UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Felipe Harrison Silva Cantarino

Criação de um Corpus Português para Auxiliar a Identificação de Notícias Verdadeiras e Falsas

Uberlândia, Brasil

2024

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Felipe Harrison Silva Cantarino

Criação de um Corpus Português para Auxiliar a Identificação de Notícias Verdadeiras e Falsas

Trabalho de conclusão de curso apresentado à Faculdade de Computação da Universidade Federal de Uberlândia, como parte dos requisitos exigidos para a obtenção título de Bacharel em Sistemas de Informação.

Orientador: Prof.ª Fernanda Maria da Cunha Santos

Universidade Federal de Uberlândia – UFU

Faculdade de Computação

Bacharelado em Sistemas de Informação

Uberlândia, Brasil

2024

Felipe Harrison Silva Cantarino

Criação de um Corpus Português para Auxiliar a Identificação de Notícias Verdadeiras e Falsas

Trabalho de conclusão de curso apresentado à Faculdade de Computação da Universidade Federal de Uberlândia, como parte dos requisitos exigidos para a obtenção título de Bacharel em Sistemas de Informação.

Prof.ª Fernanda Maria da Cunha Santos
Orientador

Rodrigo Sanches Miani
Professor

Murillo Guimarães Carneiro
Professor

Uberlândia, Brasil
2024

Dedico aos meus pais, Antonia de Lourdes da Silva Cantarino e Marcelo Rosa Cantarino e minha irmã Gabriela Sthefany Silva Cantarino, agradeço pelo incentivo aos estudos e suporte para realização da graduação.

Agradecimentos

Agradeço primeiramente aos meus familiares, pelo apoio durante a graduação, em todos os meus estudos e na vida, desde sempre.

Agradeço também a todos os professores que contribuíram na minha jornada de aprendizagem, desde a infância até a graduação.

Dedico também um agradecimento especial à minha orientadora, a Professora Fernanda Maria da Cunha Santos, que teve um papel importante na confecção do trabalho e também na escolha do tema, por meio das suas orientações. Agradeço pelo apoio, confiança e aprendizados em nossa jornada nesse trabalho.

Resumo

O trabalho aborda o desafio da detecção de notícias falsas (*fake news*) em um contexto marcado pela rápida disseminação de informações pela internet. Dessa forma, o objetivo é a criação de um novo corpus, constituído por notícias textuais, atuais, extraídas de sites da Internet e escritas na linguagem portuguesa do Brasil. Cada texto será classificado como sendo "Verdade" ou "Falso". A criação do novo corpus, denominado BoatosBr, contou com o auxílio de *Web-Crawlers*, responsáveis por obter textos de boatos já classificados. A etapa de validação do corpus BoatosBr foi formalizada pelos testes de um modelo computacional formado por técnicas de PLN e por algoritmos de classificação de AM. Os algoritmos implementados foram *Naive Bayes*, *Support Vector Machine* e *Random Forest*. Os resultados obtidos indicaram que o novo corpus é capaz de viabilizar diferentes estudos, podendo ser preciso para ser testado por sistemas computacionais automatizados pelo AM para detectar *fake news*.

Palavras-chave: *fake news*, corpus, Processamento de Linguagem Natural, *crawler*

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

AM	Aprendizado de Máquina
CSS	*Cascading Style Sheets*
HTML	*HyperText Markup Language*
IA	Inteligência Artificial
MIT	*Massachusetts Institute of Technology*
NLP	*Natural Language Process*
PLN	*Processamento de Linguagem Natural*
SVM	*Support Vector Machine*
TF-IDF	*Term Frequency – Inverse Document Frequency*

Sumário

1 Introdução

As notícias falsas, conhecidas popularmente como *fake news*, são descritas como textos contendo, intencionalmente, informações falsas e criados para manipular a opinião das pessoas sobre fatos, eventos e declarações reais (CITS, 2016). Com o crescimento da internet e das redes sociais, a forma e velocidade de comunicação foi alterada, permitindo a disseminação de informações numa escala exponencial. Dessa forma, as *fake news* ganharam cada vez mais espaço, como foi comprovado por um estudo feito pelo *Massachusetts Institute of Technology* (MIT) que revelou que notícias falsas têm 70% maiores chances de serem compartilhadas em comparação às notícias verdadeiras (OFFICE, 2018).

Consequentemente, sabe-se que as *fake news* são responsáveis por provocar desinformação sobre determinado assunto, radicalizar pensamentos e gerar pânico na sociedade. Além da desinformação, também impactam negativamente em áreas como politica, saúde, tecnologia e até mesmo na vida pessoal dos indivíduos, como, por exemplo, no caso do "Linchamento de Fabiane Maria de Jesus" em 2014 (WIKIPEDIA, 2018), que resultou na morte dessa mulher, devido às informações falsas divulgadas nas redes sociais.

Nesse contexto, saber a veracidade das informações se torna importante. No ano de 2024, a detecção de *fake news* ainda depende em sua maioria da análise manual feita por sites responsáveis por identificar e desmentir essas informações, como: Agência Lupa [1], Fato ou Fake [2], E-farsas [3] e Boatos.org [4]. Entretanto, embora existam esses sites, o trabalho manual não é suficiente para cuidar de todas as notícias, visto que as *fake news* se espalham numa escala muito maior que sua detecção. Nesse sentido, a utilização de tecnologias apresentam-se como forte aliada para auxiliar na detecção de notícias falsas.

Com base nessa ideia, uma solução para detectar *fake news*, são sistemas computacionais constituídos por algoritmos de Inteligência Artificial (IA), que possuem capacidades superiores às humanas para analisar grandes quantidades de informações de maneira rápida e eficiente.

Para o desenvolvimento de um sistema utilizando uma IA, principalmente, com algoritmos baseado em aprendizado supervisionado, uma etapa crucial é a escolha do conjunto dos dados destinados ao seu treinamento. No contexto da detecção de *fake news*, é ideal encontrar um conjunto de dados balanceado, composto por notícias verdadeiras e falsas na mesma língua, como o português. Isso garante que a eficiência do modelo computacional seja adequada para a linguagem especificada.

[1] Acesso em <https://lupa.uol.com.br/>
[2] Acesso em <https://g1.globo.com/fato-ou-fake/>
[3] Acesso em <https://www.e-farsas.com/>
[4] Acesso em <https://www.boatos.org/>

Na busca por um conjunto de dados, o trabalho Silva et al. (2020) propôs a criação de um desses conjuntos de dados e demonstrou bons resultados na detecção de *fake news*. Além desse trabalho, foram realizadas buscas em fontes de bases de dados como: Kaggle [5], Dados Gov.br [6], Github [7] e textos científicos. Por fim, foram encontrados três conjuntos de dados para treinamento, disponíveis publicamente para utilização, porém apenas dois tinham na mesma proporção textos verdadeiros quantos falsos.

1.1 Objetivo

Diante da escassez de conjuntos de dados em português, o presente trabalho propõe a criação de um novo corpus atualizado e que contenha informações falsas e verdadeiras, em mesma proporção.

1.1.1 Metodologia

O corpus proposto terá duas categorias de classificação, sendo elas "Verdadeiro" e "Falso". O objetivo será de fornecer uma base de dados nova, diversificada e atualizada para treinamento dos algoritmos de IA especializadas em detectar *fake news* em português.

A escolha das ferramentas tecnológicas para a criação de um novo corpus devem garantir que todas as etapas de construção e validação do corpus sejam realizadas. Assim, neste trabalho, serão utilizadas as seguintes tecnologias:

- Linguagem de programação Python, versão 11.3;
- Ferramentas de desenvolvimento como: Jupyter Notebook e Visual Studio Code;
- Framework Beautiful Soup 4 [8];
- Técnicas de Processamento de Linguagem Natural (PLN)
- Técnicas de Aprendizado de máquina

Após a construção do corpus, o mesmo será formatado de diferentes formas para auxiliar os outros desenvolvedores que queiram testar e explorar a base de dados. Além disso, será realizado testes sucintos com alguns algoritmos de Aprendizado de Máquina (AM) para analisar e comprovar a utilidade do corpus para o público acadêmico.

[5] Acesso em <https://www.kaggle.com/datasets>
[6] Acesso em <https://dados.gov.br/home>
[7] Acesso em <https://github.com>
[8] Disponível em <https://beautiful-soup-4.readthedocs.io/en/latest/>

1.2 Organização do trabalho

Esse trabalho está estruturado da seguinte forma: no Capítulo 2 será apresentado a fundamentação teórica necessária para o entendimento dos principais conceitos envolvidos neste estudo. O Capítulo 3 descreve outros trabalhos, com propostas que fundamentaram na construção deste. Já, as seguintes etapas do desenvolvimento da pesquisa foram detalhados no Capítulo 4. No Capítulo 5, é feita a exibição e análise dos resultados encontrados. Finalmente, no Capítulo 6, são feitas conclusões e sugestões de trabalhos futuros.

2 Fundamentação Teórica

Neste capítulo será apresentado o conteúdo teórico necessário para o desenvolvimento deste trabalho. Inicialmente, será descrito o conceito dos diferentes tipos de informações falsas presentes nas mídias digitais. Na sequência, haverá detalhes sobre as técnicas para coleta dos textos digitais e as técnicas de PLN para a manipulação e formatação dos textos, assim como, explicações sobre os algoritmos de aprendizado de máquinas implementados durante a validação do corpus.

2.1 Tipos de informações falsas

A disseminação de informações falsas contribui para a desinformação em relação a um determinado assunto ou conceito. Por consequência, sua propagação prejudica a compreensão verdadeira de temas, uma vez que pode ser apresentada de diferentes maneiras.

Segundo a pesquisa Kumar e Shah (2018), as informações falsas podem ter diferentes classificações, sendo com base na intenção do autor, categorizadas como **"má informação"** e **"desinformação"**, ou com base em seu conteúdo, distinguindo entre informações **baseadas em opiniões** e informações **baseadas em fatos**. Além dessas, o trabalho Santos (2022), observou e considerou que as informações falsas podem ser classificadas, também, como **humorísticas**. Nas seções seguintes serão detalhados os diferentes tipos de informações falsas.

2.1.1 Má informação

Corresponde às informações falsas que não foram criadas com a intenção de enganar, mas que acabam fazendo isso devido à má interpretação, falta de atenção, erros de escrita ou falta de conhecimento do autor. Como resultado, acabam divulgando más informações sobre um assunto (KUMAR; SHAH, 2018).

Um exemplo de má informação é demonstrado na Figura 1, na qual a má interpretação de uma lei provocou o surgimento e divulgação de informações falsas. Conforme detalhado em Redução de tributos sobre gás de cozinha e combustíveis [1], a nova lei propunha a redução dos impostos sobre o óleo diesel e o gás liquefeito de petróleo (GLP) nas refinarias, mas não mencionava a redução nos postos de gasolina, nem que o valor do imposto deveria ser devolvido.

[1] Disponível em <https://www.gov.br/receitafederal/pt-br/assuntos/noticias/2021/marco/reducao-de-tributos-sobre-gas-de-cozinha-e-combustiveis>

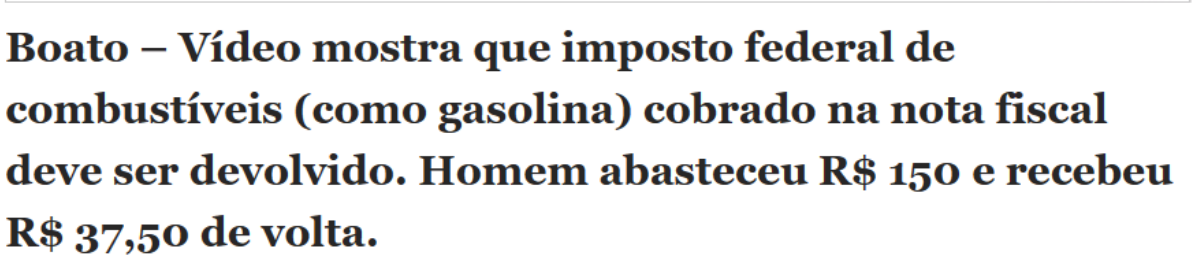


Figura 1 – Exemplo de uma má informação.

Fonte: Extraído de E-farsas [2]

2.1.2 Desinformação

As informações falsas criadas com a intenção de enganar são classificadas como desinformação, segundo o estudo Kumar e Shah (2018). Esse tipo de informação é produzido e moldado conforme as motivações do autor, sejam elas de natureza política ou econômica. Um exemplo dessa desinformação, pode ser visto na Figura 2.

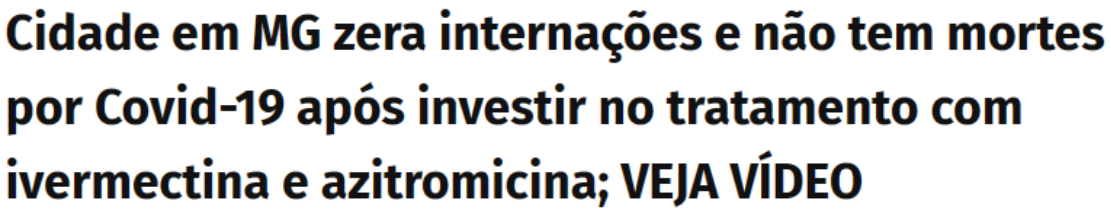


Figura 2 – O título exemplifica a desinformação.

Fonte: Extraído de claudioandreopoeta [3]

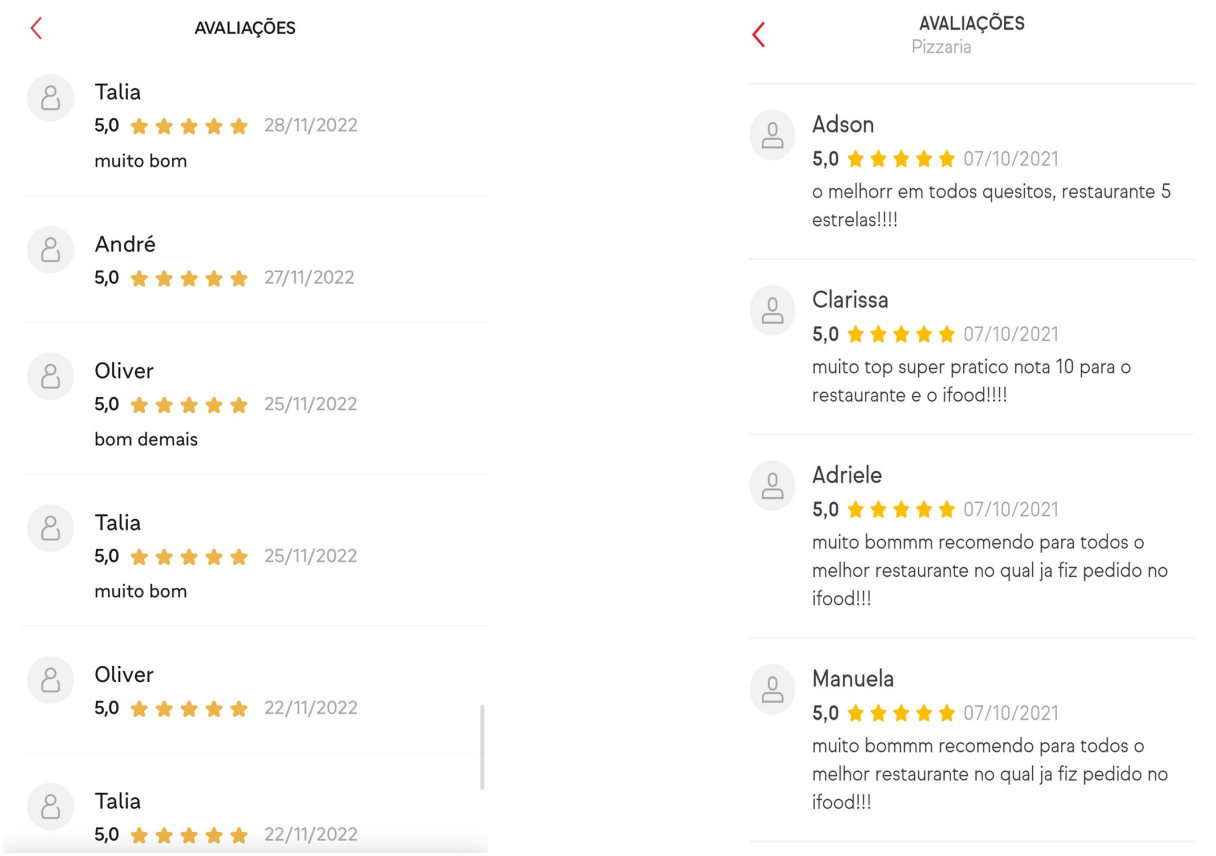
2.1.3 Baseado em opiniões

Nesse tipo de informação falsa, o conteúdo expressa uma opinião mentirosa daquela que foi dita ou escrita pelo autor, na qual não há uma base verdadeira. Dessa forma são criadas para manipular a decisão ou opinião do leitor (KUMAR; SHAH, 2018). Com a popularização do comércio eletrônico via internet, também conhecido como *e-commerce*, esse tipo de informação falsa tornou-se mais frequente, por meio de avaliações falsas de produtos, as *fake reviews*. Esta estratégia busca impulsionar as vendas, uma vez que a presença de avaliações positivas por parte de outros consumidores tendem a incentivar os potenciais compradores. Entretanto, quando essa avaliação é falsa, pode levar a uma

[2] Disponível em: <https://www.e-farsas.com/posto-de-gasolina-devolve-dinheiro-do-imposto-para-cliente-sera-verdade.html>

[3] Disponível em: <https://www.claudioandreopoeta.com.br/2021/03/cidade-em-mg-zera-internacoes-e-nao-tem.html>

decepção do consumidor, que acaba adquirindo um produto que não atende suas necessidades ou, em situações extremas, não recebe o produto adquirido. As imagens da Figura 3 são exemplos de *fake reviews*.



(a) Exemplo de avaliação falsa com usuários repetidos.

(b) Exemplo de avaliação falsa com avaliações repetidas.

Figura 3 – Exemplos de *fake reviews*.

Fonte: Adaptado pelo Autor.

2.1.4 Baseado em fatos

Corresponde as informações falsas que buscam contradizer, confundir ou descontextualizar uma informação verdadeira. São criadas para que o leitor tenha dificuldade de distinguir entre o conteúdo real e o falso, fazendo com que muitos acreditem no conteúdo falso. Esse é o tipo mais comum de informação falsa, pois nele engloba as *fake news* e os rumores ou boatos. Nas figuras 4 e 5 são apresentados exemplos de *fake news*.

Boato é uma declaração cuja veracidade não é rapidamente ou nunca foi confirmada, passada de pessoa para pessoa. Na maioria das vezes, os boatos, tem conotações negativas, mas não são bons nem ruins (UNHCR, 2022).

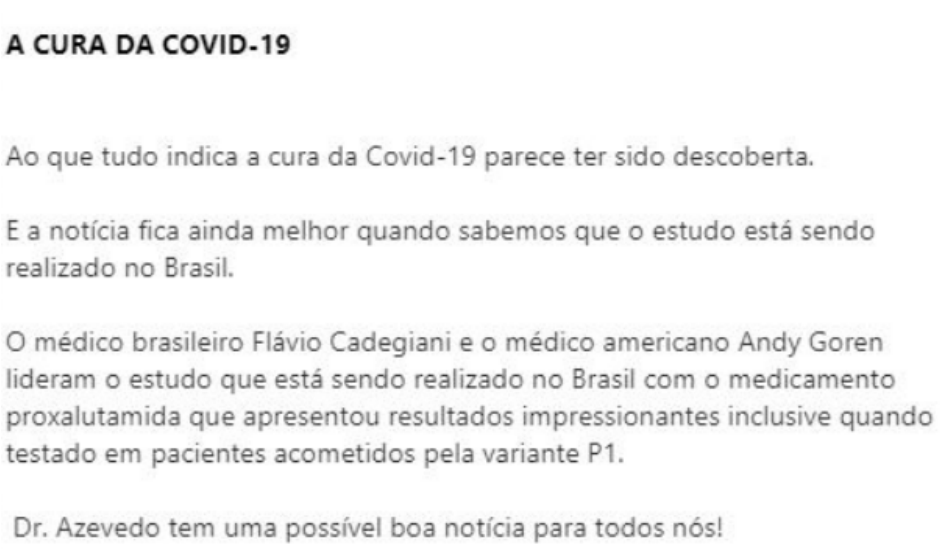


Figura 4 – Exemplo de informação falsa baseada em fato.

Fonte: Extraído de Whatsapp. [4]

Figura 5 – Exemplo de informação falsa divulgada em mensagem.

Fonte: Extraído de Whatsapp. [5]

[4] Disponível em: <https://www.e-farsas.com/wp-content/uploads/cura-covid-fake-1.jpg>
[5] Disponível em: <https://www.e-farsas.com/wp-content/uploads/vacinacao-ministerio.jpg>

2.1.5 Humorística

As informações humorísticas, como paródias e sátiras, não são criadas com a intenção de enganar o leitor, mas sim para proporcionar entretenimento (SANTOS et al., 2020). No entanto, essas formas de humor podem levar o leitor a acreditar em informações falsas, pois apresenta informações fictícias que se referem a elementos da realidade.

Frequentemente, conteúdos humorísticos são divulgados como se fossem verdadeiros, em vez de ser apenas entretenimento (RUBIN et al., 2016), conforme demonstrado na figura 6. Isso ocorre, muitas das vezes, devido à facilidade de compartilhamento, que é mais simples do que verificar a origem da informação, um processo que demanda mais tempo e esforço.

Figura 6 – Exemplo de notícia humorística divulgada como verdade.

Fonte: Adaptado de E-farsas. [6]

2.2 Web Scraping

Técnica de extração de dados na internet, que envolve a captura e o armazenamento de informações para realizar análises, posteriormente (ZHAO, 2017). Esta abordagem permite a coleta de grande quantidade de dados de maneira rápida e padronizada. Dessa forma, é um recurso muito utilizado por programas que necessitam consultar páginas da internet, frequentemente, para acompanhar atualizações de dados ou para coletar muitas informações. Esta técnica passou por aprimoramentos com o passar dos anos, e em 2024, é possível encontrar uma diversidade de ferramentas especializadas, tornando a coleta de dados mais acessível e eficiente.

Um recurso utilizado na extração de dados são os chamados *crawlers*, que correspondem a programas responsáveis por explorar páginas da internet para possibilitar a coleta de informações. Esses programas percorrem várias páginas da web, partindo de uma url inicial, presente em uma lista com páginas a serem visitadas. Conforme estas páginas são acessadas, essa lista é expandida com novas URLs encontradas (KAUSAR; DHAKA; SINGH, 2013). Esse recurso coleta as informações das páginas através de arquivos em *HyperText Markup Language* (HTML), localizando os dados desejados a partir de *tags* de marcação HTML, seletores de *Cascading Style Sheets* (CSS) e *XPath* do elemento.

[6] Disponível em: <https://www.e-farsas.com/wp-content/uploads/Guedes-Viagra.jpg>

Essa técnica tem um papel importante neste trabalho, por possibilitar a coleta dos textos para confecção do corpus.

2.3 Inteligência Artificial

O termo refere-se a uma área da informática que se dedica ao desenvolvimento de sistemas e algoritmos capazes de executar tarefas que se assemelham à inteligência humana, tais como classificação, aprendizado a partir de dados, identificação de padrões e resolução de problemas (COPELAND, 2023). Essa tecnologia demonstra a capacidade de solucionar problemas complexos numa velocidade e competência superior à humana, demonstrando ser um ótimo recurso para realização de identificação de notícias falsas.

A pesquisa em IA iniciou por volta da década de 1950, com contribuições de Alan Turing. Inicialmente, se projetava, de forma teórica, sistemas inteligentes para aplicações como xadrez e resolução de tarefas simples. O primeiro projeto prático de IA, desenvolvido por Christopher Strachey, possuía capacidade de jogar partidas completas de damas (COPELAND, 2023).

Com a expansão e exploração de estudos inovadores, as subáreas da IA adquiriram crescentes capacidades na resolução de tarefas extremamente complexas. Em 2024, é possível encontrar uma ampla atuação de algoritmos de IA em tarefas e atividades rotineiras da população. Alguns exemplos dessas aplicações e suas funcionalidades:

- Compreensão e resposta a perguntas, como o ChatGPT; [7]
- Criação de imagens com base em texto, como Midjourney; [8]
- Criação de vídeos, como Steve.ai; [9]
- Análise e avaliação de raios-x, na saúde, como o HarpiAI; [10]
- Aprendizagem adaptativa em aplicativos educacionais, como o Mondly. [11]

Dentre os exemplos acima citados, este trabalho teve o estudo direcionado para as técnicas de Processamento de Linguagem Natural (PLN) aliada aos algoritmos de AM, os quais atuarão como classificadores dos trechos digitais.

[7] Disponível em <https://chat.openai.com/>
[8] Disponível em <https://www.midjourney.com/>
[9] Disponível em <https://www.steve.ai/>
[10] Disponível em <https://harpiai.com>
[11] Disponível em <https://www.mondly.com/vr>

2.4 Modelos de aprendizagem de máquina

Modelos de Aprendizagem de Máquina (AM) correspondem a algoritmos que utilizam técnicas matemáticas e estatísticas, para aprender a partir de um conjunto de dados e, consequentemente, resolver problemas de classificação e regressão, o que seriam muito difíceis de serem solucionados por softwares tradicionais de determinados paradigmas (GOODFELLOW; BENGIO; COURVILLE, 2016).

Dessa forma, existem alguns modelos de aprendizagem que incluem os algoritmos: *Naive Bayes*, *Support Vector Machine (SVM)* e *Random Forest*, que também serão utilizados no presente trabalho, por serem conhecidos e utilizados em outros trabalhos relacionados. As características mais importantes de cada algoritmo são descritas nas subseções a seguir.

2.4.1 *Naive Bayes*

O Algoritmo de *Naive Bayes* tem como base o teorema probabilístico de Bayes, que consiste em avaliar a probabilidade de determinado evento ocorrer com base em outros eventos ocorridos (IBM, 2024). No contexto de treinamento de IA, esse algoritmo fornece uma maneira de calcular a probabilidade de um objeto ser de uma classe com base no comportamento de outras classes/eventos já avaliados. Logo, um objeto é classificado num dos rótulos disponíveis, baseando-se nas probabilidades calculadas, segundo a equação apresentada na Figura 7.

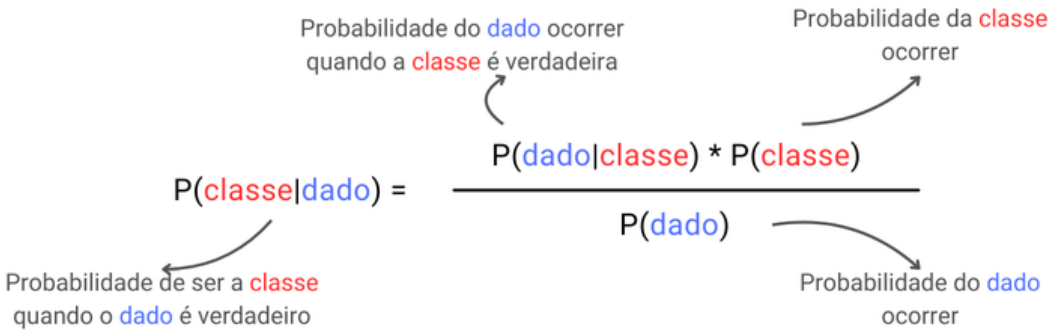


Figura 7 – Equação de *Naive bayes*

Elaborada pelo autor

Dessa forma, o algoritmo *Naive Bayes* possibilita separar as palavras dos textos e classificá-los segundo os rótulos identificados, mesmo apresentando uma disposição como a exibida na Figura 8.

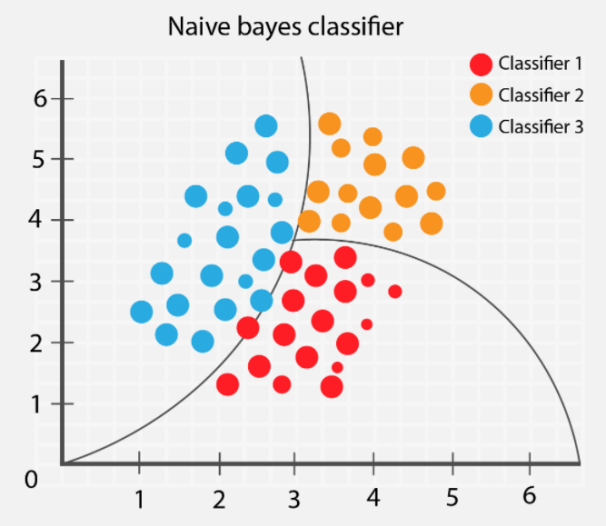


Figura 8 – Exemplo de classificação de dados com algortimo *Naive bayes*

Fonte: Extraído de Linkedin [12]

2.4.2 *Support Vector Machine (SVM)*

O algoritmo *Support Vector Machine* (SVM), busca em seu processo, a criação de um hiperplano capaz de separar os dados em diferentes classes, ou, em rótulos. Esse algoritmo funciona, reduzindo a distância entre os pontos mais distantes entre as classes, chamados de pontos de vetor, para conseguir encontrar pontos próximos que permitam a classificação correta de um novo item (VINICIUSSEGATTO, 2023). Um exemplo dessa separação por hiperplano pode ser vista na Figura 9.

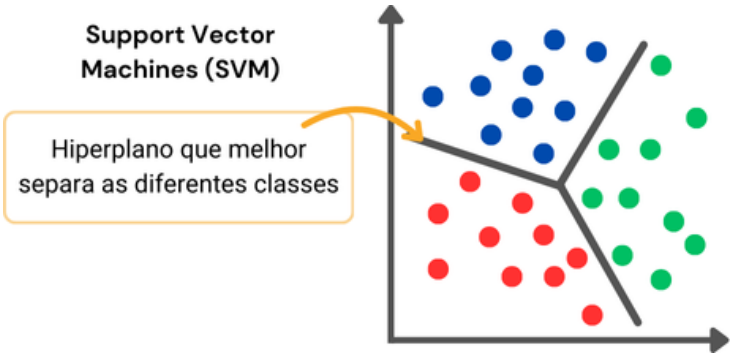


Figura 9 – Exemplo de classificação de dados com o algortimo SVM.

Fonte: Adaptado de Spot Intelligence [13]

[12] Disponível em: <https://www.linkedin.com/pulse/naive-bayes-classifier-chandrahasa-sreeramaneni/>

[13] Disponível em: <https://i0.wp.com/spotintelligence.com/wp-content/uploads/2024/05/support-vector-machine-svm.jpg?>

2.4.3 *Random Forest*

O algoritmo *Random Forest* opera pela construção de uma "floresta" de árvores de decisão, que são modelos responsáveis por filtrar o conjunto de dados em cada entroncamento ou nível da árvore com base nos atributos, até alcançar uma classificação ou rotulação final. Desse modo, durante o processo de classificação, cada árvore fornece uma previsão para o novo dado. A decisão final é determinada pela contagem das previsões individuais, sendo definida pela classificação que recebeu a maior quantidade de votos entre as árvores, conforme demonstrado na Figura 10.

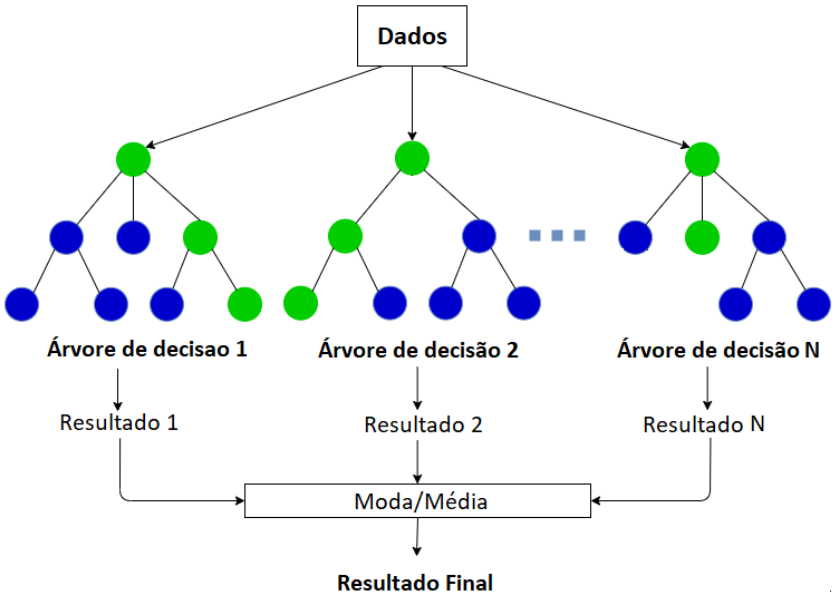


Figura 10 – Exemplo classificação de dado com algoritmo *Random Forest*

Fonte: Extraído de statplace [14]

2.5 Processamento de Linguagem Natural

É o termo que corresponde a técnica de permitir que computadores consigam compreender e trabalhar com a linguagem utilizada pelos seres humanos (IBM, 2023). Essa abordagem envolve técnicas de manipulação do texto, que irão permitir os sistemas computacionais fazerem a interpretação esperada para o problema em questão. Essa reformulação é realizada por meio de técnicas como o *stemming* e o *lemmatization*, que reduzem as palavras à sua forma base, e a tokenização, que separa as sentenças em grupos de palavras. Dessa forma, possibilita uma melhor compreensão e manipulação da linguagem pelos softwares.

[14] Disponível em: <https://statplace.com.br/blog/modelos-de-machine-learning-utilizando-o-pacote-caret/>

No mundo moderno, esta técnica é muito utilizado em conjunto com as IA, uma vez que, permitem o aprendizado pela extração e compreensão das enormes quantidades de dados processados. Dessa forma, esta abordagem é utilizado no desenvolvimento de *chatbots*, detecção de *spam*, tradução de texto e interação humano-computador por meio de texto.

2.5.1 Técnicas de Tokenização

As técnicas de tokenização dividem o texto em partes menores, chamados tokens, para melhor ser analisado pelos sistemas computacionais (IBM, 2023). Dentre as técnicas de tokenização, para este trabalho, incluem os algoritmos *Term frequency – Inverse Document Frequency* (TF-IDF) e *Word2Vec*. As características mais importantes de cada técnica são descritas a seguir.

2.5.1.1 *TF-IDF (Term frequency – Inverse Document Frequency)*

É uma ponderação de termos na qual é associado para cada termo de um documento um peso que quantifica a importância deste termo na descrição do conteúdo do documento, permitindo fornecer pesos aos documentos de uma coleção. Esta ponderação é obtida pelo produto da ponderação TF com a ponderação IDF, onde:

- **TF**: o peso é determinado pela alta frequência de um termo no documento, ou seja, documentos que possuem termos que repetem muitas vezes tem maior valor de TF.
- **IDF**: o peso é determinado pela baixa frequência de um termo na coleção de documentos, em outras palavras, quanto menos um termo aparece em todos os documentos, maior será seu valor de IDF.

Nesse sentido, em resumo, documentos com termos mais frequentes e mais raros dentro da coleção possuem um valor de TF-IDF maior.

2.5.1.2 *Word2Vec*

É uma técnica de Processamento de Linguagem Natural que transforma palavras em vetores de números de forma que as palavras relacionadas semanticamente fiquem próximas umas das outras. Essa técnica funciona treinando uma rede neural que prevê o contexto de palavras numa frase, interpretando as relações semânticas entre as palavras.

Uma representação dessa técnica pode ser analisada na Figura 11. Nela, é possível notar, pelas cores definidas para cada palavra, que simulam o vetor de números, a similaridade entre as palavras *"woman"* e *"man"* quando comparadas com a palavra *"king"*, onde é possível observar a quantidade de cores semelhantes entre essas palavras.

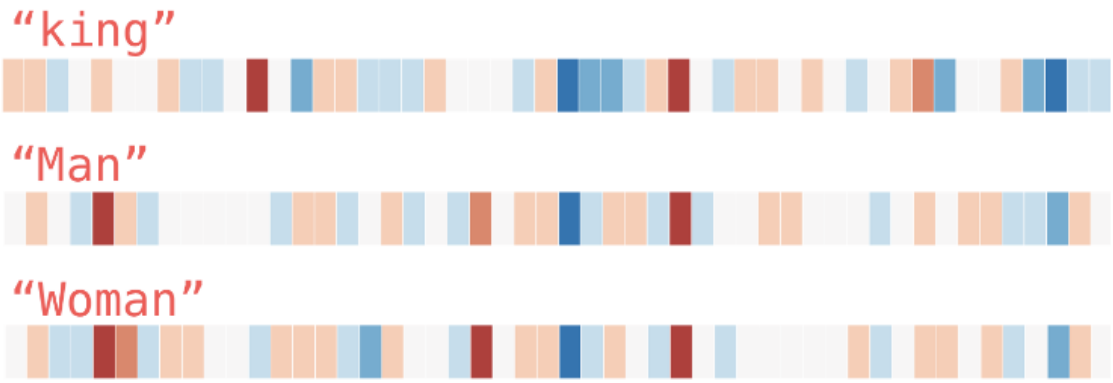


Figura 11 – Representação visual da comparação entre as palavras transformadas pelo Word2Vec.

Fonte: Extraído de https://jalammar.github.io [15]

[15] Disponível em: <https://blogdozouza.wordpress.com/2019/03/29/o-word2vec-ilustrado/>

3 Trabalhos Relacionados

Nesta seção, serão apresentados alguns trabalhos que direcionaram seus estudos à criação de *corpus* e à aplicação de algoritmos AM, com o foco na detecção de notícias falsas. Existem outros estudos relacionados ao tema proposto, além de bases de dados privadas, entretanto, os trabalhos apresentados a seguir compartilham semelhanças com o objetivo deste estudo e as bases de dados estão disponíveis publicamente para análise e utilização.

A tese de doutorado **"Detecção automática de notícias falsas em português"** (SANTOS, 2022) e o artigo Santos et al. (2020) apresentaram o primeiro corpus de referência na área de detecção de notícias falsas na língua português do Brasil, denominado Fake.Br. Esse corpus é composto por 7200 textos obtidos do período de 2016 a 2018, e sua proporção de notícias verdadeiras e falsas podem ser vista na Figura 12. Para a construção do corpus Fake.Br foi feita uma coleta manual de notícias falsas em sites que continham notícias que eram totalmente falsas. A escolha destes sites seguiu a ferramenta Monitor do Debate Político no Meio Digital [1] que informa as características tanto de layout quanto de conteúdo que podem definir um local que reproduza conteúdo falso.

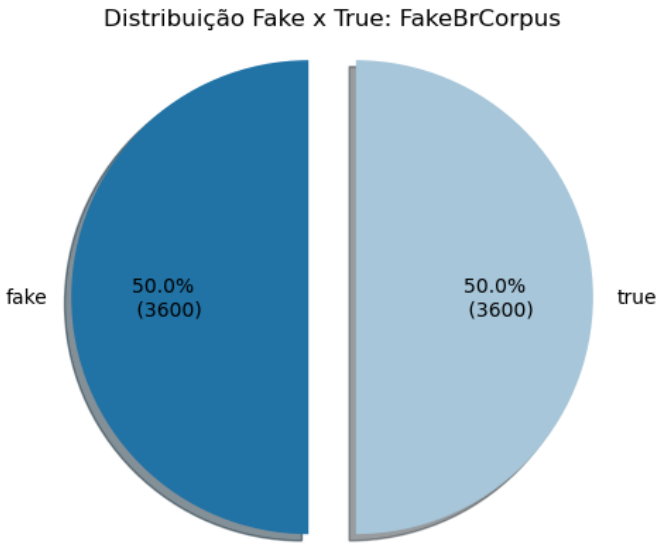


Figura 12 – Percentual de textos falsos e verdadeiros no corpus Fake.Br.

Fonte: Elaborada pelo autor.

Os principais assuntos das notícias desta base de dados foram divididas nas categorias: política, TV e celebridades, sociedade e cotidiano, ciência e tecnologia, economia,

[1] Disponível em <https://www.monitordigital.org>

e religião. A proporção de textos, em cada categoria, pode ser vista na Figura 13.

Categoria	Número de exemplos	%
Política	4.180	58,0
TV e celebridades	1.544	21,4
Sociedade e cotidiano	1.276	17,7
Ciência e tecnologia	112	1,5
Economia	44	0,7
Religião	44	0,7

Figura 13 – Proporção de textos em cada categoria no corpus Fake.Br.

Fonte: Extráida de Santos (2022)

Com base na coleta inicial, as notícias verdadeiras foram obtidas de maneira semi-automática, por meio de *web-crawlers* utilizando os termos em destaque e repetitivos presentes nos conteúdos falsos. Então, para cada notícia falsa, foi aplicada uma medida de similaridade lexical entre eles e os textos obtidos pelo *web-crawlers*. A medida de proximidade escolhida foi a similaridade do cosseno, cujo valor é um número no intervalo [0-1], em que o valor 0 indica que os textos são completamente diferentes e o valor 1 indica que os textos são completamente similares. A partir desse valor, escolheu-se a notícia verdadeira que tinha o maior valor para garantir que as notícias falsas e verdadeiras estivessem de fato relacionadas ao mesmo assunto.

Na metodologia proposta em Santos (2022), destacou três abordagens textuais para análise de uma notícia: abordagens baseadas em atributos linguísticos; abordagens baseadas em conteúdo e abordagens baseadas na estrutura do texto. Para a validação e classificação das notícias, aplicou diferentes técnicas de AM sob cada uma das abordagens: ávores de decisão, Naive Bayes, *Randon Forest*, *K-Nearest Neighbors* (KNN), *Support Vector Machines* (SVM) e redes neurais profundas, com as arquiteturas *Long Short-Term Memory* (LSTM) e *Bidirectional Encoder Representations from Transformers* (BERT).

O trabalho ***FakeRecogna: A New Brazilian Corpus for FakeNews Detection*** Garcia, Afonso e Papa (2022) tem o objetivo de criar um novo corpus brasileiro atualizado e classificá-lo em verdadeiro ou falso. De modo a alcançar este objetivo, utilizou-se da coleta de notícias falsas e verdadeiras com técnicas de mineração de dados por meio de *web-crawlers*. Com base nessa coleta, os textos obtidos forma pre-processados utilizando técnicas de PLN. Como resultado, foi obtido um conjunto de dados maior e mais atualizado que o *FakeBrCorpus*, composto por 11902 textos abrangendo o período de 2019 a 2021. A proporção quanto a quantidade de textos verdadeiros e falsos pode ser vista na Figura 14. Para avaliar a base de dados, foram implementados e testados os algoritmos de AM: Naive Bayes, Optimum-Path Forest, SVM e uma rede neural convolucional.

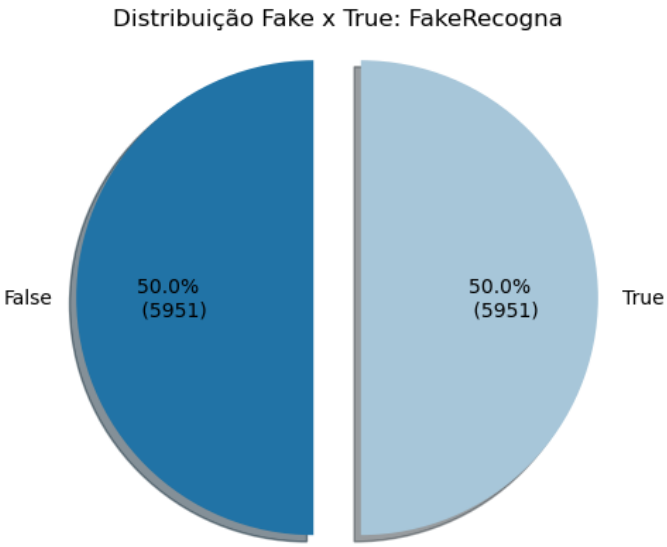


Figura 14 – Percentual de textos falsos e verdadeiros no corpus FakeRecogna.

Fonte: Elaborada pelo autor.

Também é relevante mencionar o projeto americano ***PolitiFact*** Politifact (2018), uma organização de notícias nacional sem fins lucrativos, criada em 2007, com foco na verificação de fatos. O projeto é composto por uma equipe de editores e jornalistas responsáveis por avaliar e julgar notícias, deixando de lado suas próprias opiniões, para apresentar os fatos de maneira imparcial. O *PolitiFact* utiliza uma abordagem diferente de classificação das informações, oferecendo seis classificações possíveis, em ordem decrescente de veracidade. São elas: *True* (2149), *Mostly True* (2676), *Half True* (2765), *Mostly False* (2539) e *False* (2601).

4 Desenvolvimento

Nesse capitulo será dado uma visão geral sobre as etapas tomadas para o desenvolvimento do trabalho.

4.1 Descrição geral do trabalho

O trabalho foi dividido e realizado nas seguintes etapas:

- **Coleta dos dados para composição da base:** Extração dos textos falsos das fontes de checagem dos fatos e os textos verdadeiros de outras fontes.
- **Classificação dos dados obtidos:** Análise dos textos coletados e classificação em "Verdade" ou "Falso".
- **Modelagem da base de dados usando técnicas de PLN:** Limpeza e formatação dos textos para auxiliar na utilização da base de dados.
- **Análise e complementação do corpus:** Análise dos dados obtidos e extração de mais informações para complementar a base.
- **Validação da base de dados:** Realização de teste sucintos com algoritmos de Aprendizado de máquina, para comprovar a utilidade da base.

Nas seções a seguir serão explicadas com mais detalhes cada uma das etapas.

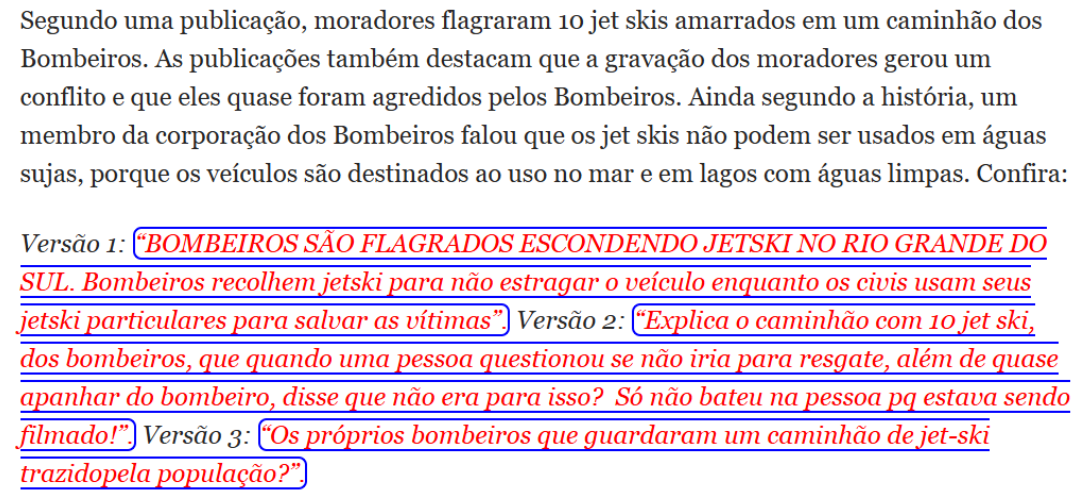
4.2 Coleta dos dados

Na primeira etapa do método proposto, foi realizado a coleta de textos para composição do corpus, através de ferramentas para mineração de dados, por meio da utilização de *web-crawlers*.

Para obter as informações falsas, foram escolhidos sites responsáveis por identificar e desmentir *fake news* mencionados no capítulo 1. Dessa forma, inicialmente, foi realizado uma análise manual e visual nesses sites, para verificar a composição dos textos e identificar padrões, tanto na escrita quanto na estrutura do site. Essa etapa é importante, pois facilitará a coleta de dados pelo *web-crawler*, uma vez que esse recurso coleta informações de sites a partir do arquivo HTML.

Após a análise desses sites, foi observado que a maioria dos conteúdos dos sites apenas apresentavam a checagem das notícias, não fornecendo o texto original do boato

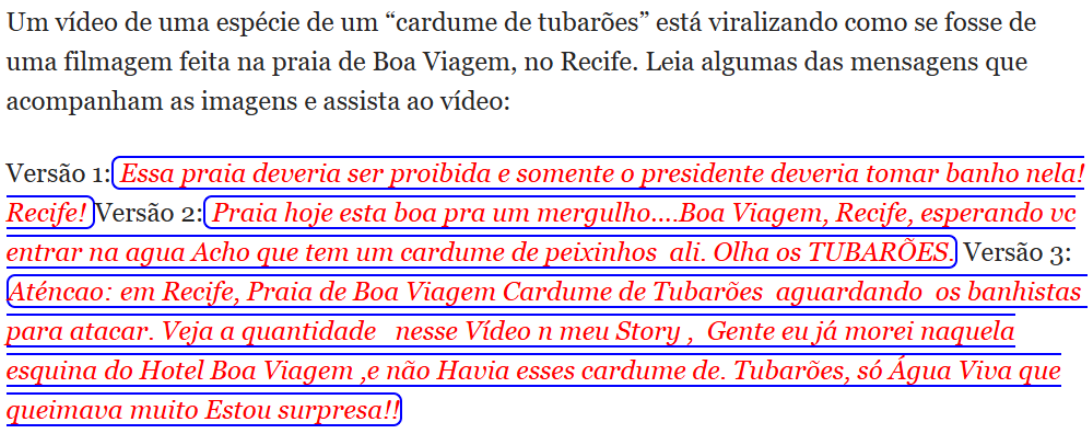
que estava circulando na internet, o que dificulta o acesso a informações falsas. Entretanto, entre as opções, o site Boatos.org [1]era o único que disponibilizava o texto original da *fake news*, em formato de texto e padronizado, como pode ser visto nas Figuras 15 e 16.



Segundo uma publicação, moradores flagraram 10 jet skis amarrados em um caminhão dos Bombeiros. As publicações também destacam que a gravação dos moradores gerou um conflito e que eles quase foram agredidos pelos Bombeiros. Ainda segundo a história, um membro da corporação dos Bombeiros falou que os jet skis não podem ser usados em águas sujas, porque os veículos são destinados ao uso no mar e em lagos com águas limpas. Confira:

Versão 1: *"BOMBEIROS SÃO FLAGRADOS ESCONDENDO JETSKI NO RIO GRANDE DO SUL. Bombeiros recolhem jetski para não estragar o veículo enquanto os civis usam seus jetski particulares para salvar as vítimas".* *Versão 2:* *"Explica o caminhão com 10 jet ski, dos bombeiros, que quando uma pessoa questionou se não iria para resgate, além de quase apanhar do bombeiro, disse que não era para isso? Só não bateu na pessoa pq estava sendo filmado!".* *Versão 3:* *"Os próprios bombeiros que guardaram um caminhão de jet-ski trazidopela população?".*

Figura 15 – Primeiro exemplo de padronização na estrutura (circulado em azul). Extraído de Boatos.org [2]



Um vídeo de uma espécie de um "cardume de tubarões" está viralizando como se fosse de uma filmagem feita na praia de Boa Viagem, no Recife. Leia algumas das mensagens que acompanham as imagens e assista ao vídeo:

Versão 1:*Essa praia deveria ser proibida e somente o presidente deveria tomar banho nela! Recife!* Versão 2:*Praia hoje esta boa pra um mergulho....Boa Viagem, Recife, esperando vc entrar na agua Acho que tem um cardume de peixinhos ali. Olha os TUBARÕES.* Versão 3: *Aténcao: em Recife, Praia de Boa Viagem Cardume de Tubarões aguardando os banhistas para atacar. Veja a quantidade nesse Vídeo n meu Story , Gente eu já morei naquela esquina do Hotel Boa Viagem ,e não Havia esses cardume de. Tubarões, só Água Viva que queimava muito Estou surpresa!!*

Figura 16 – Segundo exemplo de padronização na estrutura (circulado em azul). Extraído de Boatos.org [3]

Dessa forma, foi decidido a utilização do site Boatos.org para coletar notícias, pois possuia o texto original do boato e seguia um padrão na estrutura do site, facilitando a busca da informação pelo *web-crawler*.

[1] Acesso em <https://www.boatos.org>

[2] Disponível em <https://www.boatos.org/brasil/bombeiros-do-rio-grande-do-sul-escondem-jet-skis-e-nao-usam-em-resgates-nas-enchentes.html>

[3] Disponível em: <https://www.boatos.org/brasil/video-mostra-cardume-tubaroes-praia-boa-viagem-no-recife.html>

Quanto às notícias verdadeiras, optou-se pelo site G1 [4], por ser um dos portais de notícias mais conhecidos no Brasil, garantindo assim sua confiabilidade para notícias verdadeiras.

Após definir a fonte de dados, foi implementado um código em linguagem Python utilizando a biblioteca "Beautiful Soup 4" para criar um *crawler*, responsável por acessar os sites e coletar as informações necessárias. A lógica utilizada para a coleta de dados nesta etapa, pode ser visualizado no diagrama apresentado na Figura 17.

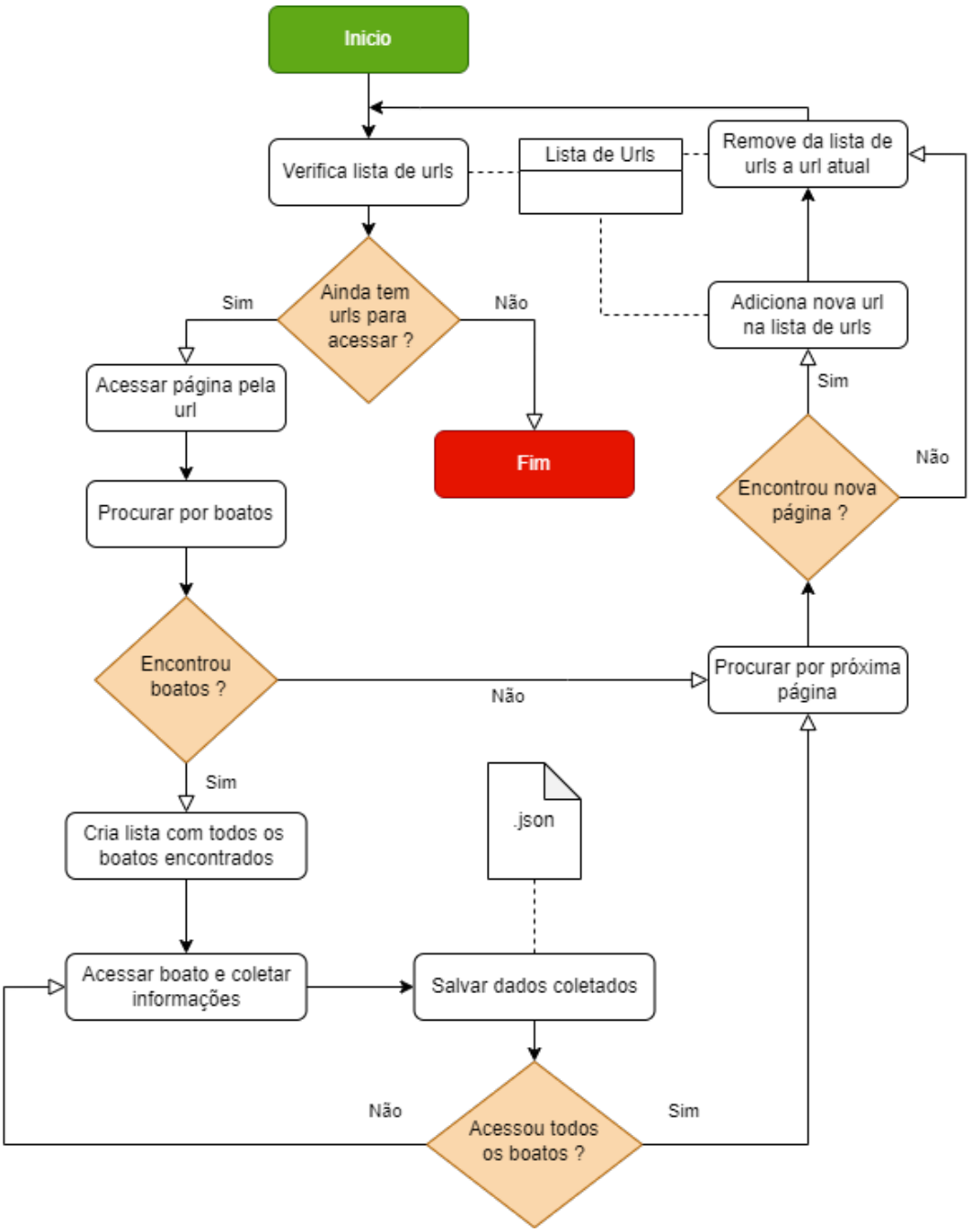


Figura 17 – Diagrama de fluxo da coleta de informações pelo *crawler*.

Fonte: Elaborada pelo autor.

[4] Acesso em <https://g1.globo.com/>

Com base no fluxo apresentado no diagrama, são fornecidos, inicialmente, *urls* iniciais que serão acessadas pelo *crawler* para buscar por boatos, por meio dos seletores CSS obtidos da análise manual realizada no site, dessa forma, essas *urls* serão acessados para extração dos dados e confecção da base.

Dos boatos encontrados, são armazenados as seguintes informações:

- *url* para acesso;
- data de publicação;
- nome do site;
- categorias, como, por exemplo, política, saúde, mundo, entre outras;
- texto original do boato que está circulando na internet.

Essas informações, são salvas num arquivo *.json* para análise. Após o processo de coleta de informações das urls, o *crawler* busca na página e extrai novas *urls* para acesso, que são adicionadas a lista de *urls* iniciais, enquanto a url já acessada é removida. O processo se repete até não ter mais urls na lista, para o crawler acessar.

4.3 Classificação dos boatos

Na segunda etapa da metodologia proposta, os textos que foram coletados na etapa anterior serão rotulados entre "Verdade" ou "Falso". Essa classificação é realizada de forma semi manual, uma vez que, os textos extraídos já possuíam uma pré-classificação com base no seu site de origem. Sendo assim, os textos extraídos do portal de notícia, G1, foram classificados como "Verdade", no entanto, os textos do site de checagem não poderiam ser todos considerados como "Falso", uma vez que utilizam várias classificações, como descrito no site Boatos.org (2024):

> "Na última atualização da nossa metodologia, criamos novas classificações de conteúdos. Elas são mostradas em tarjas ao final de todos os textos e são as seguintes: "Fake news", "Boato sem comprovação", "Golpe", "Enganoso", "Exagerado", "Verdadeiro", "Real com erros" e "Em apuração"."

Dessa forma, a estratégia utilizada para realizar a classificação foi de criar, a partir de uma análise manual dos textos, uma lista de frases para cada classificação ("Verdade" ou "Falso"), como podem ser visto na Figura 18. A lógica de utilização dessas listas consistiu em verificar se o texto possui pelo menos uma das frases da lista, caso possua, recebeu a classificação correspondente a lista.

```
FRASES_FALSO = [
    "Fake news",
    "fake news",
    "Boato sem comprovação",
    "Golpe",
    "Enganoso",
    "Exagerado",
    "é falso",
    "é falsa",
    "não é verdade",
    "não procede",
    "não há provas"
]

FRASES_VERDADE = [
    "Verdadeiro",
    "Real com erros",
    "É fato",
]
```

Figura 18 – Imagem das listas para classificação de textos em falsos ou verdadeiros.

Fonte: Elaborada pelo autor.

Observando a imagem, a quantidade de frases na lista rotulada como “Verdade” foi menor. Consequentemente, houve uma maior quantidade de frases para a classificação como “Falso”, demonstrando assim, a maior frequência de *fake news* em comparação a boatos verdadeiros nos sites de checagem.

4.4 Normalização do corpus

Na terceira etapa foi realizada a formatação dos textos coletados, de modo a auxiliar outros desenvolvedores que forem explorar essa base de dados, pois a mesma será disponibilizada ao público. Sob o conjunto de dados foi utilizado a remoção das *stopwords*, uma técnica de PLN. Ela consiste na remoção de palavras irrelevantes para a análise do texto, como, por exemplo: ‘o’, ‘a’, ‘os’, ‘as’, ‘de’, ‘para’, ‘com’, ‘sem’, ‘em’, ‘um’, ‘uma’ e entre outras. Para facilitar esse processo, foi utilizado a biblioteca NLTK da linguagem *Python*, que possui pré-definido um conjunto de *stopwords* em português.

4.5 Análise e complementação do corpus

Com o intuito de fornecer um corpus com metadados, foi elaborado uma análise sob o conjunto de textos para extrair informações relevantes que caracterizem as duas classes que se desejam identificar e rotular. Assim, a partir da natureza das informações, foi identificado um padrão textual entre alguns boatos. Por exemplo, eram frequentemente utilizadas palavras em maiúsculas, emojis e frases sensacionalistas, como pode ser visto nas Figuras 19 e 20.

Versão 1: *MOR43S EXPEDIU UMA ORDEM DE PRISÃO PARA A DEPUTADA PAULA BELMONTE MORAES SOLICITOU SERVIÇOS DE UM HACKER PARA CHANTAGEAR TEMER SIGAM COMPARTILHEM, INDIGNEM-SE, O BRASIL FOI DOMINADO POR UMA FACÇÃO CRIMINOSA, SUPR3M4, CORRUPT4 E VINGATIVA! SOMO NÓS O POVO, QUE TEREMOS QUE AGIR* Versão 2: *Moraes ordena PRISÃO de deputada corajosa que expôs a chantagem do ministro para entrar no STF*

URGENTE URGENTÍSSIMO! HACKERS INVADEM COMPUTADORES E CELULARES DE POLÍTICOS , em todo Brasil, para vasculharem sobre a tragédia no RS com o objetivo de contabilizar as mortes e enterros de corpos de vítimas da enchente e das Tsunamis causadas pela abertura das comportas das Barragens sobre os municípios afetados. AS INFORMAÇÕES SÃO ESTARRECEDORAS!

Figura 19 – Exemplos de boatos com palavras maiúsculas destacadas em vermelho.

Fonte: Elaborada pelo autor.

Versão 1: •||*TEMPLO RELIGIÃO ÚNICA*||• ✝️•✝️•✝️•✝️•✝️•✝️•☦️•✝️•✝️ *SAUDAÇÕES FILHOS E CRIATURAS DE DEUS* 🙏 *QUE A PAZ DO DEUS TODO PODEROSO REPOUSE EM VÓS.* 🥹😊 *O ASSUNTO QUE EU TRAGO PARA VOCÊS HOJE É MUITO GRAVE,* 😳 *ENQUANTO MUITOS AINDA ESTÃO NO SONO* 😴 *É MUITA COISA A ACONTECER.* 😖 *MUITOS DE VOCÊS SABEM QUE COM A CHEGADA DO "ANTI-CRISTO" TODOS PODERÃO ADORAR UMA ÚNICA *RELIGIÃO CERTO?*

Informações importantes do WhatsApp para membros do grupo 👇
Informações virais no WhatsApp

1. ✓ **= mensagem enviada**
2. ✓✓ **= mensagem alcançada**
3. **Dois azuis =** ✔✔ **mensagem lida.**
4. **Três azuis =** ✔✔✔ **O governo tomou nota da mensagem.**
5. **Dois azuis e um vermelho** ✓✓✓ **= O governo pode tomar medidas contra você**
6. **Um azul e dois vermelhos = o governo verifica suas informações**
7. **Três** ✓✓✓ **vermelhos = O governo entrou com uma ação contra ele e ele receberá uma intimação em breve.**

Seja um cidadão responsável e compartilhe com seus amigos O importante é enviar o próximo grupo mais cedo… Compartilhe 🙏 **Benevolente** 🔅🔅🔅🔅🔅🔅🔅🔅🔅

Figura 20 – Exemplos de boatos com emojis circulados em vermelho.

Fonte: Elaborada pelo autor.

Dessa forma, uma extração de pistas linguísticas foi processada sob os textos do corpus para obter informações mais relevantes de cada texto e complementar a base de dados. Foram acrescentadas as seguintes informações:

- **Quantidade de emojis**;
- **Quantidade de palavras**;
- **Quantidade de palavras em maiúsculo**;
- **Quantidade de verbos**.

Os metadados adicionados a base de dados, a torna mais completa e fornece uma maior diversidade em sua utilização, possibilitando uma análise com mais informações.

Além de uma base mais completa, foi elaborado um código que é capaz de realimentar a base de dados, realizando desde a coleta de novos textos até essa etapa. Dessa forma, com esse código disponibilizado, é possível manter a base de dados atualizada.

4.6 Validação do corpus

Na etapa final do método proposto, foram realizados testes com algoritmos de aprendizado de máquina para verificar a eficácia da base de dados construída.

Primeiramente, para a avaliação, é importante e necessário, processar os textos para um formato que os algoritmos consigam aprender, ou seja, transformar palavras para valores numéricos. Para isso, a PLN fornece técnicas de tokenização de textos, como foi mencionada na seção 2.5.1. Para esse trabalho, foi decidido a utilização de duas técnicas: *TF-IDF* e *Word2Vec*.

Após a conversão dos textos em valores numéricos, foi realizado o treinamento dos seguintes algoritmos de aprendizagem: *Naive Bayes*, *Support Vector Machine (SVM)* e *Random Forest*. Todos os três algoritmos foram implementados usando a biblioteca *Scikit-learn* do *python*, e as características mais relevantes destes foram explicados na seção 2.4.

Os resultados obtidos pelos algoritmos de AM serão apresentados no capítulo 5, assim como uma análises detalhadas do desempenho de cada um.

5 Resultados

Neste capítulo, serão apresentados os resultados obtidos no processo de criação do novo corpus, como dados estatísticos e o desempenho originado pelos algoritmos de AM. Ademais, serão comparados os resultados do novo corpus com os resultados obtidos pelos corpus Fake.Br.

5.1 O novo Corpus: BoatosBr

Ao fim das etapas de confecção deste trabalho, foi desenvolvido um novo corpus brasileiro voltado para o treinamento de IA para a detecção de *fake news* em textos escritos na língua portuguesa do Brasil. Essa nova base de dados foi apelidada de **Corpus BoatosBr**, pois armazena boatos em português, e faz referência à principal fonte dos dados, o site de checagem Boatos.org.

Este novo corpus é composto no total por 3427 textos digitais, abrangendo boatos entre o período de 2022 a 2024. A proporção de boatos verdadeiros e falsos pode ser vista na Figura 21.

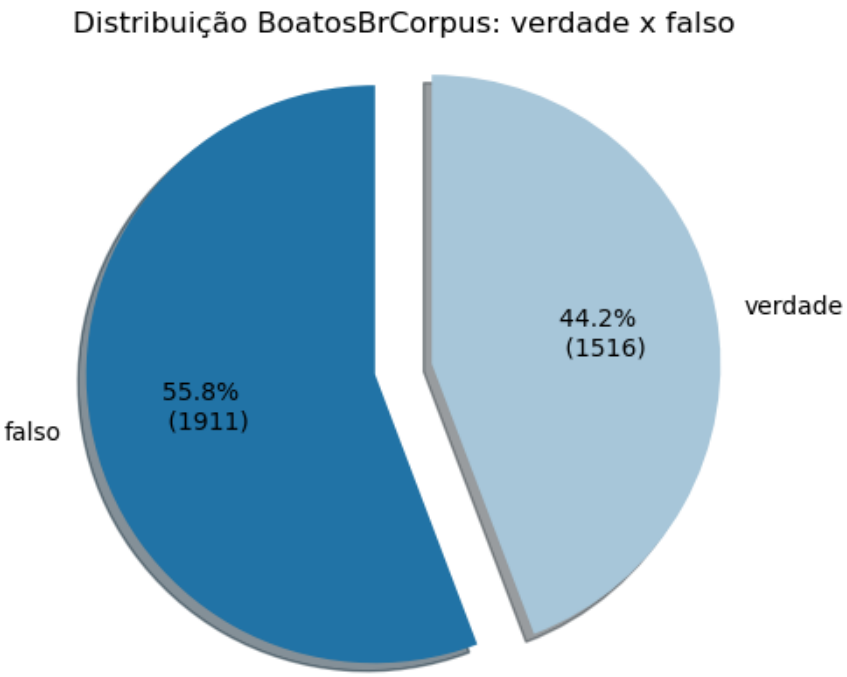


Figura 21 – Percentual de textos falsos e verdadeiros no corpus BoatosBr.

Fonte: Elaborado pelo autor.

Como pode ser visto, a base não está balanceada entre as possíveis classes, assim

como o tamanho dos textos, nos quais é possível encontrar diferença na quantidade de palavras em cada texto do corpus. A Tabela 1 mostra o quantitativo de palavras presentes nos textos.

Rótulo	Palavras no maior texto	Palavras no menor texto	Média de palavras
Verdade	5579	23	611.937
Falso	1309	6	34.242

Tabela 1 – Tabela de análise da quantidade de palavras nos textos da base

Fonte: Elaborada pelo autor.

Além disso, os textos digitais podem ser divididos em categorias, segundo seus principais assuntos: política, sociedade, saúde, mundo, ciência e tecnologia, religião e economia.

O **Corpus BoatosBr** está disponível on-line e de livre uso, podendo ser acessado pelo endereço <https://github.com/Felipe-Harrison/boatos-br-corpus>. O Corpus está dividido em três pastas principais, cada uma contendo um modelo diferente da base de dados, em formato *.json*:

- **base_simples**: Esta pasta contém a versão simples do corpus, com menos informações sobre cada texto. Será encontrado os seguintes campos:
  - **url**: URL de onde foi obtido o texto.
  - **data-publicacao**: data em que foi publicada a checagem do texto.
  - **origem**: de qual site foi retirado.
  - **categorias**: categorias do boato, por exemplo: política, saúde, mundo e entre outras.
  - **texto**: texto original que está circulando na internet.
  - **texto-normalizado**: texto normalizado e limpo, obtido na etapa 4.4.
  - **rotulo**: atributo alvo da previsão, podendo ser **'verdade'** ou **'falso'**.
- **base_completa**: Esta pasta contém a versão desenvolvida na etapa 4.5, sendo assim a mais completa do corpus, com o maior número de informações sobre cada texto. Além dos campos presentes na versão simples, encontra-se os seguintes dados adicionais:
  - **num-emojis**: quantidade de emojis
  - **num-verbos**: quantidade de verbos
  - **num-palavras**: quantidade de palavras
  - **num-palavras-maiusculas**: quantidade de palavras em maiúsculo

- **base_processada**: Esta pasta contém a mesma versão da base_completa, porém, nessa versão os campos **rotulo**, **data-publicacao**, **origem** e **categorias** foram transformados em valores numéricos.

5.2 Validação do Corpus BoatosBr

Com base nos testes descritos na Seção 4.6, foi possível avaliar a eficácia da base de dados em resolver o problema para o qual foi criada: treinar algoritmos de AM para detectar *fake news* em textos da língua portuguesa.

Na realização dos testes, foram utilizados todos os 3.427 textos da base de dados presente na pasta *base_completa* do Github. Essa base foi escolhida por possuir os textos sem as *stopwords* e por conter os metadados. Na realização dos testes de validação, foram pré-processados os textos para as fases de treinamento e de teste dos algoritmos AM. Nestas fases, o corpus é dividido em dois conjuntos: o conjunto de treinamento, quando os algoritmos de AM utilizam dos textos da base para aprender como classificar os rótulos; e o conjunto de testes, quando será realmente validado o desempenho dos algoritmos treinados. Logo, para este estudo, foi definido três percentuais para o conjunto de treinamento e de teste, respectivamente: 90% e 10%, 80% e 20% e 70% e 30%. A sequência destes percentuais serão exibidos no eixo das abscissas, caracterizando a parte destinada para teste, de todos os gráficos que seguem.

A Figura 22 mostra os resultados da acurácia, uma medida de qualidade, ao executar a técnica TF-IDF junto com os algoritmos *Naive Bayes*, SVM e *Random Forest* sob os textos do conjunto de teste.

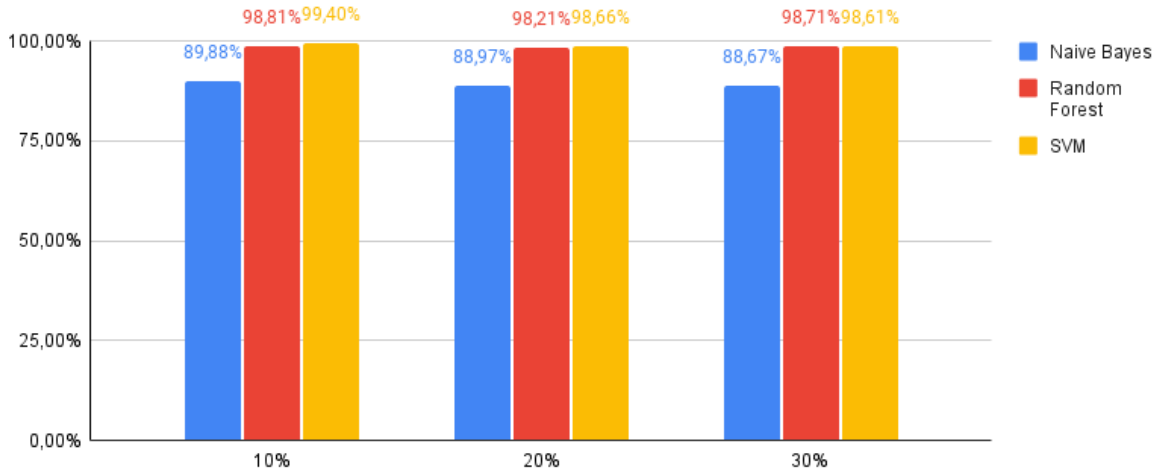


Figura 22 – Percentuais da medida acurácia obtido pelos algoritmos de AM usando TF-IDF.

Fonte: Elaborado pelo autor.

Pelos valores esboçados na Figura 22, o classificador *Naive Bayes* apresentou resultados inferiores aos demais classificadores, em todas as proporções de subdivisão do corpus. Comparativamente, os números apurados pelos classificadores SVM e *Random Forest* foram semelhantes com a acurácia superior à 98%. Ou seja, a média da acurácia do algoritmo *Random Forest* foi de 98,57%, já a média do SVM foi 98,89%.

Na sequência, foram realizados testes com o algoritmo *Word2vec*, mencionado na seção 5.2, e que necessitou avaliar e definir o tamanho do vetor de palavras. Dessa forma, foi definido diferentes tamanhos de vetores, que foram 200, 300 e 500 palavras, com o propósito de encontrar a dimensão adequada para a base de dados em estudo. Os resultados gerados pelos testes com o *Word2vec* aliado aos algoritmos *Naive Bayes*, SVM e *Random Forest*, sob o conjunto de teste, podem ser vistos nas Figuras 23, 24 e 25.

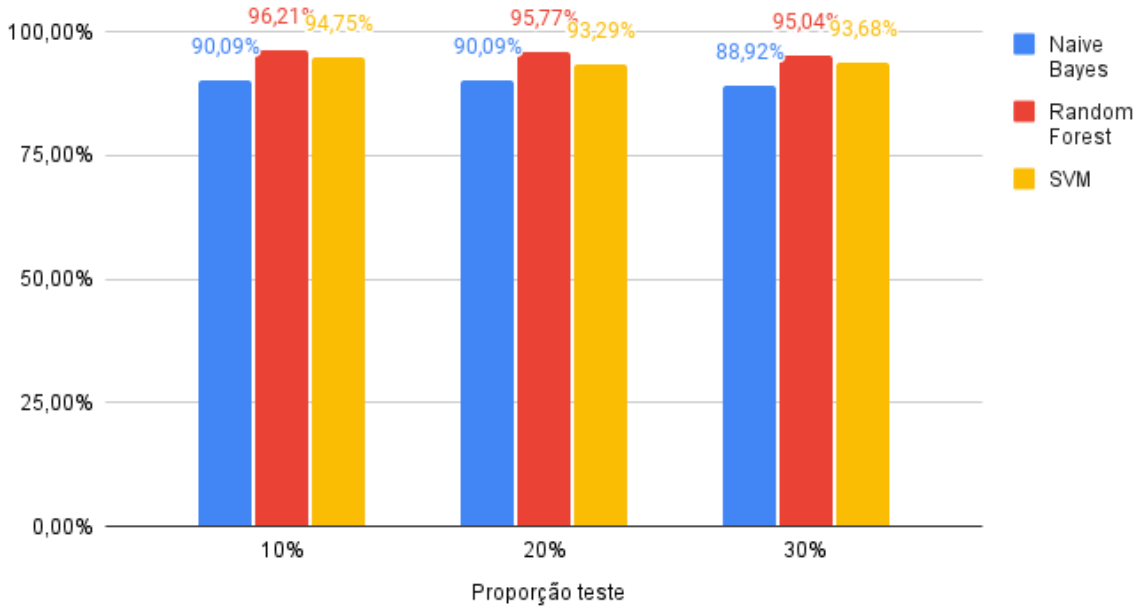


Figura 23 – Percentuais da medida acurácia obtido pelos algoritmos de AM usando Word2Vec com vetor de 200 palavras.

Fonte: Elaborado pelo autor.

De modo geral, os testes realizados com o algoritmo *Word2vec* apresentaram resultados significativos com valores da acurácia a partir de 88%. O classificador *Naive Bayes* apresentou resultados inferiores comparado aos resultados dos demais classificadores, para todas as três dimensões do vetor de palavras, com uma média de 89% de acurácia. Além disso, nas Figuras 23, 24 e 25, é possível observar que o algoritmo *Random Forest* e o SVM tiveram o desempenho muito semelhantes, alcançando uma acurácia média de 95% para o primeiro e 93% para o segundo. É importante destacar que a diferença dos tamanhos dos vetores de palavras do algoritmo *Word2vec* não influenciou nos resultados da acurácia.

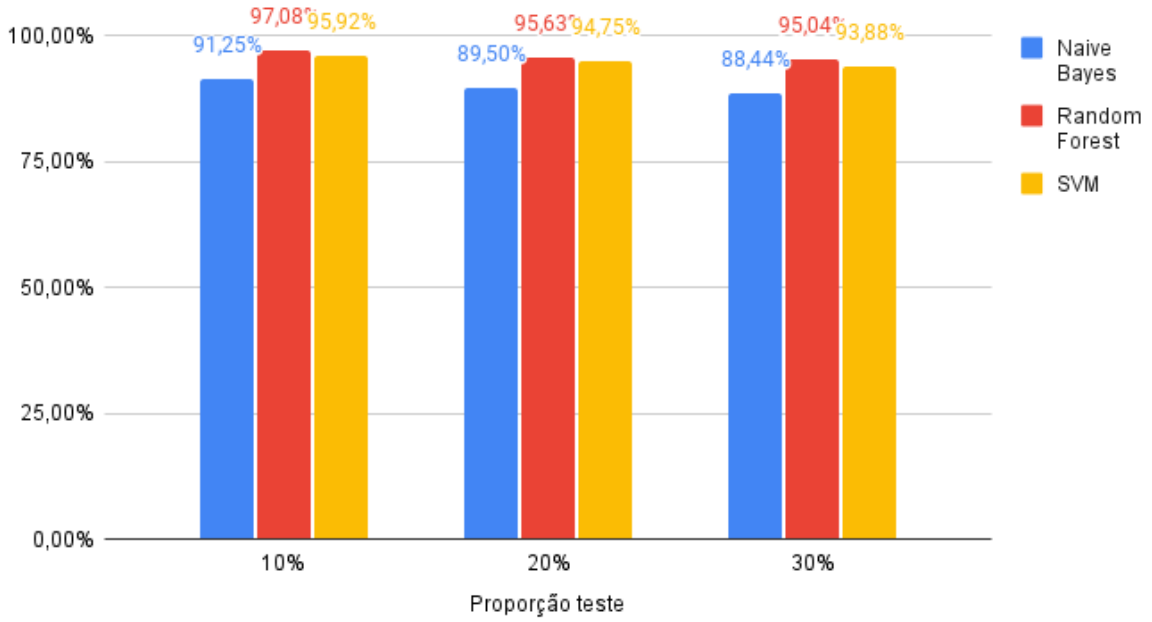


Figura 24 – Percentuais da medida acurácia obtido pelos algoritmos de AM usando Word2Vec com vetor de 300 palavras.

Fonte: Elaborada pelo autor.

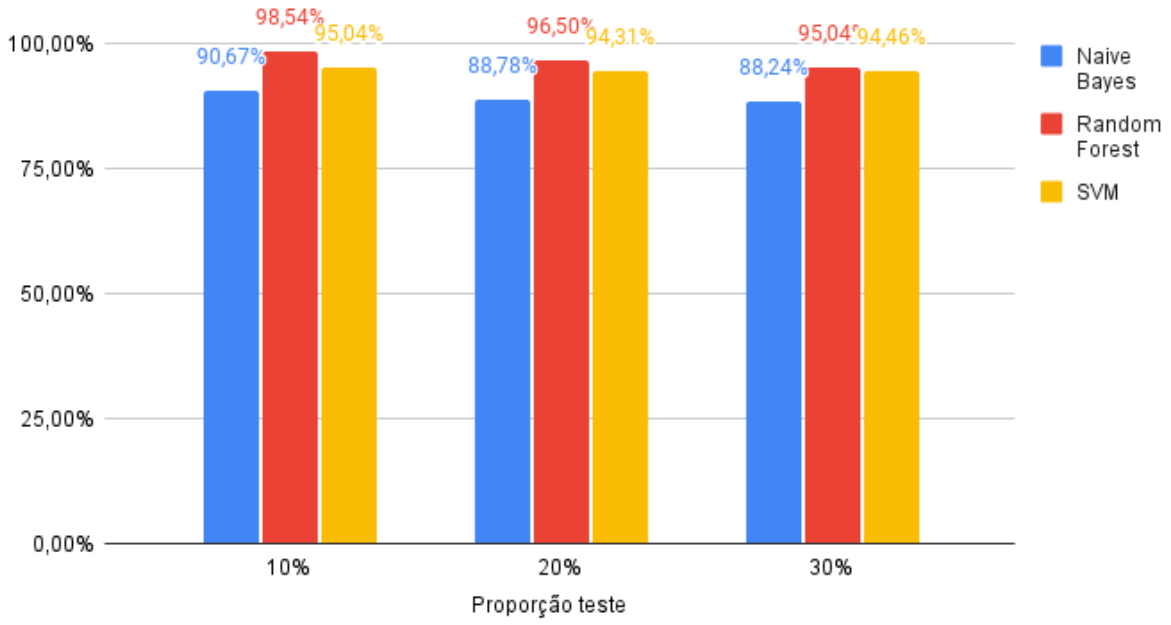


Figura 25 – Percentuais da medida acurácia obtido pelos algoritmos de AM usando Word2Vec com 500 palavras.

Fonte: Elaborada pelo autor.

Além dos testes e dos resultados alcançados com o corpus BoatosBr, foi realizada uma comparação das medidas de avaliação precisão e acurácia entre os corpus **BoatosBr**

e **Fake.Br**, e apresentado na tabela 2. Nesta comparação, foram implementados os modelos computacionais constituídos pelos algoritmos *TF-IDF*, na tokenização, e SVM, na classificação dos textos digitais. Ambos corpus foram subdivididos entre conjunto de treinamento, compreendendo 70% da quantidade total do corpus, e conjunto de teste, os 30% restantes. O corpus **Fake.Br** foi obtido pelo trabalho Santos et al. (2020).

Modelo Computacional (TF-IDF + SVM)			
Corpus	Precisão		Acurácia
	Falso	Verdade	
Corpus BoatosBr	98,456%	98,818%	98,608%
Corpus Fake.br	89,000%	89,000%	89,000%

Tabela 2 – Tabela comparativa entre os valores resultantes de um modelo computacional aplicados aos corpus Fake.Br e BoatosBR.

Fonte: Elaborada pelo autor.

Analisando a Tabela 2 é possível notar que os percentuais do corpus BoatosBr foram superiores à do outro corpus. No entanto, essa diferença pode ser justificada pelas características dos textos que compõe a base de dados, como: o ano de criação, o tamanho, o assunto tratado do texto que está relacionado ao estilo de escrita. E essas características interferem no comportamento do modelo computacional para a classificação, já que este usa dos próprios textos do corpus para aprender como identificar uma *fake news*.

6 Conclusão

Com o avanço das tecnologias e da velocidade de propagação da comunicação, a disseminação de *fake news* aumentou numa escala superior à sua detecção. Dessa forma, a utilização de metodologias automáticas para identificar e validar essas informações se tornou importante, como por exemplo, o uso de sistemas computacionais com inteligências artificiais. Nesse sentido, o objetivo do trabalho foi de criar uma nova base de dados atualizada na língua portuguesa do Brasil.

Existem sites que atuam como agências checadoras de fatos que são publicados na Internet, como o Boatos.org. Esta agência, em especial, teve um papel importante durante o desenvolvimento do trabalho, pois facilitou a busca por notícias falsas e verdadeiras já validadas. Assim, usando a linguagem de programação *Python* foi criado *web-crawlers*, responsáveis em coletar os textos que formaram o novo corpus que foi nomeado por **BoatosBr Corpus**. O novo corpus é composto por mais de 3000 textos.

Após a formação do corpus, criou-se um modelo computacional estruturado em técnicas de PLN e algoritmos de classificação de AM para validar a constituição da base de dados BoatosBr. Então, estágios primários da análise PLN foram aplicados aos textos digitais, com a pretensão de extrair computacionalmente informações de um documento textual. Para complementar, foram disponibilizados informações adicionais sobre os textos, como: quantidades de emojis, de palavras, de palavras em maiúsculo, de verbos; para criar mais possibilidades de estudos aos usuários de trabalhos futuros.

Em relação a classificação pelo AM, foram escolhidos três algoritmos clássicos presentes na literatura, e que atingiu valores para a acurácia a partir de 85% para mais. Estes testes foram realizados apenas com os textos digitais, sem considerar as informações adicionais sobre os mesmo. Logo, conclui que o corpus está apto a ser disponibilizado e utilizado em outros estudos.

Entretanto, quanto a distribuição quantitativa da base de dados, o novo corpus não ficou balanceado entre os rótulos verdadeiros e *fake*, assim como o tamanho dos textos ficaram com tamanhos diferentes. Estas são medidas importantes a serem implementadas na continuação deste trabalho.

Por fim, sugere para trabalhos futuros a adição de novos textos de diferentes sites para enriquecer o BoatosBr, além de diversificar os textos presentes na base, visto que possui apenas duas origens. Além disso, sugere explorar diferentes metadados dos textos presentes para acrescentar características dos mesmos e viabilizar diferentes opções de estudos.

Referências

BOATOS.ORG. **Sobre**. 2024. <https://www.boatos.org/sobre>. Acesso em: 30 jun. 2024. Citado na página 31.

CITS. **What is Fake News?** . 2016. <https://www.cits.ucsb.edu/fake-news/what-is-fake-news>. [Online; accessed 06-Setembro-2023]. Citado na página 11.

COPELAND, B. **Artificial Intelligence**. 2023. <https://www.britannica.com/technology/artificial-intelligence>. [Online; accessed 29-Outubro-2023]. Citado na página 19.

GARCIA, G. L.; AFONSO, L. C. S.; PAPA, J. P. Fakerecogna: A new brazilian corpus for fake news detection. In: PINHEIRO, V.; GAMALLO, P.; AMARO, R.; SCARTON, C.; BATISTA, F.; SILVA, D.; MAGRO, C.; PINTO, H. (Ed.). **Computational Processing of the Portuguese Language**. Cham: Springer International Publishing, 2022. p. 57–67. ISBN 978-3-030-98305-5. Citado na página 26.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep Learning**. [S.l.]: MIT Press, 2016. <http://www.deeplearningbook.org>. Citado na página 20.

IBM. **Natural Language Processing - IBM**. 2023. <https://www.ibm.com/br-pt/topics/natural-language-processing>. [Online; accessed 29-Outubro-2023]. Citado 2 vezes nas páginas 22 e 23.

_____. **O que são classificadores Naïve Bayes?** 2024. <https://www.ibm.com/br-pt/topics/naive-bayes>. Acesso em: 22 jul. 2024. Citado na página 20.

KAUSAR, M. A.; DHAKA, V.; SINGH, S. K. Web crawler: a review. **International Journal of Computer Applications**, Foundation of Computer Science, 244 5 th Avenue,# 1526, New York, NY 10001 . . . , v. 63, n. 2, p. 31–36, 2013. Citado na página 18.

KUMAR, S.; SHAH, N. **False Information on Web and Social Media: A Survey**. 2018. Citado 2 vezes nas páginas 14 e 15.

OFFICE, M. N. **Study: On Twitter, false news travels faster than true stories** . 2018. <https://news.mit.edu/2018/study-twitter-false-news-travels-faster-true-stories-0308>. [Online; accessed 06-Setembro-2023]. Citado na página 11.

POLITIFACT. **Principles of the Truth-O-Meter: Politifact's Methodology**. 2018. Disponível em: <https://www.politifact.com/article/2018/feb/12/principles-truth-o-meter-politifacts-methodology-i/>. Citado na página 27.

RUBIN, V. L.; CONROY, N.; CHEN, Y.; CORNWELL, S. Fake news or truth? using satirical cues to detect potentially misleading news. In: **Proceedings of the second workshop on computational approaches to deception detection**. [S.l.: s.n.], 2016. p. 7–17. Citado na página 18.

SANTOS, R. L.; SILVA, R. M.; ALMEIDA, T. A.; PARDO, T. A. Towards automatically filtering fake news in portuguese. **Expert Systems with Applications**, v. 146, p. 113199, 2020. ISSN 0957-4174. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0957417420300257>. Citado 3 vezes nas páginas 18, 25 e 40.

SANTOS, R. L. d. S. **Detecção automática de notícias falsas em português**. Tese (Doutorado) — Universidade de São Paulo, São Carlos, SP, 6 2022. Disponível em: <https://www.teses.usp.br/teses/disponiveis/55/55134/tde-14072022-165613/pt-br.php>. Citado 3 vezes nas páginas 14, 25 e 26.

SILVA, R. M.; SANTOS, R. L.; ALMEIDA, T. A.; PARDO, T. A. Towards automatically filtering fake news in portuguese. **Expert Systems with Applications**, v. 146, p. 113199, 2020. ISSN 0957-4174. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0957417420300257>. Citado na página 12.

UNHCR, t. U. R. A. **RUMORS AND MISINFORMATION**. 2022. <https://www.unhcr.org/innovation/wp-content/uploads/2022/02/Using-Social-Media-in-CBP-Chapter-6-Rumours-and-Misinformation.pdf>. [Online; accessed 28-Agosto-2024]. Citado na página 16.

VINICIUSSEGATTO. **SVM, ou Support Vector Machine.** 2023. Https://medium.com/liga-mackenzie-de-ia-ciência-de-dados/svm-ou-support-vector-machine-7efcabdcc7be. Acesso em: 23 jul. 2024. Citado na página 21.

WIKIPEDIA. **Linchamento de Fabiane Maria de Jesus**. 2018. <https://pt.wikipedia.org/w/index.php?title=Linchamento_de_Fabiane_Maria_de_Jesus&oldid=66411281>. [Online; accessed 06-Setembro-2023]. Citado na página 11.

ZHAO, B. Web scraping. **Encyclopedia of big data**, Springer Living ed. Cham, v. 1, 2017. Citado na página 18.