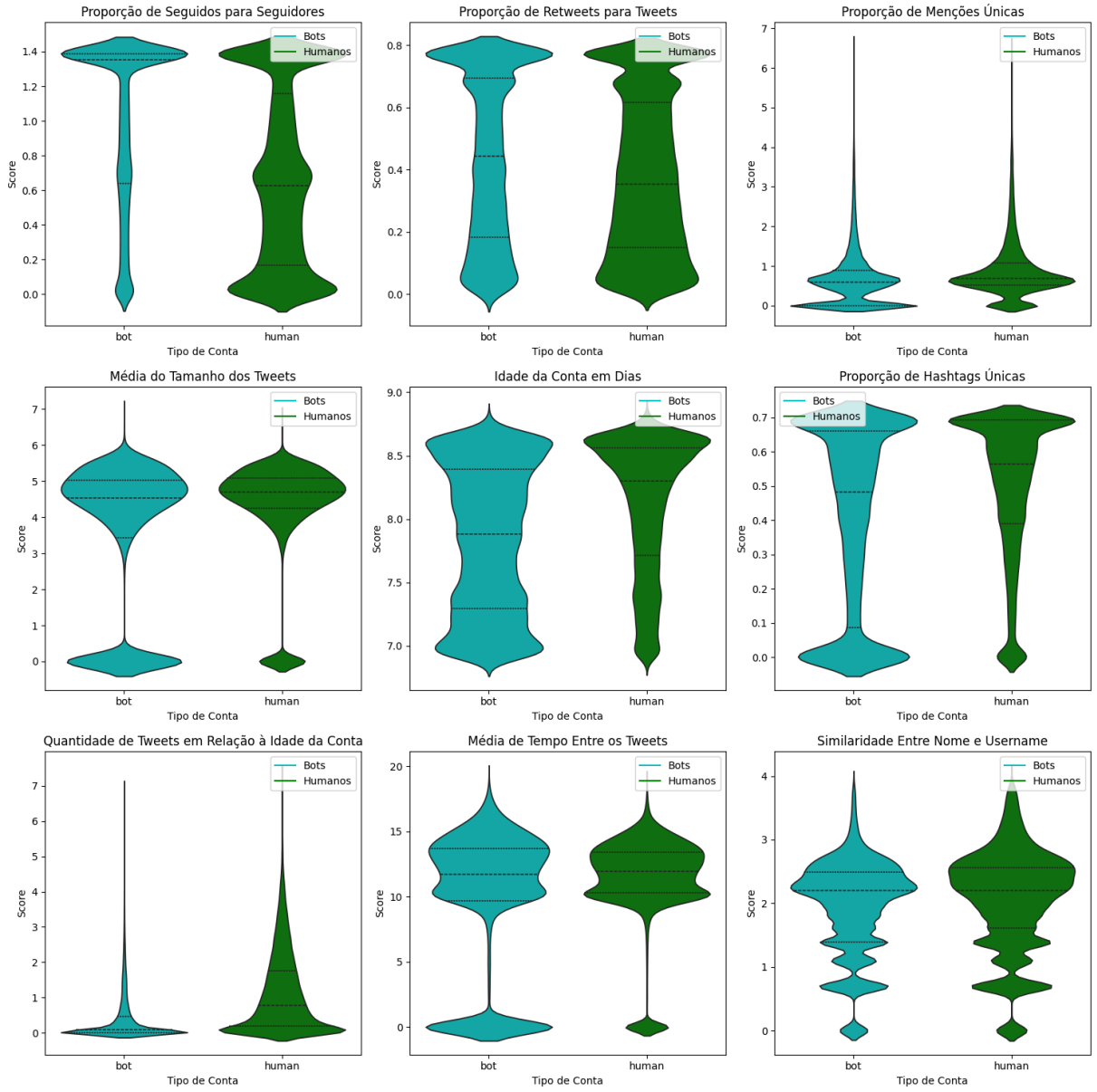


Figura 5. Violin plots das nove regras definidas separados entre bots e humanos

6. **Utilização de Hashtags Únicas:** A diversidade no uso de hashtags revelou diferenças moderadas entre bots e humanos. Bots tendem a utilizar hashtags de maneira mais repetitiva e limitada, enquanto humanos exibem maior variação. Embora os histogramas e violin plots tenham mostrado leve deslocamento dos humanos para valores mais altos, a sobreposição significativa entre os grupos reduziu o impacto dessa métrica na classificação.
7. **Quantidade de Tweets em Relação à Idade da Conta:** Essa métrica destacou-se por sua eficácia em identificar padrões de atividade. Bots apresentaram proporções muito baixas de tweets por idade da conta,

indicando alta frequência de postagens automatizadas ou atividade limitada em contas novas. Humanos, por outro lado, mostraram maior variação e taxas médias mais altas. Box plots confirmaram essas tendências, com humanos exibindo uma faixa interquartil ampla e bots concentrados próximos de zero. Essa métrica é altamente relevante para distinguir os dois grupos.

8. **Média de Tempo de Postagem entre um Tweet e Outro:** A análise do tempo médio entre postagens revelou uma diferença clara entre bots e humanos. Bots tendem a postar com maior frequência, resultando em intervalos mais curtos e padronizados, enquanto humanos possuem cadências mais espaçadas e orgânicas. Histogramas mostram picos significativos para bots em intervalos curtos, enquanto humanos apresentaram uma distribuição mais ampla. Box plots e violin plots confirmaram esses padrões, tornando essa métrica útil para a classificação.
9. **Similaridade entre Nome do Usuário e Username:** A similaridade entre nome e username apresentou diferenças mínimas entre bots e humanos. Bots mostraram valores ligeiramente mais baixos, indicando menor consistência ou personalização, mas a sobreposição entre os grupos foi significativa. Box plots mostraram que, embora essa métrica contribua como característica adicional, seu impacto na classificação geral é limitado.

4.3.2.4. Conclusão sobre as regras de detecção

As regras definidas são, em sua maioria, eficazes para distinguir bots de humanos no Twitter, com algumas métricas demonstrando maior impacto na capacidade de diferenciação. Regras de alto impacto, como a proporção de seguidos para seguidores, a média de tempo entre postagens, a proporção de retweets e a quantidade de tweets em relação à idade da conta, são particularmente relevantes e devem receber maior peso em um sistema de detecção. Métricas de impacto moderado, como o tamanho médio dos tweets, as menções por usuário e a idade da conta, também contribuem para a diferenciação, mas com menor prioridade. Por fim, regras de baixo impacto, como a proporção de hashtags únicas e

a similaridade entre nome e nome de usuário, mostraram-se menos eficazes e podem receber menor peso na classificação.

Considerando o que foi analisado, então, decidiu-se atribuir pesos a cada regra para aplicá-las no algoritmo de categorização. Regras que caem nos grupos de Alto, Moderado e Baixo impacto têm pesos diferentes, mesmo que com a mesma classificação de impacto, pois apresentaram-se na análise como mais significativas. A ideia é que estes pesos sejam ajustados conforme novas análises e validações de dados surjam em trabalhos futuros, por isso, serão variáveis dentro do sistema.

Tabela 6. Resumo do impacto e dos pesos iniciais atribuídos às regras

Regra	Impacto	Peso
Proporção de seguidos para seguidores	Alto	1,5
Média de tempo entre postagens	Alto	1,5
Proporção de retweets	Alto	1,3
Quantidade de tweets em relação à idade	Alto	1,4
Tamanho médio dos tweets	Moderado	1,3
Menções por usuário	Moderado	1,2
Idade da conta	Moderado	1,1
Proporção de hashtags únicas	Baixo	0,8
Similaridade entre Nome e Nome de Usuário	Baixo	0,6

4.3.3. Função de classificação

Para o desenvolvimento da função de classificação, foi levado em conta os pesos atribuídos na avaliação das regras e também os dados gerados em cima da amostra de contas, que já tiveram outliers tratados e passaram por transformação logarítmica.

A primeira etapa consistiu em definir os limites máximos para os scores das regras, com o limite inferior padronizado em zero, pois todos os cálculos resultaram em valores positivos. O limite superior foi estabelecido com base no 95º percentil de cada métrica, utilizando a função de quartis do Pandas. Esse ponto de corte destaca os valores mais altos (5% superiores) de cada regra, focando na distribuição central e minimizando a influência de outliers, permitindo uma interpretação robusta das métricas.

A escolha do percentil permite que o modelo seja ajustado conforme a base de dados cresce, recalculando os limites periodicamente para refletir melhor as

características dos dados. Com isso, o sistema de classificação pode adaptar-se a mudanças nos padrões de comportamento de bots e humanos, aprimorando a precisão da função de classificação ao longo do tempo.

Tabela 7. Limites superiores atribuídos às regras.

Regra	Limite
Proporção Seguindo/Seguidores	1,3883
Proporção de Retweets	0,77242
Menções por Usuário	2,2824
Tamanho Médio dos Tweets	5,527
Idade da Conta	8,6557
Proporção de Hashtags Únicas	0,69315
Número de Tweets em Relação à Idade da Conta	2,913
Tempo Médio entre Postagens	15,322
Similaridade entre Nome e Nome de Usuário	3,0445

A segunda etapa seguiu a normalização das métricas, para que fossem colocadas em um intervalo comum entre 0 e 1, para melhor visualização. A fórmula de normalização foi $normalized\ value = \frac{value - min}{max - min}$, sendo $min = 0$ e $max = limite\ superior\ da\ regra.$ Algumas métricas foram invertidas, pois quanto mais próximo de zero, mais indicam comportamento humano; já outras têm o comportamento oposto e foram normalizadas diretamente. As regras que foram invertidas foram Idade da Conta, Número de Tweets em Relação à Idade da Conta e Similaridade entre Nome e Nome de Usuário.

Ao final, a função de categorização foi desenvolvida da seguinte maneira:

- Receber configurações de limites, pesos e os dados brutos de análise de perfil de um usuário
- Para cada regra contida na análise de perfil
    - **Normaliza** os valores das métricas entre 0 e 1.
    - **Ajusta** as pontuações das métricas onde necessário (inversão).
    - **Calcula** a pontuação total ponderada.
- Retorna pontuação total final

Tabela 8. Pseudo-código da função de classificação

```
FUNÇÃO categorizar(configuração, análisePerfil)
    limites <- configuração.limits
    pesos <- configuração.weights
    pontuaçãoFinal <- 0

    PARA cada regra EM análisePerfil
        chave, valor <- regra

        SE chave ESTÁ EM camposParaIgnorar
            CONTINUAR

        peso <- pesos[chave]
        limite <- limites[chave]
        SE chave ESTÁ EM regrasInvertidas
            pontuação <- (1 - normalizar(valor, 0, limite)) * peso
        SENÃO
            pontuação <- normalizar(valor, 0, limite) * peso

        pontuaçãoFinal <- pontuaçãoFinal + pontuação

    RETORNAR pontuaçãoFinal
FIM FUNÇÃO
```

5. ANÁLISE E RESULTADOS

5.1. Avaliação inicial da classificação

Foi feita uma avaliação usando como base o amostra utilizado durante todo o trabalho. A ideia é verificar se os cálculos da função de classificação refletem os dados que foram utilizados e se é necessário algum ajuste na função, seja na lógica, nos pesos ou nos limites. Para avaliar isso foi comparado as pontuações, armazenadas em *accountScore*, com as classificações prévias do dataset, armazenadas em *accountLabel*. Por exemplo, iremos verificar se o usuário X do tipo de conta bot tem um *accountScore* que caia dentro da pontuação esperada para contas deste tipo.

Inicialmente consultou-se os dados estatísticos do dataset, dividido entre bots e humanos. Os resultados mostram que a média de *accountScore* é mais alta para bots, sendo de 6,41 do que para humanos sendo de 5,70, o que sugere que a função de classificação está conseguindo diferenciar as contas, pelo menos em um

nível médio. Além disso, o intervalo interquartil também é mais alto para bots, reforçando a ideia de que os bots tendem a ter pontuações mais altas.

No entanto, há uma sobreposição significativa, especialmente no 75º percentil para humanos (6,51), que está muito próximo da mediana para bots (6,47). Isso indica que, embora as pontuações de bots sejam geralmente mais altas, alguns humanos também recebem pontuações relativamente elevadas, o que pode gerar falsos positivos.

Tabela 9. Estatísticas para bots e humanos após a primeira classificação

Métrica	Bots	Humanos
Média	6,417	5,701
Desvio padrão	1,080	1,172
Mínimo	0,992	1,027
1º quartil	5,694	4,885
Mediana	6,470	5,670
3º quartil	7,173	6,510
Máximo	9,912	10,200

O gráfico de distribuição sugere que há uma diferença entre as pontuações médias de bots e humanos, mas ela é sutil. A distribuição das pontuações para bots parece estar um pouco deslocada para valores mais altos em comparação com a de humanos, o que indica que o sistema de classificação é capaz de captar algumas nuances entre os tipos de conta. No entanto, como as distribuições se sobrepõem bastante, há um risco significativo de erro, especialmente em contas com scores próximos à média de ambos os grupos.

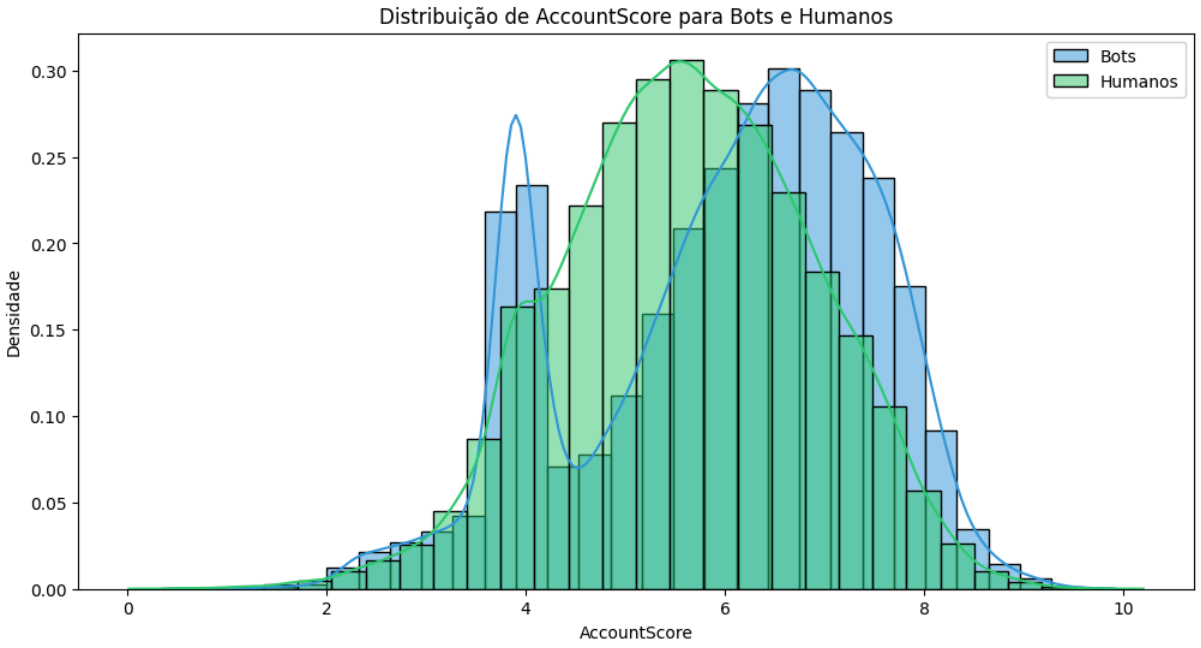


Figura 6. Histograma exibindo a distribuição dos valore de pontuação para bots e humanos

Para avaliar mais profundamente a classificação, foi utilizado a Curva ROC[17] e a AUC[18], são métricas usadas para avaliar o desempenho de classificadores binários, como no caso de diferenciar entre contas de bots e humanas. A Curva ROC é um gráfico que relaciona a taxa de verdadeiros positivos (TPR) com a taxa de falsos positivos (FPR) em vários limites de decisão. A TPR, ou sensibilidade, representa a proporção de bots corretamente identificados como bots, enquanto a FPR indica a proporção de contas humanas incorretamente classificadas como bots. Quanto mais a curva se aproxima do canto superior esquerdo (alta TPR e baixa FPR), melhor é o desempenho do classificador.

A AUC (Área Sob a Curva) fornece uma métrica única para avaliar o modelo: quanto mais próxima de 1, melhor a capacidade do classificador de distinguir entre as classes. Uma AUC de 1 indica um classificador perfeito, enquanto uma AUC de 0,5 indica um classificador aleatório. Utilizar a Curva ROC e a AUC permite avaliar a capacidade do modelo, independentemente de um ponto de corte específico, e é especialmente útil em cenários com desbalanceamento entre classes ou quando os custos de erro variam.

---

[17] Representa graficamente a relação entre a Taxa de Verdadeiros Positivos (TPR) e a Taxa de Falsos Positivos (FPR) para diferentes limites de classificação.
[18] Mede a área sob a Curva ROC, indicando a capacidade geral do modelo de distinguir entre classes. Um valor de AUC próximo de 1 indica alto desempenho do modelo.

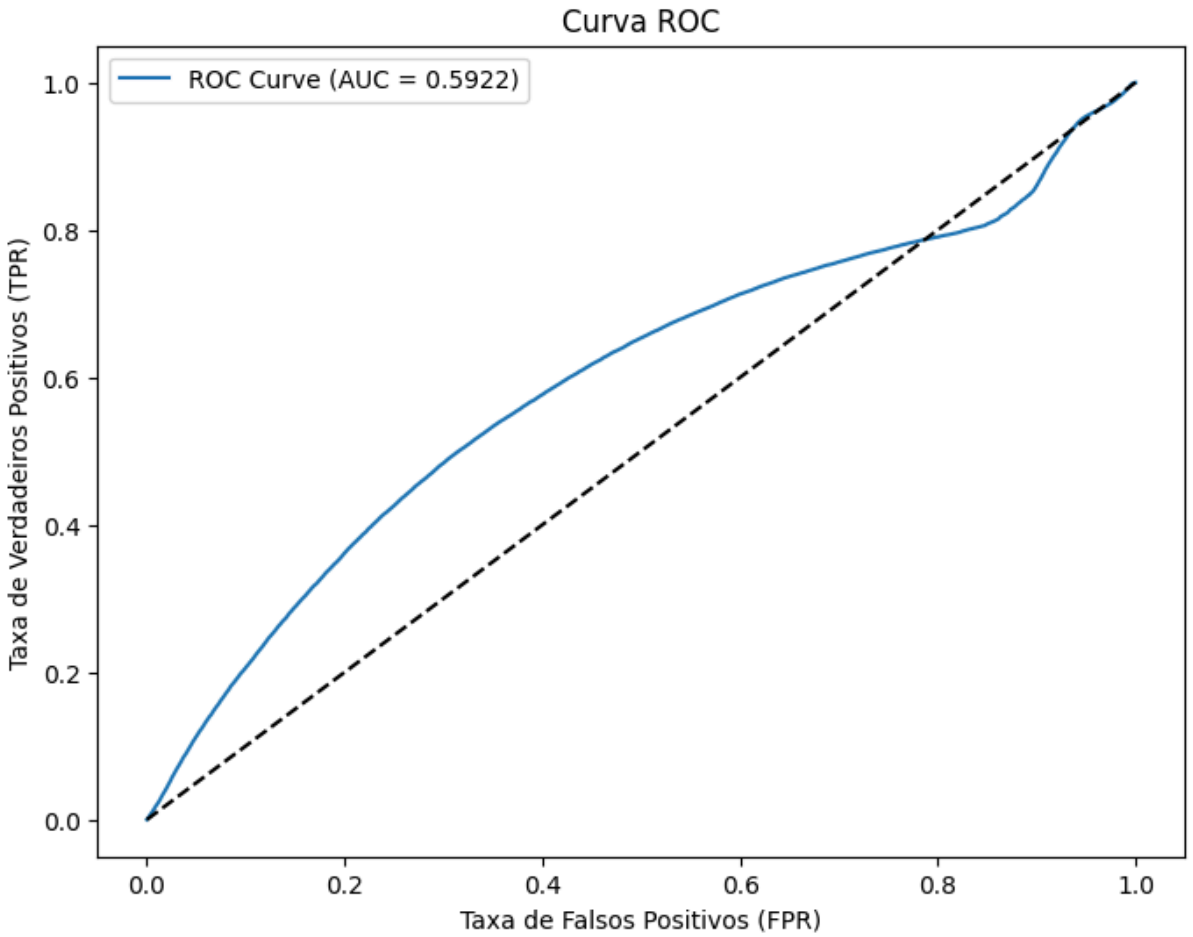


Figura 7. Curva ROC e área sob a curva

Com os pesos, limites superiores e a lógica inicial da função como, fora definida e desenvolvida na construção da ferramenta, a curva ROC gerada não se aproxima do canto superior esquerdo, que indicaria uma alta TPR e uma baixa FPR, o que é desejável em um modelo de classificação. A área sob a curva ROC (AUC) é de 0.5922, o que indica uma capacidade de classificação um pouco acima de um modelo aleatório (que teria uma AUC de 0.5). Interpretando essas duas métricas, conclui-se que a função de classificação está acertando levemente mais do que uma classificação aleatória, mas está longe de ser um classificador confiável.

5.2. Análise de impacto das regras

Para realizar as alterações necessárias na função de classificação, previamente, foi empregado o uso de três técnicas de machine learning e estatística para analisar o grau de influência das regras no dataset todo.

5.2.1. Análise de correlação com a variável alvo

Esta é uma técnica estatística que mede a força e a direção da relação linear entre cada variável explicativa e a variável alvo [30]. Neste contexto, a correlação indica o grau de associação de cada regra com a classificação da conta. Valores de correlação mais altos, sendo positivos ou negativos, sugerem uma relação mais forte com o tipo de conta, ajudando a identificar quais regras são mais impactantes.

Para esta análise, regras como a contagem de tweets em relação a idade da conta (*tweetCountToAccountAgeScore*), proporção de seguidores para seguidos (*followingToFollowerRatioScore*), e idade da conta (*accountAgeScore*) mostram correlações mais pronunciadas, sugerindo que têm um impacto mais evidente em diferenciar contas de bots e humanos.

5.2.2. Coeficientes de Regressão Logística

A regressão logística é um modelo estatístico usado para prever a probabilidade de uma variável binária com base em uma ou mais variáveis explicativas [31]. Os coeficientes da regressão logística representam a força e a direção da associação entre cada variável e a probabilidade de uma conta ser classificada como bot. Coeficientes positivos indicam que o aumento naquela regra aumenta a chance de a conta ser bot, enquanto coeficientes negativos indicam uma diminuição dessa chance.

Nesta análise, o uso de hashtags e a contagem de tweets em relação a idade da conta aparecem com coeficientes mais elevados, indicando forte impacto no resultado do modelo logístico.

5.2.3. Importância das variáveis com Random Forest

O modelo de Random Forest é um método de aprendizado de máquina baseado em um conjunto de árvores de decisão. A importância das variáveis no Random Forest é calculada com base na redução de impureza em cada divisão das árvores e reflete o impacto preditivo de cada variável no modelo. Esse método é valioso porque considera interações não lineares e relações complexas entre as variáveis, que uma simples correlação linear não captura. Variáveis com alta importância no Random Forest são aquelas que mais contribuem para a precisão do modelo, indicando sua relevância na classificação final [32].

A resposta do modelo para as variáveis mais importantes foi contagem de tweets em relação a idade da conta, idade da conta, e a proporção de seguidos para seguidores destacam-se como as mais importantes no modelo.

Tabela 10. Relação das regras e os métodos utilizados para analisar suas importâncias

Métrica	**Correlação**	**Regressão Logística**	**Random Forest**
tweetCountToAccountAgeScore	0,37	0,35	0,25
accountAgeScore	0,24	0,30	0,21
followingToFollowerRatioScore	-0,33	-0,20	0,18
hashtagUsageScore	0,18	0,50	0,09
retweetToTweetRatioScore	-0,11	-0,15	0,08

Ao final da análise de importância, notou-se que, comparada com os pesos iniciais que foram dados as regras, pode-se afirmar que, regras como idade da conta e uso de hashtags foram subestimadas, sendo mais relevantes do que o inicialmente analisado, e que regras como proporção de seguidos para seguidores e proporção de retweets foram superestimadas se mostrando menos impactantes do que realmente são.

Tabela 11. Atualização do impacto e peso das regras pós otimização

Regra	**Impacto**		**Peso**	
	Antes	**Depois**	**Antes**	**Depois**
Proporção de seguidos para seguidores	Alto	Moderado	1,5	1,1
Média de tempo entre postagens	Alto	Moderado	1,5	1,1
Proporção de retweets	Alto	Moderado	1,3	1,2
Quantidade de tweets em relação à idade	Alto	Alto	1,4	1,5
Tamanho médio dos tweets	Moderado	Moderado	1,3	1,2
Menções por usuário	Moderado	Baixo	1,2	0,9
Idade da conta	Moderado	Alto	1,1	1,4
Proporção de hashtags únicas	Baixo	Alto	0,8	1,4
Similaridade entre Nome e Nome de Usuário	Baixo	Baixo	0,6	0,8

5.2.4. Seleção do *threshold*[19]

Para classificar cada conta como bot ou humano, foi necessário definir um threshold, um limite mínimo para definir o tipo da conta. Para realizar essa análise, primeiramente, foi rodada uma nova classificação utilizando os pesos atualizados para gerar novas pontuações de classificação. Com as novas pontuações, foram selecionados, baseando-se numa análise visual, quais limites entre 0,1 e 1 geraram um maior número de acurácia do modelo. Após essas análises visuais, foi identificado que os valores entre 0,5 e 0,7 demonstraram a melhor utilização da classificação. Dentro deste intervalo, foram analisadas as informações de precisão[20], recall[21], F1-score[22] e acurácia[23] para cada um dos valores para encontrar o melhor threshold para o modelo.

Tabela 12. Relação entre estatística e threshold

Estatística	Valor	Threshold
F1-score	0,612287	0,50
Precisão	0,66672	0,70
Recall	0,73555	0,50
Acurácia	0,586095	0,61

5.3. Avaliação da classificação pós otimização

Com os novos pesos definidos, foi aplicada a função de classificação novamente no dataset e feitos testes estatísticos para cada threshold relacionado anteriormente, e os seguintes resultados se destacaram.

- **Threshold 0,61:**
    - F1-score representando melhor equilíbrio entre precisão e recall para ambas as classes, com valores de 0,62 para humanos e 0,55 para bots.
    - Acurácia de 0,59, indicando uma classificação equilibrada considerando todas as classes.

[19] Limite estatístico.
[20] Indica quantas das classificações positivas feitas pelo modelo estão corretas.
[21] Indica a capacidade do modelo de encontrar todas as instâncias positivas.
[22] Média harmônica entre a precisão e o recall demonstrando equilíbrio entre as métricas
[23] Proporção de predições corretas em relação ao total de amostras.

- **Threshold 0,5:**
    - Recall de bots de 0,74, o melhor recall para bots, mas sacrifica a precisão de humanos e o equilíbrio geral.
    - Acurácia de 0,53, indicando maior viés para detectar bots.
    - F1-score é considerado bom para bots sendo de 0,61, porém prejudicial para humanos que tem 0,42.
- **Threshold 0,7:**
    - Precisão de bots de 0,67, destacando-se ao reduzir falsos positivos de bots.
    - Recall de humanos de 0,90, o mais alto, mas isso ocorre às custas do recall de bots ser 0,21.
    - Acurácia de 0,55, ligeiramente melhor do que o threshold 0,5, mas ainda longe do equilíbrio ideal.

Com isso, optou-se pelo threshold de 0,61 devido ao equilíbrio entre precisão e recall para ambas as classes, humanos e bots. Este valor apresentou o melhor F1-score geral, com 0,62 para humanos e 0,55 para bots, indicando uma boa harmonia entre as métricas e reduzindo tanto os falsos positivos quanto os falsos negativos. Além disso, a acurácia geral foi de 59%, o maior valor entre os thresholds testados, reforçando sua consistência na classificação.

Em comparação, thresholds como 0,5 favoreceram o recall de bots, mas sacrificaram a precisão e o equilíbrio geral, enquanto o threshold de 0,7 maximizou o recall de humanos, porém com um impacto negativo no recall de bots. Dessa forma, o threshold 0,61 foi selecionado como o mais adequado para atender ao objetivo de alcançar um modelo equilibrado e confiável na detecção de bots e humanos.

5.4. Análise do conjunto de testes

Após reorganizar a função de classificação, foi realizado um teste do modelo estatístico com um conjunto de dados de teste extraídos a partir do mesmo dataset que a amostra utilizada no desenvolvimento foi criada. A amostra contém 1000 contas de usuário, sendo 500 humanos e 500 bots.

Após a aplicação da função de classificação, utilizando um threshold de 0,61, foram geradas estatísticas para o resultado da classificação que revelaram que que o modelo apresenta um desempenho moderado, relativamente otimizado devido aos esforços de otimização.

Observando a matriz de confusão, notou-se que:

- **Verdadeiros Positivos (VP):** 228 bots foram corretamente classificados como bots.
- **Verdadeiros Negativos (VN):** 354 humanos foram corretamente classificados como humanos.
- **Falsos Positivos (FP):** 146 humanos foram erroneamente classificados como bots.
- **Falsos Negativos (FN):** 272 bots foram erroneamente classificados como humanos.

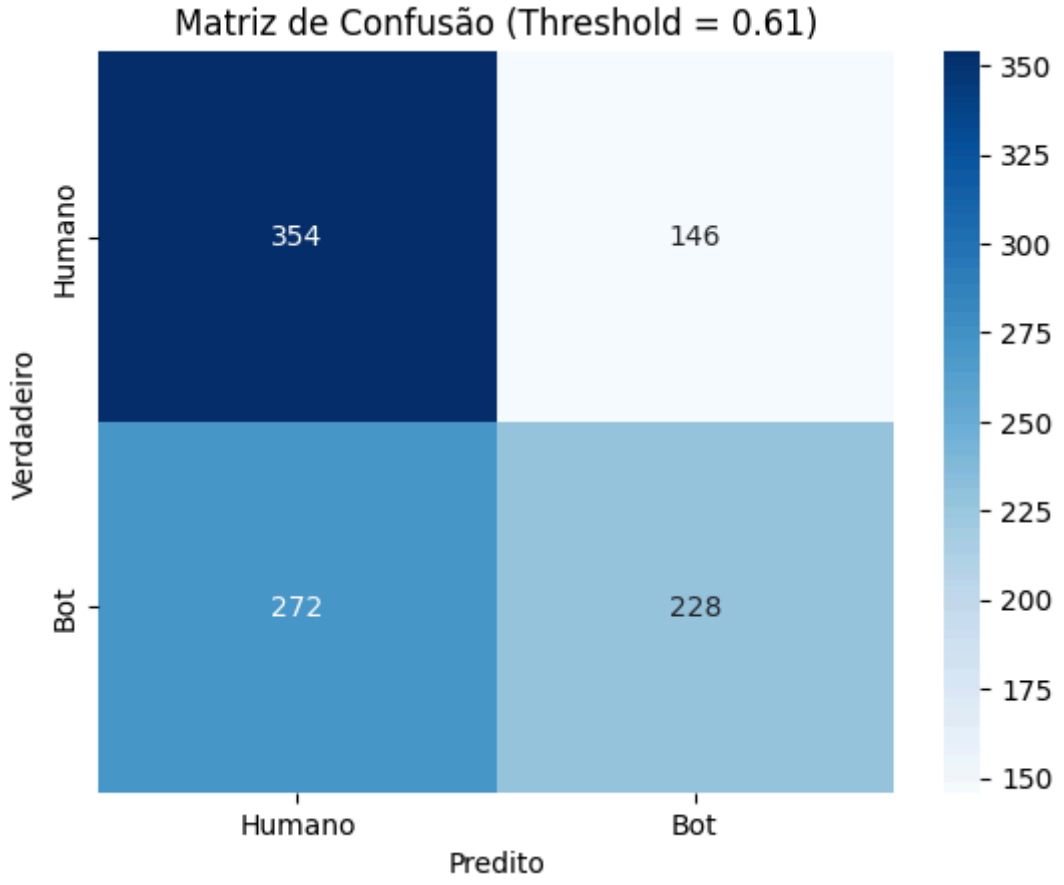


Figura 8. Matriz de confusão do conjunto de testes

Os números da matriz de confusão indicam que o modelo tem maior dificuldade em identificar bots com precisão, mas demonstra um desempenho razoável na classificação de humanos.

Nas métricas de classificação, a precisão para humanos foi de 57%, o que significa que, entre todas as contas classificadas como humanas, 57% eram, de fato, humanas. Para bots, a precisão foi um pouco maior, alcançando 61%. O recall

mostrou resultados mistos, com 71% dos humanos corretamente identificados como humanos e apenas 46% dos bots corretamente identificados como bots. O F1-Score, que equilibra precisão e recall, foi de 0,63 para humanos e 0,52 para bots, indicando um desempenho ligeiramente melhor na classificação de humanos. A acurácia geral do modelo foi de 58%.

Tabela 13. Estatísticas do conjunto de testes

	Bots	Humanos
Precisão	61%	57%
Recall	46%	71%
F1-Score	52%	63%
Acurácia	58%	

Com base nesses resultados, o threshold de 0,61 foi escolhido por seu equilíbrio entre precisão e revocação para ambas as classes, maximizando o F1-Score geral. Embora o desempenho geral do modelo ainda possa ser melhorado, esse threshold se mostrou adequado para manter um balanço entre a identificação correta de humanos e bots, o que é essencial no contexto da detecção de bots.

6. CONCLUSÃO E TRABALHOS FUTUROS

O nível de acurácia do algoritmo desenvolvido apresentou melhorias em alguns aspectos, mas ainda deixa espaço para ajustes. As análises indicaram que a reavaliação dos pesos atribuídos às regras foi uma estratégia válida, mas não suficiente para atingir níveis de desempenho altamente satisfatórios. Além disso, a inclusão de novos dados nos conjuntos de treinamento e teste mostrou-se essencial para refinar os limites superiores das métricas e recalibrar a função de classificação.

Uma abordagem que poderia ser adotada para aprimorar a interpretação das pontuações é a inclusão de uma área neutra. Scores abaixo de 0.5 indicariam bots prováveis, enquanto scores acima de 0.7 indicariam humanos prováveis, e scores entre esses valores seriam classificados como neutros. Tal adaptação pode reduzir falsos positivos e negativos, tornando a ferramenta mais confiável. Também seria fundamental incluir regras que considerem características dos perfis de usuários além das interações na timeline, como a ausência de postagens, que invalida métricas baseadas em menções, retweets e hashtags.

Considerando a limitação do modelo em classificar uma conta como humana ou não, avalia-se que foi acertada a opção de exibir ao usuário final na página da ferramenta uma escala de probabilidade ao invés de uma afirmação fixa de que aquela conta se trata deste ou daquele tipo de usuário.

Embora o objetivo principal do trabalho tenha sido desenvolver uma ferramenta que não dependesse de grandes volumes de dados ou técnicas de aprendizado de máquina, os resultados sugerem que o uso de mais dados pode ser inevitável para aprimorar a precisão das heurísticas. Assim, explorar o uso de machine learning em fases complementares do sistema pode ser uma solução prática, mesmo que isso represente uma mudança na proposta inicial.

Para trabalhos futuros, recomenda-se explorar o máximo de recursos disponíveis e incrementar a ferramenta com novas regras baseadas em comportamentos emergentes, especialmente considerando o contexto dinâmico do Twitter (agora chamado "X"), onde o comportamento dos bots e a propagação de informações falsas continuam a evoluir. A criação de uma comunidade de crowdsourcing para auxiliar na detecção de bots também poderia ser uma iniciativa valiosa, dado que usuários da rede social possuem conhecimento prático que pode complementar as análises automatizadas.

Além disso, a construção da ferramenta de forma desacoplada mantém sua viabilidade para integração com outros sistemas e redes sociais, o que é fundamental para enfrentar o problema da disseminação de informações falsas em diferentes plataformas. A detecção adaptativa e contínua é crucial, dado que as características dos bots estão em constante transformação. Em última análise, o desenvolvimento contínuo e a adaptação das ferramentas de detecção de bots são indispensáveis para mitigar os impactos das informações falsas e promover um ambiente online mais seguro e confiável.

REFERÊNCIAS

[1] POSETTI, Julie; MATTHEWS, Alice. A short guide to the history of 'fake news' and disinformation. **International Center for Journalists**, v. 7, p. 2018-07, 2018.

[2] IBGE. Diretoria de Pesquisas. **Acesso à Internet e à televisão e posse de telefone móvel celular para uso pessoal.** 2018. Pesquisa Nacional por Amostra de Domicílios Contínua 2017-2018. Disponível em: https://biblioteca.ibge.gov.br/visualizacao/livros/liv101705_informativo.pdf.

[3] RECUERO, Raquel; GRUZD, Anatoliy. Cascatas de Fake News Políticas: um estudo de caso do twitter. **Galáxia (São Paulo).** São Paulo, p. 31-47. ago. 2019. Disponível em: http://dx.doi.org/10.1590/1982-25542019239035.

[4] SHU, Kai; SILVA, Amy; WANG, Suhang; TANG, Jiliang; LIU, Huan. Fake News Detection on Social Media: A Data Mining Perspective. **Acm Sigkdd Explorations Newsletter.** New York, p. 56-66. 1 set. 2017. Disponível em: https://doi.org/10.1145/3137597.3137600.

[5] OMS. **WHO Living guideline: Drugs to prevent COVID-19**. 2021. Disponível em: https://www.who.int/publications/i/item/WHO-2019-nCoV-prophylaxes-2021-1.

[6] WU, Liang; MORSTATTER, Fred; CARLEY, Kathleen M.; LIU, Huan. Misinformation in Social Media: definition, manipulation, and detection. **Acm Sigkdd Explorations Newsletter**, v. 21, n. 2, p. 80-90, 26 nov. 2019. Association for Computing Machinery (ACM). http://dx.doi.org/10.1145/3373464.3373475.

[7] ZAROCOSTAS, John. How to fight an infodemic. **The lancet**, v. 395, n. 10225, p. 676, 2020.

[8] ZUBIAGA, Arkaitz; AKER, Ahmet; BONTCHEVA, Kalina; LIAKATA, Maria; PROCTER, Rob. Detection and Resolution of Rumours in Social Media. Acm Computing Surveys, v. 51, n. 2, p. 1-36, 2 jun. 2018. **Association for Computing Machinery (ACM).** http://dx.doi.org/10.1145/3161603.

[9] KUMAR, Srijan; SHAH, Neil. False Information on Web and Social Media: A Survey. **Association for Computing Machinery (ACM).** v. 1, n. 1, p. 1-35, abr. 2018.

[10] CONROY, Nadia K.; RUBIN, Victoria L.; CHEN, Yimin. Automatic deception detection: methods for finding fake news. **Proceedings Of The Association For Information Science And Technology**, v. 52, n. 1, p. 1-4, jan. 2015. Wiley. http://dx.doi.org/10.1002/pra2.2015.145052010082.

[11] LI, Quanzhi; ZHANG, Qiong; SI, Luo; LIU, Yingchi. Rumor Detection on Social Media: Datasets, Methods and Opportunities. **Proceedings Of The Second Workshop On Natural Language Processing For Internet Freedom: Censorship, Disinformation, And Propaganda.** Hong Kong, p. 66-75. nov. 2019.

[12] WU, Liang; LIU, Huan. Tracing Fake-News Footprints: characterizing social media messages by how they propagate. **Proceedings Of The Eleventh Acm International Conference On Web Search And Data Mining.** New York, p. 637-645. fev. 2018.

[13] BENEVENUTO, Fabrício; RIBEIRO, Filipe; ARAÚJO, Matheus. Métodos para análise de sentimentos em mídias sociais. **Sociedade Brasileira de Computação**, 2015.

[14] EKMAN, Paul. An argument for basic emotions. **Cognition & emotion**, v. 6, n. 3-4, p. 169-200, 1992.

[15] PENNEBAKER, James W.; BOYD, Ryan L.; JORDAN, Kayla; BLACKBURN, Kate. The development and psychometric properties of LIWC2015. Austin, TX. **University of Texas at Austin.** 2015.

[16] ABOKHODAIR, Norah; YOO, Daisy; MCDONALD, David W.. Dissecting a Social Botnet. **Proceedings Of The 18Th Acm Conference On Computer Supported Cooperative Work & Social Computing**, p. 839-851, fev. 2015. ACM. http://dx.doi.org/10.1145/2675133.2675208.

[17] NOBRE, Gabriel P.; ALMEIDA, Jussara M.; FERREIRA, Carlos H. G.. Caracterização de bots no Twitter durante as Eleições Presidenciais no Brasil em 2018. **Anais do Brazilian Workshop On Social Network Analysis And Mining (Brasnam)**, p. 107-118, jul. 2019. Sociedade Brasileira de Computação - SBC. http://dx.doi.org/10.5753/brasnam.2019.6553.

[18] SAKURAI, Guilherme Yukio. **Processamento De Linguagem Natural**: Detecção de Fake News. 2019. 37 f. TCC (Graduação) - Curso de Ciência da Computação, Centro de Ciências Exatas, Universidade Estadual de Londrina, Londrina, 2019.

[19] BALAGE FILHO, Pedro P.; PARDO, Thiago A. S.; ALUÍSIO, Sandra M.. An Evaluation of the Brazilian Portuguese LIWC Dictionary for Sentiment Analysis Pedro. **Proceedings Of The 9Th Brazilian Symposium In Information And Human Language Technology.** Fortaleza, p. 215-219. out. 2013.

[20] BRADLEY, Margaret M.; LANG, Peter J. **Affective norms for English words (ANEW): Instruction manual and affective ratings**. Technical report C-1, The Center for Research in Psychophysiology, University of Florida, 1999.

[21] BACCIANELLA, Stefano; ESULI, Andrea; SEBASTIANI, Fabrizio. SENTIWORDNET 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. **Lrec.** p. 2200-2204. 2010.

[22] NIELSEN, Finn Arup. A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. **Proceedings Of The Eswc2011 Workshop On 'Making Sense Of Microposts'.** p. 93-98. mar. 2011.

[23] SILVA, Mário J.; CARVALHO, Paula; SARMENTO, Luís. Building a Sentiment Lexicon for Social Judgement Mining. **Computational Processing Of The Portuguese Language, 10Th International Conference, Propor 2012.** Coimbra, p. 218-228. abr. 2012.

[24] SOUZA, Marlo; VIEIRA, Renata; BUSETTI, Débora; CHISHMAN, Rove; ALVES, Isa Mara. Construction of a Portuguese Opinion Lexicon from multiple resources. **Proceedings Of The 8Th Brazilian Symposium In Information And Human Language Technology.** Cuiabá, p. 59-66. out. 2011.

[25] DAVIS, Clayton Allen; VAROL, Onur; FERRARA, Emilio; FLAMMINI, Alessandro; MENCZER, Filippo. BotOrNot. **Proceedings Of The 25Th International Conference Companion On World Wide Web - Www '16 Companion**, [S.L.], v. 59, n. 7, p. 94-104, abr. 2016. ACM Press. http://dx.doi.org/10.1145/2872518.2889302.

[26] FERRARA, Emilio; VAROL, Onur; DAVIS, Clayton; MENZIO, Filippo; FLAMMINI, Alessandro. **Online Human-Bot Interactions: Detection, Estimation, and Characterization**. arXiv preprint arXiv:1703.03107, 2017. Disponível em: https://arxiv.org/pdf/1703.03107.

[27] SATARDAKAR, Rakesh; CHAUDHARI, Nitin. **A one-class classification approach for bot detection on Twitter. Journal of Information Security and Applications**, v. 54, p. 102538, 2020. Disponível em: https://www.sciencedirect.com/science/article/pii/S0167404820300031.

[28] ZHENG, Xianghan; ZENG, Zhipeng; CHEN, Zheyi; YU, Yuanlong; RONG, Chunming. **Detecting spammers on social networks.** Neurocomputing, v. 159, p. 27-34, 2015. Disponível em: https://www.sciencedirect.com/science/article/pii/S0925231215002106.

[29] WANG, Yuxiang; LI, Yu; GUO, Hao; CAO, Juan; DONG, Yuxiao. **TwiBot-22: Towards Graph-Based Twitter Bot Detection.** arXiv preprint arXiv:2206.04564, 2022. Disponível em: https://arxiv.org/pdf/2206.04564.

[30] MONTGOMERY, D. C.; PECK, E. A.; VINING, G. G. **Introduction to Linear Regression Analysis**. 5. ed. Hoboken, NJ: Wiley, 2012.

[31] HOSMER, D. W.; LEMESHOW, S.; STURDIVANT, R. X. **Applied Logistic Regression**. 3. ed. Hoboken, NJ: Wiley, 2013.

[32] BREIMAN, L. Random Forests. **Machine Learning**, v. 45, n. 1, p. 5-32, 2001.

[33] TWITTER cobrará por bots e outras funções de API; veja quem será afetado. **Forbes Brasil**, 08 fev. 2023. Disponível em:

https://forbes.com.br/forbes-tech/2023/02/twitter-cobrara-por-bots-e-outras-funcoes-de-api-veja-quem-sera-afetado/. Acesso em: 18 dez. 2024.

[34] X diz que encerrará operações no Brasil devido a ordens judiciais para remoção de conteúdo. **Reuters**, 17 ago. 2024. Disponível em: https://www.reuters.com/technology/x-close-operations-brazil-effective-immediately-2024-08-17. Acesso em: 18 dez. 2024.

APÊNDICES

Apêndice A - Lista de repositórios no GitHub

Abaixo estão os repositórios do projeto hospedados no GitHub, organizados de acordo com suas funções:

1. **Repositórios de Serviços**

1.1. twitter-integration

- **Descrição:** Serviço responsável pela coleta e armazenamento de dados da API do Twitter.
- **Link:** https://github.com/migueh-ufsc/twitter-integration

1.2. detection-rules

- **Descrição:** Serviço de processamento e análise heurística dos dados coletados.
- **Link:** https://github.com/migueh-ufsc/detection-rules

2. **Repositório Backend for Frontend (BFF)**
   - **Descrição:** Camada intermediária para facilitar a comunicação entre os serviços e o frontend.
   - **Link:** https://github.com/migueh-ufsc/bff-detector

3. **Repositório de Frontend**
   - **Descrição:** Interface de usuário desenvolvida para exibição dos resultados da análise de bots.
   - **Link:** https://github.com/migueh-ufsc/frontend

4. **Repositório de Scripts**

4.1. Scripts de análise de dados

- **Descrição:** Scripts utilizados no Google Colab para a análise de dados, testes das heurísticas e validação dos resultados.
- **Link:** https://github.com/migueh-ufsc/colab-scripts

4.2. Scripts de criação e atualização

- **Descrição:** Scripts utilizados para fazer chamadas para os serviços twitter-integration e detection-rules para adicionar os dados gerados na análise das bases de dados.
- **Link:** https://github.com/migueh-ufsc/scripts

Apêndice B - Acesso a página da ferramenta

Link para acesso: https://migueh-ufsc.github.io/frontend/

Migueh: uma ferramenta para detecção de social bots no X/Twitter

Indiara C. Menegat

Universidade Federal de Santa Catarina (UFSC), Departamento de Sistemas de Informação

indiara.cm@grad.ufsc.br

Resumo: *Este trabalho desenvolveu uma ferramenta web para detectar bots no X/Twitter, utilizando análise comportamental e metadados para impedir a disseminação de informações falsas. Regras heurísticas como proporção de seguidos para seguidores e atividade temporal foram aplicadas. Resultados indicam necessidade de ajustes nas regras e integração de novos dados. A arquitetura desacoplada permite adaptação para outras redes sociais, ampliando o combate à desinformação.*

Abstract: *This study developed a web tool to detect bots on X/Twitter, leveraging behavioral analysis and metadata to prevent the spread of false information. Heuristic rules, such as the ratio of followers to following and temporal activity, were applied. Results suggest the need for rule adjustments and data integration. The decoupled architecture enables adaptation to other social networks, enhancing the fight against misinformation.*

1. INTRODUÇÃO

O avanço das redes sociais na última década transformou a forma como informações são disseminadas, facilitando a comunicação global, mas também ampliando a propagação de desinformação. Eventos como as eleições americanas de 2016 e brasileiras de 2018 evidenciaram o impacto negativo da desinformação, impulsionada frequentemente por contas automatizadas, conhecidas como bots. Esses agentes desempenham um papel central na criação de câmaras de eco e na amplificação de conteúdos falsos ou maliciosos, ameaçando a integridade das interações online.

Diante desse cenário, o Twitter (agora denominado X) tornou-se um objeto de estudo relevante devido à sua popularidade e dinâmica de interação. Contudo, mudanças recentes na plataforma, incluindo restrições ao acesso da API, complicaram o desenvolvimento de ferramentas para análise e monitoramento, afetando iniciativas acadêmicas e projetos.

Este trabalho propõe a ferramenta Migueh, desenvolvida para identificar bots no X/Twitter por meio de análise comportamental e metadados. A ferramenta aplica regras heurísticas baseadas em métricas previamente estudadas, como proporção de seguidos para seguidores, frequência de retweets e intervalos entre postagens. Os resultados obtidos indicam que o Migueh pode contribuir significativamente para a detecção de bots, reduzindo o impacto da desinformação e promovendo interações mais seguras na internet.

2. TRABALHOS RELACIONADOS

2.1. Botometer

O Botometer, desenvolvido pelo Observatório de Mídias Sociais da Universidade de Indiana, utiliza aprendizado de máquina para analisar mais de mil características organizadas em categorias como rede, temporal, conteúdo e sentimento. Ele retorna uma pontuação de 0 a 5, indicando a probabilidade de uma conta ser automatizada. Sua API foi amplamente utilizada para integração com outras ferramentas e estudos acadêmicos. No entanto, as recentes mudanças no acesso à API do X/Twitter inviabilizaram a continuidade da ferramenta.

2.2. Pegabot

O Pegabot, criado pelo Instituto de Tecnologia e Sociedade do Rio de Janeiro, utiliza critérios como perfil do usuário, análise de rede e sentimentos das publicações para determinar a probabilidade de automação. A ferramenta avalia as últimas 100 publicações da conta analisada, retornando uma pontuação de 0% a 100%. Diferentemente do Botometer, o Pegabot não oferece uma API, sendo acessado exclusivamente por um portal público. As alterações no acesso à API do X/Twitter também impactaram o funcionamento desta ferramenta, levando à sua desativação.

2.3. BotSentinel

O BotSentinel é uma ferramenta gratuita projetada para rastrear e classificar contas automatizadas. Ele utiliza aprendizado de máquina e inteligência artificial, retornando uma pontuação de 0% a 100%, onde pontuações mais altas indicam maior probabilidade de comportamento malicioso. A ferramenta não revela as características específicas analisadas pelos algoritmos e oferece acesso por meio de sua plataforma ou extensões de navegador, sem suporte para APIs. Apesar de suas limitações, o BotSentinel continua sendo uma referência na análise de contas automatizadas.

2.4. Limitações e Contribuições do Migueh

As abordagens das ferramentas existentes baseiam-se em técnicas como análise de rede, processamento de linguagem natural e modelagem de propagação. Apesar de demonstrarem eficácia, enfrentam limitações, como dependência de APIs externas e dificuldade em lidar com grandes volumes de dados. A ferramenta proposta neste trabalho, o Migueh, busca superar essas limitações ao adotar uma arquitetura desacoplada e funções heurísticas adaptáveis, permitindo maior flexibilidade e escalabilidade.

3. METODOLOGIA

A ferramenta Migueh foi desenvolvida para detectar bots no X/Twitter utilizando análise comportamental e metadados. O processo seguiu etapas bem definidas para coleta, preparação e análise dos dados, definição das regras heurísticas e implementação da arquitetura da ferramenta.

3.1. Coleta e Preparação de Dados

Devido às restrições impostas pela API do Twitter e à necessidade de dados já anotados para análise, este trabalho utilizou o dataset TwiBot-22 como base. O TwiBot-22 é um conjunto de dados amplamente reconhecido por sua qualidade e cobertura, contendo contas rotuladas como bots ou humanos, além de informações

detalhadas sobre seus comportamentos. Após a seleção do dataset, os dados passaram por processos de limpeza e normalização, incluindo o tratamento de valores ausentes, preenchendo-os com a média das colunas, e a aplicação de transformações logarítmicas para ajustar a distribuição de algumas métricas

3.2. Definição de Regras Heurísticas

Foram criadas nove regras heurísticas baseadas em características comportamentais e de perfil.

Tabela 1. Regras heurísticas e suas hipóteses.

Regras	Hipótese
Proporção de seguidos para seguidores	Bots seguem mais contas do que são seguidos, gerando uma proporção elevada.
Proporção de retweets	Bots compartilham conteúdos de outras contas em grande volume.
Menções únicas	Bots mencionam diversos usuários em postagens isoladas para aumentar alcance.
Média do tamanho dos tweets	Bots postam mensagens curtas e padronizadas.
Idade da conta	Bots geralmente estão associados a contas mais recentes.
Utilização de hashtags únicas	Bots usam hashtags incomuns para alcançar nichos específicos.
Quantidade de tweets vs. idade da conta	Bots publicam frequentemente, mantendo uma alta frequência desde a criação da conta.
Média de tempo entre tweets	Bots postam em intervalos curtos, enquanto humanos têm uma periodicidade mais espaçada.
Similaridade entre nome e *username*	Bots utilizam nomes que refletem diretamente o username, evitando personalizações.

Os limites para cada métrica foram estabelecidos utilizando métodos estatísticos, como o cálculo do percentil, para minimizar a influência de outliers.

3.3. Implementação da Ferramenta

A ferramenta foi implementada com uma arquitetura modular, utilizando Node.js e TypeScript, o que permitiu desacoplar a lógica de detecção das interfaces de entrada e saída. Além disso, foram integrados serviços para visualização dos resultados e geração de relatórios. A arquitetura foi projetada para ser adaptável, permitindo expansão para outras redes sociais no futuro.

3.4. Validação

O sistema foi validado em um conjunto de dados de teste. Pesos foram atribuídos às regras baseando-se na análise exploratória das regras heurísticas e um

limite inicial foi estabelecido para definir a natureza de uma conta. A avaliação utilizou métricas como AUC e precisão para medir a eficácia das heurísticas implementadas.

4. RESULTADOS

4.1. Desempenho das Regras Heurísticas

O desempenho de cada uma das nove regras heurísticas foi avaliado individualmente para verificar sua contribuição na detecção de bots. A tabela abaixo apresenta as regras utilizadas, uma breve descrição e suas respectivas contribuições observadas nos testes.

Tabela 2. Regras heurísticas e suas contribuições observadas.

Regras	Contribuição
Proporção de seguidos para seguidores	Identificou bots com alta precisão devido ao padrão anômalo de comportamento.
Proporção de retweets	Indicador moderado, mas útil em conjunto com outras métricas.
Menções únicas	Contribuição abaixo de moderada, mas útil em cenários de spam.
Média do tamanho dos tweets	Desempenho moderado, principalmente em contas com padrões de mensagem fixos.
Idade da conta	Boa contribuição em contas mais novas, com impacto limitado em contas antigas.
Utilização de hashtags únicas	Identificou bots em contextos específicos, como campanhas de spam.
Quantidade de tweets vs. idade da conta	Um dos melhores indicadores, com alta precisão na distinção de bots e humanos.
Média de tempo entre tweets	Forte correlação com atividades bot, evidenciando padrões automatizados.
Similaridade entre nome e *username*	Contribuição baixa.

4.2. Avaliação Geral da Classificação

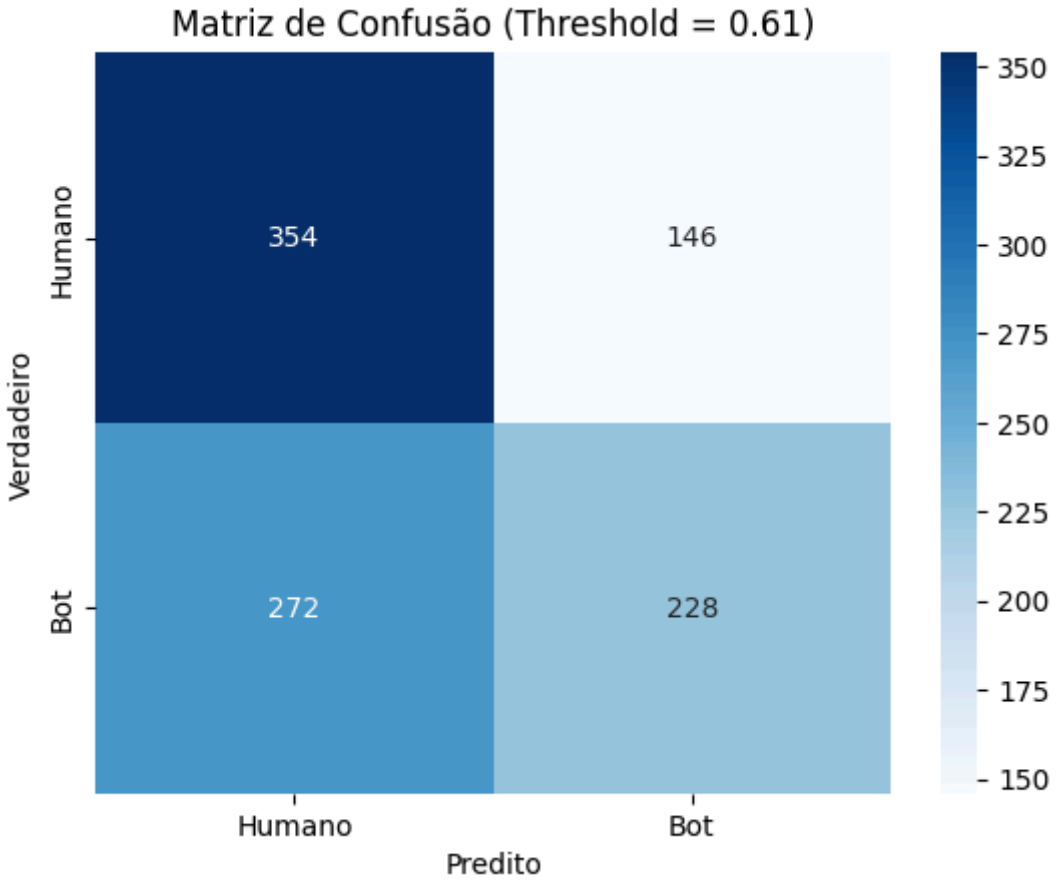
A avaliação geral da classificação foi conduzida utilizando um conjunto de testes composto por 500 bots e 500 humanos, com o objetivo de validar a eficácia das regras heurísticas e da função de classificação. Inicialmente, os pesos das regras foram otimizados com base no impacto observado em análises preliminares, atribuindo maior peso a métricas mais relevantes, como a proporção de seguidos para seguidores e a média de tempo entre postagens. Além disso, os limites superiores das métricas foram estabelecidos utilizando o 95º percentil, minimizando a influência de outliers.

A função de classificação considerou a normalização das métricas no intervalo de 0 a 1, com inversão de algumas regras, como idade da conta e proporção de tweets em relação à idade, para melhor refletir os padrões de comportamento humano. Os

scores resultantes da classificação indicaram uma média de 6,41 para bots e 5,70 para humanos, evidenciando uma distinção moderada entre os dois grupos.

A avaliação utilizou a Curva ROC para medir a capacidade de distinção entre bots e humanos, alcançando uma área sob a curva (AUC) de 0,5922, ligeiramente superior a um classificador aleatório. Foram testados diferentes thresholds para melhorar a precisão da classificação. O threshold de 0,61 foi selecionado por apresentar o melhor equilíbrio entre precisão, recall e F1-score, com uma acurácia geral de 59%.

Figura 1. Matriz de confusão da avaliação de testes com threshold de 0,61.



Os valores revelam que, embora a ferramenta consiga identificar padrões básicos entre bots e humanos, ainda há uma sobreposição significativa entre os scores, resultando em falsos positivos e negativos. Esses resultados indicam a necessidade de ajustes futuros, como o refinamento das heurísticas ou a inclusão de novos dados para melhorar a generalização e precisão da função de classificação.

5. CONCLUSÃO E TRABALHOS FUTUROS

Este trabalho apresentou a ferramenta Migueh, desenvolvida para detectar bots no X/Twitter por meio de análise comportamental e metadados. Os resultados obtidos demonstram que a abordagem heurística proposta é capaz de identificar padrões básicos de comportamento automatizado, com métricas como a proporção de seguidos para seguidores e a média de tempo entre postagens destacando-se como indicadores eficazes. Embora a AUC de 0,5922 e a acurácia geral de 59% evidenciem espaço para melhorias, a ferramenta mostrou-se funcional e escalável para outros cenários.

As contribuições deste trabalho incluem a definição e validação de nove regras heurísticas fundamentadas em comportamentos característicos de bots, proporcionando uma abordagem acessível e eficaz para identificação de contas automatizadas. Além disso, a arquitetura modular desenvolvida permite fácil adaptação da ferramenta para diferentes plataformas de redes sociais, garantindo flexibilidade e escalabilidade. Por fim, a proposta de uma metodologia baseada em heurísticas oferece uma alternativa viável ao aprendizado de máquina, especialmente em contextos com limitações de recursos, facilitando a aplicação em cenários diversos.

Para trabalhos futuros, planeja-se incorporar novas métricas que considerem padrões emergentes de bots, como detecção de clusters e análise de interações em tempo real. Além disso, a integração de métodos de aprendizado de máquina poderá complementar a abordagem heurística, aprimorando a precisão e a capacidade de generalização da ferramenta. Por fim, a expansão da ferramenta para outras plataformas de redes sociais poderá ampliar seu impacto no combate à desinformação.

6. REFERÊNCIAS

POSETTI, Julie; MATTHEWS, Alice. A short guide to the history of 'fake news' and disinformation. **International Center for Journalists**, v. 7, p. 2018-07, 2018.

IBGE. Diretoria de Pesquisas. **Acesso à Internet e à televisão e posse de telefone móvel celular para uso pessoal.** 2018. Pesquisa Nacional por Amostra de Domicílios Contínua 2017-2018. Disponível em: https://biblioteca.ibge.gov.br/visualizacao/livros/liv101705_informativo.pdf.

RECUERO, Raquel; GRUZD, Anatoliy. Cascatas de Fake News Políticas: um estudo de caso do twitter. **Galáxia (São Paulo).** São Paulo, p. 31-47. ago. 2019. Disponível em: http://dx.doi.org/10.1590/1982-25542019239035.

SHU, Kai; SILVA, Amy; WANG, Suhang; TANG, Jiliang; LIU, Huan. Fake News Detection on Social Media: A Data Mining Perspective. **Acm Sigkdd Explorations Newsletter.** New York, p. 56-66. 1 set. 2017. Disponível em: https://doi.org/10.1145/3137597.3137600.

OMS. **WHO Living guideline: Drugs to prevent COVID-19**. 2021. Disponível em: https://www.who.int/publications/i/item/WHO-2019-nCoV-prophylaxes-2021-1.

WU, Liang; MORSTATTER, Fred; CARLEY, Kathleen M.; LIU, Huan. Misinformation in Social Media: definition, manipulation, and detection. **Acm Sigkdd Explorations Newsletter**, v. 21, n. 2, p. 80-90, 26 nov. 2019. Association for Computing Machinery (ACM). http://dx.doi.org/10.1145/3373464.3373475.

ZAROCOSTAS, John. How to fight an infodemic. **The lancet**, v. 395, n. 10225, p. 676, 2020.

ZUBIAGA, Arkaitz; AKER, Ahmet; BONTCHEVA, Kalina; LIAKATA, Maria; PROCTER, Rob. Detection and Resolution of Rumours in Social Media. Acm Computing Surveys, v. 51, n. 2, p. 1-36, 2 jun. 2018. **Association for Computing Machinery (ACM).** http://dx.doi.org/10.1145/3161603.

KUMAR, Srijan; SHAH, Neil. False Information on Web and Social Media: A Survey. **Association for Computing Machinery (ACM).** v. 1, n. 1, p. 1-35, abr. 2018.

CONROY, Nadia K.; RUBIN, Victoria L.; CHEN, Yimin. Automatic deception detection: methods for finding fake news. **Proceedings Of The Association For Information Science And Technology**, v. 52, n. 1, p. 1-4, jan. 2015. Wiley. http://dx.doi.org/10.1002/pra2.2015.145052010082.

LI, Quanzhi; ZHANG, Qiong; SI, Luo; LIU, Yingchi. Rumor Detection on Social Media: Datasets, Methods and Opportunities. **Proceedings Of The Second Workshop On Natural Language Processing For Internet Freedom: Censorship, Disinformation, And Propaganda.** Hong Kong, p. 66-75. nov. 2019.

WU, Liang; LIU, Huan. Tracing Fake-News Footprints: characterizing social media messages by how they propagate. **Proceedings Of The Eleventh Acm International Conference On Web Search And Data Mining.** New York, p. 637-645. fev. 2018.

BENEVENUTO, Fabrício; RIBEIRO, Filipe; ARAÚJO, Matheus. Métodos para análise de sentimentos em mídias sociais. **Sociedade Brasileira de Computação**, 2015.

EKMAN, Paul. An argument for basic emotions. **Cognition & emotion**, v. 6, n. 3-4, p. 169-200, 1992.

PENNEBAKER, James W.; BOYD, Ryan L.; JORDAN, Kayla; BLACKBURN, Kate. The development and psychometric properties of LIWC2015. Austin, TX. **University of Texas at Austin.** 2015.

ABOKHODAIR, Norah; YOO, Daisy; MCDONALD, David W.. Dissecting a Social Botnet. **Proceedings Of The 18Th Acm Conference On Computer Supported Cooperative Work & Social Computing**, p. 839-851, fev. 2015. ACM. http://dx.doi.org/10.1145/2675133.2675208.

NOBRE, Gabriel P.; ALMEIDA, Jussara M.; FERREIRA, Carlos H. G.. Caracterização de bots no Twitter durante as Eleições Presidenciais no Brasil em 2018. **Anais do Brazilian Workshop On Social Network Analysis And Mining (Brasnam)**, p. 107-118, jul. 2019. Sociedade Brasileira de Computação - SBC. http://dx.doi.org/10.5753/brasnam.2019.6553.

SAKURAI, Guilherme Yukio. **Processamento De Linguagem Natural**: Detecção de Fake News. 2019. 37 f. TCC (Graduação) - Curso de Ciência da Computação, Centro de Ciências Exatas, Universidade Estadual de Londrina, Londrina, 2019.

BALAGE FILHO, Pedro P.; PARDO, Thiago A. S.; ALUÍSIO, Sandra M.. An Evaluation of the Brazilian Portuguese LIWC Dictionary for Sentiment Analysis Pedro. **Proceedings Of The 9Th Brazilian Symposium In Information And Human Language Technology.** Fortaleza, p. 215-219. out. 2013.

BRADLEY, Margaret M.; LANG, Peter J. **Affective norms for English words (ANEW): Instruction manual and affective ratings**. Technical report C-1, The Center for Research in Psychophysiology, University of Florida, 1999.

BACCIANELLA, Stefano; ESULI, Andrea; SEBASTIANI, Fabrizio. SENTIWORDNET 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. **Lrec.** p. 2200-2204. 2010.

NIELSEN, Finn Arup. A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. **Proceedings Of The Eswc2011 Workshop On 'Making Sense Of Microposts'.** p. 93-98. mar. 2011.

SILVA, Mário J.; CARVALHO, Paula; SARMENTO, Luís. Building a Sentiment Lexicon for Social Judgement Mining. **Computational Processing Of The Portuguese Language, 10Th International Conference, Propor 2012.** Coimbra, p. 218-228. abr. 2012.

SOUZA, Marlo; VIEIRA, Renata; BUSETTI, Débora; CHISHMAN, Rove; ALVES, Isa Mara. Construction of a Portuguese Opinion Lexicon from multiple resources. **Proceedings Of The 8Th Brazilian Symposium In Information And Human Language Technology.** Cuiabá, p. 59-66. out. 2011.

DAVIS, Clayton Allen; VAROL, Onur; FERRARA, Emilio; FLAMMINI, Alessandro; MENCZER, Filippo. BotOrNot. **Proceedings Of The 25Th International Conference Companion On World Wide Web - Www '16 Companion**, [S.L.], v. 59, n. 7, p. 94-104, abr. 2016. ACM Press. http://dx.doi.org/10.1145/2872518.2889302.

FERRARA, Emilio; VAROL, Onur; DAVIS, Clayton; MENZIO, Filippo; FLAMMINI, Alessandro. **Online Human-Bot Interactions: Detection, Estimation, and Characterization**. arXiv preprint arXiv:1703.03107, 2017. Disponível em: https://arxiv.org/pdf/1703.03107.

SATARDAKAR, Rakesh; CHAUDHARI, Nitin. **A one-class classification approach for bot detection on Twitter. Journal of Information Security and Applications**, v. 54, p. 102538, 2020. Disponível em: https://www.sciencedirect.com/science/article/pii/S0167404820300031.

ZHENG, Xianghan; ZENG, Zhipeng; CHEN, Zheyi; YU, Yuanlong; RONG, Chunming. **Detecting spammers on social networks.** Neurocomputing, v. 159, p. 27-34, 2015. Disponível em: https://www.sciencedirect.com/science/article/pii/S0925231215002106.

WANG, Yuxiang; LI, Yu; GUO, Hao; CAO, Juan; DONG, Yuxiao. **TwiBot-22: Towards Graph-Based Twitter Bot Detection.** arXiv preprint arXiv:2206.04564, 2022. Disponível em: https://arxiv.org/pdf/2206.04564.

MONTGOMERY, D. C.; PECK, E. A.; VINING, G. G. **Introduction to Linear Regression Analysis**. 5. ed. Hoboken, NJ: Wiley, 2012.

HOSMER, D. W.; LEMESHOW, S.; STURDIVANT, R. X. **Applied Logistic Regression**. 3. ed. Hoboken, NJ: Wiley, 2013.

BREIMAN, L. Random Forests. **Machine Learning**, v. 45, n. 1, p. 5-32, 2001.

TWITTER cobrará por bots e outras funções de API; veja quem será afetado. **Forbes Brasil**, 08 fev. 2023. Disponível em: https://forbes.com.br/forbes-tech/2023/02/twitter-cobrara-por-bots-e-outras-funcoes-de-api-veja-quem-sera-afetado/. Acesso em: 18 dez. 2024.

X diz que encerrará operações no Brasil devido a ordens judiciais para remoção de conteúdo. **Reuters**, 17 ago. 2024. Disponível em:

https://www.reuters.com/technology/x-close-operations-brazil-effective-immediately-2024-08-17. Acesso em: 18 dez. 2024.