

INTOLERÂNCIA EM REDE: VIGILÂNCIA COMPUTACIONAL DO DISCURSO DE ÓDIO VIA API REST

Rodrigo Oliveira

Universidade Federal Rural da Amazônia (UFRA)

rodrigop.pessoal@gmail.com

Diego Rodriguez

Universidade Federal Rural da Amazônia (UFRA)

calidevpro@gmail.com

Resumo

As redes sociais se tornaram um ambiente de grande reprodução de desigualdades, incluindo a disseminação de discursos de ódio que afetam a cidadania digital e a dignidade de grupos vulnerabilizados. Este trabalho investigou a aplicação de tecnologias no enfrentamento desse problema, implementando uma Interface de Programação de Aplicativos (API) baseada na arquitetura REST, por meio da plataforma NodeJS, para acessar o ambiente de pesquisas da rede social X (antigo Twitter) e capturar comentários públicos com potencial conteúdo ofensivo, comparando-os a um dataset de palavras aviltantes. Os resultados apontam a viabilidade de identificar comentários de ódio, ainda que com limitações na leitura completa dos dados disponibilizados pela plataforma, o que evidencia restrições de acesso que impactam o desenvolvimento tecnológico para fins de políticas públicas. Assim, este estudo revela como a tecnologia pode tanto habilitar ferramentas de monitoramento e intervenção no combate ao discurso de ódio, quanto reproduzir desigualdades ao restringir o acesso a informações, demonstrando a necessidade de políticas públicas que regulem plataformas digitais e ampliem a justiça social no ambiente virtual.

Palavras-Chave: Discurso de ódio. X. API REST.

Objetivo de Desenvolvimento Sustentável (ODS): 16 - Paz, Justiça e Instituições Eficazes

1 INTRODUÇÃO

O tema em questão "Análise de comentários aviltantes com a API do X" se encontra inserido em uma das principais áreas da Tecnologia da Informação, chamada de Interação e Sistemas Multimídia, que são programas e aplicações em que a comunicação entre homem e computador ocorre através de múltiplos meios de representação de informação (FLUCKIGER, 1995).

Este estudo se desenvolve no contexto das eleições presidenciais brasileiras de 2022, período caracterizado por intenso debate político e aumento significativo de discursos polarizados nas redes sociais. Durante este período eleitoral, observou-se uma escalada na disseminação de conteúdos ofensivos e discriminatórios, tornando ainda mais relevante a investigação de ferramentas tecnológicas capazes de identificar e monitorar tais manifestações.

As mídias sociais proporcionam para seus usuários um ambiente interativo, no qual, a possibilidade de expressar na íntegra o que se pensa, em muitos casos, resulta em divergências expressas em comentários aviltantes. Pode-se conceituar aviltante como tudo o que afeta negativamente a honra ou a dignidade de alguém (DICIO, 2022).

O comentário aviltante rompe com os direitos e garantias fundamentais, de ser inviolável a intimidade, a vida privada, a honra e a imagem das pessoas, assegurado a elas o direito a indenização pelo dano material ou moral decorrente de sua violação, conferido no seu inciso X, do Art. 5º, Dos Direitos e Garantias Fundamentais (BRASIL, 1988).

Os comentários ao afetar negativamente outro alguém, também, configura crime contra honra - calúnia, difamação e injúria, com apenações que podem variar entre um mês a três anos de detenção, sujeita, essas, acréscimos devido a agravantes, conforme apresenta, os artigos 138 a 141, Dos Crimes contra a Honra, do Decreto-Lei nº 2.848/1940.

As principais redes sociais apresentam “diretrizes da comunidade”, documentos que estabelecem as regras e comportamentos aceitáveis, assim como os inaceitáveis, por parte de seus usuários, acrescentando aos termos de serviço e políticas de privacidade de sua rede, como exemplo o antigo Twitter agora X (2022, n. p.) - no qual serão concentradas as análises e dadificações do presente trabalho, que declara: “Nossas regras foram criadas para garantir que todas as pessoas possam participar do diálogo público com liberdade e segurança” e que sejam inclusivas e autênticas, esforçam-se para minimizar o conteúdo tóxico e ilegal e dando ferramentas para controlar as interações (X, 2022, n. p.).

Uma das ferramentas que o X disponibiliza para o controle dessas interações é a plataforma do desenvolvedor que permite o primeiro contato com a API do X, voltada para o desenvolvimento de inovações para diversos casos de uso, como desenvolver soluções para empresas, ao público, para pesquisas acadêmicas e de mercado, educação, para ações sociais e outros diversos casos.

Tendo em vista as explanações, perfaz-se pertinente um estudo analítico de interações aviltantes no X através de uma api rest, para verificar até que ponto a API do X permite examinar comentários aviltantes por meio de aplicações terceiras em seu ambiente sandbox que afirma ser indicado para criar uma prova de conceito, explorar, avaliar recursos e funcionalidades instantaneamente em escala limitada (X, 2022, n. p.).

O objetivo principal deste trabalho é verificar até que ponto a API do X permite examinar comentários aviltantes por meio de aplicações terceiras em seu ambiente sandbox. Para isso, adota-se como objetivos específicos: Analisar os principais recursos e funcionalidades da API do X; criar e descrever uma aplicação com base nos recursos oferecidos pela API do X; Identificar a eficácia da aplicação em capturar tweets aviltantes.

É notável que as interações entre os usuários contidas em redes sociais nem sempre possuem o objetivo de debater assuntos de forma respeitosa. O comportamento negativo transparece em plataformas como o X, especialmente durante períodos eleitorais, quando a polarização política intensifica a propagação de discursos intolerantes.

Observa-se um aumento na quantidade de comentários carregados de preconceito, misoginia e racismo, ao equiparar os dados de denúncias entre 2019 a 2021 em que percebeu-se um crescimento de 13.405 novas denúncias de neonazismo, um acréscimo de 93% em relação ao antepenúltimo ano que registrou 1.071 casos (CENTRAL NACIONAL DE DENÚNCIAS DE CRIMES CIBERNÉTICOS, 2021).

Conforme os dados anteriores, ao fazer um recorte do X, percebe-se a crescente de casos, de 3.788 novas denúncias de algum tipo de caso de discurso de ódio, entre 2019 e 2021, com acréscimo de 51,62% em relação ao antepenúltimo ano que registrou 3.551 casos (CENTRAL NACIONAL DE DENÚNCIAS DE CRIMES CIBERNÉTICOS, 2021).

Outro fator que impulsionou a necessidade da pesquisa tem relação com a dissociação quanto à abordagem da liberdade de expressão que possui abrangências diferentes em diversos países. No X esse conceito está ligado de forma intrínseca às leis dos EUA por ser o local sede da rede social.

Entretanto, no Brasil a legislação distingue a delimitação do que é considerado liberdade de expressão (incisos IV e X, do Art. 5º da Constituição, 1988), do conceito de discurso de ódio (parágrafos 1 e 2 do Art. 20º da Constituição, 1988). Nesse contexto há a necessidade de mais ferramentas que visem proteger os usuários para que seja cumprida às leis presentes no contexto brasileiro.

Tendo em vista esse contexto, propõe-se, na elaboração deste trabalho, provocar a discussão em cima da questão: até que ponto a API do X, no pacote sandbox, permite examinar comentários aviltantes para a prova de conceito de possíveis aplicações que visem mitigar o aumento de casos de discurso de ódio, tendo em vista a sua escala limitada.

2 REVISÃO DE LITERATURA

2.1 INTERAÇÃO E SISTEMAS MULTIMÍDIA

Multimídia são programas e sistemas em que a comunicação entre homem e computador se dá através de múltiplos meios de representação de informação (p.ex.: áudio, imagem estática, animação, gráficos e texto), ou seja, multimídia é o conjunto de recursos que visam estimular todos os sentidos, porém, os mais usuais são a visão e a audição. Multimídia significa que uma informação digital pode ser representada através de áudio, vídeo e animação em conjunto com mídias tradicionais (texto, gráficos e imagens) simultaneamente (FIGUEREDO, 2010).

Sabendo desse conceito, pode-se compreender a importância dos sistemas multimídia que ajudam os seres humanos durante parte da vida, desde um canal de comunicação como aplicativos similares ao "WhatsApp" até portais de notícias como "The Washington Post". Desta forma, Willrich (2000) classifica as mídias em dois tipos, Mídias capturadas versus sintetizadas e as Mídias discretas versus contínuas.

O primeiro tipo de classificação trata de informações capturadas do mundo real como sons, imagens e vídeos, além de informações sintetizadas por computador como textos, gráficos e animações. Logo, uma aplicação multimídia é composta por pelo menos uma mídia sintetizada junto a uma capturada. O segundo tipo distingue entre mídias discretas ou estáticas, que são consideradas as de dimensão espacial (textos, imagens e gráficos), e mídias contínuas,

as quais contêm sons, vídeos e animações. Essas duas classes podem ser diferenciadas entre dependentes e independentes de tempo.

É válido citar que as áreas de aplicação dessa tecnologia estão entre os mais variados campos da informação, podendo ser parte de entretenimento como filmes, cinemas interativos, jogos, além disso, para educação como cursos online, materiais de ensino, interações técnicas entre outros. O mais comum atualmente são as redes sociais como o X, uma plataforma que visa utilizar da tecnologia para informar, debater e dar espaço a grupos para expor suas ideias ao público.

2.2 DISCURSO DE ÓDIO

A temática em questão tem sido um dos principais alvos tanto político como midiático, além disso, possui uma maior relevância nas redes sociais onde se concentra a maioria dos casos. Segundo Grossmann (2022), houve aumento na quantidade de denúncias em ambientes virtuais entre 2020 e 2021, o que pode ser justificado por um maior uso das redes devido a pandemia de Covid-19, revelando o pior lado das plataformas.

"Tivemos aumento expressivo em 2021 das denúncias relacionadas à imagens de abuso e exploração sexual de crianças e adolescentes, mesmo depois do boom do primeiro ano da pandemia, que já resultou em um aumento de mais de 100% no número de casos, mesmo comparado com essa base, o número aumentou. Também aumentou significativamente as denúncias de LGBTfobia e transfobia. E para nossa grande preocupação, a curva de denúncias e casos relacionados a neonazismo continua sendo muito ascendente e persistente: 60% de aumento nas denúncias em relação a 2020, quando já houve um aumento de mais de 100%. Os maiores números da década foram alcançados em 2020 e 2021" (GROSSMANN, 2022, n. p.).

Sabendo disso, faz-se necessário esclarecer o que pode ser considerado "Discurso de ódio" e como as empresas, donas de redes sociais, enxergam esse tipo de crime em suas plataformas para que as interações aviltantes possam ser combatidas.

Discurso de ódio ou Hate speech caracteriza-se pelo conteúdo segregacionista, fundado na dicotomia da superioridade do emissor e na inferioridade do atingido (a discriminação), e pela externalidade, ou seja, existirá apenas quando for dado a conhecer a outrem, que não o próprio emissor.

Já na visão de Schafer (2015) distingue esse fenômeno em dois modos diferentes. Hate speech in form é a definição para as formas de expressão carregando conteúdo de ódio de forma explícita, como por exemplo comentários racistas ou preconceituosos, misóginos ou que tentam diminuir ou inferiorizar minorias. Hate speech in substance é a segunda denominação para esse tipo de agressão e que se refere a modalidade velada do discurso de ódio podendo se apresentar como um argumento de proteção moral e social.

Direcionando o tema para uma rede social, em especial o X (2017), pode-se notar que ele conceitua discurso de ódio como sendo a conduta de promover violência, atacar diretamente ou ameaçar outras pessoas com base em raça, etnia, origem nacional, orientação sexual, sexo, identidade de gênero, religião, idade, deficiência ou doença grave. Essa definição consta nas regras e políticas da plataforma para alertar e proteger os usuários.

2.3 X

O X é uma plataforma de serviço de mensagens com uma proposta diferente que permite interação por meio de mensagens denominadas de tweets. Os tweets são mensagens curtas com até 280 caracteres de texto, fotos, GIFs, vídeos, links e enquetes. Cada tweet possui algumas

formas de interação, como: as respostas, os retweets, as curtidas e o compartilhamento do tweet, também chamado de engajamentos. Outras formas de interação são as Hashtags (#): antes de uma palavra-chave ou frase relevante nos Tweets que publicam para classificá-los e facilitar a exibição deles na busca do X e as menções (@): um Tweet que contém o nome de usuário de uma pessoa em qualquer lugar do corpo do Tweet (X, 2022, n. p.)

O X foi fundada em março de 2006, como um projeto paralelo de uma empresa de podcasts de São Francisco, atualmente possui mais de 5,5 mil colaboradores, em seus mais de 35 escritórios espalhados pelo mundo, que obteve uma receita de 3.716 bilhões de dólares, com 192 milhões de usuários ativos diários monetizáveis (mDAUs) em 2020, assim como de 330 milhões de usuários ativos, 211 milhões são mDAUs com 500 milhões de tweets postados todos os dias em 2021 (AHLGREN, Matt; WSR TEAM., 2022; O'REILLY, Tim; MILSTEIN, Sarah., 2011).

Na busca por prover um diálogo público com liberdade e segurança. O X estabelece regras para que ocorra um uso igualitário para cada usuário, baseado em seus princípios de promover conversas saudáveis que ajudem o mundo a aprender mais rápido e a resolver problemas comuns (X, 2022, n. p.). Por vezes, essas conversas geram muitas divergências devido a diversidade de opiniões, mesmo quando elas possuem em sua fundamentação primária um viés ofensivo, controverso, intolerante, mas ainda assim possuem o potencial de gerar uma discussão aberta e interessante ao público geral, como também uma potencial mudança de opinião.

Na ocorrência de descumprimento das regras estabelecidas em seu ambiente multimídia, o X, aplica punições baseando-se no contexto de cada ocorrência, segundo sua categoria e vigência em suas regras, considerando inúmeros fatores, como se: o comportamento é direcionado a um indivíduo, grupo ou categoria protegida de pessoas; a denúncia foi registrada pela vítima do abuso/assédio ou por um espectador; o usuário já é conhecido por violar as políticas do X; a violação é grave; o conteúdo pode ser tema de interesse público legítimo (X, 2022, n. p.).

2.4 API

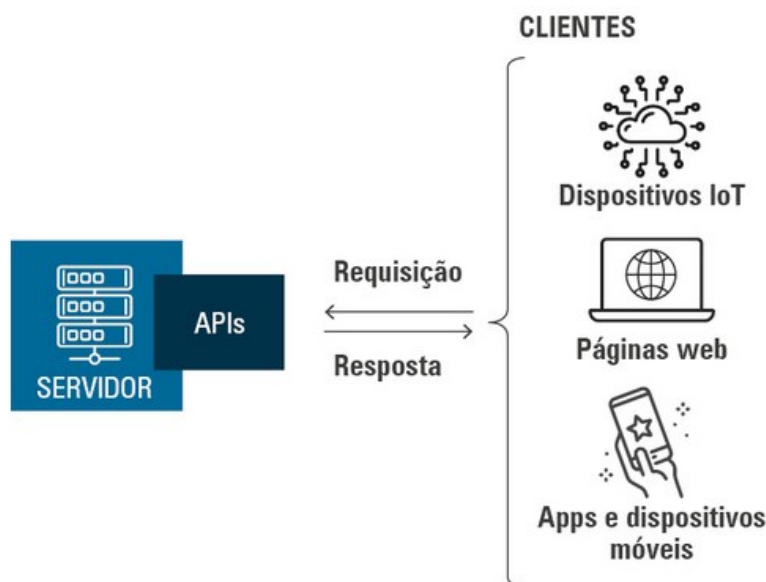
Uma Application Programming Interface (API), ou Interface de Programação de Aplicação, é uma interface que integra aplicações diferentes, com comandos predefinidos por meio de mensagens trocadas entre cliente e servidor utilizando o Protocolo de Transferência de Hipertexto (HTTP). As mensagens enviadas pelo cliente são chamadas de requests, ou requisições, como também, de solicitações, e as mensagens enviadas pelo servidor como resposta são chamadas de responses, ou respostas. (MDN, 2022; TORRES, Fernando, 2021).

As APIs representam um componente fundamental na arquitetura de sistemas modernos, especialmente no contexto de redes sociais e plataformas digitais. Segundo Fielding (2000), a arquitetura REST (Representational State Transfer) estabelece princípios para o design de APIs que promovem escalabilidade, simplicidade e interoperabilidade entre sistemas distribuídos. No contexto deste estudo, a API do X funciona como uma ponte que permite o acesso programático aos dados públicos da plataforma, possibilitando o desenvolvimento de aplicações terceiras para análise e monitoramento de conteúdo.

A importância das APIs no combate ao discurso de ódio reside na capacidade de automatizar processos de coleta e análise de dados em larga escala, algo impraticável através de métodos manuais. Torres (2021) destaca que APIs REST bem projetadas facilitam a integração entre diferentes sistemas, permitindo que pesquisadores e desenvolvedores criem

ferramentas especializadas para identificação de padrões comportamentais e conteúdos problemáticos em plataformas digitais.

Figura 01 - Diagrama de API cliente-servidor.



Fonte: TORRES, Fernando, 2021.

3 PROCEDIMENTOS METODOLÓGICOS

Para alcançar o objetivo principal deste trabalho de verificar até que ponto a API do X permite examinar comentários aviltantes por meio de aplicações terceiras em seu ambiente sandbox dentro de um contexto de eleições governamentais, contemplou analisar os recursos e funcionalidades da API do X, criar e descrever uma aplicação com base nos recursos oferecidos pela API do X e identificar a eficácia da aplicação em capturar tweets aviltantes, em um período de 9 dias, começando no dia 24 de Setembro até o dia 2 de outubro de 2022.

A escolha deste período específico se justifica por coincidir com a reta final da campanha eleitoral para a presidência do Brasil, momento caracterizado por intensa atividade nas redes sociais e maior probabilidade de ocorrência de discursos polarizados e potencialmente ofensivos. Este contexto temporal oferece um cenário ideal para testar a eficácia da API em capturar comentários aviltantes em um ambiente de alta densidade de interações políticas.

Para análise dos principais recursos e funcionalidades da API do X, primariamente, foi feita uma pesquisa utilizando o motor de busca do Google, com as palavra-chave: “api do X”, que retornou como primeiro item de resultado a webpage “Sobre as APIs do X - X Help Center” e como item vinculado “X API Documentation | Docs | X Developer Platform”.

A webpage “Sobre as APIs do X - X Help Center” explica o que são as APIs do X, que tipo de informação fica disponível com elas e algumas medidas de proteção que o X preparou para que as APIs sejam usadas (X, 2022, n. p.).

As APIs são a forma como os programas de computador “conversam” entre si para trocar informações, através de terminais (X, 2022, n. p.). Os dados disponibilizados nesses terminais têm um caráter único de compartilhamento em relação a outras mídias sociais, refletem as informações que os usuários escolhem compartilhar publicamente, permitindo amplo acesso aos dados públicos do X que os próprios usuários escolheram compartilhar (X, 2022, n. p.),

dividem-se em cinco grupos principais: Contas e usuários, Tweets e respostas, Mensagens Diretas, Anúncios e Ferramentas de publisher e SDKs.

O item vinculado “X API Documentation | Docs | X Developer Platform”, é a webpage que reúne as informações sobre as versões, os níveis de acesso, os usos e as funcionalidades da API do X, possuem três versões: a empresarial, Gnip 2.0, as versões 1.1 (v1.1) e a versão 2 (v2).

A Gnip 2.0, é uma API corporativa que oferece o mais alto nível de acesso e confiabilidade para quem depende dos dados do X. A versão 1.1 possui duas modalidades: a padrão, que é voltada para começar, testar uma integração, validar um conceito ou criar soluções, assemelhando-se a um flavor, ou amostra, da versão premium; a premium oferece acesso escalável aos dados do X e acesso rápido a recursos empresarial. A versão 2 é, atualmente, a API principal do X e inclui uma base moderna, recursos que não são encontrados nas outras versões de API fornecidas pela plataforma, como também inclui recursos com funcionalidades mais avançadas e integralização mais eficiente (X, 2022, n. p.).

Para criar e descrever uma aplicação com base nos recursos oferecidos pela API do X, foi produzido uma API Rest para auxiliar e automatizar o processo de filtragem dos dados coletados, em que foram usados um runtime NodeJS, em linguagem de programação Javascript para consumir a API principal fornecida pelo X - a v2, seguindo orientações da própria plataforma, para armazenar os dados, utilizou-se um banco de dados NoSQL, no qual foi usado o MongoDB, atualmente, segundo o DB-Engines Ranking, que classifica os sistemas de gerenciamento de banco de dados de acordo com sua popularidade, o atribui a quinta posição no ranking dos SGBDs mais usados e o primeiro na categoria NoSQL de armazenamento por documento (DB-ENGINES, 2022).

Tendo a implementação da API Rest em linguagem NodeJS utilizou-se a biblioteca ExpressJS, uma estrutura de aplicação Web enxuta e flexível que fornece um conjunto variado de recursos com uma grande quantidade de funções para métodos HTTP e middleware, funções que têm acesso ao objeto de solicitação - request, e ao objeto de resposta - response e a próxima função que será chamada - next, no ciclo de request-response da aplicação, à disposição, como também a biblioteca Axios, um cliente HTTP simples baseado em promises para o navegador e para o NodeJS, a biblioteca Nodemon, um utilitário que monitora quaisquer alterações em sua fonte e reinicia automaticamente o servidor HTTP.

Para identificar a eficácia da aplicação em capturar tweets aviltantes, a captura dos Tweets teve a utilização da plataforma web do X para analisar o ranking das trendings mais populares tendo como critério de seleção, as trendings que estivesse em alta posição no ranking e dentro do contexto das eleições presidenciais de 2022, pois como compõe Neves (2021) “[...] a crescente radicalização política e ideológica, especialmente nas redes sociais, fomenta as intolerâncias mútuas e suspende o debate construtivo ao desconhecer qualquer qualidade a ideias divergentes”, como também foi criado uma lista de palavras simples e finitas, no idioma Português brasileiro, com o único intuito de compará-las às palavras contidas nos Tweets capturados no período de coleta delimitado no presente trabalho, utilizando a biblioteca pii, que disponibiliza uma função para filtragem de palavras com base em uma lista de palavras personalizadas.

Figura 02 - Demonstração de função de filtragem com Piii.

```
const filtro = ["foo", "bar"];
const piiiFilters = require("piii-filters");
const piii = new Piii({
  filters: [
    ...Object.values(piiiFilters),
    ...filtro
  ]
});

piii.filter("foo bar baz") // resultado "* * bar"
```

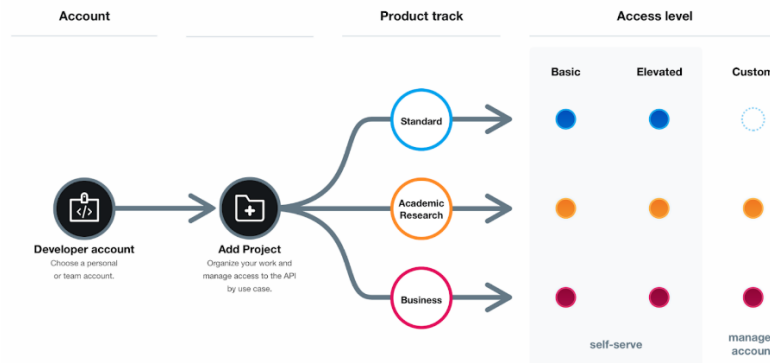
Fonte: Modificado pelo autor.

4 RESULTADOS

4.1 RECURSOS E FUNCIONALIDADES DA API DO X

O resultado obtido ao analisar os principais recursos e funcionalidades da API do X, foram de utilizar uma ferramenta de autenticação externa, de possuir 3 níveis de acesso para dimensionar o uso dos terminais da plataforma, que permite o uso de diferentes recursos em níveis: Essential, Elevated e o Academic Research, que irão gerenciar o acesso às três versões variadas disponíveis da API, os dados componentes da API e os SDKs oferecidos pela X Developer Platform.

Figura 03 - Linhas de produto e níveis de acesso.

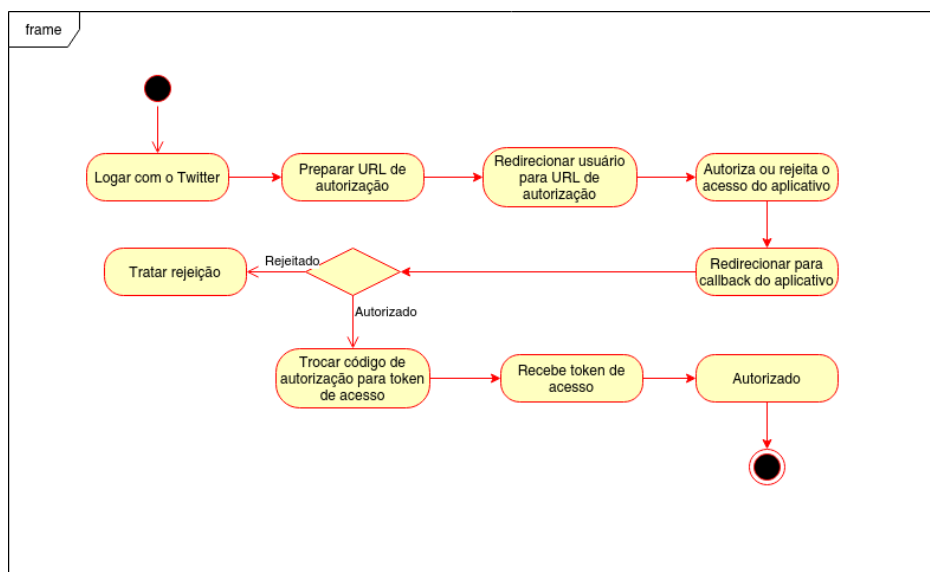


Fonte: CAIRNS, Ian; SHETTY, Priyanka, 2020.

4.1.1 Autenticação

A autenticação ocorre com a ferramenta externa OAuth 2.0, por ser um protocolo de autorização padrão do setor de desenvolvimento que fornece fluxos de autorização específicos permitindo um maior controle sobre o escopo de um aplicativo e fluxos de autorização em vários dispositivos. OAuth 2.0 permite a escolha de escopos detalhados e específicos que fornecem permissões específicas em nome de um usuário, sendo habilitada nas configurações de autenticação do aplicativo na seção de configurações do portal do desenvolvedor (X, 2022, n. p.).

Figura 04 - Diagrama de ação de autenticação.



Fonte: Modificado autor.

4.1.2 Níveis De Acesso

O nível Essential, é um nível de acesso básico e gratuito, não exige solicitação de acesso adicional, tem como fator de aquisição o cadastro de uma nova conta ou posse de uma conta cadastrada na plataforma, com esse acesso, o desenvolvedor pode adicionar um projeto e uma aplicação vinculada a esse projeto, cada aplicação pode fazer 500 mil requisições do tipo GET com duração de um mês, usando até cinco regras de pesquisa para recuperar tweets com até 512 caracteres.

O nível Elevated exige uma solicitação de acesso adicional. Após criar uma conta ou possuir uma conta existente, mas o acesso continua sendo gratuito, assim como o nível Essential, pode-se adicionar somente um projeto no Elevated, entretanto, o desenvolvedor pode vincular três aplicações ao mesmo projeto, podendo recuperar até 2 milhões de tweets por mês, usando até 25 regras de pesquisa para recuperar tweets com até 512 caracteres.

O Academic Research é uma linha de produto que possui um nível mais elevado e exige uma solicitação de acesso adicional, assim como os outros níveis, é gratuito e permite adicionar um projeto e vinculá-lo a uma aplicação e recuperar até 10 milhões de tweets por mês, usando 1000 regras para recuperar tweets com 1024 caracteres, tem o foco no público que desenvolve pesquisa acadêmicas.

Tabela 1 - Níveis de acesso e Academic Research.

	Essential	Elevated	Academic Research
Acesso	Grátis	Grátis	Grátis
Acesso extra	Não	Sim	Sim
Projetos	1	1	1
Aplicação	1	3	1
Tweet por mês	500.000	2.000.000	10.000.000
Regras	5	25	1000
Limite de caracteres	512	512	1024

Fonte: autores.

4.1.3 Dados da API

Nos terminais da API do X, existe uma grande quantidade de dados - do tipo objetos, disponíveis para solicitação, como os Tweets, Users, Spaces, Lists, Media, Polls e Places.

4.1.3.1 Tweets

O objeto Tweet é a base da plataforma do X e pode conter outros objetos Tweet, tem uma longa lista com 21 propriedades principais, como mostrado em seguida no quadro:

Quadro 01 - Propriedades do objeto Tweet.

Campos	Descrição
id	Campo do identificador único que pode ser usado para recuperar um Tweet específico.
text	Campo da mensagem do Tweet que pode ser usado para extração de palavras-chave, análise e classificação de sentimentos como outros casos de uso.
edit_history_tweet_id	Campo que armazena uma lista de identificadores únicos das versões editadas dos Tweets.
lang	Campo que identifica o idioma de origem do Tweet.
source	Campo que mostra o nome do aplicativo do qual o usuário tweetou.
in_reply_to_user_id	Campo que indica se o Tweet representado é uma resposta, nesse caso, conterá o <i>id</i> do autor do Tweet original.
entities	Campo que indica os objetos que fornecem informações adicionais sobre hashtags, urls, menções de usuários e cashtags associados a um Tweet.
author_id	Campo que indica o identificador exclusivo do usuário que postou este Tweet, ou seja, corresponde ao <i>id</i> do objeto do tipo User.
referenced_tweets	Campo com uma lista de Tweets aos quais faz referência, em casos de ser um Retweet, um Retweet com comentário - também conhecido como Tweet Citado - ou uma Resposta, ele incluirá o Tweet relacionado mencionado por seu pai.
public_metrics	Métricas de engajamento público do Tweet no momento da solicitação.
context_annotations	Campo que contém informações de contexto sobre o Tweet.
created_at	Campo que indica a data e hora em que o Tweet foi criado.
attachments	Campo que especifica o tipo de anexo presente neste Tweet no caso de possuí-lo, ou seja, um outro tipo de objeto, como: Polls, Users e os demais.
possibly_sensitive	Este campo indica que o conteúdo pode ser reconhecido como sensível. O autor do Tweet pode selecionar dentro de suas próprias preferências de conta e escolher "Marcar a mídia que você tweetou como tendo material que pode ser sensível" para que cada Tweet criado depois tenha esse sinalizador definido.
conversation_id	Campo que indica o <i>id</i> do Tweet que originou a conversa, incluindo respostas diretas, respostas de respostas.
edit_controls	Campo que, quando presente, indica por quanto tempo o Tweet pode ser editado e o número de edições restantes, propriedade que compõe o objeto Tweet na ocorrência de edições. Os tweets são editáveis apenas nos primeiros 30 minutos após a criação e podem ser editados até cinco vezes.
non_public_metrics	Campo de métricas de engajamento não públicas para o Tweet no momento da solicitação.
organic_metrics	Campo de métricas de engajamento, rastreadas em um contexto orgânico, para o Tweet no momento da solicitação.
promoted_metrics	Campo de métricas de engajamento, rastreadas em um contexto promovido, para o Tweet no momento da solicitação.
answer_settings	Campo que mostra quem pode responder a um determinado Tweet. Os campos retornados são "everyone", "mentioned users" e "followers".
withheld	Campo que, quando presente, contém detalhes de retenção de conteúdo retido.

Fonte: Modificado pelo autor.

Figura 05 - Exemplo de um objeto Tweet.

```
{
  "data": [ {
    "text": "We believe the best future version of our API will come
from building it with YOU. Here's to another great year with everyone who
builds on the Twitter platform. We can't wait to continue working with you
in the new year. https://t.co/yvxdK6aOo2",
    "edit_history_tweet_ids": ["1212092628029698048"],
    "lang": "en", "source": "Twitter Web App",
    "in_reply_to_user_id": "2244994945",
    "entities": { "urls": [
      { "start": 222, "end": 245, "url": "https://t.co/yvxdK6aOo2",
        "expanded_url":
"https://twitter.com/LovesNandos/status/1211797914437259264/photo/1",
        "display_url": "pic.twitter.com/yvxdK6aOo2",
        "media_key": "16_1211797899316740096"
      }
    ]},
    "annotations": [ {
      "start": 42, "end": 44, "probability": 0.5359,
      "type": "Other", "normalized_text": "API"
    }, { ... } ],
    "author_id": "2244994945",
    "referenced_tweets": [ {
      "type": "replied to",
      "id": "1212092627178287104"
    } ],
    "id": "1212092628029698048",
    "public_metrics": {
      "retweet_count": 7, "reply_count": 3,
      "like_count": 38, "quote_count": 1
    },
    "context_annotations": [ {
      "domain": {
        "id": "29", "name": "Events [Entity Service]",
        "description": "Real world events. "
      },
      "entity": {
        "id": "1186637514896920576",
        "name": " New Years Eve"
      },
      { ... } ],
    "created_at": "2019-12-31T19:26:16.000Z",
    "attachments": { "media_keys": ["16_1211797899316740096"]
    },
    "possibly_sensitive": false
  } ],
  "includes": { "tweets": [ { ... } ] }
}
```

Fonte: Modificado pelo autor.

4.1.4 SDK

Os SDKs englobam a API do X, fazendo-se necessário a posse de uma conta de desenvolvedor para autenticar as solicitações, como mostrado no item 4.1.1, usando as credenciais do aplicativo de desenvolvedor, localizado no projeto da X Developer Platform.

A X Developer Platform, disponibiliza 2 SDKs oficiais para desenvolvedores TypeScript/Javascript e Java, no qual utilizou-se o TypeScript/Javascript. Permitindo que o desenvolvimento ocorresse de modo mais eficiente, eliminando a necessidade de programar manualmente as complexidades em torno da API do X, fazendo-se uso das funções pré-criadas para os endpoints da versão 2 utilizadas, como também simplificou o processo de autenticação.

Os SDKs disponibilizados são criados e mantidos pela equipe da Developer Platform, o que garante a manutenção e atualização constante com novas versões da API do X.

4.2 CRIAÇÃO E DESCRIÇÃO DA APLICAÇÃO COM BASE NOS RECURSOS OFERECIDOS PELA API DO X

No presente capítulo será abordado a implementação de uma API REST utilizando Node.js e recursos disponibilizados pelo X para o acesso aos dados públicos da plataforma.

4.2.1 Descrição da API.

O desenvolvimento foi feito em cima de um sistema operacional Mac OS 13.0, codinome Ventura, para escrever os códigos Javascript, como também para executar os comandos de inicialização da aplicação, foi usado o editor de texto Visual Studio Code e o terminal embutido presente no editor.

O padrão de projeto utilizado foi o Model-View-Controller (MVC), que sofreu adaptações para utilização apenas dos models e dos controllers da aplicação. O banco de dados utilizado para salvar os tweets na aplicação foi MongoDB na versão 6.0.

A máquina usada para realizar as implementações foi o modelo MacBook Air, com chip de processamento M1, tecnologia de arquitetura ARM, da fabricante Apple Inc., com 8GB de RAM e 256GB de armazenamento.

4.2.2 Implementação

Para a implementação dos códigos da API foi utilizado o Express, uma biblioteca de código aberto que permite a criação de inicialização de um servidor na plataforma Node.js. Essa biblioteca foi utilizada, pois possui um conjunto de métodos utilitários HTTP para criar uma API REST robusta, rápida e fácil.

Ao ser executado com o CLI do node já é possível verificar através de logs a api online no host padrão e na porta configurada, já podendo ser feitas as requisições para a API.

Figura 06 - Server.js.

```
const express = require("express")
const { mainController } = require("../controller");
const server = () => {
  const app = express()
  const port = 4000
  mainController(app)
  app.listen(port, () => {
    console.log("Api online em http://localhost:" + port)
  })
}
module.exports = {
  server
}
```

Fonte: Autor, 2022.

Para a conexão com o banco de dados foi utilizado o pacote Mongoose que faz parte do ambiente para desenvolvimento mongo com Javascript em tempo de execução. Com essa

biblioteca foi possível fazer a ligação e guardar os dados dos comentários do X com as informações que foram usadas para gerar os resultados.

Figura 07 - Database.js.

```
const mongoose = require("mongoose")
mongoose.connect(mongoose.connect(process.env.DATABASE_URI);
const databaseInit = (callback) => mongoose.connection.once("open", callback);
module.exports = {
  databaseInit
}
```

Fonte: Autor, 2022.

Para a parte da modelagem dos dados salvos no banco foi utilizado a classe Schema que serve para criar uma interface dos dados que serão salvos e organizar os campos de acordo com cada tipo de dado.

Figura 08 - Tweet.js.

```
const { Schema, Model, model } = require("mongoose");
const tweetSchema = new Schema({
  id:Number,
  url:String,
  text:String,
})
module.exports = {
  TweetModel: model("tweets", tweetSchema)
}
```

Fonte: Autor, 2022.

As rotas criadas para as requisições da Api utilizam os métodos do protocolo HTTP, as quais são do tipo GET e POST.

Requisições GET requisitam apenas dados, e não devem ter outro efeito. Sempre que se insere um endereço no navegador web, é realizada uma requisição GET junto ao servidor com o intuito de receber a representação para aquela URL. O verbo POST tem o papel de se comunicar com o servidor para indicar a criação de recursos como a adição de um novo documento ao banco de dados, assim retornando ao cliente apenas o código de status da requisição o qual indica se houve sucesso ou falha na requisição.

Para a busca e criação dos tweets foi usada a rota "/tweets" da api, conectada a uma função que há inicialmente a instância da classe Client, importada do "X-api-sdk", para criar um cliente autenticado com as chaves de API para consultas ao ambiente sandbox do X, logo após isso, há os parâmetros do termo da trending selecionada e um contador de retorno do número de postagem que foram enviados como parâmetros da requisição. Com os tweets recuperados, há uma verificação no banco para cada postagem com finalidade de encontrar possíveis repetições e assim impedir da mesma postagem ser salvas duas ou mais vezes. O retorno da rota consiste em uma simples mensagem de sucesso que indica com o status 200 que não houve nenhum problema na requisição.

Figura 09 - Get-tweets-index.js.

```
app.post("/tweets", async (req, res) => {
  try {
```

```

const client = new Client(BEARER_TOKEN);
let termo = "micheque"
const stream = await client.tweets.tweetsRecentSearch({
  "query": termo,
  "max_results": 100,
  "tweet.fields": [ "id", "text", ],      });
for (let v of stream.data) {
  let has = await TweetModel.findOne({ id: v.id })
  if (!has) {
    await TweetModel.create({
      ...v,
      url: "https://X.com/OficialDeborahA/status/" + v.id
    })
  }
}
res.send("sucesso")
} catch (error) {
  console.log(error)
}
})

```

Fonte: Autor, 2022.

Para a filtragem dos comentários do X foi utilizado, através da rota "/filter" uma função que utiliza uma biblioteca de filtros customizados para buscas em textos. O filtro possui uma lista de palavras aviltantes que podem ser utilizadas na comparação dos comentários capturados e salvos no banco. Logo após a filtragem dos comentários é feito o cálculo de porcentagem utilizando o total de documentos no banco e total de comentários que possuem algum tipo de comentário ofensivo. O retorno da rota consiste no total de tweets armazenados no banco, o total de comentários ofensivos filtrados, a porcentagem de comentários ofensivos em relação ao total de documentos no banco e a lista dos textos aviltantes. Todos esses dados retornam da rota através de uma estrutura de dados JSON.

Figura 10 - Get-filter-index.js.

```

app.get("/filter", async (req, res) => {
  try {
    let tweets = await TweetModel.find()
    let bad = []
    const piiii = new Piiiii({
      filters: [
        ...Object.values(piiiiiFilters),
        ...badwords
      ]
    });
    tweets.map(t => {
      let has = piiiii.has(t.text)
      if (has) {
        bad.push(t.text)
      }
    })

    let calc = bad.length * 100 / tweets.length
    res.json({
      total_tweets: tweets.length,
      total_hate_speech_tweets: bad.length,

```

```
        hate_speech_percent:calc,  
        bad,  
    })  
  
    } catch (error) {  
        console.log(error)  
    }  
})
```

Fonte: Autor, 2022.

5 DISCUSSÃO

Esta seção apresenta uma análise crítica dos resultados obtidos, contextualizando-os dentro do panorama mais amplo das políticas públicas e desigualdades digitais no Brasil.

Os achados deste estudo revelam limitações significativas no acesso aos dados da API do X, o que levanta questões importantes sobre a democratização do acesso à informação para fins de pesquisa e desenvolvimento de políticas públicas. A restrição nos dados disponibilizados pela plataforma não apenas compromete a eficácia de ferramentas de monitoramento, mas também reproduz desigualdades no acesso à informação, privilegiando atores com maior poder econômico e técnico.

Do ponto de vista das políticas públicas, os resultados evidenciam a necessidade de marcos regulatórios que busquem regulamentar, sem controlar excessivamente, as plataformas digitais. A regulamentação deve garantir maior transparência e acesso aos dados para fins de pesquisa acadêmica e desenvolvimento de ferramentas de combate ao discurso de ódio, preservando, contudo, a liberdade de expressão e a inovação tecnológica. O controle excessivo pode levar à censura e à limitação do debate democrático, enquanto a regulamentação adequada estabelece diretrizes claras para promover um ambiente digital mais justo e igualitário.

As limitações identificadas na captura de apenas 3,8% de tweets com conteúdo potencialmente ofensivo podem estar relacionadas não apenas às restrições técnicas da API, mas também às barreiras estruturais que impedem o desenvolvimento de ferramentas mais sofisticadas de análise contextual. Esta situação perpetua desigualdades no combate ao discurso de ódio, uma vez que permanece em exposição conteúdos prejudiciais sem mecanismos adequados de monitoramento.

A interface entre tecnologia e políticas públicas se torna ainda mais crítica quando consideramos que o discurso de ódio afeta desproporcionalmente grupos já marginalizados na sociedade. As ferramentas de monitoramento e intervenção devem ser desenvolvidas com o objetivo primordial de promover a justiça e a igualdade no ambiente virtual, garantindo que todos os usuários tenham acesso a um espaço digital seguro e respeitoso. Isso demanda uma resposta coordenada entre poder público, academia e sociedade civil, sempre pautada pelo equilíbrio entre proteção e liberdade.

6 CONSIDERAÇÕES FINAIS

Podemos concluir após a filtragem de um total de 1193 Tweets armazenados localmente, que aproximadamente 3.8%, foram comentários contendo alguma palavra aviltante, ou seja, foram encontradas somente 46 tweets com palavras aviltantes. Assim podemos constatar que é possível coletar, armazenar e manipular os dados obtidos por meio da Api do X, no entanto as limitações quanto a análise e mapeamento de contexto desses comentários faz com que o resultado não seja de total confiança.

A não confiabilidade se deu por uma série de barreiras que reduziram a margem de captura dos textos em seu processo de seleção. Um dos fatores limitantes, quanto à verificação de comentários aviltantes, está no número reduzido de caracteres disponibilizado nos tweets, que afetaram a análise do contexto ao qual o comentário pertence e como consequência comprometeu a eficácia em capturar um discurso de ódio velado, tal como expresso por Schafer (2015). Além disso, a análise de contexto é por si, complexa, ao ponto de não ter sido encontrado formas da tecnologia capturar tal compreensão, em tempo hábil.

Este estudo demonstra que, embora seja tecnicamente viável desenvolver ferramentas de monitoramento de discurso de ódio utilizando APIs de redes sociais, existem limitações estruturais significativas que comprometem sua eficácia. Estas limitações não são meramente técnicas, mas refletem questões mais amplas relacionadas à governança digital e à necessidade de políticas públicas que promovam maior transparência e acesso aos dados das plataformas.

A pesquisa evidencia a urgência de se estabelecer marcos regulatórios que equilibrem a proteção da privacidade dos usuários com a necessidade de acesso a dados para fins de pesquisa acadêmica e desenvolvimento de políticas públicas de combate ao discurso de ódio. Sem essa regulamentação das diretrizes, políticas e termos das plataformas, persistem as desigualdades, mecanismos ineficazes na proteção contra a violência digital.

Para futuras pesquisas, recomenda-se a exploração de abordagens metodológicas que combinem análise quantitativa e qualitativa, bem como o desenvolvimento de parcerias entre academia, poder público e organizações da sociedade civil para pressionar por maior transparência das plataformas digitais. Somente através de uma abordagem multidisciplinar e colaborativa será possível desenvolver ferramentas verdadeiramente eficazes para o combate ao discurso de ódio e a promoção da justiça social no ambiente digital.

Por fim, este trabalho contribui para o campo da Administração ao demonstrar como questões tecnológicas se entrelaçam com desafios organizacionais e de políticas públicas, evidenciando a necessidade de uma abordagem integrada que considere tanto aspectos técnicos quanto sociais no desenvolvimento de soluções para problemas complexos da sociedade contemporânea.

7 REFERÊNCIAS

Aviltante. (2022). In Dicio, Dicionário Online de Português. Porto: 7Graus. Recuperado em 14 setembro, 2022, de <https://www.dicio.com.br/aviltante/>

Ahlgren, M., & WSR Team. (2022). 50+ Twitter statistics & facts for 2022. Website Rating. Recuperado em 21 setembro, 2022, de <https://www.websiterating.com/research/twitter-statistics/>

Bradshaw, S., Brazil, E., & Chodorow, K. (2019). MongoDB: The definitive guide: powerful and scalable data storage (3a ed.). O'Reilly Media.

Constituição da República Federativa do Brasil de 1988. (1988). Brasília, DF. Recuperado em 25 agosto, 2022, de http://www.planalto.gov.br/ccivil_03/Constituicao/Constituicao/

Cairns, I., & Shetty, P. (2022, julho 16). Chegou uma nova e melhor API do Twitter. Twitter Blog. Recuperado em 3 setembro, 2022, de https://blog.twitter.com/pt_br/topics/product/2019/chegou-uma-nova-e-melhor-api-do-twitter

Central Nacional de Denúncias de Crimes Cibernéticos. (2021). Indicadores da Central Nacional de Denúncias de Crimes Cibernéticos. SaferNet Brasil. Recuperado em 3 maio, 2022, de <https://indicadores.safernet.org.br/>

DB-Engines. (2022). DB-Engines ranking. Recuperado em 15 novembro, 2022, de <https://db-engines.com/en/ranking>

Fielding, R. T. (2000). Architectural styles and the design of network-based software architectures. Tese de doutorado, University of California, Irvine, CA, USA. Recuperado em 13 outubro, 2022, de <http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm>

Figueredo, R. T. (2010). Apostila de sistemas multimídia [Apostila]. Petrolina. Recuperado em 20 outubro, 2022, de <https://central3.to.gov.br/arquivo/453373/>

Fluckiger, F. (1995). Understanding networked multimedia: applications and technology. Prentice Hall.

Grossmann, L. O. (2022, fevereiro 8). Safernet: Discurso de ódio na internet disparou em 2020 e 2021. Convergência Digital. Recuperado em 19 maio, 2022, de <https://www.convergenciadigital.com.br/Internet/Safernet%3A-Discurso-de-odio-na-internet-disparou-em-2020-e-2021-59372.html>

Haverbeke, M. (2018). Eloquent Javascript: a modern introduction to programming (3a ed.). No Starch Press.

MDN Web Docs. (2022). Uma visão geral do HTTP. Mozilla. Recuperado em 8 novembro, 2022, de <https://developer.mozilla.org/pt-BR/docs/Web/HTTP/Overview>

MongoDB. (n.d.). O que é o MongoDB? Recuperado em 8 novembro, 2022, de <https://www.mongodb.com/pt-br/what-is-mongodb>

Moraes, W. B. (2021). Construindo aplicações com NodeJS (3a ed.). Novatec Editora.

O'Reilly, T., & Milstein, S. (2011). The Twitter book (2a ed.). O'Reilly Media.

Schäfer, G., Leivas, P. G. C., & Santos, R. H. (2015). Discurso de ódio: da abordagem conceitual ao discurso parlamentar. Revista de Informação Legislativa, 52(207), 143-158. Recuperado em 15 junho, 2022, de https://www12.senado.leg.br/ril/edicoes/52/207/ril_v52_n207_p143.pdf

Shevat, A., & Penn, S. (2022, novembro 15). Construa o que vem por aí com a plataforma de desenvolvedor do Twitter. Twitter Blog. Recuperado em 13 outubro, 2022, de https://blog.twitter.com/pt_br/topics/product/2021/construa-o-que-vem-por-ai-com-a-plataforma-de-desenvolvedor-do-t

Torres, F. E. (2021). Desenvolvimento de API REST. Editora Senac São Paulo.

Twitter. (2022, novembro 16). Documentation. Twitter Developer Platform. Recuperado em 4 outubro, 2022, de <https://developer.twitter.com/en/docs/twitter-api>

Twitter. (2022, novembro 16). Sobre o Twitter. Recuperado em 4 outubro, 2022, de <https://about.twitter.com/pt>

Twitter. (2022, novembro 16). Sobre as APIs do Twitter. Twitter Help Center. Recuperado em 4 outubro, 2022, de <https://help.twitter.com/pt/rules-and-policies/twitter-api>

Twitter. (2022, novembro 16). Tools and libraries. Twitter Developer Platform. Recuperado em 4 outubro, 2022, de <https://developer.twitter.com/en/docs/twitter-api/tools-and-libraries/sdks/overview>