Detecting Misinformation in Tweets Related to COVID-19

Ramon Sousa da Cruz, Gilberto Nunes Neto, Rafael Torres Anchiêta

[1] Instituto Federal do Piauí - Campus Picos
Laboratório de Inteligência Artificial Robótica e Automação (LIARA)
Av. Pedro Marques de Medeiros 64600-000 – Picos – PI – Brasil

ramon.sousa111.rr@gmail.com, {gilberto.nunes,rta}@ifpi.edu.br

Abstract. *The spread of misinformation brought and still brings several problems for society, being considered an infodemia by the World Health Organization (WHO). Most of the works to deal with misinformation are focused on the English language. In order to fulfill this gap, this work investigates strategies based on supervised machine learning to detect misinformation in tweets written in the Portuguese language. Moreover, a manually annotated corpus for this task was created, aiming to evaluate the developed approaches and compare them with related works. The achieved results are competitive with related works, indicating that the approach produces an interesting baseline for the built corpus.*

Resumo. *A propagação de desinformação trouxe e ainda traz diversos problemas para a sociedade, sendo considerada uma infodemia pela Organização Mundial da Saúde (OMS). A grande maioria dos trabalhos desenvolvidos para lidar com desinformação são focados para a língua inglesa. A fim de preencher essa lacuna, este trabalho investiga estratégias baseadas em aprendizado de máquina supervisionado para detectar desinformação em tweets escritos na língua portuguesa. Além disso, criou-se um corpus que foi manualmente anotado para esta tarefa, a fim de avaliar as abordagens desenvolvidas e compará-las com trabalhos relacionados. Os resultados alcançados são competitivos com trabalhos correlatos, indicando que a abordagem produz um interessante baseline para o corpus construído.*

1. Introdução

A pandemia do novo coronavírus trouxe diversos problemas sociais para a população em todo o mundo, como aumento de estresse, ansiedade, sintomas depressivos, insônia em profissionais da saúde, entre outros [Spoorthy et al. 2020]. Um outro problema que veio junto com a pandemia foi o aumento da disseminação de desinformação, sendo, esse problema, considerado uma infodemia e uma grande ameaça à saúde pública [Roozenbeek et al. 2020].

De maneira geral, desinformação pode ser definida como a circulação de informações falsas [Zubiaga et al. 2018]. Um termo mais frequente que a comunidade de pesquisa adota é *false* (ou *fake*) *news* [Lazer et al. 2018, Pierri et al. 2020]. É importante destacar que, neste trabalho, não é feita nenhuma alegação sobre a intenção dos fornecedores de informações, sejam acidentais ou maliciosas. Na Figura 1, é apresentado

um exemplo de conteúdo amplamente compartilhado no Twitter contendo desinformação. O contexto da mensagem na figura era propagar que o número de mortes pela COVID-19 estava sendo inflado a fim de que as pessoas voltassem as suas rotinas de trabalho.

> *"Gente ! O primo do porteiro aqui do prédio morreu pq foi trocar o pneu do caminhão e o pneu estourou no rosto dele. Receberam o atestado de óbito como se fosse o covid 19. Eles estão indignados"*

Figura 1. Exemplo de desinformação postada no Twitter.

Além de levar mensagens enganosas as pessoas, a propagação de desinformação levou centenas de pessoas a morte [Islam et al. 2020]. Essas pessoas morreram por beber metanol ou produtos de limpeza à base de álcool, acreditando, erroneamente, que os produtos são/eram uma cura para o vírus. Pode-se ver que esse tipo de conteúdo causou e ainda está causando um impacto extremamente negativo na sociedade.

Existem diversos trabalhos quem lidam com desinformação em textos de redes sociais, propondo métodos para identificar e combater a disseminação de desinformação [Zhou and Zafarani 2020]. No entanto, a maioria deles focam na língua inglesa, criando uma lacuna de recursos e métodos em comparação com outras línguas. Neste trabalho, com o objetivo de preencher essa lacuna, desenvolveu-se uma abordagem baseada em aprendizado de máquina para detectar *tweets* que contenham desinformação escritos na língua portuguesa. Implementaram-se duas estratégias, uma baseada em *features* superficiais e outra baseada em *embeddings*, que foram utilizadas para treinar algoritmos supervisionados a predizerem se um *tweet* é uma desinformação ou não. Para avaliar essas estratégias, criou-se um *corpus* com 12.027 *tweets* que foram manualmente anotados por 3 anotadores. Além disso, compararam-se as abordagens desenvolvidas com métodos da literatura, alcançando resultados competitivos.

As principais contribuições deste trabalho são: (i) um corpus anotado manualmente para a tarefa de detecção de desinformação e (ii) um método *baseline* que prediz se um *tweet* contém desinformação.

O restante do trabalho está organizado da seguinte forma. Seção 2 mostra os principais trabalhos correlatos com esta pesquisa. Na Seção 3, é detalhado a construção do *corpus*. Seção 4 detalha os métodos desenvolvidos para predizer desinformação. Na Seção 5, é apresentado os experimentos conduzidos e os resultados obtidos. Por fim, Seção 6 conclui o artigo indicando alguns trabalhos futuros.

2. Trabalhos relacionados

Nesta seção, os principais trabalhos que lidam com desinformação/*fake news* na língua portuguesa serão apresentados. Para a língua inglesa, os *corpora* são maiores e os melhores resultados são obtidos com redes profundas e modelos de língua.

[Monteiro et al. 2018] criaram o primeiro *corpus* de *fake news* para o Português, `Fake.Br`. O *corpus* contém 7.200 notícias, sendo 3.600 verdadeiras e 3.600 falsas. Os autores utilizaram esse *corpus* para avaliar diferentes *features* e classificadores na tarefa de detecção de *fake news*. O melhor resultado encontrado foi utilizando *features* superficiais mais a emotividade do texto junto com o classificador de Máquina de Vetores de Suporte, atingindo 0.89 de acurácia e *f-score*.

[Faustini and Covões 2019] produziram dois *corpora*, um de mensagens do aplicativo WhatsApp e outro de mensagens do Twitter. Além disso, eles avaliaram esses recursos utilizando uma técnica chamada de *One-Class Classification* (OCC), onde o modelo é treinado com apenas uma classe. Os autores compararam o algoritmo *DCDistanceOCC* com algoritmos tradicionais de aprendizagem de máquina, usando os *corpora* produzidos. Como resultado, o algoritmo *DCDistanceOCC* obteve uma performance similar aos algoritmos tradicionais de aprendizagem de máquina.

[Cabral et al. 2021] construíram um *corpus* de mensagens escritas em Português do aplicativo WhatsApp, `FakeWhatsApp.BR`. Esse *dataset* possui 5.284 mensagens, 3.091 falsas e 2.193 verdadeiras. Além do recurso, os autores aplicaram diferentes técnicas de aprendizado de máquina para avaliar o *dataset* construído. O melhor método alcançou 0.73 de *f-score* utilizando *features* superficiais com unigrama e bigramas alimentadas no classificador Máquina de Vetores de Suporte.

3. Construção do Corpus

Uma vez que existem poucos *corpora* anotados relacionados à covid-19 e desinformação para a língua portuguesa e a fim de desenvolver um classificador de desinformação, primeiramente, construiu-se um *corpus* com textos contendo desinformação e informação verdadeira. O *corpus* foi criado com textos do Twitter. Para isso, utilizou-se a biblioteca snscrape[1] e extraíram-se 14.000 *tweets* escritos em Português Brasileiro usando *hashtags* relacionadas a COVID-19 entre os meses de março e setembro de 2020, coletando 2.000 *tweets* por mês.

Após a coleta, os *tweets* foram manualmente rotulados em desinformação e informação verdadeira por 3 anotadores. O rótulo final do *tweet* foi escolhido através do voto majoritário. Na etapa de anotação, eliminaram-se 1.971 *tweets* que não tratavam sobre a COVID-19. Na Tabela 1, são apresentadas algumas informações sobre corpus manualmente anotado.

Tabela 1. Informações sobre o Corpus.

Rótulo	Tweets	Média Tokens	Média URLs	Média Emojis
Desinformação	543	29,95	0,53	0,14
Info. verdadeira	11.484	27,18	0,96	0,15

A partir da Tabela 1, pode-se ver que o *corpus* é muito desbalanceado em relação ao rótulo desinformação. Embora a quantidade de desinformação seja menor do que a quantidade de informação verdadeira, a primeira possui um alcance muito maior do que a última, devido a existência de *bots*[2] e do mecanismo de *retweetar* um *tweet* [Himelein-Wachowiak et al. 2021]. Esse mecanismo faz com que a desinformação se propague de maneira mais rápida do que a informação verdadeira [Vosoughi et al. 2018].

[1] `https://github.com/JustAnotherArchivist/snscrape`

[2] De maneira geral, é um *software* robô que automatiza algumas rotinas.

A partir do *corpus* anotado, extraíram-se alguns dados (estatísticas), como a média de: *tokens*, URLs e emojis. Pode-se observar na tabela acima que os valores tanto para desinformação quanto para informação verdadeira são similares. Dessa forma, os *tweets* parecem ter uma estrutura semelhante, sendo necessário olhar para o seu conteúdo a fim de conseguir distinguir um *tweet* com desinformação de outro com informação verdadeira, tornando a tarefa de classificação automática mais desafiadora.

4. Estratégias desenvolvidas

Para criar um classificador de desinformação, organizou-se o método em três etapas: pré-processamento, extração de *features* e classificação. Na primeira etapa, aplicou-se o algoritmo de stemização Removedor de Sufixo da Língua Portuguesa (RSLP) [Orengo and Huyck 2001] a fim de stemizar os *tweets* do *corpus*[3]. A stemização é o processo de reduzir palavras flexionadas a sua raiz, não necessariamente idêntica a raiz morfológica da palavra.

Na segunda etapa, extraíram-se *features* superficiais e *embeddings* do *corpus* pré-processado. Como *features* superficiais, adotou-se a ponderação *Term Frequency-Inverse Document Frequency* (TF-IDF), apresentado na Equação 1.

$$TF - IDF = tf_{k,j} \times log\left(\frac{N}{df_k}\right) \tag{1}$$

onde:

- $tf_{k,j}$ - número de vezes que o termo k aparece em um documento j;
- N - número total de documentos;
- df_k - número de documentos com o temo k.

De acordo com [Rajaraman and Ullman 2011], TF-IDF é um número estatístico que reflete a importância de uma palavra em uma coleção de documentos. Essa estatística é frequentemente utilizada como um fator de ponderação na área de Recuperação de Informação (RI) [Baeza-Yates and Ribeiro-Neto 2013]. Com esta estratégia, extraíram-se 24.400 *features* do *corpus*.

Para as *features* baseadas em *embeddings*, utilizou-se a abordagem *Paragraph Vector* [Le and Mikolov 2014]. *Embeddings* são representações de vetores densos de números reais que podem corresponder a uma palavra. *Paragraph Vector* é um modelo não supervisionado que aprende essas representações vetoriais a partir de um texto, através de redes neurais [Le and Mikolov 2014]. Nesse modelo, cada *tweet* é representado por um vetor denso de números reais, sendo que cada vetor possui o mesmo tamanho. Além disso, o modelo possui duas formas para aprender representações vetoriais. A primeira é chamada de *Distributed Memory Model of Paragraph Vectors* (PV-DM) onde o vetor de parágrafos e os vetores de palavras são concatenados para predizer a próxima palavra em um contexto. A segunda é chamada de *Distributed Bag of Words version of Paragraph Vector* (PV-DBOW) onde o modelo prediz palavras a partir de um parágrafo. PV-DM e PV-DBOW são apresentados nas Figuras 2 e 3, respectivamente.

[3]Avaliaram-se outras formas de pré-processamento, como: remoção de *stopwords*, normalização de palavras e filtragem de links, emojis, emoticons, entre outros. Entretanto, os resultados não melhoraram.

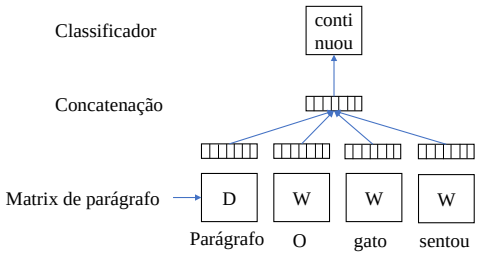

Figura 2. PV-DM.

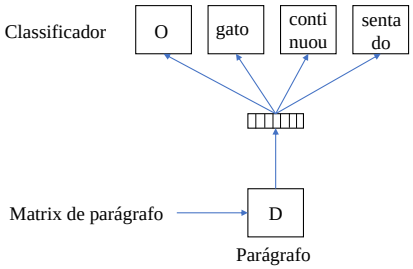

Figura 3. PV-DBOW.

Os dois modelos foram utilizados para aprender representações vetoriais do *corpus* de *tweets*. Dessa forma, treinaram-se modelos com tamanhos de vetores 50, 100 e 300 com auxílio da biblioteca Gensim[4].

Por fim, na terceira etapa, alimentaram-se vários algoritmos de aprendizado de máquina a partir das *features* extraídas com o objetivo de treinar classificadores a predizerem se um *tweet* é uma desinformação ou não.

No que segue, são detalhados os experimentos realizados e os resultados obtidos.

5. Experimentos e Resultados

As *features* superficiais e *embeddings* foram utilizadas para alimentar alguns algoritmos tradicionais de aprendizado de máquina supervisionado, como: Naïve Bayes, Árvores de Decisão, Máquina de Vetores de Suporte e Redes Neurais. Para treinar os algoritmos acima, utilizou-se a ferramenta Scikit-Learn [Pedregosa et al. 2011].

A fim de treinar os algoritmos para predizerem se um *tweet* é uma desinformação ou não, adotou-se a técnica de validação cruzada estratificada com 5 grupos.A validação cruzada divide aleatoriamente um conjunto de instâncias do *corpus* em *K* grupos de tamanhos aproximadamente iguais. O primeiro grupo é tratado como um conjunto validação, enquanto o método é ajustado (treinado) nos $k-1$ grupos restantes [James et al. 2013].

Para avaliar a performance dos algoritmos, utilizaram-se as métricas macro *f-score* e acurácia balanceada (AC) [Brodersen et al. 2010]. A primeira trata as classes com pesos iguais, sendo adequada quando se tem um *corpus* desbalanceado, enquanto a segunda observa para a média da métrica *precision* de cada classe do *corpus*. Para computar essas métricas, é necessário calcular o valor de *precision*, *recall* e *specificity*, definidas nas Equações 2, 3 e 4, respectivamente.

$$prec = \frac{TP}{(TP+FP)} \quad (2) \qquad rec = \frac{TP}{(TP+FN)} \quad (3) \qquad spec = \frac{TN}{(TN+FP)} \quad (4)$$

onde: TP, FP e FN significam *true positive*, *false positive* e *false negative*, respectivamente.

A partir das equações acima, é calculado o valor da acurácia balanceada, Equação 5, e macro *f-score*, Equação 6. Os valores 0 e 1 nas funções *precision* e *recall*

[4]`https://radimrehurek.com/gensim/`

indicam o rótulo do *corpus*, ou seja, 0 aponta para o rótulo de desinformação, enquanto que 1 aponta para o rótulo de informação verdadeira.

$$AC = \frac{(recall + specificity)}{2} \tag{5}$$

$$\begin{aligned} Macro - Precision(MP) &= \frac{(precision(0) + precicion(1))}{2} \\ Macro - Recall(MR) &= \frac{(recall(0) + recall(1))}{2} \\ Macro - Fscore &= \frac{2 \times MP \times MR}{(MP + MR)} \end{aligned} \tag{6}$$

Com o valor da métricas, pode-se comparar os resultados das estratégias desenvolvidas, conforme mostrado na Tabela 2.

Tabela 2. Resultados das estratégias desenvolvidas.

Feature	**Classificador**	**F-score**	**AC**
Superficial	Naïve Bayes	$0,56$	$0,57$
	Árvore de Decisão	$0,61$	$0,60$
	Máquina de Vetores de Suporte	$0,58$	$0,55$
	Perceptron Multicamadas	**0,65**	**0,61**
Embeddings	Naïve Bayes	$0,45$	$0,52$
	Árvore de Decisão	$0,49$	$0,49$
	Máquina de Vetores de Suporte	$0,48$	$0,50$
	Perceptron Multicamadas	$0,48$	$0,50$

A partir da tabela acima, observa-se que o melhor resultado foi obtido com o classificador Perceptron Multicamadas (MLP) alimentado pelas *features* superficiais[5]. É importante dizer que não foram realizados ajustes nos parâmetros dos algoritmos de classificação, ou seja, utilizaram-se os parâmetros padrões da ferramenta Scikit-Learn. O pré-processamento ajudou a melhorar os resultados em 1% para cada algoritmo. Embora seja uma contribuição pequena, optou-se por mantê-lo. Além disso, acredita-se que as *embeddings* obtiveram resultados piores do que as *features* superficiais devido ao tamanho e o desbalanceamento do *corpus*.

Com base no resultado obtido na tabela acima, comparou-se o melhor resultado alcançado com o método de [Monteiro et al. 2018] e [Cabral et al. 2021], desenvolvidos para identificar *fake news*. Para isso, treinaram-se os modelos desenvolvidos pelos autores no *corpus* de *tweets* e avaliaram-se as abordagens utilizando a técnica de validação cruzada estratificada com 5 grupos, adotando as métricas macro *f-score* e acurácia balanceada (AC). Na Tabela 3, é apresentado o resultado da comparação.

Pode-se ver que a estratégia baseada em *features* superficiais com o algoritmo de classificação Perceptron Multicamadas superou os métodos de [Monteiro et al. 2018] e

[5]Os melhores resultados obtidos com as *embeddings* foram com vetores de dimensão 300.

Tabela 3. Resultado da comparação entre diferentes abordagens.

Abordagem	F-score	AC
[Monteiro et al. 2018]	$0,58$	$0,55$
[Cabral et al. 2021]	$0,55$	$0,53$
Nossa	**0,65**	**0,61**

[Cabral et al. 2021], mostrando que o uso dessas *features* alcançam um bom resultado *baseline* para a detecção de desinformação em textos do Twitter para a língua portuguesa. Por um lado, é importante destacar que o método de [Monteiro et al. 2018] foi desenvolvido para identificar *fake news* em textos do tipo jornalísticos, ou seja, são muito diferentes de *tweets*. Por outro lado, a estratégia de [Cabral et al. 2021] foi desenvolvida para textos de mensagens do WhatsApp, que possuem uma estrutura similar a de textos de Twitter, uma vez que ambos normalmente são curtos, desestruturados e raramente obedecem a regras gramaticais e de pontuação.

Por último, é importante dizer que não foi possível treinar o modelo do trabalho de [Faustini and Covões 2019] no nosso *corpus*. Como o método dos autores utiliza apenas um rótulo de classe para treinamento e o nosso *corpus* é bastante desbalanceado em relação a classe de desinformação, a comparação entre os métodos não seria justa, pois o método de [Faustini and Covões 2019] é treinado na classe de mensagens falsas/desinformação. Além disso, não foi possível usar o *corpus* de *tweets* de [Faustini and Covões 2019] para treinar o nosso modelo e comparar as estratégias, pois muitos *tweets* não estão mais disponíveis. Dessa forma, optou-se por treinar o nosso modelo no *corpus* de mensagens do WhatsApp desenvolvido por [Faustini and Covões 2019], visando realizar uma comparação mais justa entre as abordagens. O *corpus* de mensagens do WhatsApp possui 165 mensagens com conteúdo falso e 12 mensagens com conteúdo verdadeiro. Para comparar as abordagens, utilizaram-se as métricas macro *f-score* e acurácia balanceada (AC). Tabela 4 apresenta o resultado da comparação.

Tabela 4. Resultado da comparação entre abordagens no *corpus* do WhatsApp.

Abordagem	F-score	AC
[Faustini and Covões 2019]	0.40	0.42
Nossa	**0.48**	**0.50**

Como pode-se ver na tabela acima, nossa estratégia obteve melhores resultados do que o método de [Faustini and Covões 2019]. O método de [Faustini and Covões 2019] é treinado apenas na classe negativa, mensagens com conteúdo falso, e avaliado na classe positiva utilizando validação cruzada com 10 grupos. Portanto, seguiu-se essa mesma estratégia de avaliação para o nosso método a fim de tornar a comparação entre as abordagens mais justa.

5.1. Análise dos resultados

Com objetivo de entender os resultados obtidos pelo nosso modelo treinado no *corpus* de *tweets*, realizou-se uma pequena análise dos resultados. Primeiramente, computou-se

os valores da matriz de confusão baseado na média do resultado dos 5 grupos, conforme apresentado na Tabela 5. A partir da tabela, pode-se ver que uma das fraquezas do modelo é em predizer corretamente quando um *tweet* é uma desinformação, ou seja, quando o valor esperado para um *tweet* é uma desinformação, mas o método prediz informação verdadeira, produzindo um alto valor de falsos positivos. Acredita-se que um dos fatores para esse problema seja o desbalanceamento do *corpus*.

Tabela 5. Matriz de confusão.

		Classe predita	
		Info. verdadeira	Desinformação
Classe atual	Info. verdadeira	$2.269,6$	$27,2$
	Desinformação	$82,4$	$26,2$

Em seguida, observou-se quais palavras são as mais discriminativas para cada classe, exibido na Figura 4. Pode-se ver que: "bolsonaro", "brasil" e "govern" são as três palavras mais discriminativas para identificar desinformação, enquanto as três palavras para identificar informação verdadeira são: "mort", "brasil" e "confirm".

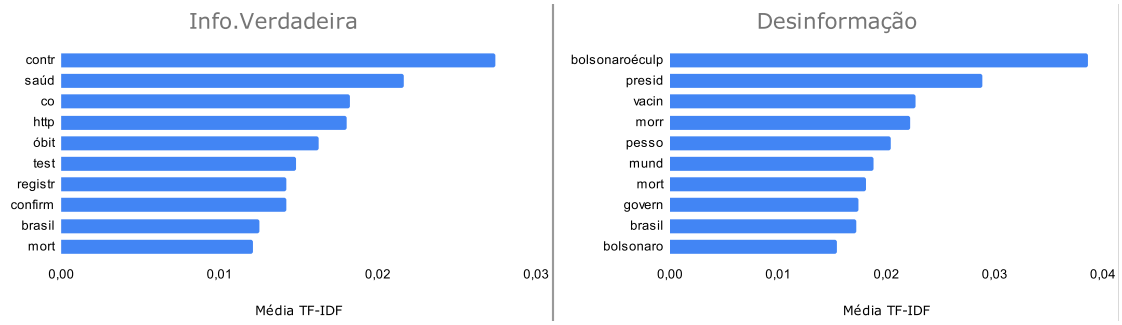

Figura 4. Palavras mais discriminativas.

Baseado nessas informações, verificou-se as sentenças classificadas erroneamente como informação verdadeira, ou seja, as sentenças deviam ter sido classificadas como desinformação, mas foram classificadas como sendo verdadeiras. Para isso, analisou-se as dez palavras com maiores valores de TF-IDF, que foram as seguintes: "únic", "http", "co", "úmid", "disp", "dissemin", "diss", "dispon", "dispens" e "disp". Como pode-se ver, a maioria dessas palavras não está na lista das palavras mais discriminativas para desinformação, mostrando que a abordagem TF-IDF é muito dependente das palavras do *tweet*. Assim, faz-se necessário o desenvolvimento de métodos mais robustos que levem em consideração informação semântica das sentenças.

O *corpus* construído e as abordagens desenvolvidas estão publicamente disponíveis em `https://github.com/Gungni/Akhasic`.

6. Conclusão

Este trabalho apresentou a construção de *corpus* de desinformação a partir de mensagens do Twitter e um método *baseline* responsável por classificar se um *tweet* é uma desinformação ou não. Além disso, comparou-se nossa estratégia com abordagens similares responsáveis por detectar *fake news*. Essa comparação mostrou que a abordagem

desenvolvida supera os métodos anteriores desenvolvidos. Por fim, realizou-se uma pequena análise dos resultados, visando obter *insights* para futuras melhorias.

Como trabalhos futuro, pretende-se: (i) investigar estratégias para o balanceamento do *corpus* como *data augmentation*, a fim de evitar problemas com estratégias baseadas em *under* ou *over sampling*; (ii) avaliar formalismos semânticos, visando analisar informações semânticas explícitas das sentenças, como, por exemplo, ontologias relacionadas ao tema covid-19 [Gritz et al. 2021] e (iii) analisar estratégias baseadas em aprendizado profundo ou modelos de língua.

Agradecimentos

Os autores agradecem ao Instituto Federal de Educação, Ciência e Tecnologia do Piauí pelo apoio neste trabalho.

Referências

Baeza-Yates, R. and Ribeiro-Neto, B. (2013). *Recuperação de Informação-: Conceitos e Tecnologia das Máquinas de Busca*. Bookman Editora.

Brodersen, K. H., Ong, C. S., Stephan, K. E., and Buhmann, J. M. (2010). The Balanced Accuracy and Its Posterior Distribution. In *Proceedings of the 20th International Conference on Pattern Recognition*, pages 3121–3124, Istanbul, Turkey. IEEE.

Cabral, L., Monteiro, J. M., da Silva, J. W. F., Mattos, C. L., and Mourao, P. J. C. (2021). FakeWhastApp.BR: NLP and Machine Learning Techniques for Misinformation Detection in Brazilian Portuguese Whatsapp Messages. In *Proceedings of the 23rd International Conference on Enterprise Information Systems*, pages 63–74, Online. SCITEPRESS.

Faustini, P. and Covões, T. (2019). Fake news detection using one-class classification. In *Proceedings of the 8th Brazilian Conference on Intelligent Systems*, pages 592–597, Salvador, Brazil. IEEE.

Gritz, R., Pereira, R., Silva, H. M., Zatti, H., Viana, L., Navarro, K., Dias, T., Oliveira, V., Souza, R., Oliveira, V., Netto, M. B., and Porto, F. (2021). An ontology based natural language processing pipeline for brazilian covid-19 emr. In *Anais do XV Brazilian e-Science Workshop*, pages 97–104, Evento Online. SBC.

Himelein-Wachowiak, M., Giorgi, S., Devoto, A., Rahman, M., Ungar, L., Schwartz, H. A., Epstein, D. H., Leggio, L., Curtis, B., et al. (2021). Bots and misinformation spread on social media: Implications for covid-19. *Journal of Medical Internet Research*, 23(5):e26933.

Islam, M. S., Sarkar, T., Khan, S. H., Kamal, A.-H. M., Hasan, S. M., Kabir, A., Yeasmin, D., Islam, M. A., Chowdhury, K. I. A., Anwar, K. S., et al. (2020). Covid-19–related infodemic and its impact on public health: A global social media analysis. *The American journal of tropical medicine and hygiene*, 103(4):1621.

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). *An introduction to statistical learning*, volume 112. Springer.

Lazer, D. M., Baum, M. A., Benkler, Y., Berinsky, A. J., Greenhill, K. M., Menczer, F., Metzger, M. J., Nyhan, B., Pennycook, G., Rothschild, D., et al. (2018). The science of fake news. *Science*, 359(6380):1094–1096.

Le, Q. and Mikolov, T. (2014). Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning*, pages 1188–1196, Beijin, China. PMLR.

Monteiro, R. A., Santos, R. L., Pardo, T. A., De Almeida, T. A., Ruiz, E. E., and Vale, O. A. (2018). Contributions to the study of fake news in portuguese: New corpus and automatic detection results. In *Proceedings of the 13th International Conference on Computational Processing of the Portuguese Language*, pages 324–334, Canela, Brazil. Springer.

Orengo, V. M. and Huyck, C. R. (2001). A stemming algorithmm for the portuguese language. In *Proceedings of the Eighth International Symposium on String Processing and Information Retrieval*, pages 186–193, Laguna de San Rafael, Chile. IEEE.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Pierri, F., Piccardi, C., and Ceri, S. (2020). Topology comparison of twitter diffusion networks effectively reveals misleading information. *Scientific reports*, 10(1):1–9.

Rajaraman, A. and Ullman, J. D. (2011). *Mining of massive datasets*. Cambridge University Press.

Roozenbeek, J., Schneider, C. R., Dryhurst, S., Kerr, J., Freeman, A. L., Recchia, G., Van Der Bles, A. M., and Van Der Linden, S. (2020). Susceptibility to misinformation about covid-19 around the world. *Royal Society open science*, 7(10):201199.

Spoorthy, M. S., Pratapa, S. K., and Mahant, S. (2020). Mental health problems faced by healthcare workers due to the covid-19 pandemic–a review. *Asian journal of psychiatry*, 51:102119.

Vosoughi, S., Roy, D., and Aral, S. (2018). The spread of true and false news online. *Science*, 359(6380):1146–1151.

Zhou, X. and Zafarani, R. (2020). A survey of fake news: Fundamental theories, detection methods, and opportunities. *ACM Computing Surveys (CSUR)*, 53(5):1–40.

Zubiaga, A., Aker, A., Bontcheva, K., Liakata, M., and Procter, R. (2018). Detection and resolution of rumours in social media: A survey. *ACM Computing Surveys (CSUR)*, 51(2):1–36.