Identificação de "*Fake News*" no contexto político brasileiro: uma abordagem computacional

Abstract. *This paper shows a computational solution's main results to analyze Brazilian fake news in a political context, in order to investigate which Machine Learning Algorithm, between Support Vector Machine and Naive Bayes, reach the best result to classify, in a natural language context, whether Brazilian political news is fake or not. The better performance was reached by combining SVM (RBF) + BOW with 80,4% accuracy, 82% precision, 76% recall, 78% of F1-Score, and 88% of AUC. The non-probabilistic algorithms proved to be better in the classification of fake news, thus, the results allow to present a path for future works.*

Resumo. *Este artigo apresenta os principais resultados de uma solução computacional para analisar as notícias falsas brasileiras em um contexto político, de modo a investigar qual algoritmo de aprendizado de máquina, entre Support Vector Machine e Naive Bayes, atinge o melhor resultado para classificar, em um contexto de linguagem natural, se uma notícia política é falsa ou não. O melhor desempenho foi alcançado pela combinação de SVM (RBF) + BOW com 80,4% de precisão, 82% de precisão, 76% de recuperação, 78% de F1-Score e 88% de AUC. Os algoritmos não probabilísticos se mostraram melhores na classificação de notícias falsas, sugerindo um caminho para trabalhos futuros nesta área de pesquisa.*

1. Introdução

O dicionário Macquarie elegeu *fake news* como a palavra do ano em 2016, por causa de seu uso na corrida presidencial norte-americana que elegeu o ex-presidente Donald Trump, onde ferramentas analíticas foram usadas em sua campanha para mapear perfis específicos e lançar notícias falsas nas redes sociais que foram compartilhadas por milhões de pessoas [Bovet e Makse 2019].

Além disso, em 2018 no Brasil houve a circulação de notícias falsas no whatsapp, compartilhado por robôs ou *bots* usado nas eleições presidenciais do até então candidato e atual presidente Jair Bolsonaro. De acordo com Abdin (2019), 75% da polução brasileira informou que utilizava a televisão para ver noticias em 2018, sendo que 66% da população brasileira afirmou ter acesso a conexões com a internet, o que faz com que o Brasil seja o país que mais utiliza a internet, perdendo apenas para os Estados Unidos. Ainda, Adbin (2019) acrescentou, com uma pesquisa realizada pela Universidade de Oxford e Reuters Institute, que 66% dos brasileiros que responderam a pesquisa utilizam as midias sociais para ver notícias (Facebook, WhatsApp, Instagram, Twiter, entre outras), destacando que desta forma tornam-se mais suscetíveis a receberem notícias falsas.

As notícias online são mais rápidas e fáceis de consumir, devido à facilidade que as pessoas têm em acessar, comentar e compartilhar e porque essas notícias são menos caras se comparadas com outros canais [Shu et al 2017]. Logo, a qualidade do conteúdo consumido online é colocada em dúvida, dado que boa parte do fluxo de acessos é proveniente de pessoas que interagem com estas notícias por meio das mídias sociais.

As facilidades e ações destacadas contribuem para o alto volume de notícias falsas, que são criadas para diversos propósitos, entre eles: ganho político, financeiro, confundir o leitor, entre outros [Shu et al 2017]. Notícias falsas não são atuais, existem há muito tempo, mesmo antes do primeiro jornal impresso em 1439 e podem ser denominadas como qualquer notícia falsa que tenha um sério impacto negativo sobre o indivíduo e a sociedade.

Páginas que compartilham notícias falsas têm maior participação dos usuários nas mídias sociais do que aquelas que têm conteúdo jornalístico real. O uso de mídias tradicionais como meio de informação diminuiu ao longo dos anos, devido principalmente ao uso de redes sociais. De 2017 a 2018, os meios de comunicação tradicionais caíram 17% em engajamento (interação), enquanto os disseminadores de notícias falsas tiveram um aumento de 61% [Bondielli et al 2019].

A pandemia do novo coronavírus é um exemplo de tema com ampla divulgação de desinformação com alto índice de engajamento, onde desde o final de janeiro de 2020, a Organização Mundial da Saúde vem lutando contra a disseminação de notícias falsas sobre ela, desde textos até vídeos que fornecem recomendações erradas para prevenir essa doença, que podem ser prejudiciais à saúde [Sharma et al 2020].

De modo a combater esse problema, técnicas de aprendizado de máquina têm apresentado bastante sucesso no reconhecimento de padrões, por esse motivo elas estão sendo cada vez mais utilizadas para a análise de texto de notícias falsas [El Naqa et al 2015]. Atualmente, o processo de identificação de vários sites que o realizam é feito manualmente. Como exemplo de outras pesquisas para identificar notícias falsas por meio de *machine learning*, há a pesquisa em que se relata como suas características são identificadas e os métodos utilizados para avaliar se um item de notícias é verdadeiro ou falso [Monteiro et al 2018].

As notícias falsas têm como característica principal a falsidade intencional ou consciente de suas publicações [Klein et al 2017]. No entanto, houve um estudo sobre o uso da palavra "notícia falsa" que foi além, dividindo-a em algumas categorias que foram utilizadas em estudos anteriores, entre elas está a categoria de notícias fabricadas, que se refere a publicações que não são completamente precisas e são publicadas no estilo de notícias para criar legitimidade [Tandoc et al 2018].

Nesta pesquisa, duas premissas foram assumidas (i) notícias falsas são notícias fabricadas e (ii) sites de grandes empresas de comunicação são adotados como fontes confiáveis de informação. Embora esses sites possam ter algum viés político em suas notícias, eles tomam cuidado com os fatos publicados para que sua imagem não seja prejudicada. Isso ocorre devido ao fato de grandes empresas não quererem serem afetadas negativamente pela disseminação de notícias falsas.

Além disso, houve uma pesquisa realizada pela Associação Brasileira de Comunicação Empresarial (ABERJE) entre 27 de fevereiro e 4 de abril de 2018 com 52 empresas nacionais e internacionais, nas quais a disseminação de notícias falsas diz respeito a 85% das empresas onde as principais preocupações são danos à marca da empresa, imagem, credibilidade e perda econômica, por isso muitas empresas evitam compartilhar notícias falsas [Adriani 2019].

2. Trabalhos Correlatos

Existem notícias falsas que podem ser facilmente detectadas pelos seres humanos, entretanto outras tentam persuadir o leitor e a distorção da informação ocorre de forma sutil, de modo que a identificação manual pode ser mais difícil, e o mesmo ocorre quando envolve uma quantidade imensa de notícias.

Logo, este estudo envolveu dois algoritmos de aprendizado de máquina SVM, que apresentou excelentes resultados nas pesquisas de Monteiro et al (2018) e *Naive Bayes*, que apresentou bons resultados nas pesquisas de Granik et al (2017), Bharadwaj et al (2019) para detectar notícias falsas na língua inglesa e Dias (2019) para detectar notçias falsas brasileiras. Deste modo, este estudo objetiva comparar e identificar qual dos dois algoritmos de aprendizado de máquina apresenta melhores resultados quando empregados no processo de classificação uma notícia brasileira, de cunho político e em português.

De acordo com Rubin, Chen e Conroy (2016) existem três tipos de notícias falsas: as satíricas, as baseadas em verdades e as fabricadas. As satíricas são comumente encontradas em sites de humor que satirizam figuras públicas. As baseadas em verdade distorcem os fatos de notícias verdadeiras para criar uma certa credibilidade, desse modo tornam-se mais críveis. As fabricadas não são necessariamente baseadas notícias reais, podendo ser baseadas em dados puramente fictícios.

Para esta pesquisa os três tipos de notícias falsas foram utilizados, entretanto no projeto atual, outros atributos além do texto de notícias em si não foram considerados. De acordo com Zhou et al (2020) e outras obras utilizadas como base para o projeto de Monteiro et al (2018) e Shu et al (2017), existem alguns atributos que fazem a ponte entre a exatidão dos modelos de detecção de notícias falsas, tais como: número de palavras, número de pontos, número de parágrafos etc.

O processamento da linguagem natural é uma área de inteligência artificial cujo objetivo é a interpretação e manipulação das línguas humanas. Esse processamento geralmente envolve traduzir linguagem natural em dados (números) que um computador pode usar para aprender sobre o mundo [Lane et al 2019] e esse processo é chamado de vetorização. Neste projeto será utilizado um comparativo entre os desempenhos dos modelos com as técnicas: *bag of words* e TF-IDF (ou *Term frequency–inverse document frequency*).

Ao utilizar a técnica *bag of words*, o texto original é transformado em um conjunto de palavras e a frequência que uma palavra aparece no texto é calculada. A saída deste método é uma matriz, onde cada coluna representa uma palavra no vocabulário e cada linha corresponde a um texto [Ghosh et al 2019], a saída será o número de vezes que a palavra aparece no mesmo.

No contexto do projeto, a técnica *Bag of words* será a representação do texto das notícias, para que os modelos possam compreender e mapear os padrões, e ser avaliado se a melhor representação para problemas de classificação de notícias falsas seria vetores de frequência de palavras ou uma vetorização que leve em conta a importância das palavras em todas as notícias, TF-IDF.

A técnica TF-IDF (Frequência do Termo Inverso da Frequência nos Documentos), é uma estatística numérica que se destina a refletir a importância de uma palavra no corpus textual. Comparado ao *Bag of words*, essa técnica não é influenciada por stop words, pois

não leva em conta apenas a frequência de uma palavra em um único texto, mas sua importância em relação a todas as outras notícias, portanto essa vetorização destaca os termos mais relevantes [Jivani 2011]. A saída é semelhante ao *Bag of words*, onde cada linha representa um texto e a coluna corresponde às palavras do vocabulário [Lane et al 2019], a diferença está nos valores da matriz.

$$FrequênciaDoTermo(i,j) = \frac{Frequência\ do\ termo\ no\ texto\ j}{Total\ de\ palavras\ no\ texto\ j} \quad (1)$$

$$IDF(i) = \log 10\left(\frac{Total\ de\ textos}{Textos\ que\ mais\ têm\ a\ palavra\ i}\right) \quad (2)$$

$$TFIDF(i) = \text{FrequenciaDoTermo(i,j)} \times \text{IDF(i)} \quad (3)$$

O TF-IDF realiza o cálculo da frequência que o termo aparece no texto e a frequência de documentos inversos (IDF), ou seja, o peso que a palavra tem em relação a todos os textos, e assim é calculado o log10 da divisão do total de textos pelos textos que possuem a palavra (Equação 2). Finalmente, para encontrar o TF-IDF, o TF (*Term Frequency*) é multiplicado pelo IDF (*Inverse Document Frequency*), Equação 3.

De acordo com Freire e Goldschmidt (2019) a detecção automática de notícias falsas pode ser interpretada como um problema de classificação binaria, onde dado uma notícia como uma entrada (ε), a tarefa de detectar notícias falsas é prever se essa notícia é falsa ou não, ou seja, f: ε de modo que f é a função de previsão.

Quanto ao funcionamento das técnicas, pode-se destacar que o algoritmo SVM (*Support Vector Machine*) é um algoritmo não probabilístico que tenta adaptar um vetor (no plano ou hiperplano) entre diferentes classes, com o intuito de encontrar uma separação robusta entre as classes (notícia falsa e notícia verdadeira). Após o modelo definir o melhor vetor de suporte, ele será capaz de prever a qual classe um novo dado pertence ao verificar a qual conjunto o dado é mais aderente [Dias 2019].

SVM lineares definem borsas lineares a partir de dados linearmente [Freire e Goldschimidt 2019], ou seja, se o conjunto de treinamento X com n objetos e seus rótulos Y, X é linearmente separável se for possível separar objetos das classes Y por um hiperplano, onde Y = {-1,1}.

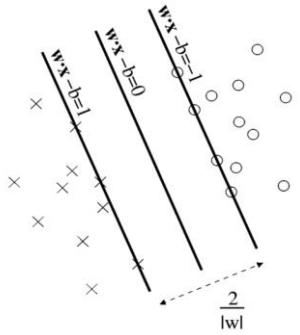

Figura 1. Representação da máquina de vetores de suporte linear

A equação do hiperplano linear é apresentada na Equação 4, onde w × x é o produto escalar entre os vetores w e x. A equação do hiperplano pode ser usada para dividir o espaço de entrada X, em duas regiões: w×x+b>0 e w×x+b<0 [Freire e Goldschimidt 2019], (Figura 1). Portanto, para problemas de classificação, a Equação 6 pode ser obtida.

$$h(x) = w \times x + b \quad (4)$$

$$g(x) = sgn\big(h(x)\big) = f(x) = \begin{cases} +1, & w \times x + b < 0 \\ -1, & w \times x + b \geq 0 \end{cases} \quad (6)$$

Há muitos casos em que não é possível dividir satisfatoriamente o treinamento estabelecido por um hiperplano, um exemplo seria a divisão deles por curvas diretas. De acordo com Hearst (1998), SVMs lidam com problemas não lineares mapeando-os de seu espaço original, para um espaço maior, chamado espaço de características, a escolha apropriada do mapeamento de conjunto de dados pode ser separada por um SVM linear. Este truque é chamado de *kernel trick* ou truque do *kernel* [Dias 2019]. Para fins de estudo, ambas as abordagens de SVM serão utilizadas no projeto para medir o desempenho de classificação de ambos os espaços.

O *Naive Bayes* é um classificador probabilístico que assume a independência entre os atributos dos dados [Harrison 2019]. Métodos probabilísticos bayesianos assumem que a probabilidade de um evento A, representado por uma variável alvo, como por exemplo, a classe “falsa”, dado um evento B, representado por valores de atributos de entrada [Freire e Goldschimidt 2019], por exemplo: as representações numéricas da notícia.

O algoritmo estima a probabilidade condicional de cada recurso de um determinado texto para cada classe com base na ocorrência desse recurso nessa classe e multiplica essas probabilidades de todos os recursos de um determinado texto para calcular a probabilidade final de classificação para cada classe [Vajjala 2020].

A e B são os eventos e P (A | B) é a probabilidade condicional de que A aconteça já que B é verdade. P (B | A) é a probabilidade condicional de que B aconteça, uma vez que A é verdadeiro e P (B) e P (A) são as probabilidades de observar A e B independentemente um do outro. Ao considerar X ($x_1, \ldots, x_n$) como os recursos de entrada pelo modelo [Dias 2019], por exemplo: representação numérica das notícias, e C como as classes de problema, por exemplo, falsa ou não falsa,.

No contexto do projeto, a probabilidade de que um item de notícia X pertença a uma classe C, falsa ou não falsa, será aquela que tem a maior probabilidade de que este item X esteja associado. Ambos os algoritmos foram selecionados devido a sua utilização em diversas pesquisas de classificação de notícias falsas, como Monteiro et al (2018) e Dias (2019), e neste projeto serão analisados a partir de um contexto de linguagem natural, de modo a identificar qual algoritmo apresenta melhor desempenho entre o probabilístico ou o não probabilístico.

Após o treinamento dos modelos, eles foram avaliados utilizando as seguintes métricas [Harrison 2019]: matriz de confusão (para facilitar o entendimento do desempenho dos classificadores e para obter os Erros Tipo I e Tipo II), acurácia (é a porcentagem de classificações corretas), precisão (é a porcentagem de predições positivas que estavam corretas [Halimu 2019], recall (é a porcentagem de valores positivos classificados corretamente), F1 score (média harmônica entre o recall e a precisão, percentual de valores positivos corretamente classificados, precisão) e curva ROC (uma das formas de fazer a análise de classificadores em problemas binários [Davis e Goadrich] e para que o algoritmo seja considerado "bom" ele precisa ter uma protuberância em relação ao canto superior esquerdo).

3. Metodologia

Ao final desta pesquisa, pretende-se responder à seguinte pergunta de pesquisa: Dentre o SVM e o *Naive Bayes*, qual algoritmo apresenta o melhor resultado para classificar, em um contexto de linguagem natural, se uma notícia política brasileira é falsa ou não? Como isso pode afetar o contexto político brasileiro?

Para a primeira pergunta foram formuladas as seguintes hipóteses de pesquisa:

> **Hipótese Nula (H0):** Não foi possível chegar a uma conclusão porque ambos os resultados foram insuficientes ou muito semelhantes para chegar a uma solução.
>
> **Hipótese Alternativa (H1):** O algoritmo classificador SVM foi melhor que o *Naive Bayes* e apresentou bons resultados.
>
> **Hipótese Alternativa (H2):** O algoritmo classificador SVM foi pior do que *Naive Bayes* e apresentou resultados ruins.

Para coleta de dados, foram utilizados os *datasets* que compõem o Fake.Br Corpus [Monteiro et al 2018] com o filtro para notícias de categoria "política", resultando em um conjunto final de 3.000 notícias.

Tabela 1. Conjunto de dados para etapas de treinamento

Origem das instâncias (notícias)	Quantidade	Origem das instâncias (notícias)	Quantidade
www.diariodobrasil.org	1880	sustentabilidade.estadao.com.br	6
g1.globo.com/	1260	saude.estadao.com.br	5
politica.estadao.com.br	578	link.estadao.com.br	4
afolhabrasil.com.br/politica/	141	educacao.estadao.com.br	3
www1.folha.uol.com.br/	81	estadao.com.br	2
www.thejornalbrasil.com.br	54	viagem.estadao.com.br	2
cultura.estadao.com.br	37	territorioeldorado.limao.com.br	2
internacional.estadao.com.br	33	ciencia.estadao.com.br/blogs	1
economia.estadao.com.br	31	topfivetv.com	1
brasil.estadao.com.br	19	emais.estadao.com.br	1
ceticismopolitico.com	14	acervo.estadao.com.br	1
esportes.estadao.com.br/	11	datafolha.folha.uol.com.br	1
alias.estadao.com.br	7		

Os dados passaram por etapas por um conjunto de etapas de pré-processamento (ver Figura 2), vetorização, treinamento nos algoritmos de classificação SVM e *Naive Bayes* (ver Figura 3) e avaliação final que incluíram o cálculo do F1 Score, curva ROC, acurácia e Área Sob a Curva (AUC).

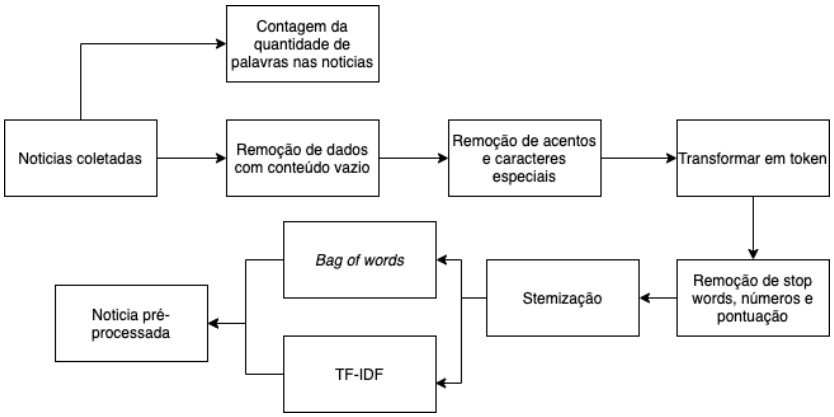

Figura 2. Processo de pré-processamento das notícias

Na etapa de pré-processamento, o conteúdo das notícias foi convertido em letras minúsculas, então elementos vazios, acentos e caracteres especiais foram removidos do conjunto de dados. Além disso, stop words (palavras que não trazem valor ao modelo), como preposições e artigos, também foram removidos usando a biblioteca NLTK. Depois disso foi feito stemização, o que reduz palavras flexionadas ao seu radical e, finalmente, vetorização, uma vez que as notícias são dadas não estruturados, *Bag of Words* e TF-IDF são usados para este processo, gerando também *bigram* e *n-gram* (Figura 2).

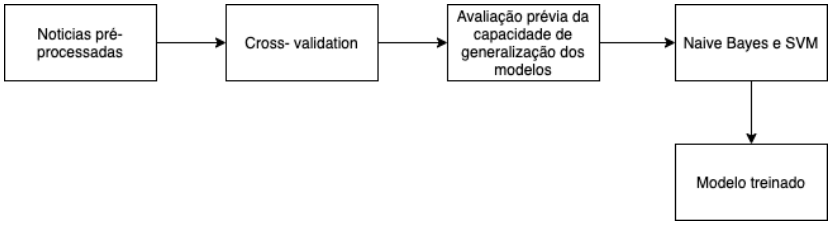

Figura 3. Processo de treinamento dos modelos

Os modelos foram treinados com *cross validation* de 5 *folds* [Rodriguez 2009], a saída desse processo é um conjunto de tamanho 3, dentro dele há um vetor para cada métrica que foi escolhida, como: precisão, recall, F1 Score e o valor AUC de cada um dos *folds* (subconjuntos). Assim no final de cada etapa a média de cada um deles foi calculada e assim ter uma avaliação prévia da capacidade de generalização dos modelos (Figura 3), e logo após isso os dados de teste são passados para o modelo treinado. Esta fase de teste dos modelos foi realizada com um conjunto de dados separado, onde a saída indicará se a notícia é ou não uma notícia falsa.

Para fins de medir o desempenho de classificação de ambos os *kernels* do SVM, será ajustado o parâmetro dele de modo que o *kernel* RBF (Função Radial Base), muito utilizado para dados não lineares, e o *kernel* linear serão testados, ambos implementados pelo algoritmo SVC da biblioteca de Scikit Learn, sendo necessário apenas ajustar o parâmetro *kernel*. E dois algoritmos da técnica *Naive Bayes* serão utilizados, sendo implementados com a biblioteca Scikit Learn, *MultinomialNB* e GaussianNB.

3.1. Dados para validação

Foi gerado um conjunto de dados para validação do modelo através de *web scraping* em python, utilizando as bibliotecas Beautifulsoup e Selenium, ambas abertas, com o Beautifulsoup projetado para análise de documentos HTML e XML e Selenium projetado para ser uma estrutura portátil para testar aplicações *web*.

Tabela 2: Conjunto de dados para etapas de teste.

Conjunto de dados de notícias do *WebSite*	Classe	Tamanho da amostra
https://contraponto.jor.br/	Falsa	291
https://www.conversaafiada.com.br/politica/	Falsa	180
https://g1.globo.com/politica/	Não Falsa	159
https://politica.estadao.com.br/noticias/	Não Falsa	131

Como mostrado na Tabela 2, foram mapeados dois sites que compartilham notícias falsas e dois sites considerados confiáveis, de acordo com a justificativa da Seção 1, para o conjunto de dados de treinamento. Neste conjunto foram coletados de cada notícia: link, título e conteúdo e foi inserido um campo indicando se essas notícias são falsas ou não.

4. Análise de dados

Durante a análise feita na base de treinamento, verificou-se que mais de 75% das notícias falsas têm menos de 218 caracteres de palavras, então notícias falsas tendem a ser mais curtas se comparadas com as não falsas e mais de 75% das notícias não falsas tem número de palavras menor que 1369 caracteres.

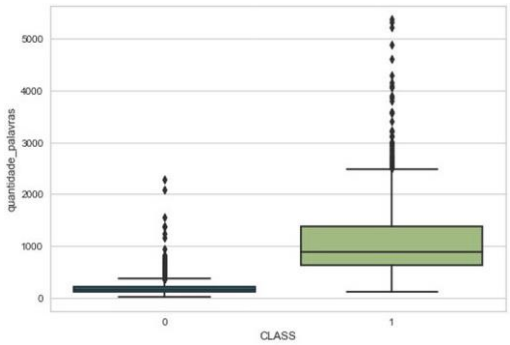

Figura 3. *Boxplot* da distribuição de quantidade de palavras

De acordo com a Figura. 3. É possível notar que a mediana das notícias não falsas, representadas pelo número 1, é muito maior do que 50% das notícias falsas, representadas pelo número 0. No entanto, é possível notar possíveis outliers, ou valores atípicos, pois há notícias falsas com mais de 1000 palavras.

Também foi descoberto que, em média, o número de palavras em notícias falsas é de 303 caracteres, enquanto em uma notícia não falsa o número aumenta para 527 caracteres. Para dar uma melhor visibilidade das palavras mais constantes em notícias falsas políticas e notícias não falsa, um *wordcloud* foi gerado e, além facilitar a visualização, pode ajudar a entender se é necessário realizar o pré-processamento novamente, para remover palavras que não vão ajudar no aprendizado do modelo.

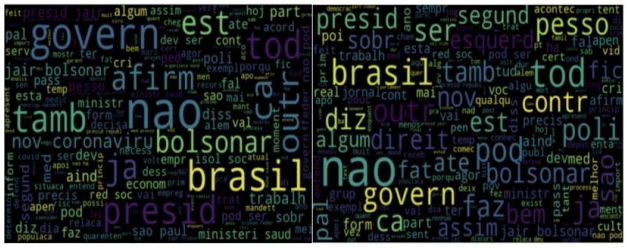

Figura 4. *Wordcloud*: (lado esquerdo) de notícias não falsas e (lado direito) de notícias falsas.

Como mostrado na Figura. 4., as palavras podem parecer estranhas, mas isso é devido ao processo de stemização, é comum as notícias falsas políticas brasileiras terem palavras como "não", "brasil", "governo", "bolsonaro", "presidente" e muitas outras. Quanto maior o tamanho das palavras no gráfico, mais recorrente é nas notícias. Nas notícias não falsas, Figura. 3 (lado direito), as palavras que mais se repetem são "não", "brasil", "presidente", entre outros.

4.1. Vetorização

Após o pré-processamento de ambos os dados (teste e treinamento), foram aprovados dois tipos de vetorização, o *Bag of Words* e o TF-IDF, de modo que foram gerados 4 tipos de vetorização. Esses vetores foram implementados ajustando o parâmetro "*n-gram*": *Bag of*

Words com *unigram*, *Bag of Words* com *bigram* e *unigram*, TF-IDF com *unigram* e TF-IDF com *bigram* e *unigram*.

4.2. *Cross validation* e teste dos modelos

A validação cruzada foi utilizada como etapa de pré-treinamento, com 5 *folds*, pois de acordo com Rodriguez (2009) é recomendado utilizar *folds* de 5 ou 10 visto que são menos tendenciosos, em uma base de 4180 exemplos, sendo 2090 notícias falsas e 2090 não falsas. Com ele é possível obter conhecimento prévio do comportamento do modelo com os dados. Durante o desenvolvimento dos algoritmos, nenhum parâmetro foi alterado, apenas no algoritmo SVM, onde o parâmetro do *kernel* foi alterado para realizar os testes com *kernel* linear visto que o default do SVC é o *kernel* RBF.

Tabela 3: Principais resultados

Descrição do modelo	Acur.	Prec.	Rec	F1	AUC	FN	FP
SVM (Linear)+ BOW	75,03	0,74	0,75	0,75	0,82	113	77
SVM (Linear)+ BOW + *Bigram*	**78,5**	**0,77**	**0,78**	**0,77**	**0,86**	**87**	**76**
SVM(Linear)+ TF-IDF	68,85	0,75	0,69	0,69	0,80	191	46
SVM(Linear)+ TF-IDF+ *Bigram*	64,7	0,72	0,70	0,65	0,88	250	18
SVM(RBF)+ BOW	**80,4**	**0,82**	**0,76**	**0,78**	**0,88**	**32**	**117**
SVM(RBF)+ BOW+ *Bigram*	67,2	0,76	0,58	0,53	0,87	7	242
SVM(RBF)+ TF-IDF	71,6	0,74	0,72	0,68	0,50	24	192
SVM(RBF)+ TF-IDF+ *Bigram*	71,2	0,75	0,71	0,67	0,50	17	202
*Multinomial*NB+ BOW	49,5	0,66	0,58	0,47	0,73	372	12
*Multinomial*NB+ BOW+*Bigram*	38,2	0,69	0,50	0,28	0,69	470	0
*Multinomial*NB+ TF-IDF	39,8	0,69	0,51	0,31	0,73	458	0
*Multinomial*NB+TF-IDF+*Bigram*	38,1	0,19	0,50	0,28	0,81	471	0
GaussianNB+ BOW	45,8	0,68	0,56	0,41	0,56	400	4
GaussianNB+ BOW+*Bigram*	42,1	0,63	0,53	0,35	0,53	6	434
GaussianNB+ TF-IDF	45,7	0,67	0,56	0,41	0,56	408	5
GaussianNB+ TF-IDF+ *Bigram*	42,1	0,63	0,53	0,35	0,53	6	434

Após o treino com a validação cruzada, o modelo foi validado utilizando um banco de dados com 761 exemplos, composto por 471 notícias falsas e 290 verdadeiras, os resultados podem ser vistos na Tabela 3 e Tabela 4.

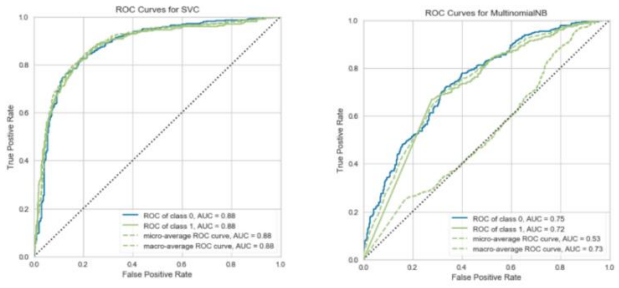

Figura 6. Curva ROC: a) SVM(RBF)+ BOW e b) *Multinomial*NB + BOW

Considerando os modelos SVM e *Naive Bayes* que tiveram os melhores desempenhos, é possível observar que o algoritmo *Naive Bayes* teve mais dificuldade em diferenciar as classes (se é falsa ou não), e pode ser visto nas colunas Falsos Positivos e Falsos Negativos, que os erros Tipo I e II são maiores, conforme mostrado na Tabela 3 e na Tabela 4, portanto pode-se concluir que apresenta um problema de *underfitting*, ou seja o algoritmo apresentou uma hipótese muito simples para a resolução do problema de

classificação de notícia falsa, resultando em valores ruins nas métricas tanto no treino quanto no teste.

Além disso, pode-se notar que a curva ROC e o AUC deste modelo é pior em comparação com os do SVM, conforme mostrado na Figura. 6. Uma forma que pode ajudar o modelo *Naive Bayes* a diferenciar as classes é com a implementação de mais dados, otimizando os parâmetros do modelo ou melhorando o pré-processamento nas notícias, por exemplo excluindo palavras que não trazem valor ao modelo, como as palavras "não" e "brasil".

5. Conclusão

Notícias falsas têm mostrado claramente sua influência na sociedade, espalhando cada vez mais desinformação sobre quaisquer temas em que estejam inseridas.

A chance de notícias falsas se propagarem rapidamente é muito maior do que as de um conteúdo verdadeiro [Vosoughi, Roy e Aral 2018], principalmente em países cujo consumo das notícias está sendo feitas a partir de mídias sociais e é exatamente por isso que estudos como esse são desenvolvidos, para ser possível mudar esse cenário e propor ferramentas que ajudem as grandes mídias sociais a combater esse tipo de notícia, evitar a desinformação da população brasileira e assim apoiar a democracia.

Conforme mostrado na Tabela 3, o SVM teve a maior precisão em relação aos outros modelos, já o algoritmo *Naive Bayes* apresentou mais dificuldade de identificar e diferenciar as classes, que podem ser vistas na curva ROC, Figura 4., e no número de Erros Tipo I (Falso Positivo) e Erros Tipo II (Falso Negativo), conforme mostrado na Tabela 4.

Assim, em um contexto de linguagem natural sem a implementação de outras características, o SVM parece ser o melhor algoritmo para classificação de notícias falsas em comparação com o algoritmo de *Naive Bayes*, confirmando que a hipótese H1 "O algoritmo classificador SVM foi melhor que *Naive Bayes* e apresentou bons resultados", levantada na seção 3.

Além disso o estudo apresentou resultados contrários do apresentado pelo Dias (2019), cuja conclusão apresentada foi que os algoritmos *Naive Bayes* apresentaram performance muito acima que o SVM, entretanto o autor não realizou nenhuma limpeza nos dados, realizando apenas a Lematização nas notícias selecionadas portanto pode-se concluir que a etapa de limpeza pré-processamentos, como: Stemização, remoção de stop words, remoção de pontuação e espaços em brancos são necessárias para uma avaliação mais assertiva no desempenho dos modelos.

Em trabalhos futuros, mais algoritmos de classificação podem ser considerados utilizados no estudo, por exemplo: redes neurais, Floresta Aleatória e regressão logística, para obter o melhor algoritmo para detectar notícias falsas no campo político.

No projeto atual, outros atributos além do texto de notícias em si não foram considerados, e de acordo com Zhou et al (2020) e outras obras utilizadas como base para o projeto, Monteiro et al (2018) e Shu et al (2017), existem alguns atributos que fazem a ponte entre a exatidão dos modelos de detecção de *fake news*, tais como: número de palavras, número de pontos, número de parágrafos etc.

Para que a análise seja mais precisa, podem ser utilizados parâmetros adicionais, como por exemplo a URL de uma página, ligada a um índice de confiança, podem ser exploradas variações nos parâmetros dos algoritmos, buscando verificar se haverá mudanças nos resultados obtidos. Há também a necessidade de se automatizar melhor o processo de coleta dos dados, de forma que um texto de uma notícia de qualquer página possa ser automaticamente extraído apenas passando o endereço para o sistema.

Referências

Abdin, L. (2019). Bots and fake news: the role of WhatsApp in the 2018 Brazilian Presidential election. Casey Robertson, 41(1).

Adriani, R. (2019). Fake News in the Corporate World: A Rising Threat. European Journal of Social Science Education and Research, 6(1), 92-110.

Bharadwaj, P., & Shao, Z. (2019). Fake news detection with semantic features and text mining. International Journal on Natural Language Computing (IJNLC) Vol, 8.

Bondielli, A., & Marcelloni, F. (2019). A survey on fake news and rumour detection techniques. Information Sciences, 497, 38-55.

Bovet, A., & Makse, H. A. (2019). Influence of fake news in Twitter during the 2016 US presidential election. Nature communications, 10(1), 1-14.

DAVIS, Jesse; GOADRICH, Mark. The relationship between Precision-Recall and ROC curves. In: Proceedings of the 23rd international conference on Machine learning. 2006. p. 233-240.

Dias, C. R. M. (2019). Towards fake news detection in Portuguese: New dataset and a claim-based approach for automated detection.

El Naqa, I., & Murphy, M. J. (2015). What is machine learning?. In machine learning in radiation oncology (pp. 3-11). Springer, Cham.

Ghosh, S., & Gunning, D. (2019). Natural Language Processing Fundamentals: Build intelligent applications that can interpret the human language to deliver impactful results. Packt Publishing Ltd.

Granik, M., & Mesyura, V. (2017, May). Fake news detection using naive Bayes classifier. In 2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON) (pp. 900-903). IEEE.

Halimu, C., Kasem, A., & Newaz, S. S. (2019, January). Empirical comparison of area under ROC curve (AUC) and Mathew correlation coefficient (MCC) for evaluating machine learning algorithms on imbalanced datasets for binary classification. In Proceedings of the 3rd international conference on machine learning and soft computing (pp. 1-6).

Harrison, Matt. (2019). Machine Learning Poket Reference. O'Relly Media, Inc. ISBN 9781492047544

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

Jivani, A. G. (2011). A comparative study of stemming algorithms. Int. J. Comp. Tech. Appl, 2(6), 1930-1938.

Klein, D., & Wueller, J. (2017). Fake news: A legal perspective. Journal of Internet Law (Apr. 2017).

Lane, Hobson. Howard, Cole. Hapke, Hannes. (2019). Natural Language Processing in Action: Understanding, analyzing, and generatint text with Python. Manning. ISBN 9781617294631

Lorena, A. C., Gama, J., & Faceli, K. (2000). Inteligência Artificial: Uma abordagem de aprendizado de máquina. Grupo Gen-LTC.

Monteiro R.A., Santos R.L.S., Pardo T.A.S., de Almeida T.A., Ruiz E.E.S., Vale O.A. Contributions to the Study of Fake News in Portuguese: New Corpus and Automatic Detection Results. In: Villavicencio A. et al. (eds) Computational Processing of the Portuguese Language. PROPOR 2018. Lecture Notes in Computer Science, vol 11122. Springer, Cham

Rodriguez, J. D., Perez, A., & Lozano, J. A. (2009). Sensitivity analysis of k-fold cross validation in prediction error estimation. IEEE transactions on pattern analysis and machine intelligence, 32(3), 569-575.

Rubin, V. L., Chen, Y., & Conroy, N. K. (2015). Deception detection for news: three types of fakes. Proceedings of the Association for Information Science and Technology, 52(1), 1-4.

Sharma, K., Seo, S., Meng, C., Rambhatla, S., Dua, A., & Liu, Y. (2020). Coronavirus on social media: Analyzing misinformation in Twitter conversations. arXiv preprint arXiv:2003.12309.

Shu, K., Sliva, A., Wang, S., Tang, J., & Liu, H. (2017). Fake news detection on social media: A data mining perspective. ACM SIGKDD explorations newsletter, 19(1), 22-36.

Tandoc Jr, E. C., Lim, Z. W., & Ling, R. Defining “Fake News”: a typology of scholarly definitions. Digital Journalism 6 (2), 137–153 (2018).

Vajjala, S., Majumder, B., Gupta, A., & Surana, H. (2020). Practical Natural Language Processing: A Comprehensive Guide to Building Real-World NLP Systems. O'Reilly Media.

Vascon, L. F. C., & de Souza, L. A. F. (2019). A violência policial em páginas de redes sociais virtuais: impactos das notícias falsas na opinião pública. Complexitas–Revista de Filosofia Temática, 3(1), 16-27.

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

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