Fake News Detection about Covid-19 in the Portuguese Language

Anísio Pereira Batista Filho[1], **Débora da Conceição Araújo** [2,3], **Máverick André Dionísio Ferreira**[3], **Paulo Salgado Gomes de Mattos Neto**[3]

[1]Engenharia da Computação, Universidade Federal do Vale do São Francisco
Juazeiro, Bahia, Brazil

[2]Colegiado de Ciência da Computação, Universidade Federal do Vale do São Francisco
Salgueiro, Pernambuco, Brazil

[3]Centro de Informática, Universidade Federal de Pernambuco (UFPE)
Recife, Pernambuco, Brazil

anisio.batistafilho@discente.univasf.edu.br, debora.caraujo@univasf.edu.br,

amaverick70@gmail.com, psgmn@cin.ufpe.br

Abstract. *Fake news propagation has been a problem noted in several areas of society, for example, in the fight against the pandemic caused by the new coronavirus (Sars-Cov-2). Combating misinformation, especially on social networks, is fundamental to control the spread of the virus and, consequently, the pandemic. Therefore, this work built supervised learning models focused on identifying fake news about the Sars-Cov-2. As a result, 18 models were built and rated, their reached 0.62%, 0.82%, and 0.47% f-score for the labels considered (news, opinion, and fake).*

Resumo. *A disseminação de notícias falsas tem sido um problema notado em diversos setores da sociedade, e vem dificultando o combate à pandemia causada pelo novo coronavírus (Sars-Cov-2). Combater desinformação sobre o Sars-Cov-2, principalmente nas redes sociais, é de fundamental importância para o controle da propagação do vírus e, consequentemente, da pandemia. Diante disso, nesse trabalho são construídos modelos de aprendizado supervisionado focados na identificação de notícias falsas sobre o novo coronavírus. Como resultados, foram construídos e avaliados 18 modelos, os quais chegaram a alcançar 0.62%, 0.82% e 0.47% de f-score para as classes consideradas (news, opinion e fake).*

1. Introdução

Uma pandemia ocorre com a rápida disseminação de uma doença. Em geral, um vírus se espalha entre países e continentes em um curto espaço de tempo, como dias ou meses [Cunha 2004]. No ano de 2009, um cenário de pandemia foi causado pela influenza A (H1N1), caracterizada por uma infecção respiratória que apresentou um baixo índice de letalidade, variando entre 0,01% e 0,03% [Kelly 2011], mas um alto potencial de contágio. Em menos de um ano, havia casos registrados da influenza em mais de 200 países. De forma mais agressiva do que a influenza A (H1N1), o novo coronavírus, o Sars-Cov-2,

ultrapassou os 200 países com registros de pessoas infectadas pelo vírus em menos de 6 meses após sua descoberta [World Health Organisation 2021].

Em agosto de 2021, havia mais de 199 milhões de pessoas com a Covid-19, doença causada pelo Sars-Cov-2, além de mais de 4.23 milhões de mortes em todo o mundo [World Health Organisation 2021]. Na América do Sul, o Brasil é o país com maior proliferação do vírus, com mais de 19.9 milhões de casos e número superior a 557 mil pessoas mortas [Ministério da Saúde 2021]. Com a proliferação da Covid-19 na América Latina, observou-se uma crescente propagação de notícias falsas, do inglês *fake news*, relacionadas à doença nas Redes Sociais (RS). Para além do âmbito da saúde, a disseminação de notícias falsas nas RS é um problema nos diversos setores da sociedade e tem motivado a condução de pesquisas no mundo todo [Zhang and Ghorbani 2020, Freire and Goldschmidt 2019]. Um fato que contribui para a proliferação desse tipo de notícia são os algoritmos que consideram as preferências do usuário pois, a partir desses, as pessoas têm muito contato com conteúdos que corroboram com sua visão de mundo, o que não significa que há evidências científicas que comprovem as informações compartilhadas.

Ao considerar o contexto delineado, este estudo teve por objetivo atuar na tarefa de identificação de notícias falsas sobre a Covid-19 no Brasil. Para tal, foram analisadas postagens presentes na rede social Twitter, pois a influência desse tipo de notícia na plataforma é alvo de pesquisa em diversos trabalhos presentes na literatura [Ajao et al. 2018, Bovet and Makse 2019, Ebeling et al. 2020]. Em suma, as principais contribuições deste artigo podem ser sumarizadas da seguinte forma: 1) Construção de uma base de dados com informações compartilhadas no Twitter sobre a Covid-19, considerando o idioma português; 2) Rotulação da base de dados construída considerando as classes *news*, *opinion* e *fake*; 3) Construção e análise de modelos computacionais para identificação das notícias falsas sobre a Covid-19 no idioma português; 4) Análises das performances dos algoritmos a partir de testes estatísticos.

2. Trabalhos Relacionados

O problema relacionado à disseminação de notícias falsas ganhou notoriedade, em 2016, após sua possível influência nas eleições presidenciais dos EUA [Reis et al. 2019]. Ao considerar o problema para o idioma português, Monteiro et al. 2018 investigaram os impactos das notícias falsas em redes sociais. Para tal, propuseram o corpus Fake.Br, um conjunto de dados rotulado para identificação de documentos contendo notícias falsas. Foram coletadas manualmente 7200 notícias, sendo exatamente 3600 verdadeiras e 3600 falsas, entre janeiro de 2016 e janeiro de 2018. Na avaliação do corpus, utilizaram as métricas precision, recall e f-score para as classes *fake* e *true* e a acurácia geral.

Na área da saúde, a proliferação de notícias falsas também se mostra um problema importante. Em Waszak et al. 2018 foram analisados os links relacionados à saúde mais compartilhados entre os anos de 2012 e 2017, no idioma polonês. Cada link foi verificado quanto à presença de notícias falsas. Os resultados mostraram que 40% dos links compartilhados com mais frequência continham textos enganosos. Os links com conteúdo enganoso foram compartilhados mais de 450.000 vezes e "vacinas" era o assunto mais recorrente. Em [Ebeling et al. 2020, Ebeling et al. 2021] foi proposto um framework para analisar e caracterizar o comportamento de grupos com posturas opostas e a relação

com a polarização política, tendo como estudo de caso o cenário polarizado do COVID brasileiro. Os autores identificaram que a polarização política, entre as classes chamadas Cloroquiners e Quarenteners, influencia os argumentos da economia e da vida e um maior apoio/oposição ao presidente. Além disso, observou-se que como um grupo, a rede de Cloroquiners é mais fechada e conectada, e os Quarenteners têm um envolvimento político mais diverso com uma comunidade de usuários polarizada apenas com políticos de esquerda e seus partidários.

Este projeto se destaca aos demais por propor uma abordagem que considera o contexto de propagação de notícias falsas acerca da pandemia de Covid-19 no Brasil, um dos países mais afetados pela doença. Entre as principais contribuições estão a construção e rotulação de uma base de dados sobre o contexto do novo coronavírus para o idioma português. Por fim, são analisados os desempenhos de algoritmos de AM para a tarefa de classificação de texto, a partir das classes: *news*, *opinion* e *fake*.

3. Materiais e Métodos

Nesta seção estão descritos os processos de construção e pré-processamento da base de dados, além das técnicas de Processamento de Linguagem Natural e algoritmos de Aprendizado de Máquina utilizados durante os experimentos.

3.1. Base de dados

Para realização deste trabalho, foi construída uma base de dados a partir de postagens realizadas na rede social Twitter sobre o contexto da Covid-19 no Brasil. Para tal, foi utilizada a API gratuita Tweepy[1] durante o processo de coleta de dados. Para coletar dados representativos, uma string de busca foi refinada por meio de pesquisa sobre o contexto do novo coronavírus. A versão final da string foi composta das seguintes palavras-chave: covid OR covid19 OR coronavirus OR coronavac OR astrazeneca OR pfizer OR sputnik OR sinovac OR jansen OR johnson&johnson OR butantan OR fiocruz OR oxford OR vacina moderna OR butanvac.

O período de captura dos tuítes deste estudo se estendeu entre os dias 05/01/2021 e 13/01/2021. As postagens repetidas (retweet) foram excluídas. Deste modo, foram recuperadas um total de 9602 postagens e, destas, 1963 foram rotuladas. Para que houvesse uma maior validação do estudo e concordância entre os rotuladores, foi elaborado um guia de anotação[2], que contém todos os detalhes da rotulação e está dividido nas seguintes seções: 1. Introdução; 2. Sobre as classes; 3. Sobre o pertencimento dos tweets para com as respectivas classes e; 4. Sobre a anotação. Em suma, dois anotadores humanos rotularam os 1963 tuítes de forma manual, e anotaram em concordância 66,23% dos dados, alcançando uma concordância moderada, de 47,19%, no índice Cohen Kappa [Cohen 1960]. Durante o processo de anotação foram consideradas três classes:

- *fake* - tuíte sobre a covid-19/coronavírus com características de notícia, mas sem notícias oficiais que corroboram com o seu conteúdo; tuítes que descredibilizam uma informação cientificamente comprovada;

[1] https://docs.tweepy.org/en/stable/api.html

[2] https://drive.google.com/drive/folders/1TW72sbFtmEa-QGOkb7TphzlgmDjc_jP7?usp=sharing

- *opinion* - tuíte que, apesar de envolver a covid-19/coronavírus, não se trata de uma informação ou notícia;
- *news* - tuíte sobre a covid-19/coronavírus com características de notícia e que há notícias oficiais corroborando com o seu conteúdo.

Para as etapas de treinamento e teste dos algoritmos, apenas as postagens rotuladas em concordância entre os dois anotadores foram utilizadas. Desse modo, a versão final da base de dados foi concluída com um total de 1300 postagens rotuladas[3], sendo: 320 (24,62%) tuítes pertencentes à classe *news*; 275 (21,15%) tuítes pertencentes à classe *fake* e; 705 (54,23%) tuítes contidos na classe *opinion*.

3.2. Pré-processamento dos dados

De acordo com [Chen et al. 2018], quando se trata de aprendizado de máquina, o pré-processamento dos textos é um procedimento-chave para o desempenho dos métodos de classificação. As técnicas utilizadas na etapa de pré-processamento de textos podem variar de acordo com as características de cada base de dados. Neste projeto foram utilizadas as seguintes técnicas:

- Normalização - compreende a normalização de termos comuns em RS que não adicionam sentido ao texto para o aprendizado de máquina, por exemplo urls, endereços de e-mail, números de telefone, nome de usuário, datas, horários, hashtag e etc.;
- Remoção de stopwords - Consiste na exclusão de palavras que, em geral, não agregam significado semântico ao texto. A remoção de stopwords atinge, principalmente, palavras que representam artigos, conjunções e preposições [Lo et al. 2005]. Para tal, foram utilizadas as stopwords presentes na biblioteca NLTK para o idioma português. Os autores deste estudo também adicionaram à lista da NLTK, stopwords da escrita informal, comumente utilizadas nas redes sociais, ex.: "vc", "ta" e outras;
- Stemming - esse método consiste em reduzir as palavras ao seu radical [Plisson et al. 2004]. Esse método pode beneficiar a classificação do texto, tanto por reduzir o vocabulário de palavras quanto por se concentrar no sentido completo do texto, em vez de analisar o significado de cada palavra de modo individual;
- Vetorização/Discretização - parte dos modelos de AM não trabalham com textos em linguagem natural, portanto existem técnicas que visam associar os elementos textuais a matrizes numéricas. Neste projeto será utilizada a técnica Term-Frequency Inverse Document Frequency (TFIDF), técnica amplamente utilizada na literatura [Sjarif et al. 2019];
- Oversampling (Over) - acrescenta amostras à classe minoritária, de modo a tornar seus exemplos mais significantes para o modelo. Nesse estudo, foi utilizada a técnica Synthetic Minority Over-sampling Technique (SMOTE) [Chawla et al. 2002, Fernández et al. 2018];
- Undersampling (Under) - remove amostras da classe majoritária com a finalidade de tornar as amostras de cada classe balanceadas [Liu et al. 2008].

Neste trabalho foram realizados experimentos considerando o Over e Undersampling de forma individual e combinada, tendo em vista que os melhores resultados podem ser alcançados a partir da combinação dessas técnicas [Chawla et al. 2002].

[3] https://drive.google.com/drive/folders/1TW72sbFtmEa-QGOkb7TphzlgmDjc_jP7?usp=sharing

3.3. Metodologia dos experimentos

Foram realizados experimentos a partir dos seguintes modelos de AM: Decision Tree (DT) [Breiman et al. 2017], Naive Bayes (NB) [Zhang 2004], Support Vector Machine (SVM) [Hearst et al. 1998], Random Forest (RF) [Breiman 2001], AdaBoost (Ada) [Hastie et al. 2009] e XGBoost (XGB) [Chen and Guestrin 2016]. O conjunto de dados foi dividido aleatoriamente em um conjunto de treinamento (com 75% dos tuítes) e outro de teste, fora da amostra (com 25%).

Os modelos foram treinados a partir do método *k-fold crossvalidation* [Bengio and Grandvalet 2004], em que se divide o conjunto de treinamento em 10 subconjuntos diferentes, utilizando 9 deles para treino e o último para validação do modelo. Este procedimento foi repetido 10 vezes, alternando os dados para o conjunto de validação. O treinamento dos modelos foi realizado através do método de Grid Search[4], do Scikit-Learn, que recebe o modelo de AM, os parâmetros iniciais e o método de crossvalidation. O Grid Search verificada as diversas possibilidades de parâmetros e retorna a combinação que atingiu os melhores resultados para o algoritmo. Os melhores parâmetros são utilizados para realizar a predição do modelo e sua performance é avaliada. Para avaliar o desempenho dos algoritmos foi utilizada a métrica f-score [Ferri et al. 2009]. O cálculo da f-score é realizado a partir de duas outras métricas, *precision* e *recall*, em que *precision* pondera o número de instâncias recuperadas que são relevantes e *recall* pondera o número de instâncias relevantes que são recuperadas, como mostra as equações 1, 2 e 3.

$$precision = \frac{VerdadeirosPositivos}{VerdadeirosPositivos + FalsosPositivos} \tag{1}$$

$$recall = \frac{VerdadeirosPositivos}{VerdadeirosPositivos + FalsosNegativos} \tag{2}$$

$$f - score = 2 * \frac{precision * recall}{precision + recall} \tag{3}$$

Para cada algoritmo, 10 resultados de f-score foram gerados para que fosse possível identificar os modelos que alcançaram desempenho estatisticamente superior. Para as análises estatísticas, foram utilizados os testes: 1) *Shapiro-Wilk* [Shapiro and Wilk 1965] - os resultados desse teste indicam se os desempenhos dos modelos seguem ou não uma distribuição normal; 2) T-Student [Student 1908] - utilizado para analisar se há diferença entre os desempenhos dos modelos quando ambos obedecem a uma distribuição normal; 3) *Wilcoxon* [Wilcoxon 1992] - utilizado para analisar se há diferença entre os modelos quando pelo menos um não apresenta dados que obedecem a uma distribuição normal.

4. Resultados e Discussões

Esta seção está subdividida em duas partes principais: em 4.1 são apresentados os resultados dos modelos construídos e análises estatísticas e; em 4.2, são analisadas as características dos algoritmos que se destacaram em termos de performance.

[4] `https://scikit-learn.org/stable/modules/grid_search.html`

4.1. Desempenhos dos Modelos Computacionais

Esta seção apresenta os resultados e análises estatísticas obtidas por meio dos experimentos realizados, descrito na seção 3.3. Todas as técnicas de pré-processamento de texto, apresentadas em 3.2, foram utilizadas para construir os conjuntos de treinamento e teste para os algoritmos, variando apenas o uso das técnicas Over e Undersampling, conforme mostra a tabela 1. A tabela 1 apresenta a média de desempenho, a partir da medida f-score, alcançada por cada classificador em relação às classes *news*, *opinion* e *fake*.

Tabela 1. F-score e desvio padrão de cada modelo por classes (*label*)

	f-score/desvio padrão		
Modelo	**news**	**opinion**	**fake**
DT	0.51/0.06	0.78/0.05	0.38/0.06
DT+Over	0.47/0.06	0.75/0.04	0.40/0.10
DT+Over+Under	0.52/0.05	0.76/0.04	0.39/0.05
NB	0.56/0.09	0.79/0.03	0.33/0.13
NB+Over	0.56/0.07	0.68/0.05	0.43/0.08
NB+Over+Under	0.55/0.06	0.68/0.06	0.44/0.07
SVM	0.60/0.06	0.81/0.03	0.44/0.12
SVM+Over	0.61/0.06	0.81/0.03	0.44/0.10
SVM+Over+Under	0.60/0.06	0.81/0.03	0.43/0.10
RF	0.54/0.07	0.80/0.02	0.32/0.11
RF+Over	**0.62/0.05**	**0.82/0.03**	0.43/0.09
RF+Over+Under	0.59/0.06	**0.82/0.02**	0.41/0.09
Ada	0.52/0.08	0.78/0.04	0.46/0.06
Ada+Over	0.54/0.08	0.79/0.04	0.44/0.06
Ada+Over+Under	0.57/0.06	0.80/0.04	**0.47/0.09**
XGB	0.56/0.07	**0.82/0.03**	0.46/0.06
XGB+Over	0.55/0.06	0.81/0.03	**0.47/0.08**
XGB+Over+Under	0.54/0.06	0.81/0.03	0.44/0.10

Ao analisar os resultados, é possível observar que para cada classe um algoritmo diferente se destacou. No que se refere a detecção de notícias verdadeiras (*news*), o algoritmo Random Forest, fazendo uso da técnica de Oversampling, obteve os melhores resultados com, em média, 0.62% de f-score e apresentando um desvio padrão de 0.05.

Em relação à classe *opinion*, três modelos se destacaram ao alcançar uma f-score média de 0.82%, sendo eles: Random Forest com Oversampling; Random Forest com Over e Undersampling e; XGBoost. Para as análises realizadas a seguir, utilizou-se como algoritmo de referência o Random Forest com Over e Undersampling, por apresentar o menor desvio padrão durante a classificação. Ao analisar a classe-alvo (*fake*), dois modelos apresentaram 0.47% de f-score média, sendo: AdaBoost com Over e Undersampling e; XGBoost com Oversampling. Para as demais análises realizadas, o modelo de referência selecionado foi o XGBoost com Oversampling, por apresentar o menor desvio padrão durante a classificação.

Para analisar se os algoritmos que apresentaram o melhor desempenho médio de f-score são estatisticamente superiores aos demais, seus desempenhos foram comparadas

aos desempenhos de todos os outros modelos. Para tal, o primeiro passo foi conhecer a natureza da distribuição dos dados por meio do teste de normalidade de *Shapiro-Wilk* (ver seção 3.3), em que a hipótese nula considera que os dados provém de uma distribuição normal. Após a aplicação do teste de normalidade, para verificar se houve, de fato, diferença estatítica nos resultados, os testes *T-Student* ou de *Wilcoxon* são utilizados para as comparações. Os testes realizados consideraram um $\alpha = 0.05$, ou seja, possuem 95% de confiança.

Foram utilizados para fins de comparação os algoritmos: Random Forest com *Oversampling* (melhor desempenho na classe *news*); Random Forest com Over e *Under-sampling* (melhor performance na classe *opinion*) e; XGBoost com Oversampling (melhor desempenho na classe *fake*). A tabela 2 detalha as comparações realizadas entre o modelo Random Forest com oversampling em relação aos demais algoritmos, no que se refere à classe *news*. Durante os testes de normalidade não foi possível rejeitar a hipótese nula quanto a distribuição das f-scores de nenhum dos modelos, portanto o teste paramétrico *T-Student* foi utilizado em todos os casos.

Tabela 2. Comparação estatística entre o modelo RF+Over e demais algoritmos em relação à classe *news*

Modelos	Teste	Valor p	Resultado
RF+Over e DT	*T-Student*	0.000	Rejeita H0
RF+Over e DT+Over	*T-Student*	0.000	Rejeita H0
RF+Over e DT+Over+Under	*T-Student*	0.000	Rejeita H0
RF+Over e NB	*T-Student*	0.084	Não rejeita H0
RF+Over e NB+Over	*T-Student*	0.038	Rejeita H0
RF+Over e NB+Over+Under	*T-Student*	0.015	Rejeita H0
RF+Over e SVM	*T-Student*	0.369	Não rejeita H0
RF+Over e SVM+Over	*T-Student*	0.562	Não rejeita H0
RF+Over e SVM+Over+Under	*T-Student*	0.396	Não rejeita H0
RF+Over e RF	*T-Student*	0.012	Rejeita H0
RF+Over e RF+Over+Under	*T-Student*	0.235	Não rejeita H0
RF+Over e Ada	*T-Student*	0.002	Rejeita H0
RF+Over e Ada+Over	*T-Student*	0.011	Rejeita H0
RF+Over e Ada+Over+Under	*T-Student*	0.042	Rejeita H0
RF+Over e XGB	*T-Student*	0.039	Rejeita H0
RF+Over e XGB+Over	*T-Student*	0.017	Rejeita H0
RF+Over e XGB+Over+Under	*T-Student*	0.006	Rejeita H0

É possível observar que o modelo Random Forest com Oversampling possui desempenho estatisticamente superior a 12 dos 18 modelos presentes nos experimentos. Desse modo, o modelo se apresentou como uma boa alternativa para a classificação de notícias a partir dos testes realizados nessa base de dados. A tabela 3 detalha as comparações realizadas entre o modelo Random Forest com Over e Undersampling em relação aos demais algoritmos, no que se refere à classe *opinion*. Assim como ocorreu nos experimentos anteriores, não foi possível rejeitar a hipótese nula durante os testes de normalidade, sendo o teste *T-Student* utilizado em todas as comparações de desempenho.

Tabela 3. Comparação estatística entre o modelo RF+Over+Under e demais algoritmos em relação à classe *opinion*

Modelos	Teste	Valor p	Resultado
RF+Over+Under e DT	*T-Student*	0.037	Rejeita H0
RF+Over+Under e DT+Over	*T-Student*	0.000	Rejeita H0
RF+Over+Under e DT+Over+Under	*T-Student*	0.005	Rejeita H0
RF+Over+Under e NB	*T-Student*	0.093	Não rejeita H0
RF+Over+Under e NB+Over	*T-Student*	0.000	Rejeita H0
RF+Over+Under e NB+Over+Under	*T-Student*	0.000	Rejeita H0
RF+Over+Under e SVM	*T-Student*	0.598	Não rejeita H0
RF+Over+Under e SVM+Over	*T-Student*	0.541	Não rejeita H0
RF+Over+Under e SVM+Over+Under	*T-Student*	0.543	Não rejeita H0
RF+Over+Under e RF	*T-Student*	0.084	Não rejeita H0
RF+Over+Under e RF+Over	*T-Student*	0.833	Não rejeita H0
RF+Over+Under e Ada	*T-Student*	0.052	Não rejeita H0
RF+Over+Under e Ada+Over	*T-Student*	0.182	Não rejeita H0
RF+Over+Under e Ada+Over+Under	*T-Student*	0.294	Não rejeita H0
RF+Over+Under e XGB	*T-Student*	0.875	Não rejeita H0
RF+Over+Under e XGB+Over	*T-Student*	0.812	Não rejeita H0
RF+Over+Under e XGB+Over+Under	*T-Student*	0.743	Não rejeita H0

Tabela 4. Comparação estatística entre o modelo XGB+Over e demais algoritmos em relação à classe *fake*

Modelos	Teste	Valor p	Resultado
XGB+Over e DT	T-Student	0.008	Rejeita H0
XGB+Over e DT+Over	*T-Student*	0.112	Não rejeita H0
XGB+Over e DT+Over+Under	*T-Student*	0.011	Rejeita H0
XGB+Over e NB	*T-Student*	0.009	Rejeita H0
XGB+Over e NB+Over	*T-Student*	0.230	Não rejeita H0
XGB+Over e NB+Over+Under	*T-Student*	0.280	Não rejeita H0
XGB+Over e SVM	*T-Student*	0.446	Não rejeita H0
XGB+Over e SVM+Over	*T-Student*	0.420	Não rejeita H0
XGB+Over e SVM+Over+Under	*T-Student*	0.357	Não rejeita H0
XGB+Over e RF	*T-Student*	0.003	Rejeita H0
XGB+Over e RF+Over	*T-Student*	0.303	Não rejeita H0
XGB+Over e RF+Over+Under	*T-Student*	0.105	Não rejeita H0
XGB+Over e Ada	*T-Student*	0.622	Não rejeita H0
XGB+Over e Ada+Over	*Wilcoxon*	0.262	Não rejeita H0
XGB+Over e Ada+Over+Under	*T-Student*	0.883	Não rejeita H0
XGB+Over e XGB	*T-Student*	0.743	Não rejeita H0
XGB+Over e XGB+Over+Under	*T-Student*	0.465	Não rejeita H0

Por meio da tabela 3 é possível observar que o modelo Random Forest com Over e Undersampling apresentou um desempenho estatisticamente superior a 5 dos 18 modelos analisados para a classe *opinion*. Essa classe obteve os melhores desempenhos gerais, pois

em 10 dos 18 modelos analisados foi alcançada uma média de f-score igual ou superior a 80% (ver tabela 1). Acredita-se que esse bom desempenho dos modelos na classe *opinion* se deve, em parte, a quantidade de exemplos rotulados com essa classe, que representa 54.23% das amostras de toda a base de dados.

A tabela 4 detalha as comparações realizadas entre o modelo XGBoost com Oversampling em relação aos demais algoritmos, no que se refere à classe *fake*. Durante os testes de normalidade, foi possível rejeitar a hipótese nula apenas nos dados do modelo AdaBoost com oversampling, sendo assim, este modelo teve seu desempenho comparado por meio do teste de *Wilcoxon* e os demais por meio do teste T-Student.

O modelo XGBoost com Oversampling, mesmo alcançando a melhor f-score média, apresentou um desempenho estatisticamente superior a apenas 4 dos 18 modelos construídos para a classe *fake*. Analisar o desempenho dos algoritmos por classe, e não apenas o desempenho geral, permite avaliar as possibilidades de melhoria, principalmente, no que se refere a predição da classe-alvo. Neste trabalho, atribui-se as dificuldades de classificação, principalmente, ao fato de a base de dados ser desbalanceada e possuir um número considerado pequeno de exemplos anotados.

4.2. Discussões dos Melhores Modelos

Conforme apresentado na seção (4.1), os algoritmos Random Forest e XGBoost obtiveram as melhores performances nas classificações, sendo o RF melhor ao lidar com as classes *news* e *opinion* e o XGBoost melhor ao lidar com a classe *fake*. Além disso, analisando a classe-alvo (*fake*), observou-se que não foi encontrada diferença estatística entre os desempenhos desses modelos. O ponto em comum entre os dois algoritmos é que ambos são *ensembles* [Dietterich 2000] baseados em árvores e, durante seus treinamentos, buscam estimar a importância de uma característica individual para o modelo por meio da métrica *Mean Decrease Gini* (MDG) [Breiman 2001]. Ao observar o número de características avaliadas pelos referidos algoritmos (RF e XGBoost) com MDG superior à zero, foi possível perceber que: o algoritmo Random Forest + Oversampling avaliou, durante o processo de treinamento, um total de 2540 características e; o XGBoost + Oversampling avaliou um total de 345. Isso evidencia uma menor dimensionalidade do modelo construído pelo XGBoost, no que se refere ao número de características consideradas preditivas, o que faz desse modelo uma boa alternativa ao lidar com matrizes esparsas, como costuma ser o caso de dados textuais.

A tabela 5 apresenta um ranking com as quinze características que foram consideradas mais preditivas (de acordo com o MDG) por cada algoritmo na tarefa de classificar os tuítes em *news*, *opinion* ou *fake*. Como pode ser observado, dentre as características mais importantes, o XGBoost pontuou as palavras: ciência - (MDG 1.892), precoce - (MDG 1.123), vachina - (MDG 1.016) e ivermectina - (MDG 0.964). Já o Random Forest pontuou: vacina - (MDG 2.458), eficácia - (MDG 1.246) e china - (MDG 0.738).

É possível perceber, a partir das características consideradas mais relevantes para a construção dos modelos que, apesar de lidar com uma base pequena e desbalanceada, ambos os algoritmos identificaram as palavras utilizadas para tratar dois dos temas-chave no enfrentamento da pandemia no Brasil, são eles: Vacinação e Tratamento precoce.

Tabela 5. Ranking de características - XGBoost e Random Forest

Top-15 de Características – XGBoost			Top-15 de Características - Random Forest		
Característica	**Descrição**	**MDG**	**Característica**	**Descrição**	**MDG**
men	1573	2,103	vacin	2499	2,458
cienc	447	1,892	eficac	841	1,246
poi	1890	1,426	brasil	326	1,100
contr	577	1,385	dos	818	1,006
tant	2367	1,333	covid	620	0,965
duvid	831	1,288	dor	816	0,897
ment	1577	1,284	men	1573	0,827
precoc	1938	1,123	chin	436	0,738
poli	1893	1,109	milho	1597	0,648
parc	1808	1,109	dobr	799	0,577
estud	971	1,104	menos	1574	0,577
vachin	2497	1,016	compr	515	0,567
tax	2373	1,004	parc	1808	0,542
leit	1464	0,979	govern	1178	0,507
ivermectin	1395	0,964	contr	577	0,476

5. Considerações Finais

Esta pesquisa aborda o problema de proliferação de notícias falsas sobre a Covid-19 considerando o idioma português. Para tal, uma base de dados proveniente do Twitter foi construída e rotulada de forma manual considerando as classes *news*, *opinion* e *fake*.

Diferentes técnicas de pré-processamento de texto antecederam a construção dos métodos de classificação analisados. Por fim, 18 modelos foram construídos a partir dos algoritmos: Decision Tree, Naive Bayes, SVM, Random Forest, AdaBoost e XGBoost. Diferentes configurações desses algoritmos foram combinadas às técnicas de Oversampling e Undersampling, devido a natureza desbalanceada da base de dados construída. O algoritmo XGBoost combinado à técnica de Oversampling apresentou o melhor desempenho médio para a classe-alvo, com 0.47% de f-score, e performance geral de 0.61% (f-score) nessa tarefa de classificação. Vale ressaltar que este estudo priorizou a análise de cada classe de forma individual para que o desempenho da classe-alvo (*fake*) fosse observado em detalhes. Ao passo que analisamos classes com uma maior quantidade de exemplos, melhores são os desempenhos dos modelos para dada classe. Por esse motivo, tem-se em 10 dos 18 algoritmos utilizados um desempenho igual ou superior a 80% de f-score quando se trata da classe *opinion*, majoritária na base de dados.

Como trabalhos futuros, pretende-se aumentar a quantidade de amostras do corpus, além de considerar técnicas que possibilitem a montagem de vetores de características com baixa dimensionalidade e alta capacidade preditiva (análises sintáticas, semânticas). Pretende-se, ainda, analisar o desempenho de métodos semi-automáticos de classificação, considerando que o volume de dados é extenso, porém a rotulação manual pode ser uma tarefa excessivamente demorada e cansativa. Por fim, espera-se com este estudo corroborar com as pesquisas que envolvem o problema de detecção de notícias falsas considerando a língua portuguesa.

Referências

Ajao, O., Bhowmik, D., and Zargari, S. (2018). Fake news identification on twitter with hybrid cnn and rnn models. In *Proceedings of the 9th international conference on social media and society*, pages 226–230.

Bengio, Y. and Grandvalet, Y. (2004). No unbiased estimator of the variance of k-fold cross-validation. *Journal of machine learning research*, 5(Sep):1089–1105.

Bovet, A. and Makse, H. A. (2019). Influence of fake news in twitter during the 2016 us presidential election. *Nature communications*, 10(1):1–14.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. (2017). *Classification and regression trees*. Routledge.

Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357.

Chen, T. and Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pages 785–794, New York, NY, USA. ACM.

Chen, Y., Zhou, B., Zhang, W., Gong, W., and Sun, G. (2018). Sentiment analysis based on deep learning and its application in screening for perinatal depression. In *2018 IEEE Third International Conference on Data Science in Cyberspace (DSC)*, pages 451–456. IEEE.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Cunha, B. A. (2004). Influenza: historical aspects of epidemics and pandemics. *Infectious Disease Clinics*, 18(1):141–155.

Dietterich, T. G. (2000). Ensemble methods in machine learning. In *International workshop on multiple classifier systems*, pages 1–15. Springer.

Ebeling, R., Sáenz, C. A. C., Nobre, J., and Becker, K. (2020). Quarenteners vs. chloroquiners: A framework to analyze how political polarization affects the behavior of groups. In *2020 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, pages 203–210. IEEE.

Ebeling, R., Sáenz, C. A. C., Nobre, J., and Becker, K. (2021). The effect of political polarization on social distance stances in the brazilian covid-19 scenario.

Fernández, A., Garcia, S., Herrera, F., and Chawla, N. V. (2018). Smote for learning from imbalanced data: progress and challenges, marking the 15-year anniversary. *Journal of artificial intelligence research*, 61:863–905.

Ferri, C., Hernández-Orallo, J., and Modroiu, R. (2009). An experimental comparison of performance measures for classification. *Pattern Recognition Letters*, 30(1):27–38.

Freire, P. and Goldschmidt, R. (2019). Combatendo fake news nas redes sociais via crowd signals implícitos. In *Anais do XVI Encontro Nacional de Inteligência Artificial e Computacional*, pages 424–435. SBC.

Hastie, T., Rosset, S., Zhu, J., and Zou, H. (2009). Multi-class adaboost. *Statistics and its Interface*, 2(3):349–360.

Hearst, M. A., Dumais, S. T., Osuna, E., Platt, J., and Scholkopf, B. (1998). Support vector machines. *IEEE Intelligent Systems and their applications*, 13(4):18–28.

Kelly, H. (2011). The classical definition of a pandemic is not elusive. *Bulletin of the World Health Organization*, 89:540–541.

Liu, X.-Y., Wu, J., and Zhou, Z.-H. (2008). Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 39(2):539–550.

Lo, R. T.-W., He, B., and Ounis, I. (2005). Automatically building a stopword list for an information retrieval system. In *Journal on Digital Information Management: Special Issue on the 5th Dutch-Belgian Information Retrieval Workshop (DIR)*, volume 5, pages 17–24.

Ministério da Saúde (2021). Coronavírus no brasil. `https://covid.saude.gov.br/`, Accessed on 08/02/2021.

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 *International Conference on Computational Processing of the Portuguese Language*, pages 324–334. Springer.

Plisson, J., Lavrac, N., Mladenic, D., et al. (2004). A rule based approach to word lemmatization. In *Proceedings of IS*, volume 3, pages 83–86.

Reis, J. C., Correia, A., Murai, F., Veloso, A., and Benevenuto, F. (2019). Supervised learning for fake news detection. *IEEE Intelligent Systems*, 34(2):76–81.

Shapiro, S. S. and Wilk, M. B. (1965). An analysis of variance test for normality (complete samples). *Biometrika*, 52(3/4):591–611.

Sjarif, N. N. A., Azmi, N. F. M., Chuprat, S., Sarkan, H. M., Yahya, Y., and Sam, S. M. (2019). Sms spam message detection using term frequency-inverse document frequency and random forest algorithm. *Procedia Computer Science*, 161:509–515.

Student (1908). The probable error of a mean. *Biometrika*, pages 1–25.

Waszak, P. M., Kasprzycka-Waszak, W., and Kubanek, A. (2018). The spread of medical fake news in social media–the pilot quantitative study. *Health policy and technology*, 7(2):115–118.

Wilcoxon, F. (1992). Individual comparisons by ranking methods. In *Breakthroughs in statistics*, pages 196–202. Springer.

World Health Organisation (2021). Who coronavirus (covid-19) dashboard. `https://covid19.who.int/`, Accessed on 08/02/2021.

Zhang, H. (2004). The optimality of naive bayes. *In: Association for the Advancement of Artificial Intelligence (AAAI)*, 1(2):3.

Zhang, X. and Ghorbani, A. A. (2020). An overview of online fake news: Characterization, detection, and discussion. *Information Processing & Management*, 57(2):102025.