DETECÇÃO DE *FAKE NEWS* NA LÍNGUA PORTUGUESA

FAKE NEWS DETECTION IN PORTUGUESE LANGUAGE

Godinho, Franklin Roberto, Almeida, João A. P., Leme, Vinicius Giovani, Johannes V.

franklin.roberto@outlook.com, palmeira.joao@gmail.com, viniciusgiovani@gmail.com, johannes.lochter@facens.br

Centro Universitário Facens - Sorocaba, SP, Brasil

Submetido em: 15 janeiro de 2022. Aceito em: 15/02/2022

RESUMO

Um fenômeno global, as notícias falsas ou *fake news*, impactam negativamente em toda sociedade, áreas como jornalismo, política, saúde e, também, a vida cotidiana das pessoas, sofrem severamente devidos os impactos causados por essas notícias falsas. Nos últimos anos o acesso a redes sociais chegou a um patamar nunca imaginado, com milhares de dados gerados todos os dias, o Brasil tem a terceira maior população de usuários de mídia social em todo mundo, são 150 milhões de usuários, o que representa 70,3% de sua população, Grandchamp (2021). O fácil acesso à internet e baixo custo, principalmente, por dispositivos móveis, promove uma alta disseminação de diversos conteúdos, o que se transformou numa gigantesca massa de dados e de difícil controle, pois usuários seguem compartilhando suas opiniões e influenciando outras pessoas com posts e notícias diversas. O objetivo deste artigo é propor a detecção das *fake news* através da inteligência artificial, mais especificamente na subárea NLP (Processamento de Linguagem Natural), baseando-se num conjunto de notícias falsas e verdadeiras, todas em português, para treinar os modelos de aprendizado de máquina. Os modelos selecionados para este trabalho foram Bag of Words, Word Embeddings+Word2Vec, LSTM+Word2Vec e BERT. Os melhores resultados foram obtidos pelo modelo Bag of Words, que teve baixo custo de processamento, ao contrário dos modelos baseados em redes neurais profundas (LSTM+Word2Vec e BERT), que tiveram bons resultados, porém alto custo de processamento. O modelo Word Embeddings+ Word2Vec também teve baixo custo de processamento, entretanto, com resultados abaixo do Bag of Words. O detalhamento dos resultados de cada modelo está exposto na seção 5 deste artigo.

ABSTRACT

A global phenomenon, the false news or fake news, impact negatively all society, areas such as journalism, politics, health, and also people's daily lives, suffer severely due to the impacts caused by these false news. In recent years the access to social networks has reached a level never imagined, with thousands of data generated every day, Brazil has the third largest population of social media users in the world, there are 150 million users, which represents 70.3% of its population, Grandchamp (2021). The easy access to the internet and low cost,

especially through mobile devices, promotes a high dissemination of various contents, which has become a gigantic mass of data and difficult to control, as users keep sharing their opinions and influencing others with posts and various news. The objective of this paper is to propose the detection of fake news through artificial intelligence, more specifically in the sub-area NLP (Natural Language Processing), based on a set of fake and true news, all in Portuguese, to train machine learning models. The models selected for this work were Bag of Words, Word Embeddings + Word2Vec, LSTM + Word2Vec and BERT. The best results were obtained by the Bag of Words model, which had low processing cost, unlike the models based on deep neural networks (LSTM + Word2Vec and BERT), which had good results but high processing cost. The Word Embeddings + Word2Vec model also had a low processing cost, however, with results below those of Bag of Words. The details of the results of each model are exposed in section 5 of this article.

Palavras-chave: *Fake news*. Processamento de linguem natural. Redes Sociais. Aprendizado de máquina. Inteligência Artificial.

1 INTRODUÇÃO

Diferentemente do que muitos imaginam, *fake news* não são um problema da atualidade, utilizadas para fins políticos, econômicos, sociais pessoais e coletivos, em rápidas pesquisas, é possível encontrar, na história, inúmeras situações onde as pessoas utilizavam esse formato de notícia para destruir reputações, espalhar boatos, rumores e até derrubar governos.

Bem antes das redes sociais, as mentiras se espalhavam de várias formas, Darnton (2017) relembra fatos importantes sobre notícias falsas, um livro, intitulado História Secreta, do historiador Procópio, repleto de verdades duvidosas, que manteve em segredo até a sua morte, para arruinar a reputação do imperador Justiano. O poeta Pietro Aretino, que tentou manipular a eleição do conclave papal de 1522, escrevendo sonetos perversos sobre todos os candidatos, menos Giulio de Médici, o patrono de Aretino. Surgimento dos *pasquins,* na Itália do século XVII, que produzia informações falsas e sensacionalistas. Os *pasquins* foram substituídos, em grande parte, por um gênero mais popular, os Canards, a gazeta cheia de boatos e notícias falsas, que circulou pelas ruas de Paris durante os 200 anos seguintes.

McGuillen (2017) identificou notícias fabricadas na Alemanha do século XIX, estas eram elaboradas por correspondentes locais, que fingiam ser repórteres enviados do exterior. Um exemplo foi nos anos 1860, Theodor Fontaine escreveu com riquezas de detalhes, de “Londres”, durante uma década, relatos pessoais sem nunca estar ali em todos esses anos.

Fatos mais recentes ou mais antigos, como os apresentados anteriormente, tem uma grande distinção, o poder massivo de disseminação pela internet, que propaga notícias a um número expressivo de pessoas muito rapidamente. (Garcia, 2020) publicou sobre o estudo do MIT (Instituto de Tecnologia de Massachusetts), que analisou 126 mil threads do Twitter, onde foi constatado que a verdade demora aproximadamente seis vezes mais do que a mentira para alcançar 1.500 pessoas e espalha-se mais longe e rapidamente.

Pontuou também sobre um relatório da consultoria Gartner que diz, em 2022 consumiremos mais boatos do que informação verdadeira.

Este artigo está organizado da seguinte forma, a seção 2 traz fatos recentes sobre *fake news* e trabalhos anteriores*,* na seção 3, o detalhamento da base de dados que foi utilizada, na seção 4, os resultados modelos de inteligência artificial utilizados para identificar as notícias falsas e, por fim, na seção 5 a discussão e conclusão de todo estudo.

2 REFERÊNCIAS ATUAIS SOBRE *FAKE NEWS*

Em 2016, as eleições presidenciais dos Estados Unidos foram marcadas pela denúncia contra o então candidato republicano, Donald Trump. O referido teria produzido e disseminado notícias falsas com o objetivo de prejudicar a candidata democrata Hillary Clinton (Mariana Giordão, 2020).

No Brasil, a última eleição presidencial também foi marcada pela propagação de notícias falsas, muitas em favor do candidato Jair Bolsonaro e contra o PT (Partido dos Trabalhadores) e Fernando Haddad, segundo colocado no pleito (Giordão, 2020).

A pandemia do novo coronavírus também foi tema para disseminação de notícias falsas, a Avaaz.org (rede para mobilização social global através da Internet), fez um estudo que aponta, cerca de 110 milhões brasileiros acreditam em pelo menos uma notícia falsa sobre a pandemia, esse número representa sete em cada dez pessoas (Mariana Giordão, 2020).

De acordo com o estudo “Desinformação on-line e eleições no Brasil”, publicado pela Fundação Getúlio Vargas (FGV) em 2020, as postagens em rede sociais (Facebook e YouTube) que desinformam são cada vez maiores, principalmente em épocas próximas as eleições. Em 7 anos foram identificadas mais de 337 mil publicações, gerando mais de 16 milhões de interações no Facebook, e mais de 23 milhões no YouTube. Segundo o estudo, a fake news mais disseminada foi a suposta venda de códigos de segurança das urnas eletrônicas para Venezuela feita pelo Tribunal Superior Eleitoral. Outra notícia falsa bastante espalhada foi a de que um hacker havia conseguido quebrar a segurança das urnas eletrônicas (FGV DAPP, 2020).

(Galho, 2021) utilizou a NLP na sua pesquisa e desenvolveu um protótipo chamado Detecção Automática de Fake News (DAFN). Utilizou a mesma base de dados empregada neste artigo *Fake.Br Corpus* (PROPOR, 2018). O protótipo desenvolvido passou pela preparação do corpus, seleção de características e, por fim, o processo de categorização de novas notícias, utilizando a técnica da similaridade por lógica difusa, que permite efetuar a categorização graduada de uma notícia para uma ou mais categorias (Galho, 2021).

Para detecção on-line de fake news, (Saad et al, 2017) utilizou análises do algoritmo N-Gram e técnicas de aprendizado de máquina. O melhor modelo obtido foi o Linear SVM, em conjunto com a técnica de extração de características (TF-IDF), chegando numa acurácia de 92%. O conjunto de dados utilizado no estudo, para notícias verdadeiras, foi extraído do site reuters.com e para notícias falsas do kaggle.com.

2.1 Identificação das *Fake News*

Existem características que ajudam a entender melhor o contexto de uma notícia falsa. A (UFRJ, 2018) publicou um artigo colocando pontos relevantes para elucidar como essa notícia pode ser identificada:

- Sátira ou Paródia: não é intencionalmente nocivo, mas pode levar à confusão do leitor;
- Conexão Falsa: título não corresponde fielmente ao conteúdo, gerando uma espécie de “clickbait” para aumentar o acesso;
- Contexto Falso: uma determinada informação quando fora de contexto pode se tornar inapropriada ou inválida com o passar do tempo;
- Conteúdo Manipulado: seja por adulteração de texto e/ou imagens, ou por tendenciar determinada opinião/visão política/ponto de vista;
- Conteúdo Enganoso: a informação é utilizada de forma a difamar a pessoa ou o assunto a que se refere;
- Conteúdo Impostor: informação é mal utilizada, moldando uma situação e criando uma inverdade com informações falsas de marcas ou pessoas;
- Conteúdo Fabricado: todo o seu conteúdo é falso, criado para enganar e prejudicar.

Para os seres humanos, mesmo conhecendo todas essas características, antes de repassarmos a notícia, é prudente observar se a origem é confiável, buscar em outras fontes sobre o mesmo assunto, a fim de confrontar as informações, observar a escrita e erros de ortografia, pois um escritor ou jornalista profissional, dificilmente cometerá erros rudimentares.

Devido a demasia de notícias surgindo a todo momento, a detecção de *fake news* utilizando a tecnologia, mais precisamente, modelos de aprendizado de máquina, com bibliotecas avançadas de NLP, aliada a alta capacidade de processamento dos computadores, são uma excelente alternativa para combater a disseminação.

3 BASE DE DADOS

Para este estudo foi utilizado um conjunto de dados intitulado *Fake.Br Corpus* (PROPOR, 2018). Este *corpus* possui 7.200 notícias, sendo 3.600 falsas e 3.600 verdadeiras, todas na língua portuguesa. As notícias falsas foram coletadas dos sites Diário do Brasil (3.338), A Folha do Brasil (190 notícias), The Jornal Brasil (65 notícias) e Top Five TV (7 notícias). As notícias verdadeiras foram coletadas utilizando um web crawler, programa que colhe conteúdo na web de forma sistematizada através do protocolo padrão da web (http/https), nos principais sites de notícias do Brasil, G1, Folha de São Paulo e Estadão. Os criadores explicam que o crawler procurou pelos substantivos e verbos presentes e palavras mais frequentes nas notícias falsas, ignorando as stop words. Encontraram 40 mil notícias verdadeiras e, por fim, fizeram uma verificação manual para garantir que a notícia falsa estava relacionada a verdadeira. A Tabela 1 abaixo demonstra os textos por categoria.

Tabela 1 - Distribuição das amostras por categoria.

Categoria	Número de Amostras	%
Política	4.180	58
TV e Celebridades	1.544	21,4
Sociedade e Notícias Diárias	1.276	17,7
Ciência e Tecnologia	112	1,5
Economia	44	0,7
Religião	44	0,7

Analisando a tabela 1 foi identificado que 58% de todas as amostras são relacionadas a política, isso demonstra que o meio político concentra grande parte das notícias, logo os impactos das fake news tendem a afetar a população eleitoral e a confiabilidade nos partidos políticos e futuros candidatos.

Utilizando a base de dados pré-processada (PROPOR, 2018), onde todas palavras foram convertidas para minúsculo, os acentos, pontuações, caracteres especiais e *stop words*, que são palavras irrelevantes como “as, e, os, para, com, sem, foi”) foram removidas. A função *WorldCloud,* da linguagem de programação *Python,* destaque as palavras mais utilizadas em todas notícias do *corpus* e são demonstradas nas Imagens 1 e 2.

Imagem 1 – Palavras que mais aparecem nas notícias falsas.

Observando a Imagem 1, fica mais evidente o forte relacionamento da política com *fake news*, pois a maioria das palavras destacadas remetem a esse tema.

Imagem 2 – Palavras que mais aparecem nas notícias verdadeiras.

Na Imagem 2, as principais palavras em destaque também remetem a política, muito disso se deve ao fato de que o *corpus* contém muitas notícias sobre política, mas isso tem uma explicação, com base nos estudos e história das *fake news*, fica evidente que grande parte das notícias são baseadas em política e o corpus utilizado neste estudo ratifica esse fato.

As Tabelas 2 e 3 demonstram dois trechos iniciais de notícias verdadeiras e falsas, aninhadas, que fazem parte do *corpus.*

Tabela 2 – Exemplo de notícia Falsa comparada a Verdadeira.

Falsa	**Verdadeira**
Cara de pau não tem limites: Zé Celso aciona MP contra Silvio Santos. O Teatro Oficina procurou o Ministério Público Federal (MPF) nesta terça-feira na tentativa de violar os direitos de Silvio Santos, que deseja erguer três torres residenciais de cem metros de altura cada uma.	O prefeito João Doria (PSDB) receberá o dramaturgo José Celso Martinez Corrêa e as arquitetas do Teatro Oficina no dia 29 para discutir o imbróglio das torres que Silvio Santos quer construir ao lado da sede da companhia.

Tabela 3 – Exemplo de notícia Falsa comparada a Verdadeira.

Falsa	Verdadeira
Hoje completou um ano da morte de um dos psicopatas mais repulsivos da era atual: Fidel Castro. O filme A Morte Do Demônio (de Sam Raimi, de 1982, refilmado em 2013) deveria ter recebido uma tradução mais decente no Brasil. The Evil Dead teria sido melhor traduzido como Mortos Demoníacos ou algo do tipo.	Cuba comemorará o primeiro aniversário da morte do emblemático líder Fidel Castro a partir de sábado (25) com uma semana de vigílias no momento em que a ilha põe em marcha um processo político que deve por fim aos 60 anos de governo dos irmãos Castro.

As duas notícias apresentadas, intituladas como Falsa e Verdadeira, demostram como é complexo para pessoas encontrarem as nuanças entre cada uma, caso não estivessem intituladas, a dificuldade em dizer qual é a falsa e qual a verdadeira seria consideravelmente difícil. Somente através de pesquisas na internet, buscando fontes confiáveis, seria mais fácil identificar. Por isso, na próxima seção (4), é proposta a ajuda da inteligência artificial para detectarmos *fake news*.

4 MÉTODOS

A detecção de fake news, mesmo por inteligência artificial, mas especificamente a subárea NLP (Processamento de Linguagem Natural), é um grande desafio, o computador precisará entender as características de comunicação através dos textos e interpretar a linguagem humana.

Existem várias técnicas empregadas em projetos de NLP, uma delas é a popular “Classificação de Texto”, amplamente usada em várias ferramentas, desde a identificação de *spam* de e-mails, tradução de texto, até análise de sentimentos (Prates, 2019). Este estudo se baseou em quatro técnicas/modelos de classificação de texto, nas subseções a seguir, é abordado um resumo de como cada técnica funciona.

4.1 *Bag of Words (BoW)*

Essa técnica não considera a ordem ou sequência das palavras, desta forma, através do *corpus*, um vocabulário de palavras é criado, é capturada as frequências de ocorrência de palavras, criando um vetor binário. As funções *WordTokenize* e *CountVectorizer* foram utilizadas para converter dados de texto em vetores. Basicamente pegam uma lista de palavras de uma frase e retornam um vetor, colocando o número zero se a palavra não estiver presente no token e colocando a contagem de token, se presente. A função *TfidfVectorizer* também foi utilizada nos experimentos e se diferencia das outras duas, porque não se concentra apenas na frequência das palavras presentes no *corpus*, mas também fornece a importância/peso das palavras.

No exemplo a seguir, duas sentenças foram criadas para um melhor entendimento de como é estruturado esse vetor, sendo que, as palavras foram

transformadas em minúsculas, as *stop words,* caracteres especiais e acentos foram removidos.

- Sentença 1: A detecção de fake news na política do Brasil.
- Sentença 2: As fake news na pandemia afetam o Brasil e o mundo.

Tabela 4 – Vetor de exemplo do Bag of Words (BoW)

Sentença	deteccao	fake	news	politica	brasil	pandemia	afetam	mundo
Sentença 1	1	1	1	1	1	0	0	0
Sentença 2	0	1	1	0	1	1	1	1

Observando a Tabela 4 foi identificado que, as palavras encontradas nas sentenças 1 e 2 são preenchidas com o número um, que representa quantas vezes a palavra apareceu na sentença e, zero no caso de a palavra não constar no texto.

A fim de melhorar os resultados do *BoW,* foram adicionados novos atributos para cada notícia do *corpus,* além da própria notícia e da classe (falsa ou verdadeira), foram incluídos o tamanho da mensagem, a quantidade de palavras e a maior palavra. Também foi utilizada a técnica *POS Tagging* para categorizar todas as palavras do vocabulário. Essa técnica cria novos atributos e define se a palavra é um adjetivo, advérbio, substantivo. Outro atributo muito importante adicionado foi a quantidade de palavras erradas na notícia, a função *SpellChecker,* trouxe esse resultado e confirmou que as notícias falsas possuem palavras muitas palavras erradas.

Para que os modelos de aprendizado de máquina pudessem ser treinados, o conjunto de dados foi dividido em 70% para treino e 30% para teste. Os algoritmos escolhidos foram *LinearSVC, Random Forest* e *Bernoulli Naive Bayes.* Nas predições realizadas, o modelo com melhor desempenho foi o LinearSVC com TFIDF, o resultado está na subseção 5.5.

4.2 *Word Embeddings*

Word Embeddings é uma técnica robusta que fornece uma representação vetorial densa de palavras, que capturam algo sobre seu significado. São também uma melhoria em relação aos esquemas de codificação de palavras do modelo de *Bag of Words* mais simples, como contagens de palavras e frequências que resultam em vetores grandes e esparsos (principalmente valores 0) que descrevem documentos, mas não o significado das palavras. *Word Embeddings* funcionam usando um algoritmo para treinar um conjunto de vetores de comprimento fixo, denso e de valor contínuo, com base em um grande corpo de texto. Cada palavra é representada por um ponto no espaço da embedding e esses pontos são aprendidos e movidos com base nas palavras que circundam a palavra-alvo (Brownlee, 2017).

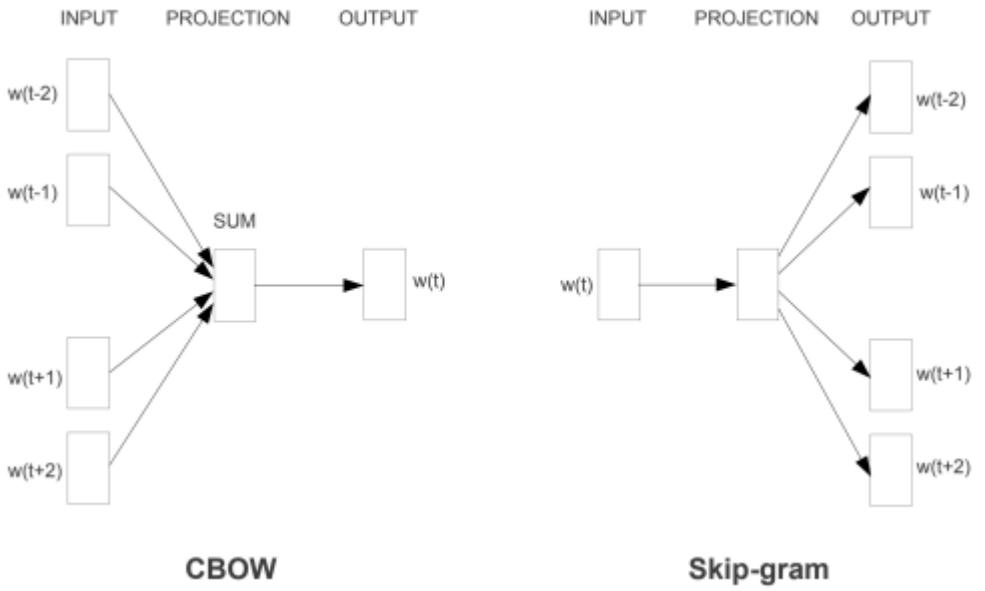
(Ahmed, 2021) descreve as principais características das embeddings:

- É uma técnica que representa as palavras individuais como vetores de valor real em um espaço vetorial predefinido.
- Esta técnica é amplamente utilizada no campo do *Deep Learning*, uma vez que as redes neurais no aprendizado profundo funcionam com valores vetoriais.
- É uma representação distribuída densa para cada palavra.

Para aprender as *Word Embeddings* a partir de um corpo de texto, existe um método estatístico criado por Tomas Mikolov, em 2013, chamado Word2vec (Brownlee, 2017). Este método foi desenvolvido principalmente para tornar a aprendizagem das *Word Embeddings* em redes neurais mais eficiente.

Word2vec tem duas versões diferentes, Continuous bag of words (CBOW) e Skip-gram (SG). Estas versões são os principais algoritmos para o Word2vec (Thanaki, 2017).

Imagem 3 – Modelos de treino Word2vec (Brownlee, 2017).



Ambos os modelos estão focados na aprendizagem das palavras dado o seu contexto de utilização local, onde o contexto é definido por uma janela de palavras vizinhas. Esta janela é um parâmetro configurável do modelo. O tamanho da janela deslizante tem um forte efeito sobre as semelhanças vectoriais resultantes. As janelas grandes tendem a produzir semelhanças mais atuais, enquanto que as janelas mais pequenas tendem a produzir semelhanças mais funcionais e sintáticas. A principal vantagem da abordagem é que podem ser aprendidas Word Embeddings de alta qualidade e de forma eficiente (baixa complexidade de espaço e tempo), permitindo que se aprendam embeddings maiores (mais dimensões) a partir de um corpus de texto muito maior (milhares de milhões de palavras) (Brownlee, 2017). Para este estudo foi utilizado o modelo Word2vec já treinado do Google (Google Code, 2013).

4.3 *LSTM*

Para utilizar o LSTM os documentos foram representados por embeddings, onde todas as redes neurais precisam ter entradas que contenham a mesma forma e tamanho. Entretanto, quando é processado e utilizado os textos como entradas para o modelo *LSTM*, nem todas as frases têm o mesmo comprimento, desta forma, existem sentenças que são mais longas ou mais curtas (Olah, 2015).

Foi definido o comprimento comum de 3599 para todas as notícias e executado o preenchimento usando a função *pad_sequences*, do Python, para deixar todas as sentenças com tamanhos iguais de comprimento, com a adição de zeros nas sentenças menores que o comprimento 3599. O modelo simples foi iniciado, onde a primeira camada é a embutida, que tem a entrada do tamanho do vocabulário, características vetoriais e comprimento da frase. Depois, adicionado 30% da camada de *dropout,* para evitar a sobreposição e a camada *LSTM*, que tem 100 neurônios e, por fim, utilizada a função de ativação *sigmoid*.

Usando vetores *Word2Vec* pré-treinados com LSTM, é necessário 12GB de RAM e 4GB de espaço HardDisk, para contornar isso, ao invés da criação de vetores de palavras, foram empregados vetores pré-treinados, que são conjuntos de dados em parte do Google *News* (cerca de 100 bilhões de palavras). O modelo contém 300 vetores dimensionais para 3 milhões de palavras e frases. (Google Code, 2013)

4.4 BERT

BERT (*Bidirectional Encoder Representations from Transformers)* é um método para representações de linguagem pré-treinada com um enorme *Corpus* para uso geral, esse método é usado para executar diversas tarefas de Processamento de Linguagem Natural (Devlin et al., 2018). Ao contrário dos modelos de Transformer, que usa dois mecanismos separados, um codificador que lê o texto de entrada e um decodificador que gera a predição para a tarefa, o BERT tem como objetivo gerar um modelo em que apenas o mecanismo de codificação é necessário. Para este trabalho, foi aplicado o BERTimbau, um modelo BERT pré-treinado para português brasileiro (Souza et al., 2019), o conjunto de dados foi dividido em três partes: treino, validação e teste e adicionamos três camadas ao fim da rede.

4.5 Métricas

Uma das métricas utilizadas na análise dos resultados é a Matriz de Confusão, que permite visualizar o desempenho do modelo de aprendizado de máquina, mais precisamente, a relação entre acertos e erros. Para melhor entendimento, a Imagem 7 destaca o que é cada valor dentro dessa matriz e os descritivos obtidos (Nogare, 2020) desses valores, esclarecem como essa técnica ajuda na análise do algoritmo. A classe 0 foi definida como *Fake News* e a classe 1 como Notícia Verdadeira.

Imagem 7 – Matriz de Confusão (Nogare, 2020).

		Valor Predito	
		Sim	**Não**
Real	**Sim**	Verdadeiro Positivo (TP)	Falso Negativo (FN)
	Não	Falso Positivo (FP)	Verdadeiro Negativo (TN)

Os descritivos a seguir ajudam no entendimento de cada valor dentro da matriz, assim é possível, de fato, confirmar como está o desempenho do modelo:

- **Positivo Verdadeiro (*True Positive* – TP):** significa que a classe prevista e observada originalmente faz parte da classe positiva;
- **Falso Positivo (*False Positive* – FP):** significa que a classe predita retornou positivo, mas a original observada era negativa;
- **Negativo Verdadeiro (*True Negative* – TN):** os valores preditos e observados fazem parte da categoria negativa;
- **Falso Negativo (*False Negative* – FN):** representa que o valor predito resultou na classe negativa, entretanto, o original observado era da classe positivo.

Outrossim, existem métricas para avaliar a matriz de confusão e são comumente utilizadas (Nogare, 2020). Através do relatório de classificação *(classification_report)* da biblioteca *metrics* do *Python*, todas as métricas para avaliação da matriz de confusão podem ser analisadas com clareza.

- **Acurácia:** Quantidade classificada como Positivos e Negativos corretamente, e pode ser formalizada em (TP + TN) / (TP + TN + FP + TN);
- **Precisão:** Quantidade Positiva classificada corretamente. E é calculada por TP / (TP + FP);
- **Recall:** Taxa de valores classificada como Positivo, comparada com quantos deveriam ser. E pode ser calculada como TP / (TP + FN);
- **F1 SCORE:** É calculado como a média harmônica entre Precisão e Recall, sendo sua formulação matemática representada por (2* TP) / (2* TP + FP + FN).

5 RESULTADOS

A seguir são demonstrados os resultados de cada método com o seu melhor modelo. Para problemas de classificação a curva ROC (*Receiver Operating Characteristic*) é excelente para analisar quão bem o modelo está nas predições. O eixo Y apresenta a sensibilidade, ou seja, a taxa de verdadeiros positivos e o eixo X a taxa de falsos positivos. O melhor cenário a ser alcançado nesse gráfico seria a taxa de falsos positivos igual a zero e a taxa de verdadeiros

positivos igual a um. A métrica AUC (*Area Under Curve*) também é apresentada nesse gráfico, o seu valor varia de 0,0 a 1,0 e quanto maior o seu resultado melhor o modelo está classificando as classes.

Imagem 4 - Curva ROC/AUC (Linear SVC) do *Bag of Words* + TFIDF.

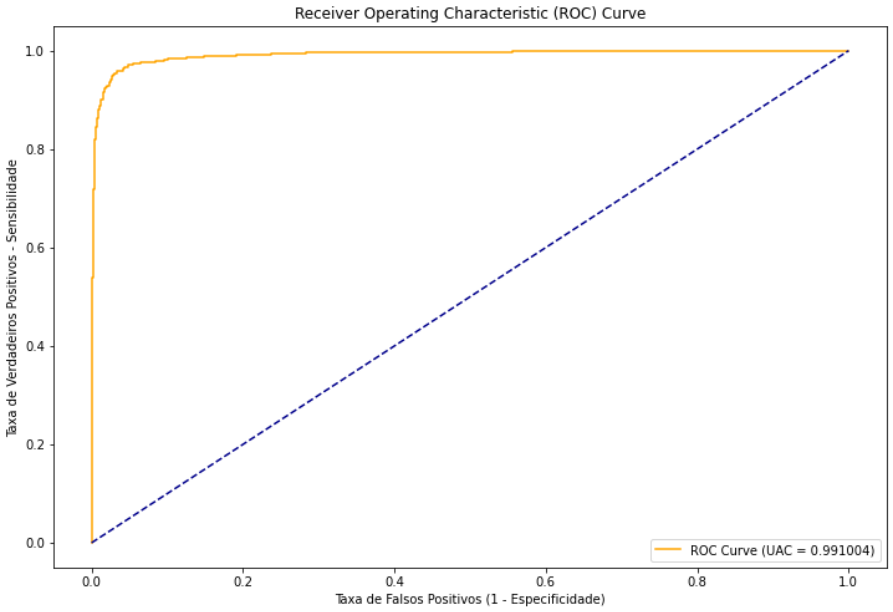


Imagem 5 - Curva ROC/AUC (Random Forest Classifier) do *Word Embeddings*.

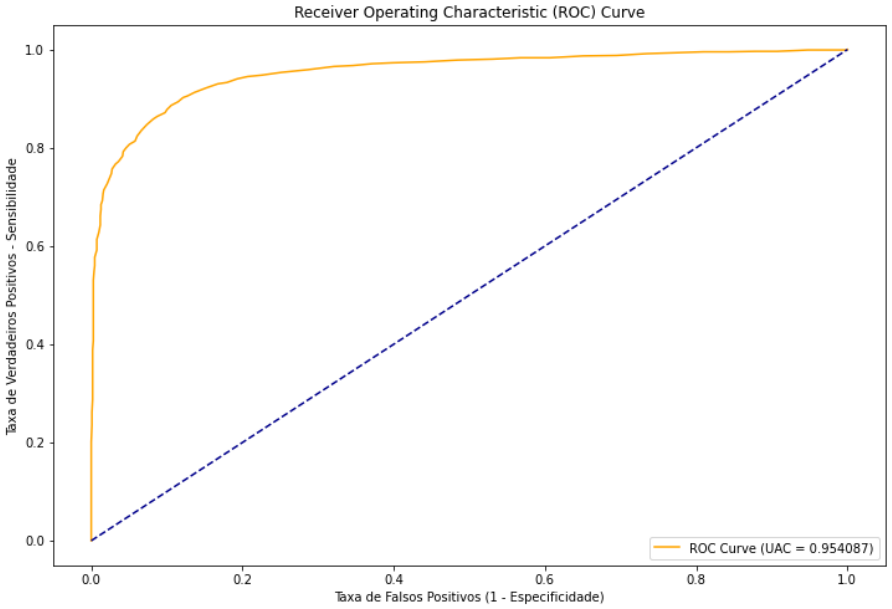


Imagem 6 – Curva ROC/UAC (LSTM).

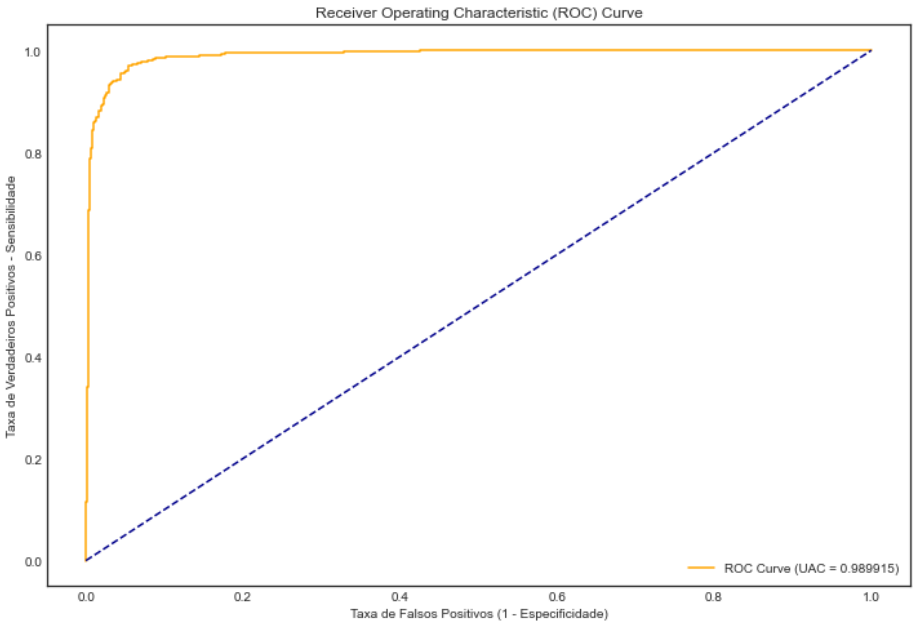
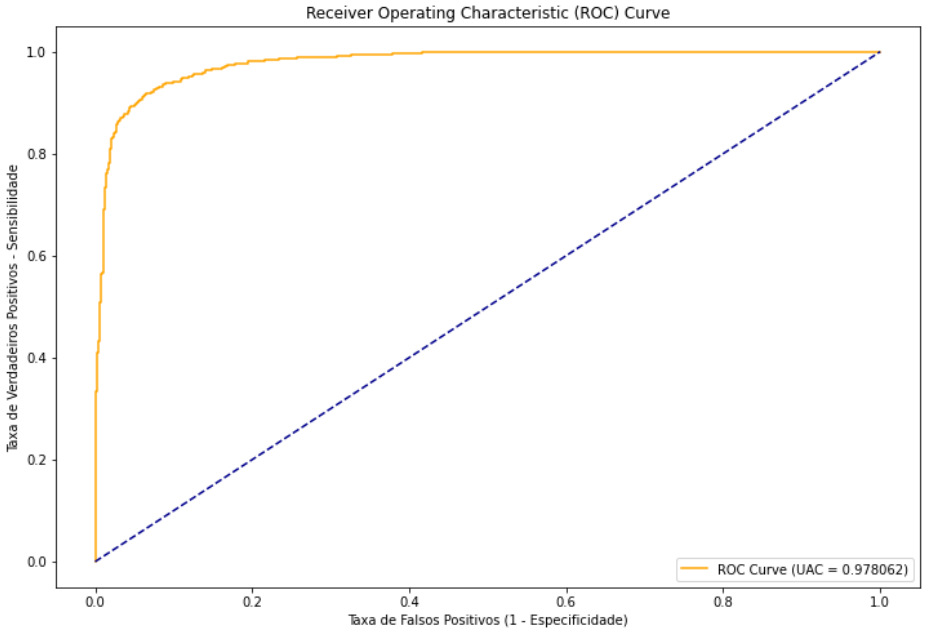


Imagem 7 – Curva ROC/UAC (BERT).



A Imagem 4 exibe excelentes resultados encontrados no Bag of Words pelo modelo Linear SVC, é possível observar que a taxa de verdadeiros positivos está muito alta, isso demostra que o modelo está acertando nas predições e a taxa de falsos positivos está bem baixa. A UAC está bem alta também, ratificando que o modelo está classificando com exatidão.

As Tabelas 5, 6, 7 e 8 demonstram os resultados da matriz de confusão.

Tabela 5 – Matriz de Confusão Linear SVC + TFIDF do modelo Bag of Words.

Predito	0	1	All
Real			
0	1047	40	1087
1	41	1032	1073
All	1088	1072	2160

TP = 1047 | TN = 1032 | FP = 41 | FN = 40

Tabela 6 – Matriz de Confusão Random Forest Classifier do modelo Word Embeddings.

Predito	0	1	All
Real			
0	1007	80	1087
1	164	909	1073
All	1171	989	2160

TP = 1007 | TN = 909 | FP = 164 | FN = 80

Tabela 7 – Matriz de Confusão LSTM.

Predito	0	1	All
Real			
0	835	65	900
1	21	879	900
All	856	944	1800

TP = 835 | TN = 879 | FP = 21 | FN = 65

Tabela 8 – Matriz de Confusão BERT.

Predito	0	1	All
Real			
0	976	111	1087
1	64	1009	1073
All	1040	1120	2160

TP = 976 | TN = 1009 | FP = 64 | FN = 111

Com foco no melhor modelo (Linear SVC + TFIDF do modelo Bag of Words), a Tabela 5 demonstra um total de 2.160 amostras, sendo que, 1.047 foram preditas corretamente como *fake news*, 1.032 preditas corretamente como notícia verdadeira, 41 foram preditas como *fake news*, mas eram notícias verdadeiras e 40 foram preditas como notícia verdadeira, porém eram *fake news*.

Através dessa análise podemos concluir que o algoritmo está com excelente desempenho nos acertos e baixos valores de erros.

A seguir a Tabela 9 demonstra os resultados finais de cada modelo.

Tabela 9 – Resultados dos melhores modelos escolhidos de cada técnica.

Modelo	Notícia	Precisão	Recall	f1-score	Acurácia
BoW+LienarSVC+TFIDF	Falsa	0.96	0.96	0.96	0.96
	Verdadeira	0.96	0.96	0.96	
LSTM+Word2Vec	Falsa	0.91	0.98	0.94	0.94
	Verdadeira	0.98	0.90	0.94	
WordEmbeddings+RF+Word2Vec	Falsa	0.86	0.93	0.89	0.89
	Verdadeira	0.92	0.85	0.88	
Bert + BERTimbau	Falsa	0.94	0.90	0.92	0.92
	Verdadeira	0.90	0.94	0.92	

Conforme a Tabela 9, os quatro modelos de classificação tiveram um ótimo desempenho nas métricas apresentadas, dentre esses modelos, o algoritmo classificação BoW+LienarSVC+TFIDF obteve melhor desempenho nos resultados da precisão, recall e no f1-score, tanto para notícia falsa, quanto para a notícia verdadeira. Também foi utilizado, como desafio, o algoritmo do modelo de classificação BERT, este apresentou excelentes resultados, entretanto, teve um alto custo processamento, desta forma, prejudicando a sua performance. Devido a esse ponto, o modelo não entrou no *ranking* de modelos recomendados para ser implementado em produção.

6 DISCUSSÃO E CONCLUSÃO

O presente artigo abordou a detecção da *fake news* na língua portuguesa e teve como principal objetivo identificar, dentre os vários modelos desenvolvidos, o com melhor performance na detecção de notícias falsas, baseando-se na técnica de Processamento de Linguagem Natural (NLP), que foi aplicada na base de dados referenciada na seção 3.

Com base nos métodos utilizados na análise, fica perceptível que os modelos de aprendizado de máquina tiveram um alto desempenho na detecção das fake news e que é um caminho promissor no combate a esse fenômeno global. Todos os resultados apresentados neste artigo, em especial, os que constam na Tabela 8, ratificam o êxito da inteligência artificial em minimizar os impactos das fake news na sociedade.

O melhor desempenho apresentado foi no modelo do *Bag of Words*, onde sua acurácia foi de 96% e, também, se comportou melhor nos resultados da precisão, *recall* e no *f1-score*, tanto para notícia falsa, quanto para a notícia verdadeira, conforme apresentado na Tabela 8. Além desse modelo ter sido significativamente melhor, também exigiu menos tempo de processamento e recurso de máquina, comparado com os modelos LSTM, *Word Embeddings* e Bert, que foram utilizados na análise. Com isso torna-se mais rentável e viável para futura implementação em ambiente produção.

Para um trabalho futuro, aconselha-se capturar uma base mais ampla de notícias na língua portuguesa, dar continuidade nos experimentos e análises com os modelos apresentados e implementar o melhor modelo em um sistema de detecção de *fake news* em tempo real.

REFERÊNCIAS

DARNTON, Robert. A verdadeira história das *fake news.* EL PAÍS. abr. 2017. Disponível em: https://brasil.elpais.com/brasil/2017/04/28/cultura/1493389536_863123.html. Acesso em: 30 nov. 2021.

MCGUILLEN, Petra. *How the techniques of 19th-century fake news tell us why we fall for it today*. *Nieman Lab*. abr. 2017. Disponível em: http://www.niemanlab.org/2017/04/how-the-techniques-of-19th-century-fake-news-tell-us-why-we-fall-for-it-today. Acesso em: 30 nov. 2021.

GARCIA, Jorge G. *Fake news* seguem padrões concretos. E os algoritmos já conseguem rastreá-los. EL PAÍS. jun. 2020. Disponível em: https://brasil.elpais.com/tecnologia/2020-06-11/fake-news-seguem-padroes-concretos-e-os-algoritmos-ja-conseguem-rastrea-los.html. Acesso em: 05 dez. 2021.

GIORDÃO, Mariana. O impacto das *fake news* na sociedade. G&A Comunicação. Jun. 2020. Disponível em: https://www.geacomunicacao.com.br/insights/o-impacto-das-fake-news-na-sociedade. Acesso em: 05 dez. 2021.

FGV DAPP. Desinformação On-line e Eleições no Brasil. FGV DAPP, Rio de Janeiro. out. 2020. Disponível em: https://democraciadigital.dapp.fgv.br/wp-content/uploads/2020/11/Relatorio-1-Texto.pdf. Acesso em: 10 dez. 2021.

UFRJ, Segurança da Informação. *Fake News:* Como Identificar e evitar a disseminação. set. 2018. Disponível em: https://www.security.ufrj.br/geral/fake-news-como-identificar-e-evitar-a-disseminacao. Acesso em: 10 dez. 2021.

PROPOR, 2018. Fake.Br Corpus. out. 2020 Disponível em: https://github.com/roneysco/Fake.br-Corpus. Disponível em: https://sites.icmc.usp.br/taspardo/PROPOR2018 -MonteiroEtAl.pdf. Acesso em: 10 nov. 2021.

BROWNLEE, Jason (2017). The *Word Embedding Model*. *Deep Learning for Natural Language Processing. (pp. 115-119).*

GOOGLE, Code. Word2Vec. jul. 2013. Disponível em: https://code.google.com/archive/p/word2vec. Acessado em: 10 dez. 2021.

AHMED, Shahid. *Word Embedding Using Python Gensim Package. Inside AIML.* dez. 2021. Disponível em: https://insideaiml.com/blog/Word-Embedding-Using-Python-Gensim-Package-1024. Acessado em: 15 dez. 2021.

GALHO, Thais Silva. Entenda a Detecção Automática de Fake News por Similaridade Difusa. set. 2021. Disponível em: https://www.eldorado.org.br/blog/entenda-a-deteccao-automatica-de-fake-news-por-similaridade-difusa-com-machine-learning-natural-language-processing-nlp. Acessado em: 10 nov. 2021.

SAAD, Sherif et al. *Detection of Online Fake News Using N-Gram Analysis and Machine Learning Techniques.* out. 2017. Disponível em: https://www.researchgate.net/publication/320300831_Detection_of_Online_Fake_News_Using_N-Gram_Analysis_and_Machine_Learning_Techniques. Acessado em: 15 dez. 2021.

PRATES, Wlademir Ribeiro. Introdução ao Processamento de Linguagem Natural (NLP). ago. 2019. Disponível em: https://cienciaenegocios.com/processamento-de-linguagem-natural-nlp. Acessado em: 15 dez. 2021.

OLAH, Christopher. *Understanding LSTM Networks. a*go. 2017. Disponível em: https://colah.github.io/posts/2015-08-Understanding-LSTMs. Acessado em: 10 nov. 2021.

DEVLIN, Jacob et al. BERT: *Pre-training of Deep Bidirectional Transformers for Language Understanding. mai. 2019. Disponível em: https://arxiv.org/pdf/1810.04805.pdf. Acessado em: 10 nov. 2021.*

SOUZA, Fábio et al. *Portuguese Named Entity Recognition using BERT-CRF. fev. 2020.* Disponível em: https://arxiv.org/pdf/1909.10649.pdf. Acessado em: 10 nov. 2021.

NOGARE, Diego. Performance de *Machine Learning* – Matriz de Confusão. abr. 2020. Disponível em: https://diegonogare.net/2020/04/performance-de-machine-learning-matriz-de-confusao. Acessado em: 20 out. 2021.