CENTRO UNIVERSITÁRIO CHRISTUS

SISTEMAS DE INFORMAÇÃO

MIRELLA GADELHA SANTOS

DETECÇÃO DE FAKE NEWS: UM COMPARATIVO ENTRE OS MODELOS DE APRENDIZADO SUPERVISIONADO PASSIVE AGRESSIVE E MULTINOMIAL NAIVE BAYES

FORTALEZA

2020

MIRELLA GADELHA SANTOS

DETECÇÃO DE FAKE NEWS: UM COMPARATIVO ENTRE OS MODELOS DE APRENDIZADO SUPERVISIONADO PASSIVE AGRESSIVE E MULTINOMIAL NAIVE BAYES

Trabalho de Conclusão de Curso (TCC) apresentado ao curso de Sistemas de Informação do Centro Universitário Christus, como requisito parcial para obtenção do grau de bacharel em Sistemas de Informação.

Orientador: Prof. Dr. Daniel Nascimento Teixeira

FORTALEZA

2020

Dados Internacionais de Catalogação na Publicação
Centro Universitário Christus - Unichristus
Gerada automaticamente pelo Sistema de Elaboração de Ficha Catalográfica do
Centro Universitário Christus - Unichristus, com dados fornecidos pelo(a) autor(a)

S237d Santos, Mirella Gadelha.
Detecção de Fake News: Um comparativo entre modelos de aprendizado supervisionado utilizando as técnicas Cross-Validation e Hold-Out / Mirella Gadelha Santos. - 2020.
72 f. : il. color.

Trabalho de Conclusão de Curso (Graduação) - Centro Universitário Christus - Unichristus, Curso de Sistemas de Informação, Fortaleza, 2020.
Orientação: Prof. Dr. Daniel Nascimento Teixeira.

1. Fake News. 2. Aprendizado Supervisionado. 3. Processamento de Linguagem Natural. I. Título.

CDD 005

MIRELLA GADELHA SANTOS

DETECÇÃO DE FAKE NEWS: UM COMPARATIVO ENTRE OS MODELOS DE APRENDIZADO SUPERVISIONADO PASSIVE AGRESSIVE E MULTINOMIAL NAIVE BAYES

Trabalho de Conclusão de Curso (TCC) apresentado ao curso de Sistemas de Informação do Centro Universitário Christus, como requisito parcial para obtenção do grau de bacharel em Sistemas de Informação.

Aprovada em:

BANCA EXAMINADORA

Prof. Dr. Daniel Nascimento Teixeira (Orientador)
Centro Universitário Christus (Unichristus)

Prof. Ms. David Kenned Ferreira Andrade Viana
Centro Universitário Christus (Unichristus)

Prof. Ms. Felipe Timbó Brito
Centro Universitário Christus (Unichristus)

AGRADECIMENTOS

Agradeço primeiramente a Deus por ter me dado forças e a oportunidade de estar realizando este projeto.

Aos meus pais, Maria Jucileide e Francisco, por uma vida inteira de incentivo e apoio inimagináveis. Obrigada por sempre estarem ao meu lado nos momentos difíceis, por acreditarem no meu potencial e por todos os sacrifícios feitos para que eu pudesse ter uma educação de qualidade.

A minha irmã Mylena por todo o amor, companheirismo e por me ajudar da melhor forma possível a melhorar este trabalho.

Ao meu orientador, professor Daniel Nascimento Teixeira, pela competência e por todo o apoio neste trabalho. Obrigada por pacientemente me ensinar sobre diversos assuntos que não seriam tão fáceis de compreender sem a sua ajuda e, além disso, obrigada por ajudar a tornar este um projeto o qual eu tenho orgulho.

A todos os meus professores que sempre me incentivaram e são minha motivação para seguir na carreira acadêmica. Agradeço profundamente por me ensinarem, educarem e, acima de tudo, por me fazerem inquieta.

"Que nada nos defina, que nada nos sujeite. Que a liberdade seja a nossa própria substância."

(Simone de Beauvoir)

RESUMO

Com a popularização da internet e consequentemente com a facilidade de acesso às informações disponíveis no âmbito online, sobretudo em virtude do uso das redes sociais digitais, o consumo e a disseminação de informações falsas vêm trazendo à tona uma constante preocupação em todo o mundo em relação ao seu potencial de circulação e as consequências negativas que as chamadas *fake news* podem acarretar. Preocupado com a gravidade que o tema apresenta à nossa sociedade e em um esforço para auxiliar no combate à crescente desinformação, este trabalho propôs a detecção automática de *fake news* utilizando inteligência artificial e algoritmos de aprendizado supervisionado para classificação de notícias falsas no idioma português brasileiro. Assim, foram considerados dois algoritmos de classificação: o *Multinomial Naive Bayes*, baseado na regra de *Bayes*, e o *Passive Agressive*. Para analisar o desempenho destes classificadores também foram utilizadas duas técnicas de validação de dados: a *Hold-Out* e *Cross-Validation*. Ambos os algoritmos alcançaram resultados satisfatórios ao classificar as notícias, entretanto, os resultados para o algoritmo de classificação *Passive Agressive* se mostraram superiores obtendo uma acurácia de $91,81\%$ ao utilizar a técnica *Hold-Out* e $92,29\%$ ao empregar a técnica *Cross-Validation*.

Palavras-chave: Fake News. Aprendizado de Máquina. Processamento de Linguagem Natural.

ABSTRACT

The popularization of the Internet and the facility of access to all the available information online, mostly because of the use of social media, has brought to surface the consumption and the spread of Fake News in this age and, with that, the constant worry about their capacity of circulation or even the negative consequences that could come with them. Concerned with the importance of this theme to our society and with and effort to to combat misinformation, this Paper proposes the automatic detection of Fake News using a system based on artificial intelligence and supervised machine learning algorithms to classify Fake News written on portuguese language. Thus, two algorithms were considered in the classification. The Multinomial Naive Bayes, based on the Bayes rule, and the Passive Agressive. Also, to analyze the performance of these classifiers, two techniques of data validation were used: the Hold-Out and the Cross-Validation. Both algorithms achieved satisfactory results on the task to classify the news. However, the results of the Passive Agressive were superiors and obtained a 91,81% accuracy when using the Hold-Out technique and a 92,29% accuracy with the Cross-Validation technique.

Keywords: Fake News. Machine Learning. Natural Language Processing.

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

1.1 Contextualização e delimitação do tema

Segundo a definição do dicionário Aurélio (FERREIRA, 2014) entende-se por notícia o relato de acontecimento atual, de interesse público geral, ou de determinado segmento da sociedade, veiculado em jornal, rádio, televisão, etc. Essas notícias geralmente são repassadas por fontes oficiais, que obedecem um conjunto de normas éticas com o intuito de garantir que as informações sejam divulgadas sem alterações ou detalhes falsos. Hoje, com a popularização da internet, o número de informações produzidas cresce em grandes proporções. De acordo com um estudo feito por Barbosa e O'Reilly (2011), a cada dia do ano de 2012 o mundo produziu mais informação escrita do que toda a que existia antes de 2003.

Pode-se dizer que esse avanço digital acabou gerando um estado de hiperconectividade, o que consequentemente levou a popularização do acesso à informação, grande parte dela disposta de forma gratuita. Levando em conta este cenário, é possível que informações falsas ou boatos sejam gerados e difundidos na web. Assim, elas podem levar as pessoas a acreditarem em algo equivocado e, como resultado, encorajá-las a propagarem essas notícias, o que cria uma cadeia de desinformação capaz de levar desde a ações inofensivas, como por exemplo, a monetização de cliques ou até a situações mais graves, como a difusão de ameaças e discursos de ódio. Tudo isso resulta em uma influência a diversos campos da sociedade, sejam nos âmbitos sociais, econômicos, políticos, etc.

As *fake news* são notícias com conteúdo falso e sem embasamento, criadas com o intuito de enganar e manipular o leitor (TANDOC *et al.*, 2017). Antigamente, o trabalho de checar a veracidade das informações era exclusivamente da imprensa, ou seja, a mesma era responsável tanto pela comunicação de fatos novos como também pela checagem de notícias. Entretanto, é importante dizer que esse cenário tem mudado, fazendo com que o leitor passe a desempenhar esse papel investigativo por si próprio, já que o consumo de notícias dadas por redes sociais tem sido favorecido se comparado a outros veículos de comunicação.

Em 2016, um departamento da Universidade de Oxford, a *Oxford Dictionaries*, elegeu a "*pós-truth*", ou pós-verdade, como a palavra do ano. Além de eleger o termo, a instituição definiu o que é a "pós-verdade": um substantivo "que se relaciona ou denota circunstâncias nas quais fatos objetivos têm menos influência em moldar a opinião pública do que apelos à emoção e a crenças pessoais" (HOLZHACKER, 2017). Algumas situações contribuíram diretamente para

a criação do termo *pós-truth*, um exemplo disso foi a eleição de Donald Trump em 2016 como presidente dos Estados Unidos e o referendo que decidiu pela saída da Grã-Bretanha da União Europeia. Em ambas as campanhas houve um grande número de compartilhamento de *fake news* em plataformas digitais que foram de significativa importância para que as informações falsas tivessem alcance e legitimidade (HOLZHACKER, 2017).

O uso de redes sociais tem contribuído de forma significativa para a propagação de notícias falsas. Outro fator importante para entender a divulgação de *fake news* é o uso de softwares ou robôs que produzem conteúdos e interagem com usuários em contas de redes sociais. No cenário brasileiro, um estudo produzido pela Fundação Getúlio Vargas mostrou que, em 2014, 11% das discussões durante o período eleitoral foram geradas por perfis falsos ou controlados por robôs. Entre os perfis de apoio ao senador Aécio Neves 19% das interações foram geradas por robôs, seguido de 17% das interações de perfis apoiadores da ex-presidente Dilma Roussef (FABIO, 2016).

Ao analisar 14 milhões de mensagens no Twitter, os pesquisadores identificaram que "poucas contas são responsáveis por uma grande parcela do tráfego que traz conteúdo enganoso" (SHAO *et al.*, 2017).

> Essas contas são provavelmente *bots*, e descobrimos várias estratégias de manipulação que usam. Primeiro, os *bots* são particularmente ativos em amplificar notícias falsas no estágio inicial de propagação, antes que o conteúdo seja viral. Em segundo, os *bots* miram usuários influentes através de respostas e menções. (SHAO *et al.*, 2017).

Vale ressaltar que as *fake news* não precisam necessariamente ter um cunho político. Elas podem se manifestar em outros diversos setores da sociedade, como saúde e segurança, por exemplo. Segundo Shu *et al.* (2017), a *fake news* não se trata apenas de uma informação pela metade ou mal apurada, mas de qualquer informação falsa intencionalmente divulgada para atingir interesses de indivíduos ou grupos.

1.2 Problematização

As notícias em geral possuem um papel crucial no repasse de qualquer tipo de informação e consequentemente impactam diretamente a sociedade e a forma como entendemos determinados acontecimentos. Pode-se dizer que as *fake news* são criadas com o intuito de enganar seus leitores e consequentemente legitimar um ponto de vista com uma informação falsa, o que pode levar ao compartilhamento extremamente rápido de informações não verdadeiras para

uma quantidade elevada de pessoas, impactando negativamente a sociedade com um conteúdo viral e até mesmo nocivo. Em um esforço para combater a crescente desinformação, vários sites de checagem de fatos foram implantados para expor ou confirmar histórias. Esses sites desempenham um papel crucial no combate as notícias falsas, mas exigem análises especializadas que inibem uma resposta oportuna (MIHALCEA; STRAPPARAVA, 2009).

Diante disso, pode-se dizer que encontrar *fake news* antes que elas gerem consequências reais pode acabar sendo uma tarefa difícil, isto porque atualmente os sites de mídias sociais ainda dependem bastante de editores humanos, que muitas vezes não conseguem acompanhar o grande fluxo de notícias. Além disso, as técnicas de análise atuais, em alguns casos, dependem da verificação externa dos fatos. Nesse contexto, pode ser difícil averiguar os casos de histórias mais recentes visto que, muitas vezes, quando se consegue provar a veracidade de uma notícia, o dano já foi generalizado (SCIENTIFC AMERICAN BRASIL, 2018).

Desse modo, tem-se como pergunta de pesquisa "Como elaborar uma aplicação que seja capaz de auxiliar no trabalho investigativo jornalístico e também no combate a desinformação gerada pela disseminação de informações falsas na web?".

1.3 Pressupostos

As máquinas e os sistemas inteligentes estão executando tarefas que até recentemente eram prerrogativas dos humanos, em alguns casos com resultados mais rápidos e mais assertivos. Elas, porém, ainda estão restritas a prever cenários com base em grandes conjuntos de dados e a executar tarefas específicas (KAUFMAN; SANTAELLA, 2020). Diante desse cenário, a inteligência artificial tem se mostrado uma ferramenta computacional capaz de auxiliar na identificação e verificação de *fake news* para o jornalismo, sites agregadores de conteúdos e a sites de mídias sociais, além de ser capaz de fornecer segurança acerca da confiabilidade do conteúdo a ser verificado.

1.4 Objetivos

1.4.1 Objetivo geral

Este trabalho tem como objetivo principal construir um modelo de classificação utilizando aprendizagem de máquina para detectar *fake news* e implementar esse mesmo modelo de aprendizado em uma aplicação web. O principal problema considerado, e que o trabalho visa

abordar, é como a tecnologia pode ser capaz de encontrar padrões em notícias falsas e também verdadeiras, a fim de criar um modelo de predição que possa automaticamente detectar uma notícia falsa utilizando o texto da própria notícia como entrada.

1.4.2 Objetivos específicos

1. Utilizar aprendizado de máquina supervisionado com o intuito de detectar *fake news*.
2. Produzir comparativo entre os modelos de aprendizado supervisionado *Multinomial Naive Bayes* e *Passive Agressive* a fim de buscar a técnica mais eficiente que será utilizada para a classificação de notícias verdadeiras e falsas.
3. Avaliar a capacidade de generalização dos modelos de classificação, a partir de um conjunto de dados, utilizando as técnicas *Cross-Validation* e *Hold-Out*.
4. Desenvolver um *chatbot*, a partir do conhecimento adquirido neste trabalho, capaz de checar a veracidade de uma notícia.

1.5 Justificativa

A popularização de termos como *"fake news"*, "pós-verdade"e "desinformação"tem trazido à tona uma preocupação com a veracidade e a confiabilidade das informações disseminadas na web (RIPOLL; MATOS, 2017). Desta forma, o presente trabalho se justifica pela necessidade urgente do debate em torno de informações falsas que circulam na internet e pela importância de desenvolver formas de auxiliar no combate à desinformação.

A pesquisa propõe-se a discutir o uso da tecnologia como ferramenta de auxílio na verificação de *fake news* para o jornalismo, sites agregadores de conteúdos e a sites de mídias sociais. Diante desta motivação, o trabalho visa desenvolver modelos preditivos utilizando aprendizado supervisionado para classificar notícias verdadeiras e falsas. Por fim, após o desenvolvimento dos modelos de classificação, os mesmos serão avaliados e comparados com o intuito de medir a eficácia e determinar qual atende melhor ao propósito deste projeto.

1.6 Estrutura do trabalho

O trabalho está organizado em seis capítulos, de forma a apresentar o conteúdo mais claramente, conforme os parágrafos a seguir.

No Capítulo 2 são apresentados alguns conceitos necessários para o entendimento

dos capítulos subsequentes.

No Capítulo 3 são apresentados alguns dos trabalhos que serviram como base para o desenvolvimento deste projeto.

No Capítulo 4 é apresentada a técnica proposta pelo trabalho, explicando como foi o processo de desenvolvimento dos modelos de aprendizado para a detecção de *fake news*.

No Capítulo 5, são apresentados os resultados obtidos dos algoritmos estudados. Os mesmos serão analisados por meio da utilização de algumas métricas de desempenho, a fim de demonstrar a eficácia dos modelos de aprendizado.

Por fim, no Capítulo 6, são apresentadas as conclusões sobre o trabalho e são propostas algumas ideias para trabalhos futuros.

2 REFERENCIAL TEÓRICO

2.1 As *fake news* e sua disseminação por meio das mídias digitais

O termo *fake news* tem o significado definido pelo Dicionário de Cambridge (2018) para indicar falsas histórias que aparentam ser notícias e são capazes de se espalhar pela internet ou outras mídias, sendo geralmente criadas com o intuito de influenciar pontos de vista políticos ou como piadas. As *fake news* constituem uma espécie de "imprensa marrom"(*yellow press*) intencionalmente veiculando conteúdos falsos, sempre com a intensão de obter algum tipo de vantagem, seja financeira (mediante receitas oriundas de anúncios), política ou eleitoral (CARVALHO; KANFFER, 2018).

Shu *et al.* (2017) apresentam duas características-chave para classificar as fake news: (1) a falta de autenticidade e (2) seu propósito de enganar. Logo, pode-se dizer que esse tipo de notícia não se trata apenas de uma informação pela metade ou mal apurada, mas de informação falsa, criada e divulgada intencionalmente a fim de atingir interesses de indivíduos ou grupos. Recuero e Gruzd (2019) compreende três elementos essenciais para a definição de uma *fake news*: (1) o componente de uso da narrativa jornalística e dos componentes noticiosos; (2) o componente da falsidade total ou parcial da narrativa; e (3) a intencionalidade de enganar ou criar falsas percepções por meio da propagação dessas informações na mídia social. A circulação de notícias falsas, deste modo, atua diretamente na produção de desinformação, de modo particular, na internet, embora não seja o único ambiente usado para isso (SHAO *et al.*, 2017).

Pode-se dizer que a disseminação de notícias falsas não é um fenômeno recente na história da humanidade. Na idade moderna, as *fake news* já faziam sucesso na cobertura midiática. Em um artigo para o The New York Review of Books, Darnton (2017) relembra que Pietro Aretino,

> tentou manipular a eleição pontifícia de 1522, escrevendo sonetos perversos sobre todos os candidatos (exceto o seu favorito, Medici) e colando-os para o público admirar no busto de uma figura conhecida como Pasquino, em Roma. O "pasquinade" então se transformou em um gênero comum de difundir notícias desagradáveis, a maioria delas falsas, sobre figuras públicas.

No contexto social estabelecido a partir da evolução da tecnologia, assim como também da contínua utilização das mídias e das redes sociais e do nível de influência destas sobre o indivíduo, pode-se dizer que estes fatores criaram um ambiente propício para o crescimento do fenômeno *fake news*. Esse termo surge com o intuito de descrever o fenômeno da ampla

divulgação de notícias falsas ocorrido nas redes sociais. Apesar de não ser uma invenção contemporânea, foi por meio das redes sociais que ganharam espaço para proliferação, alcançando a dimensão apresentada na atualidade. Pode-se atribuir sua rápida ascensão a quatro fatores: a descentralização da informação, a polarização política, a crise de confiança nas instituições e o crescimento do pensamento individualista (POUBEL, 2018).

> Atualmente as instituições políticas mundiais passam por uma crise de confiança decorrente de vários fatores, tendo por reflexo o avanço de ideias radicais em todo o mundo. Como consequência, essa crise acarreta inevitavelmente a polarização política, tendo os indivíduos se posicionado entre dois pensamentos extremos. Com o surgimento de um novo meio de comunicação, as redes sociais, o homem conquistou um espaço de exposição de suas ideias e interesses particulares, no qual há a prevalência de seus gostos, promovendo o fortalecimento de seu comportamento individualista. As redes sociais, portanto, se apresentaram como meio de manifestação do indivíduo de sua posição política, com total liberdade de ataque ao pensamento oposto, e é neste contexto de conflito que prosperam as fake news (TEIXEIRA *et al.*, 2018).

2.2 Tipologia das *fake news*

Com a popularização das mídias sociais nos últimos anos e, consequentemente, com o aumento da criação de notícias falsas, foi possível notar que as *fake news* possuem propriedades e atributos diferentes. Wardle (2017) definiu essa variedade de notícias em sete categorias diferentes como forma de classificar e facilitar a detecção das mesmas, são elas:

1. Sátira ou paródia
2. Falsa conexão
3. Conteúdo enganoso
4. Falso contexto
5. Conteúdo impostor
6. Conteúdo manipulado
7. Conteúdo fabricado

2.2.1 Sátira ou paródia

Segundo Burfoot e Baldwin (2009), notícias do tipo sátira ou paródia tendem a imitar artigos de notícias verdadeiras, incorporando ironia como uma tentativa de fornecer informações e, ao mesmo tempo, criticar ou ridicularizar utilizando o humor como ferramenta. Para Thu e Aung (2018), a sátira é uma forma de comunicação implícita e pode ser definida como uma arte literária que tem o propósito de ridicularizar e desprezar. Esse tipo de linguagem pode

ser encontrado extensivamente em vários canais: literatura, televisão, internet, mídias sociais, quadrinhos e desenhos animados. Devido ao seu uso generalizado, a manipulação da sátira é atualmente uma das tarefas mais desafiadoras na linguística computacional, processamento de linguagem natural e análise de sentimentos de multimídia social (THU; AUNG, 2018).

De acordo com Serra (2018), a sátira nem sempre é humorística, já a paródia tende a criar um efeito cômico, ridicularizando o tema escolhido (ver Figura 1). Ainda segundo a autora, existem três técnicas mais utilizadas na sátira. São elas: a diminuição, inflação e a justaposição.

> A diminuição reduz a grandeza do que está sendo satirizado com o intuito de torná-lo ridículo ou de sobressair os defeitos criticados, a inflação é o oposto, o objeto satirizado tem seus aspectos exagerados ou aumentados, já a justaposição tenta igualar o nível de importância de coisas claramente desiguais (SERRA, 2018).

Figura 1 – Exemplo de Sátira ou Paródia.

Fonte: Sensacionalista (2014).

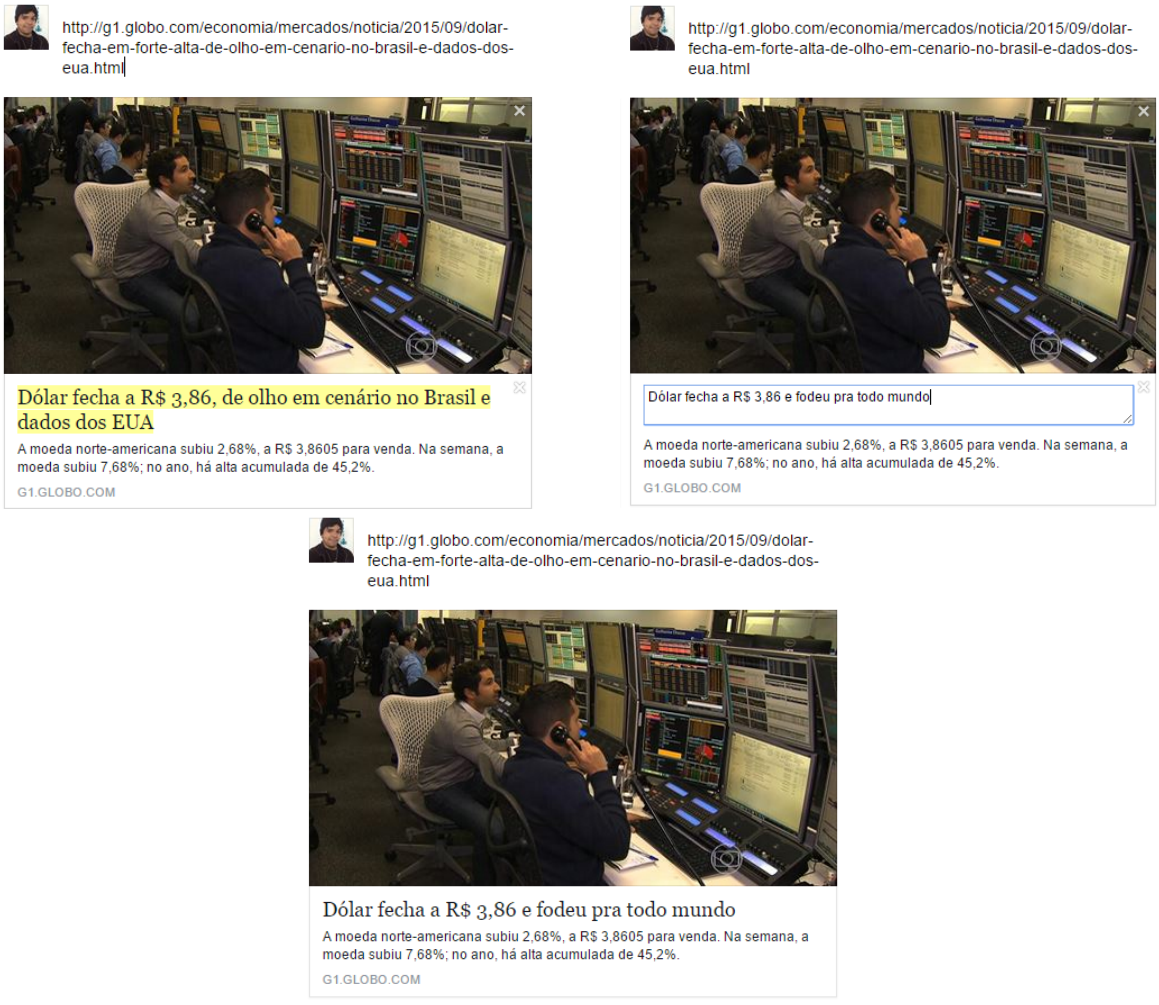
2.2.2 Falsa conexão

Segundo Wardle (2017), falsa conexão é um tipo de *fake news* que apresenta divergências entre a chamada da notícia e o conteúdo mostrado. Esse tipo é usado para atrair a

atenção do usuário com manchetes, imagens ou legendas chamativas, que quando clicadas levam a matérias não condizentes com o conteúdo apresentado.

A rede social Facebook permitia, até 2017, que os usuários da rede modificassem os títulos de notícias antes de compartilhá-las, o que poderia dar margem para a propagação de notícias falsas, já que era possível para o usuário adicionar uma manchete mais sensacionalista ou até mesmo mentirosa. Um exemplo disso foi o caso de uma postagem que teve o título "Dólar fecha a RS 3,86, de olho em cenário no Brasil e dados dos EUA"alterado, apenas como uma piada, para "Dólar fecha em RS 3,86 e fodeu pra todo mundo", como pode ser visto na Figura 2. Mesmo com o título da notícia adulterado, a mesma foi repostada mais de 5 mil vezes no Facebook. Uma pesquisa publicada no HAL, arquivo de acesso aberto multidisciplinar para o depósito e disseminação de documentos de pesquisa, confirmou que 59% de todos os links compartilhados nas redes sociais não são clicados e lidos (GABIELKOV *et al.*, 2016).

Figura 2 – Exemplo de Falsa Conexão.



Fonte: Camarossi (2015).

2.2.3 Conteúdo enganoso

De acordo com Wardle (2017), conteúdo enganoso pode ser entendido como um tipo de notícia que usa informações falsas com o propósito de difamar o assunto ou a pessoa em questão. É frequente o uso desse tipo de *fake news* principalmente no meio político. Um caso que ganhou grande repercussão foram as notícias falsas atribuídas a vereadora Marielle Franco, logo após sua morte a tiros no Rio de Janeiro em março de 2018. Entre algumas das falsas notícias divulgadas estava a do deputado federal Alberto Fraga (ver Figura 3), onde o mesmo afirmava que Marielle teria vínculo com uma facção criminosa e seria casada com um narcotraficante. A base das *fake news* atribuídas a Marielle tinha cunho político, buscando difamar e marginalizar suas lutas e conquistas, entretanto, todas as notícias foram desmentidas por voluntários que criaram um site para contar a verdade dos fatos.

Figura 3 – Exemplo de Conteúdo Enganoso.

Fonte: O Globo (2018).

2.2.4 Falso contexto

Esse tipo de *fake news* refere-se a conteúdos verdadeiros, mas que são utilizados em um contexto errado com o propósito de atrair mais atenção para a notícia. Um exemplo desse caso aconteceu em 2019 com novas queimadas na Amazônia. Um post feito pelo presidente francês Emmanuel Macron na rede social Twitter dizia que a Amazônia estava "queimando", onde juntamente desta informação foi anexada uma imagem com uma área sob fogo (ver Figura 4). Entretanto, segundo um programa de verificação de fatos feito pelo G1 (2019), a imagem é

bem mais antiga do que o contexto das queimadas em questão. A fotografia foi feita em 2013 pelo fotográfo Loren McIntyre, quando o mesmo esteve na Amazônia em expedições desde a década de 70, trabalhando pela National Geographic. O que caracteriza esse tipo de *fake news* é o fato das imagens existirem, não serem manipulações digitais, porém serem utilizadas fora de contexto, o que faz com que sirvam ao propósito da notícia falsa (SERRA, 2018).

Figura 4 – Exemplo de Falso Contexto.

Fonte: G1 (2019).

2.2.5 Conteúdo de impostor

Segundo Serra (2018), os conteúdos impostores são frequentemente sites que imitam portais de jornalismo profissional para dar maior credibilidade à notícia fabricada. O site do G1 foi vítima dessa prática ao ter seu site copiado tanto em aparência quanto em conteúdo (ver Figura 5), o que levava os leitores assíduos e que conheciam o site verdadeiro a julgar a página semelhante com o mesmo nível de confiabilidade. Nesse caso específico, o site falso utilizava

uma notícia verdadeira que foi publicada no portal do G1 em 2015, mas que foi modificada na página falsa. O portal *fake* tinha a mesma aparência do G1 e a matéria tinha o mesmo título e fotos da notícia original, mas com variação no conteúdo. Um dos poucos indícios que denunciavam a falsificação era a URL da página (CATRACA LIVRE, 2017).

Figura 5 – Exemplo de Conteúdo Impostor.

Fonte: Catraca Livre (2017).

2.2.6 Manipulação de conteúdo

A manipulação de conteúdo se dá quando informações verdadeiras, imagens ou vídeos são manipulados com o propósito de enganar ou criar notícias virais. De acordo com Wardle (2017), o conteúdo manipulado é bem parecido com o do falso contexto mas neste tipo de notícia o conteúdo é utilizado para benefícios próprios ou para tentar desinformar os usuários sobre determinado assunto verdadeiro.

Um vídeo postado pelo chefe de comunicação da Casa Branca foi o primeiro na história do Twitter a ser classificado como conteúdo manipulado. Dan Scavino, diretor de mídias sociais da Casa Branca, publicou um vídeo (ver Figura 6) com um dos potenciais opositores de Donald Trump nas eleições deste ano, 2020. O vídeo mostra o democrata dizendo que os

eleitores "só podem reeleger"o presidente republicano, Donald Trump, entretanto, o conteúdo foi manipulado para remover o contexto no qual o discurso do Joe Biden foi expressado. Na verdade, o mesmo teria dito: "Desculpem, nós só podemos reeleger Donald Trump se ficarmos de fato presos nessa lógica de atirar uns nos outros aqui", referindo-se às divisões internas de seu partido.

O vídeo ainda se encontra disponível na plataforma, mas foi classificado pelo administradores da rede social como "mídia manipulada". No entanto, apesar da classificação, o vídeo já conta com mais de seis milhões de visualizações e foi compartilhado pelo próprio presidente dos EUA, Donald Trump.

Figura 6 – Exemplo de Conteúdo Manipulado.

Fonte: Twitter (2020).

2.2.7 Conteúdo fabricado

De acordo com Zannettou *et al.* (2018), o conteúdo fabricado é um tipo de notícia que não tem base em fatos reais, o que leva este a ser o tipo mais grave de informações falsas, pois seu único objetivo é desinformar o público. Serra (2018) diz que de todos os sete tipos de *fake news*, o conteúdo fabricado é o mais praticado, já que seu conteúdo consiste totalmente em informações inventadas. Uma das *fake news* desta categoria que ganharam maior repercussão

foi a de que o papa teria anunciado apoio ao Donald Trump durante as eleições presidenciais de 2015 nos Estados Unidos (ver Figura 7).

Figura 7 – Exemplo de Conteúdo Fabricado.

Fonte: Serra (2018).

2.3 *Fake news* na era da informação digital

Atualmente, a informação está presente em abundância nos mais diversos meios, principalmente nas formas digitais, características do que se convencionou chamar de "Web"a rede de conexões e informações desenvolvida com o advento da internet (CHOUDHURY, 2014). O atual contexto informacional dá-se a partir da constante produção, disseminação e consumo no âmbito digital principalmente por meio dos compartilhamentos em redes sociais, tais como Facebook e Twitter, e em aplicativos de mensagens instantâneas como, por exemplo, o Whatsapp.

> Esse tráfego de informações por meio de novas formas de acesso e produção de conteúdo, porém, tem possibilitado o consumo e disseminação de informações falsas, distorcidas, manipuladas, servindo às mais diversas finalidades pessoais e institucionais. A popularização de termos como "fakenews", "pós-verdade"e "desinformação"tem trazido à tona uma recente preocupação com a veracidade e a confiabilidade das informações disseminadas na web, as quais acabam formando opiniões e construindo pretensos conhecimentos, baseados em informações falsas ou imprecisas (RIPOLL; MATOS, 2017).

O aumento de ferramentas tecnológicas como smartphones, tablets, notebooks, etc,

possibilitou o desenvolvimento de uma sociedade que está conectada a todo momento por meio do fácil acesso e tráfego de informação. A vida social é constantemente transferida do espaço físico para o espaço virtual e a nova realidade passa a ser a representação imagética, a sua virtualização (BAUDRILLARD, 1999).

De acordo com um estudo realizado pelo Instituto de Tecnologia de Massachusetts, publicado pela revista Science e noticiado pelo Correio Braziliense (2018), as *fake news* se disseminam 70% mais rápido que as notícias verdadeiras. Enquanto estas atingem uma média de mil pessoas, as notícias falsas, quando mais populares, alcançam um público de até cem mil indivíduos. As mídias sociais, por estarem cada vez mais sendo responsáveis pela disseminação de uma grande quantidade de notícias, estão não somente se entrelaçando com o jornalismo, mas estão a ponto de se tornarem fontes primárias de notícias (ROCHLIN, 2017). A preferência pela rapidez em publicar a notícia em detrimento da precisão e o uso de várias estratégias para obter cliques e compartilhamentos têm favorecido a ampla disseminação das *fake news* (MÜLLER; SOUZA, 2018). Segundo Souza (2017):

> Ao longo de sua história, o jornalismo sempre conviveu em menor ou maior grau com notícias falsas. Boatos publicados sem apuração, notícias pagas para favorecer alguém, notícias simplesmente inventadas em veículos sensacionalistas – tudo isso não vem de hoje e foi algo com que a imprensa sempre buscou lidar. No entanto, com a internet, a proliferação das notícias falsas aumentou exponencialmente (SOUZA, 2017)

Para Kovach e Rosenstiel (2003) a notícia é o principal produto do jornalismo e se sustenta por meio da necessidade do ser humano: o instinto de percepção. Os autores reforçam ainda o compromisso do jornalismo com a verdade e o consentimento entre os profissionais sobre a importância de apurar bem os fatos, buscar a exatidão, a equidade e a verdade, em um trabalho contínuo que está na essência das notícias.

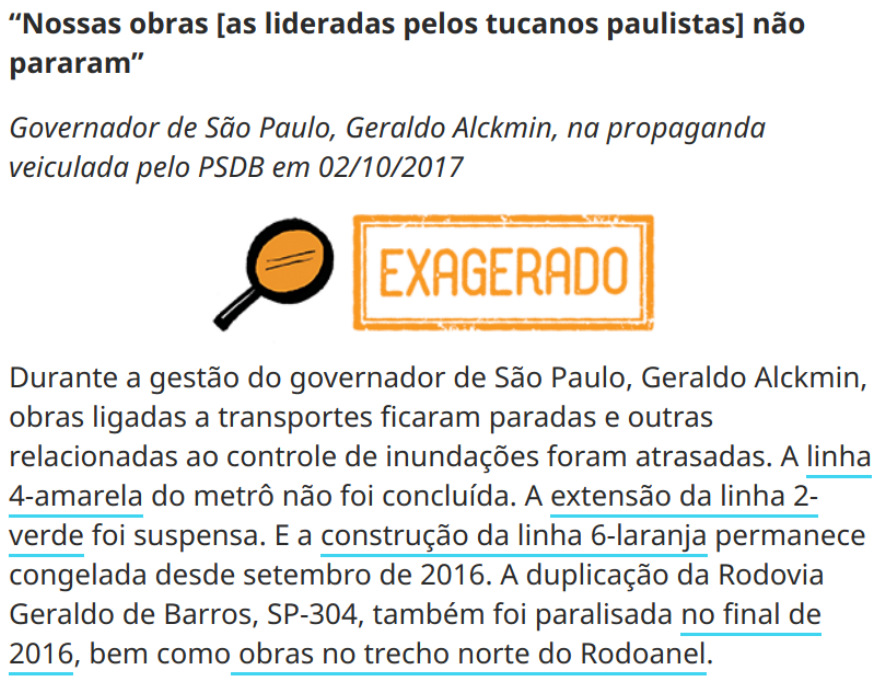
2.4 Informação e *fact-checking*

Segundo Serra (2018), as técnicas e ferramentas usadas no *fact-checking* consistem no trabalho do jornalismo investigativo. A verificação de notícias refere-se a confirmação de fatos e dados difundidos em artigos replicados pela internet. Esta prática consiste em avaliar a exatidão de declarações, principalmente de discursos políticos, com o objetivo de detectar erros, imprecisões e mentiras.

Diante de um cenário caótico de informações, as agências de notícias e os meios de comunicação passaram a criar ou investir em ferramentas de apuração e checagem de fatos a fim de garantir que informações mal apuradas e inverídicas não sejam disseminadas. Os serviços e as iniciativas de *fact-checking* alcançaram uma importância tão grande, que passaram a ser lideradas por uma organização mundial: a *International Fact-Checking Network*, ou IFNC. No Brasil, apenas três agências são certificadas por esta organização, são elas: Lupa, Truco e Aos Fatos. O intuito é que as instituições credenciadas estabeleçam compromissos com apartidarismo e equidade, transparência das fontes, transparência de financiamento da organização, transparência de método e correções francas e amplas.

Um exemplo de checagens de matérias foi feita pela Agência Lupa após Geraldo Alckmin, o então governador de São Paulo e mais tarde candidato à presidência do Brasil pelo PSDB, falar que obras do seu governo "nunca eram paralisadas". A agência então utilizou de fontes referenciais para confrontar a situação e mostrar que a fala do político não estava correta. Para isso, a iniciativa recorreu a informações encontradas em diários oficiais e no site do governo do Estado de São Paulo.

Figura 8 – Governador de São Paulo, Geraldo Alckmin, na propaganda veiculada pelo PSDB em 02/10/2017.



Fonte: Lupa (2017).

Outro exemplo de iniciativa desenvolvida com o intuito de alertar sobre conteúdos duvidosos disseminados na internet, esclarecendo quais notícias são verdadeiras e quais são

falsas, é o "Fato ou Fake"do G1 (ver Figura 9). Este serviço, lançado em 2018, trata-se de uma seção editorial do portal G1, onde jornalistas são responsáveis por fazerem um monitoramento diário com o intuito de identificar mensagens suspeitas compartilhadas nas redes sociais e por aplicativos como o *WhatsApp*. O serviço também conta com um *bot* nas redes sociais Facebook e Twitter responsável por divulgar os conteúdos que já tenham sido verificados pelos jornalistas da Globo, informando o que é falso e o que é verdadeiro.

Figura 9 – Fato ou *Fake*: serviço de checagem de conteúdos suspeitos.

Fonte: Lupa (2017).

Diante deste cenário, têm-se crescido a discussão sobre como a tecnologia pode vir a ser nova fonte de verificação automatizada de *fake news* para o jornalismo, sites agregadores de conteúdos e a sites de mídias sociais. De acordo com Scientifc American Brasil (2018), algoritmos de análise linguística adotam uma abordagem diferente, analisando atributos quantificáveis como estrutura gramatical, escolha de palavras, pontuação e complexidade, o que poderia vir a auxiliar e acelerar o trabalho de diversos profissionais, levando em consideração que um algoritmo pode trabalhar a detecção de notícias falsas de forma rápida, além de também poder ser utilizado na análise de vários tipos de matérias.

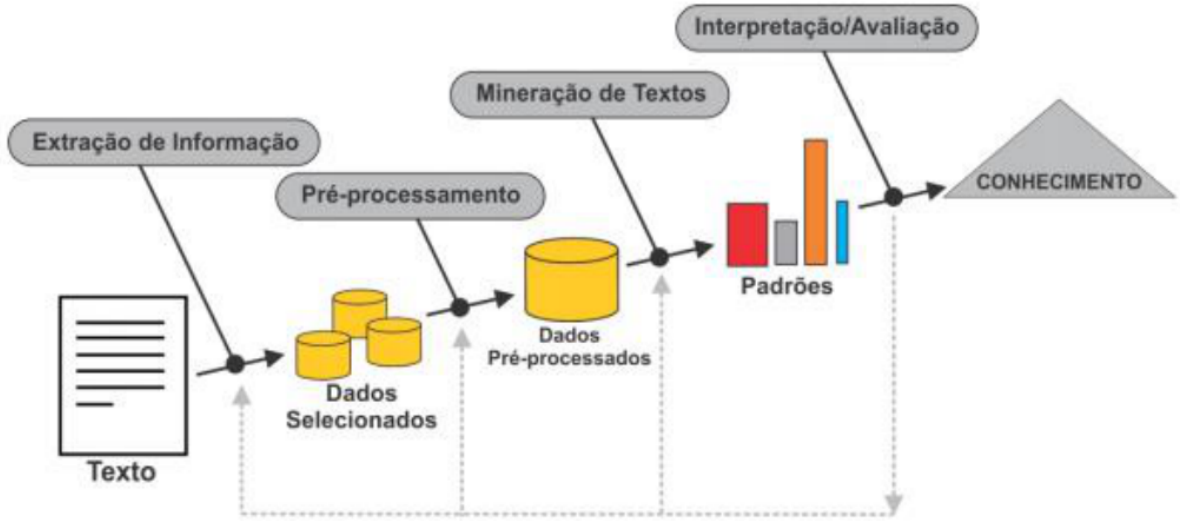
2.5 Processamento de Linguagem Natural (PLN)

O Processamento de Linguagem Natural (PLN, em inglês *Natural Language Processing*), é uma subárea da ciência da computação, inteligência artificial e linguística, que estuda os problemas da geração e compreensão da linguagem humana (COSTA, 2019). A PLN também consiste no desenvolvimento de modelos computacionais para a realização de tarefas que dependem de informações expressas em alguma língua natural como, por exemplo, tradução e interpretação de textos, busca de informações em documentos e interface homem-máquina (PEREIRA, 2009).

Segundo Yse (2019), a PLN representa o tratamento automático da linguagem humana natural, como a fala e texto. Entretanto, o valor real por trás dessa tecnologia vem dos cenários em que ela é implementada. O autor destaca alguns desses casos como, por exemplo, o uso da tecnologia por organizações como Yahoo e Google para classificar e filtrar e-mails a partir da análise de textos, onde o intuito seria o de evitar que *spams* cheguem na caixa de entrada do usuário. Outro caso de uso, também evidenciado pelo autor, foi a utilização da PLN no auxílio do processo de identificação de notícias falsas, desenvolvido pelo Instituto de Tecnologia de Massachusetts, com um sistema capaz de determinar se uma fonte é precisa ou politicamente tendenciosa, detectando assim se a mesma pode ser confiável ou não.

O processamento da linguagem segue uma cadeira de ações, conforme a Figura 10, que se iniciam na leitura de arquivos brutos de texto, selecionando dados de interesse a serem observados e minerando informações por meio da pré-seleção realizada. Depois desta fase, os padrões obtidos por meio da etapa anterior são então utilizados para obter algum conhecimento específico.

Figura 10 – Etapas do processamento de linguagem natural.



Fonte: Mondoni (2019).

A PLN, apesar dos avanços, ainda é caracterizada como um problema difícil na ciência da computação. Entender a linguagem humana é compreender não somente as palavras, mas também os conceitos e como eles estão ligados para criar significado. Apesar de ser uma das coisas mais fáceis para os humanos aprenderem, a linguagem e a sua ambiguidade é o que torna a PLN um problema tão difícil para os computadores dominarem (AHO, 2003). Os algoritmos de processamento de linguagem natural são tipicamente baseados em algoritmos de aprendizado de máquina. Essa tecnologia evita que seja preciso codificar manualmente grandes conjuntos de regras, já que a mesma utiliza da aprendizagem automática para treinar um modelo de processamento. Ela faz isso através da análise de um conjunto de exemplos (ou seja, um corpus grande, como um livro, até uma coleção de sentenças). Em geral, quanto mais dados forem analisados, mais preciso será o modelo (PEREIRA, 2009).

2.5.1 TF-IDF

O TF-IDF (*Term Frequency-Inverse Document Frequency*) trata-se de uma medida estatística que tem como intuito indicar a importância de um termo, palavra ou frase específica para determinado documento (CASAROTTO, 2019). Este cálculo é o resultado do produto de dois fatores: o TF (*term frequency*), que refere-se à frequência de uma palavra dentro do universo de documentos; e IDF (*inverse document frequency*), o número total de documentos existentes dividido pelo número total de documentos em que uma determinada palavra aparece. Ao verificar

a semelhança entre dois documentos, por exemplo, é intuitivo inicialmente analisar as palavras que estão presentes em ambos e verificar o número de vezes que cada termo aparece em cada um dos documentos, sendo este processo feito pelo fator TF. Além disso, visto que alguns termos são bastante comuns como, por exemplo, artigos e conjunções, também é necessário ponderar a frequência com que esses termos ocorrem, para isso utiliza-se o fator IDF (CAVALCANTI *et al.*, 2011).

O cálculo do TF-IDF dá-se a partir da Equação 2.1, sendo t_j a frequência de um termo em dado documento d_i multiplicado pelo fator de ponderação, que varia entre 0 e o $\log N$, onde N é o número de documentos do conjunto de dados e $d(t_j)$ é o número de documentos onde há a ocorrência do termo pelo menos uma vez (CAVALCANTI *et al.*, 2011).

$$tfidf(t_j, d_i) = freq(t_j, d_i) x \log \frac{N}{d(t_j)} \tag{2.1}$$

Ao aplicar a fórmula, o valor do fator de ponderação é igual a 0 quando o termo aparece em todos os documentos, mas quando aparece em apenas um, ele é $\log N$. Na equação, o valor do $tdidf$ favorece a frequência do termo t_j no documento d_i e desfavorece a sua frequência na coleção $d(t_j)$, ou seja, as particularidades de cada documento são acentuadas (MALTA; KUROIVA, 2019). Dentro dessa lógica, o TF-IDF serve para processar a linguagem utilizada nos conteúdos. A sua utilização não consiste em dar sentido aos termos, mas sim entender a sua importância ao dar pesos diferentes para cada um destes (CASAROTTO, 2019).

2.6 Python e algoritmos de processamento de linguagem natural

A linguagem Python teve seu desenvolvimento iniciado em 1989 no Instituto Nacional de Pesquisa em Matemática e Ciência da Computação, CWI (Centrum Wiskunde & Informatica), na Holanda, por Guido van Rossum (PYTHON, 2020). Com uma sintaxe bastante similar a linguagem C, Python foi desenvolvido para preencher as lacunas existentes entre a linguagem C e o shell, por isso é considerada uma linguagem interpretada, orientada a objetos e interativa.

Python tem uma grande biblioteca padrão que contém classes, métodos e funções para realizar essencialmente qualquer tarefa. A ferramenta segue a filosofia de "baterias incluídas", o que significa que tudo o que é preciso para rodar uma aplicação está, na maioria das vezes, presente na instalação básica da plataforma, sem a necessidade de se instalar bibli-

otecas adicionais (PYSCIENCE-BRASIL, s.d.). Essa característica permite sua utilização em diversos cenários de diferentes complexidades como, por exemplo, automatização de gerenciamento de infraestrutura, desenvolvimento de aplicações web e, no caso deste trabalho, análise e processamento de dados.

2.6.1 Natural Language Toolkit (NLTK)

O NLTK foi originalmente criado em 2001 como parte de um curso de linguística computacional do Departamento de Ciência da Computação e Informação da Universidade da Pensilvânia. NLTK é uma plataforma utilizada para desenvolver programas, na linguagem de programação Python, que trabalham com dados de linguagem humana para aplicação em processamento de linguagem natural (BARBOSA *et al.*, 2017).

O NLTK foi desenvolvido com quatro objetivos principais: simplicidade, consistência, extensibilidade e modularidade (NLTK, 2018). O módulo NLTK trata-se de um conjunto de ferramentas cujo o propósito é trabalhar com processamento de linguagem natural. Para isso, a ferramenta conta com um conjunto de bibliotecas que podem ser utilizadas em diferentes situações como, por exemplo, para separar sentenças em um parágrafo, separar palavras dentro de cada sentença, reconhecer padrões em um texto e também criar modelos de classificação que permitam fazer uma análise a partir de um conjunto de dados.

O NLTK trabalha com algumas bibliotecas essenciais no tratamento de dados, são elas: *tokenização*, *stopwords* e *stemmer*. A tokenização, também conhecida como segmentação de palavras, quebra a sequência de caracteres em um texto localizando o limite de cada palavra, ou seja, os pontos onde uma palavra termina e outra começa (INDURKHYA; DAMERAU, 2010). As *stopwords* são palavras que podem ser consideradas descartáveis para a compreensão do sentido de um texto, ou seja, são semanticamente irrelevantes para a construção do modelo, logo podem ser removidas na fase de pré-processamento dos dados (BARBOSA *et al.*, 2017). Já o *stemmer* refere-se a técnica de remover sufixos e prefixos de uma palavra, chamada *stem*.

2.6.2 Scikit-Learn

O scikit-learn fornece uma biblioteca, ou seja, uma coleção de classes de funções, de aprendizado de máquina de código aberto para a linguagem de programação Python (PEDREGOSA *et al.*, 2011). A biblioteca provê algoritmos de aprendizado supervisionado e não supervisionado através de interfaces pré-definidas. Além da classificação, foco deste trabalho,

o scikit-learn contém outros módulos, tais como: regressão e clusterização. Por se tratar de uma biblioteca para mineração e análise de dados, a mesma conta com uma série de pacotes necessários para o seu aproveitamento, como por exemplo, o matplotlib[1], o numpy[2] e o pandas[3]. Neste projeto, todos os classificadores utilizados e testados foram providos pelo scikit-learn.

2.7 Aprendizado de máquina

De acordo com Costa (2019), o aprendizado de máquina, ou AM, é uma área da inteligência artificial que visa estudar algoritmos que sejam capazes de aprender a resolver problemas de maneira autônoma. AM estuda métodos computacionais para adquirir novos conhecimentos, novas habilidades e novos meios de organizar o conhecimento já existente (MITCHELL *et al.*, 1997). Os algoritmos conseguem reconhecer e extrair padrões de um grande volume de dados, através de um modelo de aprendizado que é criado com objetivo de tornar suas decisões e predições mais exatas possíveis com base nos padrões descobertos (MARSLAND, 2015). Mitchell *et al.* (1997) afirma ainda que um entendimento detalhado dos algoritmos de AM pode levar também a um melhor entendimento da capacidade, e incapacidade, do aprendizado humano.

AM destina-se a estudar formas de programar um computador para que este seja capaz de entender padrões e a partir deles, tomar decisões automáticas sobre um determinado assunto. Muitas são as aplicações que necessitam desempenhar este tipo de atividade, como por exemplo, classificação automática de documentos, reconhecimento de caracteres escritos manualmente, extração de conhecimentos de dados biológicos, etc. Desta forma, para que estas atividades possam ser executadas é necessário que a aplicação obtenha uma base de conhecimento sobre a qual são analisadas as hipóteses e, após a tomada de decisão, seja realizada uma ação sobre elas (PIVETTA, 2013).

2.7.1 A hierarquia do aprendizado

Segundo Monard e Baranauskas (2003), a indução é a forma de inferência lógica que permite obter conclusões genéricas a partir de conjuntos de fatos ou exemplos particulares, podendo assim ser caracterizada como o raciocínio que parte de um conceito específico e o

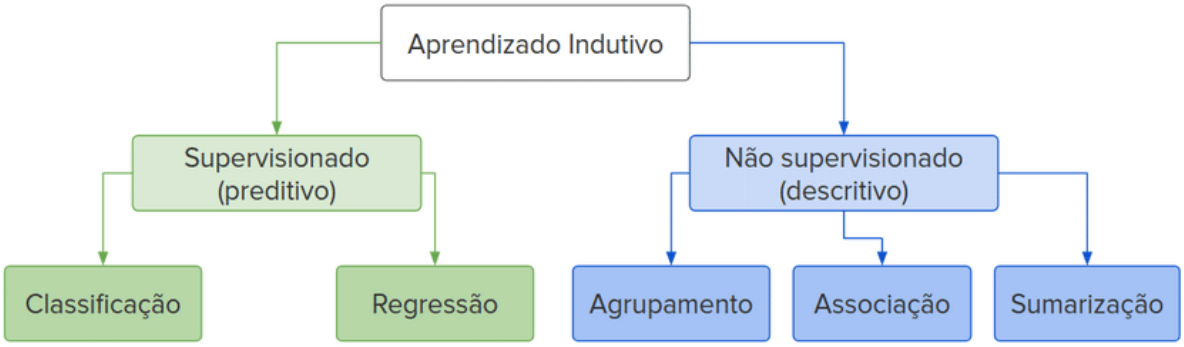
[1] <https://matplotlib.org/>
[2] <https://www.numpy.org/>
[3] <https://pandas.pydata.org/>

generaliza. Na indução, as hipóteses geradas através da inferência indutiva podem ou não preservar a verdade, visto que se o número de exemplos for insuficiente, ou se estes não forem bem selecionados, as hipóteses obtidas podem ser de pouco valor. Entretanto, apesar disso, a indução continua sendo considerada um dos principais meios de criar novos conhecimentos e predizer eventos futuros (MONARD; BARANAUSKAS, 2003).

O paradigma mais difundido do aprendizado indutivo de máquina é conhecido como aprendizado indutivo a partir de exemplos. Ele consiste em induzir descrições gerais de conceitos utilizando um dado conjunto de exemplos específicos desses mesmos conceitos (NICOLETTI, 1994). Diversos algoritmos de aprendizado de máquina utilizam o aprendizado indutivo para realizar generalizações e especializações na indução de uma hipótese, a partir de exemplos. Nesse caso, um sistema que aprende por meio de exemplos recebe como dados informações na forma de situações específicas, cada uma delas devidamente classificada, e produz, como resultado, uma hipótese que generaliza aquelas situações inicialmente fornecidas (NICOLETTI, 1994).

De acordo com Monard e Baranauskas (2003), o aprendizado de máquina indutivo pode ser dividido em duas categorias: aprendizado supervisionado e não supervisionado (ver Figura 11). No aprendizado supervisionado, o objetivo é induzir conceitos a partir de um determinado conjunto de exemplos que já estão rotulados com uma classe conhecida. Nesse tipo de aprendizado, se as classes possuem valores discretos, o problema é conhecido como classificação. Porém, caso as classes possuam valores contínuos, o problema é classificado como regressão (MOHRI *et al.*, 2018). Já no aprendizado não supervisionado, os exemplos não possuem uma rotulação pré-definida. Desta forma, a base de treinamento é formada através da similaridade entre as informações, sendo o modelo probabilístico o responsável por encontrar o resultado (NILSSON, 2009).

Figura 11 – A hierarquia do aprendizado de máquina.



Fonte: Barreto (2018).

A escolha de qual tipo de aprendizado utilizar, supervisionado ou não supervisionado, depende da tarefa e dos exemplos estarem ou não rotulados com o atributo classe. Neste trabalho, o intuito é construir um classificador que possa determinar corretamente se uma notícia é verdadeira ou falsa, baseado em um conjunto de dados com saídas já rotuladas. Para isso, será utilizado o método de classificação do aprendizado supervisionado.

2.7.2 Aprendizado de máquina supervisionado

No aprendizado supervisionado é fornecido a um algoritmo de aprendizado, denominado indutor, um conjunto de dados de treinamento $E = \{E_1, E_2, ...E_N\}$, onde cada exemplo $E_i \in E$ possui um rótulo associado que define a classe a qual o mesmo pertence. Formalmente, pode-se dizer que cada exemplo $E_i \in E$ pode ser representado por uma tupla

$$E_i = (\overrightarrow{x_i}, y_i) \tag{2.2}$$

O vetor $\overrightarrow{x_i}$ corresponde aos valores que representam as características, ou atributos, do exemplo E_i e y_i equivale a um dos possíveis valores do atributo Y associado a classe E_i. A Tabela 1 representa a forma geral de um conjunto de exemplos E com N exemplos e M atributos e está representada no formato atributo-valor, padrão utilizado como entrada pela maioria dos algoritmos de aprendizado de máquina. No modelo atributo-valor, as colunas $(A_1, ...A_M)$ da tabela representam os diferentes atributos, as linhas $(E_1, ...E_N)$ os diferentes exemplos e Y o atributo que assume os valores da classe de cada exemplo E_i.

Tabela 1 – Conjunto de dados no formato atributo-valor.

	A_1	A_2	...	A_M
E_1	x_{11}	x_{12}	$\dots$	x_{1M}
E_2	x_{21}	x_{22}	$\dots$	x_{2M}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
E_N	x_{N1}	x_{N2}	$\dots$	x_{NM}

Fonte: Batista e Monard (2004).

O objetivo do aprendizado supervisionado é induzir um mapeamento geral dos vetores $\overrightarrow{x}$ para valores y. Portanto, o sistema de aprendizado deve construir um modelo, $y = f(\overrightarrow{x})$, de um função desconhecida f, também chamada de função conceito, que permite predizer valores y para exemplos previamente não vistos. Entretanto, o número de exemplos utilizados para a criação do modelo não é, na maioria dos casos, suficiente para caracterizar

completamente essa função f. Na realidade, os sistemas de aprendizado são capazes de induzir uma função **h** que aproxima f, ou seja, $\mathbf{h}(\overrightarrow{x}) \approx f(\overrightarrow{x})$. Desta forma, **h** é chamada de hipótese sobre a função conceito f (BATISTA; MONARD, 2004).

Para este trabalho, dispõe-se de um grande conjunto de dados previamente classificados, os quais podem ser utilizados no treinamento do classificador, tornando-se útil então o emprego do aprendizado supervisionado. Dentre os algoritmos que utilizam este tipo de aprendizado para a obtenção da base de conhecimento, alguns merecem destaque quando empregados na classificação textual. A seguir, são apresentados alguns destes algoritmos.

2.8 Aprendizado de máquina supervisionado e classificadores

2.8.1 Naive Bayes

O classificador *Naive Bayes* trata-se de uma técnica de classificação baseada em algoritmos de aprendizado de máquina capazes de fornecer predições associadas a valores de probabilidades (WANKE *et al.*, 2014). O *Naive Bayes* também pode ser definido como um método probabilístico de aprendizado supervisionado, derivado da regra de Bayes (proposto por Thomas Bayes), que tem como objetivo encontrar a probabilidade a *posteriori*. O classificador *bayesiano* apresenta uma maneira de calcular a probabilidade da ocorrência de um evento, baseando-se em probabilidades obtidas da análise de eventos passados. Estas informações são utilizadas para construir a base de conhecimento da aplicação, ou seja, o conjunto de informações responsáveis pela classificação correta das hipóteses.

O propósito do classificador *Naive Bayes* é verificar se uma amostra analisada pertence ou não a uma determinada classe. A obtenção desta resposta realiza-se através de uma análise estatística das informações coletadas sobre as instâncias fornecidas. Mitchell *et al.* (1997) apresenta o seu funcionamento através da Equação 2.3, onde c é a categoria, d o documento (no caso deste trabalho, as notícias), $P(c)$ e $P(d)$ probabilidades a *priori* e $P(c|d)$ a *posteriori*. A probabilidade a *priori* não tem informações sobre outros eventos, enquanto que a *posteriori* é uma probabilidade condicional e considera a ocorrência de eventos passados.

$$P(c|d) = \frac{P(c)P(d|c)}{P(d)} \tag{2.3}$$

A probabilidade de um documento d, uma notícia, estar em determinada categoria

c é dada pela fórmula 2.4, sendo $P(c)$ a probabilidade *a priori* de um documento d estar na categoria c e $P(t_k|c)$ a probabilidade de um termo k do documento ocorrer num documento de classe c.

$$P(c|d)\alpha P(c)\prod_{k=1}^{n_d} P(t_k|c) \tag{2.4}$$

O algoritmo *Naive Bayes* assume a hipótese de independência condicional, ou seja, o mesmo supõe que os atributos são considerados não correlacionados. Isso significa que a presença de um determinado atributo não tem nenhuma relação com os outros. Os termos $\{t1,t2,...,tn_d\}$ são *tokens* que estão presentes no vocabulário usado no processo de classificação, onde n_d é a quantidade de *tokens* do vocabulário presentes no documento d. Por exemplo, na frase "Casa do Estudante em Fortaleza abriga jovens que apostaram na educação para superar dificuldades", os *tokens* do documento são representados por {Casa, Estudante, Fortaleza, abriga, jovens, apostaram, educação, superar, dificuldades}, logo, $n_d = 9$.

O intuito do uso do *Naive Bayes* é, basicamente, identificar a melhor classe para cada documento analisado. A fórmula usada pela maioria das implementações deste algoritmo de classificação é dada pela Equação 2.5 (MANNING *et al.*, 2008).

$$c_{map} = \arg\max[\log\hat{P}(c)\sum_{k=1}^{n_d}\log\hat{P}(t_k|c)] \tag{2.5}$$

O símbolo ^ utilizado em $P(c)$ e $P(t_k|c)$ representa as estimativas realizadas a partir de um conjunto de treinamento. A maximização é feita com o intuito de obter a classe, para o documento, cuja probabilidade é a maior. O logaritmo da soma é utilizado para evitar o *underflow*, já que as probabilidades se tornam números cada vez menores ao serem realizadas sucessivas multiplicações, o que não seria possível de representar pelo computador (CAMARGO; BRAGHETTO, 2016).

Para calcular a probabilidade a *priori* é utilizada a frequência relativa, ou seja, o resultado obtido da divisão entre o número de documentos (N_c) na classe c e o total de documentos (N). Logo, $\hat{P}(c)$ é dado pela equação 2.6.

$$\hat{P}(c) = \frac{N_c}{N} \tag{2.6}$$

A probabilidade condicional de um termo ser de determinada classe é calculada por meio da equação 2.7 (CAMARGO; BRAGHETTO, 2016). Na fórmula, o vocabulário é representado pela letra V, já o T_{ct} simboliza a quantidade de vezes que o termo t ocorre na classe c no conjunto de treinamento. O número 1 somado no numerador e denominador é chamado de *Laplace Smoothing*, utilizado como forma de correção que consiste em adicionar dados à base de treinamento para que não haja probabilidades iguais a 0 (MANNING *et al.*, 2008). Como na Equação 2.4 é usado um produtório, se uma das probabilidades fosse nula, a probabilidade a *posteriori* seria nula (CAMARGO; BRAGHETTO, 2016).

$$\hat{P}(t|c) = \frac{T_{ct} + 1}{\sum_{t' \varepsilon V}(T_{ct'} + 1)} \tag{2.7}$$

Uma das principais vantagens do classificador *Naive Bayes* é sua rapidez de uso. Isso porque é o algoritmo mais simples entre algoritmos de classificação (KOTSIANTIS; PINTELAS, 2004). Entretanto, apesar de ser um método de fácil uso e de utilizar a hipótese de independência condicional, o que não é semelhante ao que acontece nos casos reais, a classificação textual feita pelo *Naive Bayes* costuma ser boa e, em alguns casos, até melhor que métodos mais sofisticados (CAMARGO; BRAGHETTO, 2016).

2.8.2 Multinomial Naive Bayes

O classificador *Multinomial Naive Bayes*, também conhecido como *Multinomial NB*, é uma variante do modelo *Naive Bayes*, mas com a utilização da distribuição multinomial para cada termo, ou seja, as palavras de cada sentença. Desse modo, o algoritmo *Multinomial NB* funciona bem para dados que podem ser contados, se tornando um bom modelo para classificação de tópicos, por exemplo (RUSSELL; NORVIG, 2004). A distribuição multinomial é parametrizada por $\theta_{ci} = (\theta_{c1}, \theta_{c2}, \ldots, \theta_{cn})$, onde a probabilidade θ_{ci} é dada pela seguinte equação:

$$\hat{\theta}_{ci} = \frac{N_{ci} + \alpha}{N_c + \alpha n} \tag{2.8}$$

A equação dá-se, basicamente, pelo número de vezes que uma palavra i aparece em documentos de uma classe c (N_{ci}) dividido pelo número total de ocorrências de palavras em uma classe (N_c). Onde n representa o número total de palavras e alfa é uma constante que contabiliza

os recursos que não estão presentes nas amostras de aprendizado, impedindo assim que haja probabilidade igual a 0.

2.8.3 Passive Agressive

O *Passive Agressive* é um meta-algoritmo de classificação binária comumente utilizado para aprendizados em larga escala, onde não é necessária uma taxa de aprendizado, visto que, basicamente, o mesmo itera sobre as predições e a partir delas analisa o modelo construído e, caso seja necessário, modifica para adaptar a predição incorreta (SONONE, 2018).

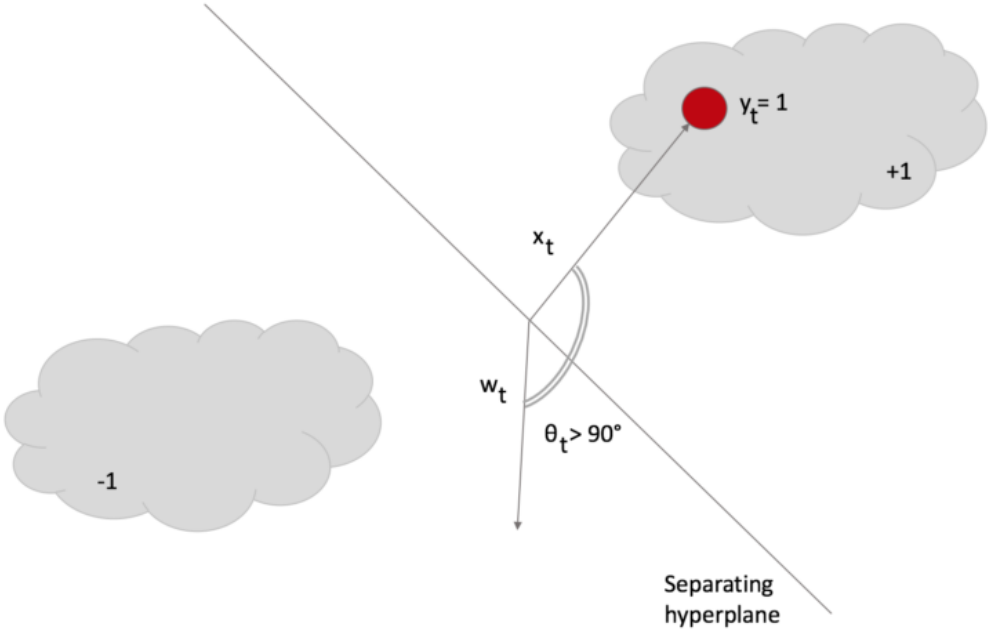
A classificação binária, usada para prever qual de duas classes um documento está associado, ocorre em uma sequência de rodadas. Em cada rodada, o algoritmo *Passive Agressive* observa uma instância de dados e prevê a sua classe usando a hipótese atual, podendo esta ser +1 ou -1. Depois da predição ter sido feita, a verdadeira classe é revelada e o algoritmo sofre então uma perda instantânea que reflete o grau em que a previsão estava incorreta. No final de cada rodada, o algoritmo usa o par instância-classe recém obtido a fim de melhorar a regra de previsão atual para as rodadas seguintes que irão vir.

O algoritmo *Passive Agressive* funciona por meio da seguinte regra:

$$\begin{cases} \bar{w}_{t+1} = argmin_{\bar{w}} \dfrac{1}{2}||\bar{w} - \bar{w}_t||^2 + C\xi \\ L(\bar{w}; \bar{x}_t, \bar{y}_t) \leq \xi \end{cases} \tag{2.9}$$

Assumindo que a variável x_t trata-se da instância de dados então apresentada e que a variável w_t refere-se ao vetor utilizado para realizar as predições, pode-se dizer então que o vetor w_t é utilizado para determinar a classe da instância x_t. Se o valor previsto é correto, a perda da função é igual a 0 e o valor de *argmin* é igual a w_t, o que significa que o algoritmo é passivo, ou seja, indica que a classificação foi feita corretamente.

Figura 12 – Representação visual do algoritmo *Passive Agressive*.

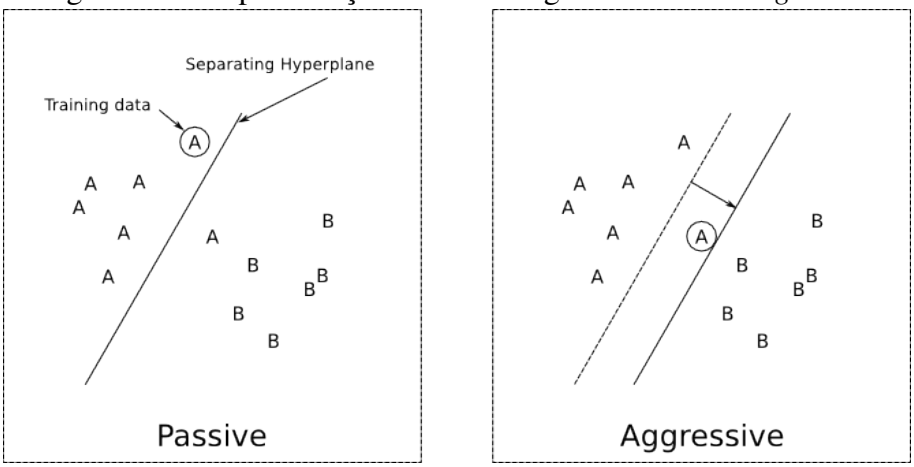


Fonte: (SIDESWIPE, 2020).

Na Figura 12 pode ser visto um exemplo de uma classificação feita incorretamente. O ângulo $\theta > 90^o$ indica que o produto escalar é negativo, sendo a instância x_t então classificada como -1, entretanto, o seu valor verdadeiro é +1, logo, pode-se dizer que a classificação foi executada de forma incorreta. Nesse caso, a regra de atualização torna-se agressiva devido ao fato de que o algoritmo precisa encontrar um novo vetor w_t o mais próximo possível do vetor anterior, caso contrário, o conhecimento existente da hipótese atual é imediatamente perdido.

Basicamente, o algoritmo *Passive Agressive* opera criando um vetor que possui uma "zona"utilizada para balizar a posição das predições em categorias representadas por -1, 0 e 1, que tem como intuito separar as amostras de dados, de forma que as predições sejam posicionadas em uma das zonas definidas pelo vetor (ver Figura 13). Ao realizar uma nova predição, o valor resultado é analisado em relação a origem e ao valor esperado, caso a predição seja positiva o modelo é alterado, do contrário permanece o mesmo (CRAMMER *et al.*, 2006).

Figura 13 – Representação visual do algoritmo *Passive Agressive*.



Fonte: (SIDESWIPE, 2020).

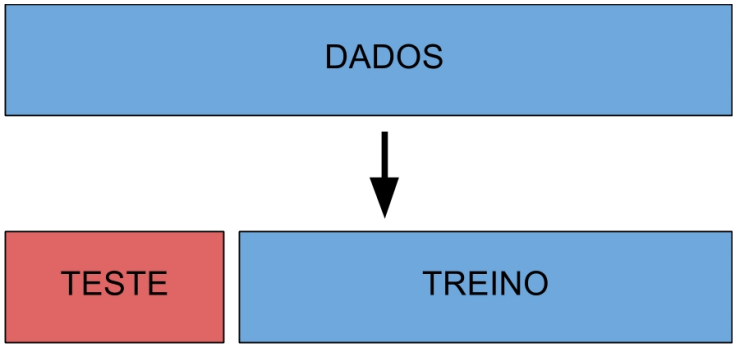
2.9 Algoritmos de fragmentação

Nas subseções que seguem serão abordados alguns algoritmos de fragmentação, utilizados neste trabalho, como é o caso do *Hold-Out* e o *Cross-Validation*.

2.9.1 Algoritmo Hold-Out

O método de avaliação *Hold-Out*, também chamado de *Percent Split* (WITTEN; FRANK, 2002), consiste em dividir um único conjunto de dados em dois subconjuntos disjuntos de treinamento e teste, sendo o primeiro conjunto utilizado na construção do modelo de classificação e o segundo utilizado para calcular a precisão deste mesmo modelo. Algumas proporções sugeridas na divisão do conjunto são, por exemplo, 70%-30% e 50%-50% (primeira e segunda fases de classificação).

Figura 14 – Representação visual do método *Hold-Out*.



Fonte: (Laboratório de Estatística e Geoinformação - LEG/UFPR, 2016).

O algoritmo recebe como entrada a informação referente à percentagem na qual os

dados serão divididos e a partir disso cria um subconjunto de treinamento com *x*% do tamanho da base de dados fornecida como entrada, onde *x* trata-se da percentagem dada. Depois disso, com o restante dos dados é então criado um subconjunto de dados de teste. Definido os conjuntos de treino e teste, o algoritmo de classificação é então aplicado sobre o subconjunto de treinamento. A partir disso, acontece um laço de repetição que itera *j* vezes, sendo *j* o número de instâncias do subconjunto de teste. A cada iteração deste laço, é processada uma predição do classificador sobre a instância de teste corrente (PIMENTA *et al.*, 2009).

O método *Hold-Out* trabalha de forma simples e por isso apresenta algumas limitações. Uma delas é de que caso sejam poucos os dados com valores conhecidos de classe, menos ainda serão aqueles usados na construção do modelo, devido a divisão do conjunto original em dois subconjuntos de treino e teste. Outro problema está no cálculo do erro, visto que ao separar apenas um conjunto para testes, a estimativa de erro do algoritmo pode ser tendenciosa caso seja escolhido um conjunto de testes que produzam resultados muito bons ou muito ruins, impactando diretamente no desempenho do modelo de classificação e na precisão deste.

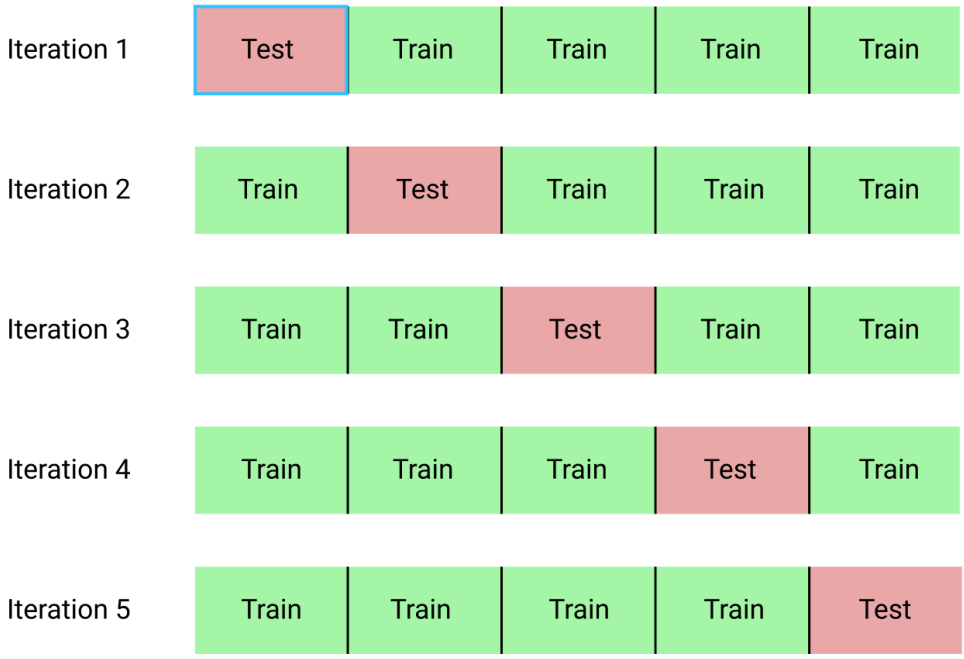
2.9.2 Algoritmo Cross Validation K-Fold

A técnica *K-Fold* do algoritmo *Cross-Validation* trabalha dividindo aleatoriamente o conjunto inicial de dados em *i* subconjuntos, definido pelo número de *folds*. Destes *i* subconjuntos, um é retido para servir como dados de teste (conjunto de teste) e o restante dos *i-1* subconjuntos são usados como dados de treinamento (conjunto de treinamento) (PIMENTA *et al.*, 2009). Este processo é repetido *i* vezes, onde cada subconjunto será utilizado pelo menos uma vez como conjunto de teste.

Na Figura 15, pode-se levar em consideração um cenário de *5-Fold Cross-Validation*, sendo o conjunto de dados dividido em cinco partes. Na primeira iteração a primeira parte será utilizada para teste e as partes restantes serão utilizadas para treinamento do modelo de aprendizado, gerando assim uma métrica de avaliação. Na segunda iteração, a segunda parte será utilizada para teste enquanto as demais para treino. Esse processo é então repetido cinco vezes até que todo o conjunto de dados passe pelo processo de treino e teste, onde ao final será gerada uma métrica de avaliação média para o modelo.

O *Cross-Validation K-Fold* soluciona algumas fraquezas do método *Hold-Out* (TAN *et al.*, 2006). Algumas de suas vantagens são o uso de mais objetos na construção do modelo (primeira fase de classificação), além da possibilidade de utilização de todo o conjunto de dados,

Figura 15 – Representação visual do K-Fold Cross-Validation.



Fonte: Vasconcellos (2018).

de modo alternado e sem repetição, na avaliação (segunda fase de classificação). Entretanto, uma desvantagem desta técnica é o custo computacional gerado ao executar diversas vezes o mesmo algoritmo de classificação.

2.10 Medidas de desempenho

A fim de definir quais métricas devem ser levadas em consideração ao avaliar o desempenho dos modelos de classificação abordados neste trabalho, é necessário ter um entendimento do problema e de como métricas em questão são geradas. No contexto de uma classificação binária, elas se derivam dos conceitos apresentados na matriz de confusão da Figura 16.

Figura 16 – Representação visual de uma Matriz de Confusão.

		Valor Verdadeiro	
		Classe Positiva	Classe Negativa
Valor previsto	Classe Positiva	**VP** Verdadeiro Positivo	**FP** Falso Positivo
	Classe Negativa	**FN** Falso Negativo	**VN** Verdadeiro Negativo

Fonte: Silva (2018).

Uma matriz de confusão, também conhecida como tabela de confusão, trata-se de uma matriz 2x2 que permite verificar o desempenho do método de aprendizagem, exibindo a quantidade de instâncias classificadas correta e erroneamente em cada uma das classes em questão, como pode ser visto na Figura 16 (STEHMAN, 1997).

De acordo com Faceli *et al.* (2011), a avaliação de medidas de desempenho em duas classes pode ser classificada como verdadeiro positivo (VP), que indica a quantidade de documentos que foram classificados como positivos e realmente são da classe de positivos; verdadeiro negativo (VN), que indica a quantidade de documentos que foram classificados como negativos e realmente são da classe de negativos; falso positivo (FP), também conhecido como *Type I Error*, que indica a quantidade de documentos que foram classificados como positivos, mas são da classe de negativos; e por fim, o falso negativo (FN), também conhecido como *Type II Error*, que indica a quantidade de documentos que foram classificados como negativos, mas são da classe dos positivos.

A partir da matriz de confusão pode-se extrair algumas medidas que podem ser utilizadas para avaliação de desempenho do modelo de classificação (FACELI *et al.*, 2011), são elas:

- **Acurácia**: Indica a performance geral do modelo. Dentre todas as classificações, quantas o modelo classificou corretamente, dado pela soma dos valores da diagonal principal da matriz.

$$ac(\hat{f}) = \frac{VP+VN}{VP+VN+FP+FN} \tag{2.10}$$

- **Precisão**: Indica a proporção de exemplos positivos classificados corretamente dentre todas as classificações marcadas como positivos.

$$prec(\hat{f}) = \frac{VP}{VP+FP} \tag{2.11}$$

- **Recall**: Indica a proporção de acerto na classe positiva dentre todas as situações de classe positiva como valor esperado.

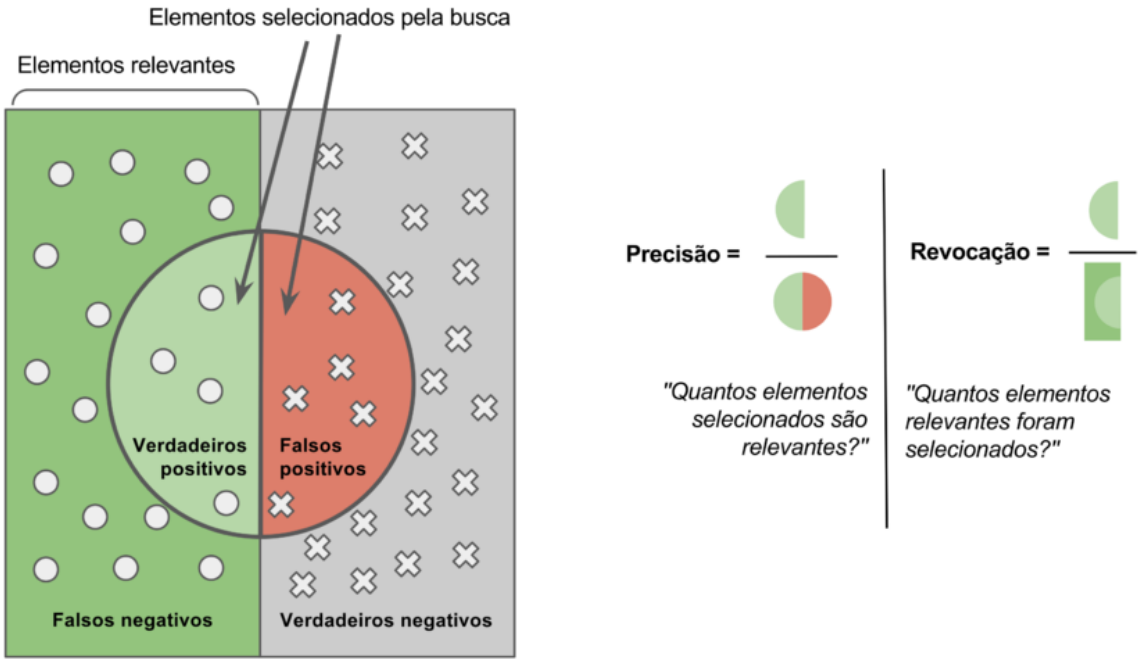
$$recall(\hat{f}) = \frac{VP}{VP+FN} \tag{2.12}$$

- **F1 Score**: Indica a média harmônica entre as medidas de precisão e *recall*.

$$f1(\hat{f}) = \frac{2 * TP}{2 * TP + FP + FN} \tag{2.13}$$

Tanto a precisão quanto o *recall* são consideradas métricas bastante importantes e complementares para medir o desempenho de um classificador. Essas duas medidas estão associadas ao conceito de relevância, sendo que a precisão mede a qualidade ou exatidão do algoritmo, ao passo que a revocação mede sua completude. A aplicação convencional dessas medidas é conhecida na área da tecnologia como recuperação de informação, ou *information retrieval*. Nesse contexto, enquanto a precisão mede a quantidade de objetos recuperados que são relevantes, o *recall* mede a quantidade de objetos relavantes que foram recuperados (FERRARI; SILVA, 2017), (ver Figura 17).

Figura 17 – Métricas de avaliação: Precisão e *Recall*.

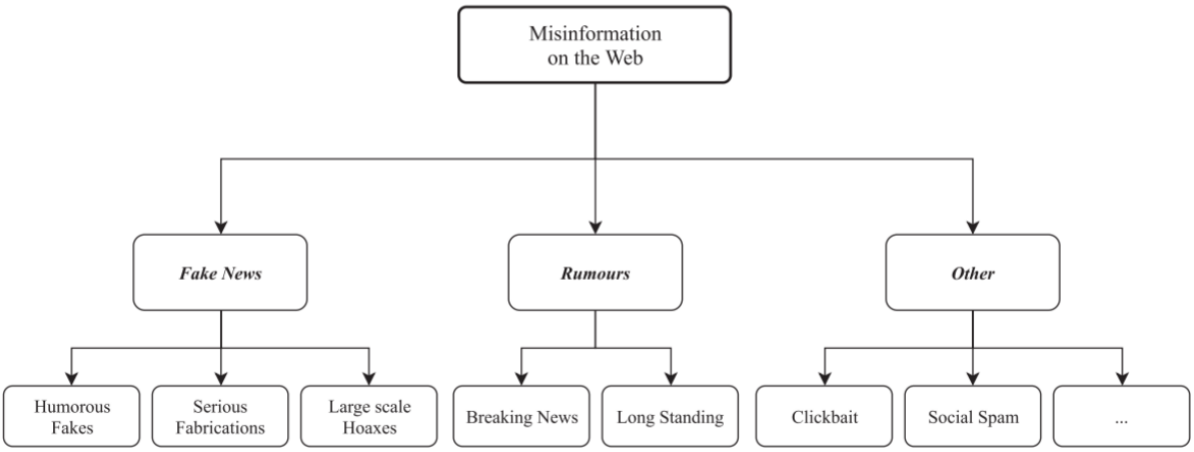


Fonte: Wikipedia (2016).

3 TRABALHOS CORRELATOS

No estudo realizado por Bondielli e Marcelloni (2019), o objetivo principal é a detecção de rumores e notícias falsas. Para isso, os autores verificaram duas abordagens de extração de atributos: baseado no conteúdo e no contexto do texto (ver Figura 18). Estudos que consideram textos publicados em mídias sociais devem utilizar a abordagem baseada em contexto, visto que esta se mostrou mais eficaz na detecção devido ao tamanho curto dos textos. Nos trabalhos que consideram atributos baseados no conteúdo, como é o caso desta pesquisa, são utilizados métodos de análise léxica (n-gramas), sintática (Part of Speech Tagging) e semântica (análise de sentimento).

Figura 18 – Tipos de extração de atributos usados na pesquisa para detecção de *fake news* de Bondielli e Marcelloni (2019).



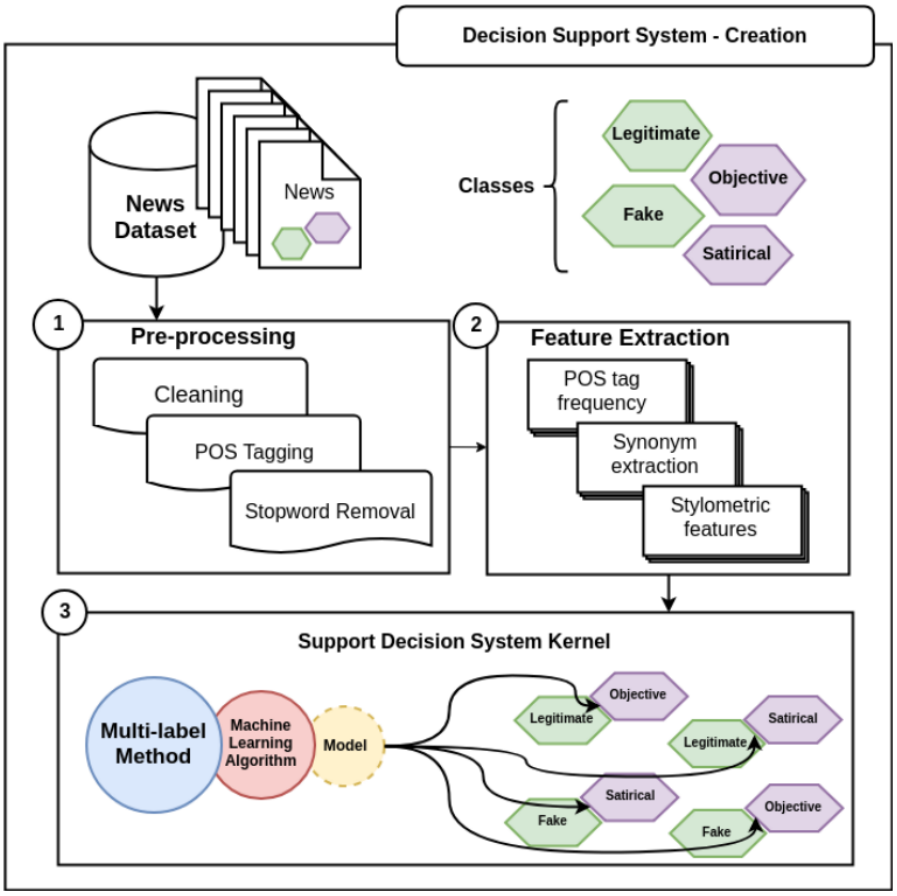
Fonte: Bondielli e Marcelloni (2019).

Na pesquisa desenvolvida por Monteiro *et al.* (2018), foram analisadas manualmente 7.200 notícias e pôde-se identificar que os percentuais de classes gramaticais foram próximos para notícias falsas e verdadeiras. No trabalho, foi calculada a quantidade de erros gramaticais e verificou-se que notícias falsas possuem 10 vezes mais erros que notícias verdadeiras. Outras variáveis também foram analisadas como forma de encontrar padrões nos textos, como por exemplo, pausalidade, incerteza, emotividade e não imediatismo do texto. Além do resultado obtido, o trabalho de Monteiro *et al.* (2018) resultou na criação de um corpus de notícias chamado Fake.Br, que possibilita a identificação de alguns padrões na escrita de *fake news* e que será utilizado nesta pesquisa. Os autores também utilizaram o algoritmo de classificação SVM como meio de categorizar as notícias e obtiveram uma acurácia de 89%.

Morais *et al.* (2019) aborda a detecção de *fake news* com independência de linguagem,

extraindo características em comum das estruturas de diferentes línguas. O esquema do método (ver Figura 19) aplicado no trabalho é constituído, basicamente, em três etapas: 1. fase de pré-processamento da notícia, onde são trabalhados métodos para o tratamento do texto e da informação; 2. fase de computação e extração de características das notícias, onde o intuito é construir uma estrutura que possa ser aplicada em um modelo de aprendizado de máquina; 3. fase de classificação, onde as notícias são classificadas em: verdadeiro, falso ou sátira, consequentemente detectando as *fake news*.

Figura 19 – Representação esquemática do método aplicado na pesquisa de Morais *et al.* (2019).



Fonte: Morais *et al.* (2019).

Os trabalhos apresentados possuem uma influência motivadora e servem de inspiração para a metodologia aplicada neste trabalho. Nesta pesquisa, será adicionada outras variáveis e métricas para validação e identificação de notícias falsas não contempladas nos estudos anteriores, além da utilização de diferentes algoritmos de classificação daqueles utilizados nos trabalhos que foram mencionados. Assim, ratificando conclusões de estudos anteriores e os incrementando com informações relevantes.

4 METODOLOGIA

Quanto aos fins, neste trabalho, a pesquisa é de natureza exploratória; quanto à estratégia, o trabalho constitui-se em um estudo de caso único (YIN, 2015; MARTINS, 2006), cuja coleta de dados ocorreu por meio de pesquisa documental. De acordo com Yin (2015), o estudo de caso "é uma investigação empírica que investiga um fenômeno contemporâneo dentro de seu contexto de vida real".

O principal objetivo deste trabalho é desenvolver um modelo de aprendizado capaz de classificar notícias como verdadeiras ou falsas. Para isto, é utilizado o aprendizado de máquina, uma área que possibilita a realização de uma atividade de maneira independente por uma aplicação. Dentre os tipos de aprendizado de máquina existentes, optou-se pelo aprendizado supervisionado, visto a possibilidade de uso de um corpus de dados desenvolvido por Monteiro *et al.* (2018) que pode ser empregado no treinamento e na validação da aplicação.

Nesta pesquisa, foram analisados dois algoritmos de aprendizado supervisionado utilizados na classificação textual, são eles: o *Multinomial NB* e *Passive Agressive*. Estes algoritmos foram propostos durante o Projeto de Trabalho de Conclusão de Curso (TCC), onde será realizado uma comparativo entre as duas técnicas e, por fim, o que apresentar melhor desempenho, será utilizado no desenvolvimento de uma aplicação que será responsável por detectar *fake news*.

4.1 Frameworks utilizados

4.2 Obtenção do conjunto de dados

Para realizar o treinamento dos algoritmos de aprendizado de máquina utilizados nesta pesquisa, que serão responsáveis pela detecção de *fake news* em notícias, é necessário uma base de dados com notícias reais e notícias falsas para que os métodos preditivos possam assim encontrar padrões e regularidades que irão servir de base no processo de classificação do aprendizado supervisionado. Neste trabalho, foi utilizado o "Fake.Br Corpus", que trata-se de um conjunto de dados de notícias verdadeiras e falsas criado por Monteiro *et al.* (2018). A Tabela 2 apresenta trechos de notícias coletadas e já classificadas como reais ou falsas. É importante notar que a notícia verdadeira não necessariamente desmente a notícia falsa, como apresentado nos exemplos da Tabela 2.

Tabela 2 – Exemplos de notícias verdadeiras e falsas coletadas.

Notícia Real	Notícia Falsa
Michel Temer não quer o fim do Carnaval por 20 anos. Notícias falsas misturam proximidade dos festejos, crise econômica e medidas impopulares do governo do peemedebista.	Michel Temer propõe fim do carnaval por 20 anos, “PEC dos gastos”. Michel Temer afirmou que não deve haver gastos com aparatos supérfluos sem pensar primeiramente na educação do Brasil. A medida pretende cancelar o carnaval de 2018.
Ingresso feminino barato como marketing ‘não inferioriza mulher’, diz juíza do DF. Afirmação consta em decisão sobre preços diferentes para homens e mulheres em festa no Lago Paranoá. ‘Prática permite que mulher possa optar por participar de tais eventos sociais’, diz texto.	Acabou a mordomia ! Ingresso mais barato pra mulher é ilegal. Baladas que davam meia entrada para mulher, ou até mesmo gratuidade, estão na ilegalidade agora. Acabou o preconceito com os homens nas casas de show de todo o Brasil.

Fonte: Monteiro *et al.* (2018).

O conjunto de dados utilizado para treinamento possui 3.600 notícias reais e 3.600 notícias falsas. Todas elas foram alinhadas em duas condições: por assunto e por tamanho, na busca por evitar o viés sobre os dados. Todos os detalhes utilizados para a obtenção dos conjuntos de dados podem ser encontrados e analisados no trabalho "Contributions to the Study os Fake News in Portuguese: New Corpus and Automatic Detection Results", (MONTEIRO *et al.*, 2018).

Desta maneira, o conjunto de dados obtidos neste trabalho acabou dividido em 6 categorias de notícias: Política, TV e Celebridades, Sociedade e Notícias Diárias, Ciência e Tecnologia, Economia e, por último, Religião. A distribuição de textos por categoria pode ser vista na Tabela 3.

Tabela 3 – Distribuição de textos por categoria no Fake.Br Corpus.

Categoria	Quantidade	Porcentagem
Política	4180	58,0
TV e Celebridades	1544	21,4
Sociedade e Notícias Diárias	1276	17,7
Ciência e Tecnologia	112	1,5
Ecnomia	44	0,7
Religião	44	0,7
Total	7200	100

Fonte: Monteiro *et al.* (2018).

Neste trabalho, no entanto, foi necessário realizar algumas alterações no conjunto de dados obtidos. Essa ação foi tomada pois notou-se que haviam, embora representando número

pequeno, algumas notícias repetidas. Sendo assim, elas foram removidas e outras foram alteradas para manter o alinhamento do conjunto de dados.

Além disso, com o objetivo de evitar viés na classificação devido à variação de tamanho entre as notícias verdadeiras e falsas, os experimentos finais foram realizados em cima de uma base de dados igualmente dividida nestas duas categorias. Sendo assim, cada par de notícias foi truncado no mesmo número de palavras da menor notícia, como pode ser visto nas Tabelas 4 e 5.

Tabela 4 – Notícias originais, sem truncamento.

Notícia Falsa	Notícia Verdadeira
Janaína, a mulher que representa os brasileiros honestos, denuncia: "Querem acabar com o Sérgio Moro". O PT é uma vergonha para a política. Não há motivos para duvidar da advogada e professora de direito Janaína Paschoal. É só lembrar as declarações divulgadas pelos próprios petistas. Eles disseram: "O partido é o inimigo número 1 da Operação Lava Jato "Abaixo um vídeo onde Janaína Paschoal, na reunião na Comissão do Impeachment, lembra claramente das várias ações impetradas por parlamentares do PT e do PCdoB contra o juiz Sérgio Moro. O que Janaína demonstra claramente é a famosa "tática de assédio jurídico"	Alvo da Lava Jato, Renan sofre resistência no PMDB para liderar a bancada. Senador, que vai deixar presidência do Senado em fevereiro, tem o nome cotado para assumir a liderança do partido ou a Comissão de Constituição e Justiça. O futuro do presidente do Senado, Renan Calheiros (PMDB-AL), após o fim de seu mandato à frente da Casa, em fevereiro, ainda está indefinido. A probabilidade maior é a de que Renan ocupe a liderança do PMDB na Casa – função que já desempenhou outras duas vezes – mas dentro da bancada há resistência ao nome dele, principalmente por conta dos processos do senador na Justiça e os inquéritos de que é alvo na Lava Jato. Peemedebistas ouvidos pelo G1 dizem que o fato de ele ser investigado pode desestabilizar e constranger o partido caso assuma a liderança da bancada. Em dezembro, o STF tornou Renan réu pelo crime de peculato [...]

Fonte: Monteiro *et al.* (2018).

Tabela 5 – Notícias após o truncamento.

Notícia Falsa	Notícia Verdadeira
Janaína, a mulher que representa os brasileiros honestos, denuncia: "Querem acabar com o Sérgio Moro". O PT é uma vergonha para a política. Não há motivos para duvidar da advogada e professora de direito Janaína Paschoal. É só lembrar as	Alvo da Lava Jato, Renan sofre resistência no PMDB para liderar a bancada. Senador, que vai deixar presidência do Senado em fevereiro, tem o nome cotado para assumir a liderança do partido ou a Comissão de Constituição e Justiça. O futuro do

declarações divulgadas pelos próprios petistas. Eles disseram: "O partido é o inimigo número 1 da Operação Lava Jato "Abaixo um vídeo onde Janaína Paschoal, na reunião na Comissão do Impeachment, lembra claramente das várias ações	presidente do Senado, Renan Calheiros (PMDB-AL), após o fim de seu mandato à frente da Casa, em fevereiro, ainda está indefinido. A probabilidade maior é a de que Renan ocupe a liderança do PMDB na Casa – função que já desempenhou outras duas

Fonte: Monteiro *et al.* (2018).

4.3 Pré-Processamento

Para desenvolver um classificador automático utilizando aprendizado de máquina é necessário que atributos textuais das notícias, do conjunto de dados de treinamento, sejam extraídos e representados em um formato numérico (MONTEIRO *et al.*, 2018). Neste trabalho, com intuito de criar um vetor de características de cada notícia, foi utilizada a técnica *Term Frequency - Inverse Document Frequency* (TF-IDF). Essa técnica consegue avaliar a importância de uma palavra contida em um texto e sua relevância em todo o conjunto de treino. A sua utilização, nesta fase da pesquisa, é medir o quão importante uma palavra é no contexto de todas as notícias do conjunto de dados, onde quanto mais recorrente for a palavra, menor é sua importância.

Considerando que, neste trabalho, as *stopwords*[1] não estão sendo removidas, palavras como "de", "para", "foi", que ocorrem bastante nas notícias coletadas da base de dados, podem vir a ter uma importância maior do que os termos que ocorrem raramente. Por este motivo, a utilização do TF-IDF surge como alternativa para calcular a frequência das palavras e também definir seu grau de importância no conjunto de dados de treinamento.

Desta forma, para se extrair os atributos das notícias e construir o vetor de treinamento que será utilizado, nas próximos etapas do aprendizado, pelos algoritmos de classificação, foram utilizadas as bibliotecas Pandas e Scikit-Learn do Python 3.6.

O primeiro passo foi a utilização da biblioteca Pandas para a leitura do arquivo "news.csv", que contém todas as notícias que foram coletadas e que servirá como conjunto de dados de treinamento para o algoritmo de aprendizado, que fará a detecção de *fake news*. Depois disso, para uma melhor organização, foi necessário separar a coluna de saídas (*fake* e *true*) em

[1] Palavras consideradas irrelevantes ao texto, pois não tem sentido próprio quando consideradas sozinhas. Como exemplos temos os artigos e as preposições (OLIVEIRA, 2019).

uma lista chamada *"news_type"* e remover a coluna *"type"* do *dataframe*[2] *news*, formando assim, uma base de dados apenas com as notícias a serem treinadas (ver Figura 20).

Figura 20 – Leitura do arquivo .csv das notícias.

```
news = pd.read_csv("news-list.csv", sep=";")
news_type = news.type
news = news.drop("type", axis=1)
```

Fonte: Próprio Autor.

O segundo passo foi configurar os conjuntos de dados de treinamento e de teste por meio do modelo *train_test_split* da biblioteca Scikit-Learn (ver Figura 21). No código, a variável *news["data"]* é passada como primeiro parâmetro da função, já que a divisão será feita em cima desse conjunto de dados; a coluna *"news_type"* é a lista que foi criada na fase anterior e o que se deseja classificar, agora associada a variável *"type_train"*; a variável *"test_size"*, atribuída ao valor 0.30, significa que 30% dos dados será dividido em um conjunto de testes. Por fim, os dados de treino e de teste são armazenados nas variáveis *news_train* e *news_test*, respectivamente.

Figura 21 – Definição dos conjuntos de dados de treino e teste.

```
news_train, news_test, type_train, type_test = train_test_split(
  news['data'], news_type, test_size = 0.30)
```

Fonte: Próprio Autor.

No último passo, foi criado um vetorizador atribuído a variável *"train"* (ver Figura 22), com a representação numérica de ocorrência das palavras do conjunto de treinamento. Para isso, foi utilizado o método TfidfVectorizer da biblioteca Scikit-Learn. O parâmetro *lowercase* foi passado com o intuito de transformar todas as palavras em minúsculas e assim limitar as suas ocorrências, já o parâmetro *"max_df"* foi utilizado com o objetivo de remover palavras que apareçam em mais de 70% das notícias, isso porque a ocorrências dessas palavras pode influenciar diretamente no modelo de classificação, gerando o que pode-se chamar de viés.

Por fim, um exemplo de como ficaria a matriz de atributos, que será utilizada como base de treinamento na fase de aprendizado dos algoritmos de classificação, pode ser visto na Tabela 6 onde os dados da matriz foram normalizados.

[2] Estrutura bidimensional de dados, como uma planilha.

Figura 22 – Criação dos vetores de classificação.

```
tfidf_vectorizer = TfidfVectorizer(max_df=0.7)
tfidf_train = tfidf_vectorizer.fit_transform(news_train)
tfidf_test = tfidf_vectorizer.transform(news_test)
train = tfidf_train.toarray()
```

Fonte: Próprio Autor.

Tabela 6 – Exemplo da matriz de atributos gerada com TF-IDF.

"expuls"	**"kat"**	**"abr"**	**"pmdb"**	**"legend"**	...	**Classificação**
1	1	0,815625	0,76111111	0,525	...	Fake
0,38095238	0,15873016	0,51785714	0,36243386	1	...	Real
0,09756098	0	0	0	0	...	Fake
0	0	0	0	0	...	Real

Fonte: (OLIVEIRA, 2019)

4.4 Treinamento

Feita a montagem da matriz de atributos, gerada na fase de pré processamento, inicia-se a fase de treinamento. Nesta etapa foram utilizados os classificadores *MultinomialNB* e o *Passive Agressive* da biblioteca *Scikit-Learn* para Python 3.6.

Os experimentos foram feitos utilizando dois algoritmos de classificação, são eles: o *Multinomial NB* (ver Figura 23) e o *Passive Agressive* (ver Figura 24). O método *fit()*, utilizado por ambos os classificadores, é o responsável por aprender e descobrir padrões entre os dados previsores e categóricos, isto é, ele possui a função de treinar os modelos de predição. O método funciona com dois parâmetros, em que *"train"* é a matriz de atributos (o conjunto de observações a ser aprendido) e *"type_train"* é o resultado pretendido, ou seja, as saídas *fake* e *true*. Aplicando o método *fit()*, a informação na matriz de atributos é relacionada às saídas da variável *"type_train"*, para que, ao receber uma nova informação com as mesmas características já conhecidas na matriz, seja possível predizer corretamente a saída da mesma.

Figura 23 – Treinamento utilizando o algoritmo *Multinomial NB*.

```
model_nb = MultinomialNB()
model_nb.fit(train, type_train)
```

Fonte: Próprio Autor.

Figura 24 – Treinamento utilizando o algoritmo *Passive Agressive*.

```
model_linear = PassiveAggressiveClassifier(max_iter=1000)
model_linear.fit(train, type_train)
```

Fonte: Próprio Autor.

Feito esse processo, foi feita a predição em cima do conjunto de testes gerado também na fase de pré processamento. Na próxima etapa, cada algoritmo será avaliado de acordo com diferentes métricas em cada uma de suas execuções, são elas: acurácia, precisão, *recall* e *F1 score*.

5 RESULTADOS

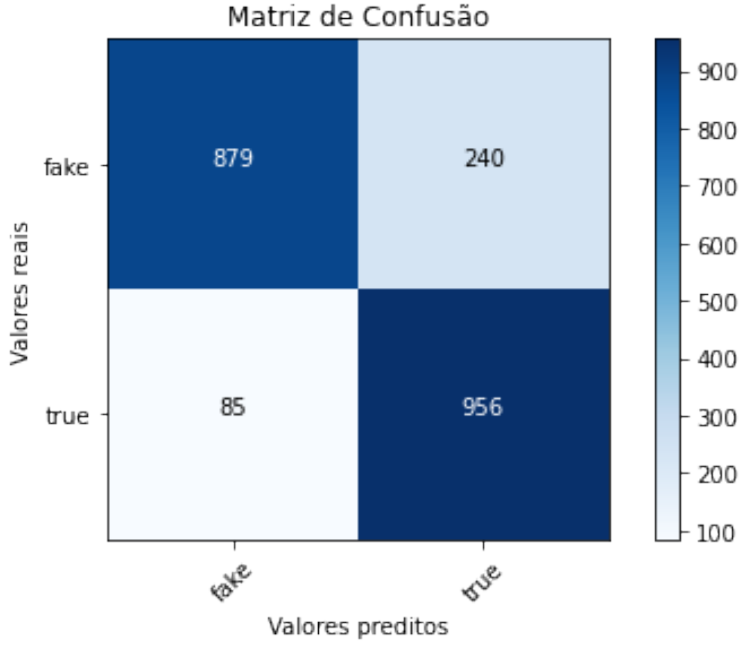
5.1 Resultados do classificador *MultinomialNB*

Este capítulo tem o objetivo de avaliar a classificação automática das notícias como verdadeiras ou falsas. Os resultados são obtidos com a execução dos classificadores *Multinomial NB* e *Passive Agressive* associados às técnicas *Cross-Validation* e *Hold-Out*. Para efeitos de comparação, serão utilizadas as métricas Precisão, *Recall* e *F1 Score*.

5.1.1 MultinomialNB associado à técnica Hold-Out

Com 30% do conjunto de dados (2160 notícias selecionadas, randomicamente, na fase de pré-processamento) sendo utilizado para avaliação do modelo de treinamento, neste primeiro experimento aplicando o algoritmo de classificação *Multinomial NB* no processo de aprendizado de máquina, obteve-se o seguinte resultado na matriz de confusão:

Figura 25 – Resultado da matriz de confusão do algoritmo *Multinomial NB*.



Fonte: Próprio Autor.

Na Figura 25 é representado o resultado da classificação do algoritmo *Multinomial NB* em uma matriz de confusão. Na horizontal têm-se os valores preditos por meio do classificador e na vertical têm-se os valores reais do conjunto de dados. O classificador obteve um total de 879 acertos ao prever a classe *fake* para notícias realmente falsas e um total de 956 acertos ao

prever a classe *true* para notícias de fato verdadeiras. Com estes resultados obtidos através do processo de classificação, foi possível calcular algumas métricas de avaliação de desempenho do classificador, como pode ser visto na Tabela 7.

Tabela 7 – Resultado do algoritmo *Multinomial NB* utilizando técnica *Hold-Out*.

	Precisão	Recall	F1 Score
Fake	0.9118	0.7855	0.8440
True	0.7993	0.9183	0.8547
Média	0.8555	0.8519	0.8493
Acurácia	0.8495		

Fonte: Próprio Autor.

O algoritmo de classificação *Multinomial NB* apresentou uma acurácia de 84% para predições corretas de todas as previsões possíveis. A métrica Precisão, usada para avaliar quão bem o modelo de classificação trabalhou, obteve 91% e 79% de predições realizadas corretamente para as classes *fake* e *true*, respectivamente, resultando em uma média de 85%. A métrica *Recall*, usada para avaliar o desempenho do classificador para prever positivos, ou seja, a classe que se quer prever, conseguiu um resultado de 78% para a classe *fake* e 91% para a *true*, resultando também em uma média de 85%. Por fim, a métrica *F1 Score*, responsável por calcular a média das métricas Precisão e *Recall*, teve apenas uma leve diferença para os resultados preditos para as classes *fake* e *true*, obtendo o resultado de 84% para predições corretas de classe *fake* e 85% para predições corretas de classe *true*, resultando em uma média final de 84%.

5.1.2 MultinomialNB associado à técnica Cross-Validation K-Fold

Na técnica *Cross Validation K-Fold*, o conjunto de dados foi dividido aleatoriamente em 5 *folds*, ou partições ($S_1, \ldots, S_5$), disjuntos, sendo estes aproximadamente do mesmo tamanho. Após a divisão do conjunto de dados, foram executadas cinco iterações de indução e teste, uma para cada *fold*, utilizando o classificador *Multinomial NB*. Na primeira iteração, foi induzido o classificador com os conjuntos de dados $S_2, \ldots S_5$, sendo este depois testado com o conjunto S_1. Na segunda iteração, foram utilizados os conjuntos $S_1, S_3, \ldots, S_5$, sendo o classificador depois testado com o conjunto S_2. Na terceira iteração, foram utilizados os conjuntos $S_1, S_2, S4, S_5$, sendo o classificador depois testado com o conjunto S_3. Na quarta iteração, foram utilizados os conjuntos S_1, S_2, S_3, S_5, sendo o classificador depois testado com o conjunto S_4. Por fim, na quinta e última iteração, foram utilizados os conjuntos S_1, S_2, S_3, S_4, sendo o classificador depois

testado com o conjunto S_5.

Depois de dividir o conjunto de dados em cinco subconjuntos e testá-los, os resultados obtidos, em cada execução das partições, podem ser vistos na Tabela 8. Para cada uma das cinco partições foram avaliadas quatro métricas de desempenho, são elas: Acurácia, Precisão, *Recall* e *F1 Score*. Por fim, o algoritmo de classificação *Multinomial NB*, associado à técnica *cross validation k-fold*, apresentou uma acurácia de 86% para predições corretas, valor este obtido por meio da média da acurácia de cada uma das partições.

Tabela 8 – Resultado do algoritmo *Multinomial NB* utilizando técnica *Cross-Validation K-Fold*.

	Acurácia	Precisão	Recall	F1 Score
k-fold 1	0.8715	0.8729	0.8715	0.8714
k-fold 2	0.8520	0.8553	0.8520	0.8517
k-fold 3	0.8743	0.8771	0.8743	0.8740
k-fold 4	0.8784	0.8798	0.8784	0.8783
k-fold 5	0.8583	0.8610	0.8583	0.8580
Média	0.8669	0.8692	0.8669	0.8667

Fonte: Próprio Autor.

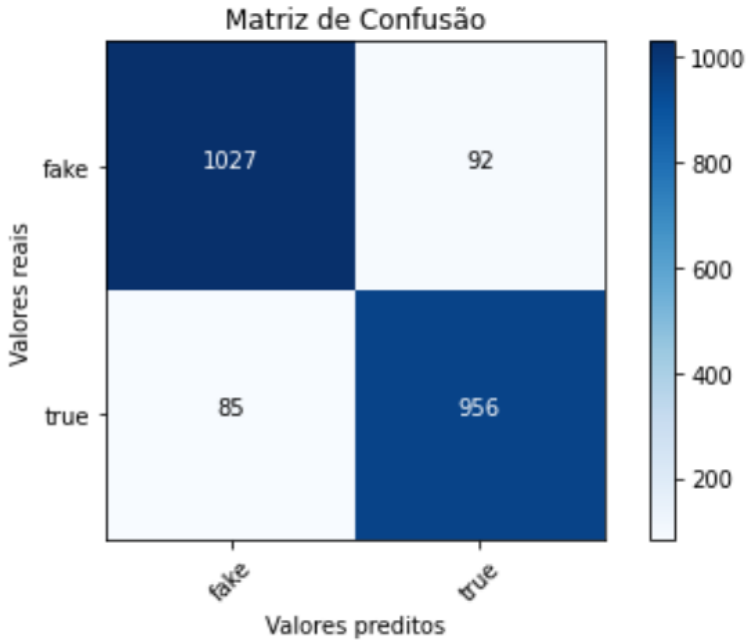
5.2 Resultados do classificador *Passive Agressive*

5.2.1 Passive Agressive associado à técnica Hold-Out

O experimento realizado aplicando o algoritmo de classificação *Passive Agressive* utilizou a mesma amostragem de 30% do conjunto de dados (2160 notícias selecionadas, randomicamente, na fase de pré-processamento) utilizada na avaliação do algoritmo *Multinomial NB*. O resultado da matriz de confusão deste experimento pode ser visto na Figura 26.

Na Figura 26 é representado o resultado da classificação do algoritmo *Passive Agressive* em uma matriz de confusão. Na horizontal têm-se os valores preditos por meio do classificador, e na vertical têm-se os valores reais do conjunto de dados. O classificador obteve um total de 1027 acertos ao prever a classe *fake* para notícias realmente falsas e um total de 954 acertos ao prever a classe *true* para notícias de fato verdadeiras. Com estes resultados obtidos através do processo de classificação, foi possível calcular algumas métricas de avaliação de desempenho do classificador, como pode ser visto na Tabela 9.

Figura 26 – Resultado da matriz de confusão do algoritmo *Passive Agressive*.



Fonte: Próprio Autor.

Tabela 9 – Resultado do algoritmo *Passive Agressive* utilizando técnica *Hold-Out*.

	Precisão	Recall	F1 Score
Fake	0.9236	0.9178	0.9207
True	0.9122	0.9183	0.9153
Média	0.9179	0.9181	0.9180
Acurácia	0.9181		

Fonte: Próprio Autor.

O algoritmo de classificação *Passive Agressive* apresentou uma acurácia de 91% para predições corretas de todas as previsões possíveis. A métrica Precisão, usada para avaliar quão bem o modelo de classificação trabalhou, obteve 92% e 90% de predições realizadas corretamente para as classes *fake* e *true*, respectivamente, resultando em uma média de 91%. A métrica *Recall*, usada para avaliar o desempenho do classificador para prever positivos, ou seja, a classe que se quer prever, atingiu um resultado de 91% para ambas as classes *fake* e *true*, resultando também em uma média de 91%. Por fim, a métrica *F1 Score*, responsável por calcular a média das métricas Precisão e *Recall*, obteve o resultado de 92% para predições corretas de classe *fake* e 91% para predições corretas de classe *true*, resultando em uma média final de 91%.

5.2.2 Passive Agressive associado à técnica Cross-Validation K-Fold

Na técnica *Cross Validation K-Fold*, o conjunto de dados foi dividido aleatoriamente em 5 *folds*, ou partições ($S_1, \ldots, S_5$), disjuntos, sendo estes aproximadamente do mesmo tamanho. Após a divisão do conjunto de dados, foram executadas cinco iterações de indução e teste, uma para cada *fold*, utilizando o classificador *Passive Agressive*. Na primeira iteração, foi induzido o classificador com os conjuntos de dados $S_2, \ldots S_5$, sendo este depois testado com o conjunto S_1. Na segunda iteração, foram utilizados os conjuntos $S_1, S_3, \ldots, S_5$, sendo o classificador depois testado com o conjunto S_2. Na terceira iteração, foram utilizados os conjuntos $S_1, S_2, S4, S_5$, sendo o classificador depois testado com o conjunto S_3. Na quarta iteração, foram utilizados os conjuntos S_1, S_2, S_3, S_5, sendo o classificador depois testado com o conjunto S_4. Por fim, na quinta e última iteração, foram utilizados os conjuntos S_1, S_2, S_3, S_4, sendo o classificador depois testado com o conjunto S_5.

Depois de dividir o conjunto de dados em cinco subconjuntos e testá-los, os resultados obtidos, em cada execução das partições, podem ser vistos na Tabela 10. Para cada uma das cinco partições foram avaliadas quatro métricas de desempenho, são elas: Acurácia, Precisão, *Recall* e *F1 Score*. Por fim, o algoritmo de classificação *Passive Agressive*, associado à técnica *cross validation k-fold*, apresentou uma acurácia de 92% para predições corretas, valor este obtido por meio da média da acurácia de cada uma das partições.

Tabela 10 – Resultado do algoritmo *Passive Agressive* utilizando técnica *Cross-Validation K-Fold*.

	Acurácia	Precisão	Recall	F1 Score
k-fold 1	0.9347	0.9284	0.9312	0.9333
k-fold 2	0.9159	0.9161	0.9125	0.9145
k-fold 3	0.9243	0.9215	0.9187	0.9229
k-fold 4	0.9305	0.9270	0.9284	0.9291
k-fold 5	0.9090	0.9063	0.9090	0.9090
Média	0.9229	0.9199	0.9199	0.9218

Fonte: Próprio Autor.

5.3 Comparação entre os classificadores *Multinomial NB* e *Passive Agressive*

Na Tabela 11 pode-se ver o comparativo entre os dois modelos de classificação, *Multinomial NB* e *Passive Agressive*, baseado nos resultados obtidos através da técnica de validação *Hold-Out* e utilizando as quatro métricas adotadas neste projeto para avaliação de

desempenho destes classificadores.

Tabela 11 – Resultados obtidos dos algoritmos *Multinomial NB* e *Passive Agressive* utilizando a técnica *Hold-Out*.

Algoritmo	Acurácia	Precisão	Recall	F1 Score
Multinomial NB	0.8495	0.8555	0.8519	0.8493
Passive Agressive	**0.9181**	**0.9179**	**0.9181**	**0.9180**

Fonte: Próprio Autor.

Através desses resultados, é possível identificar que o modelo *Passive Agressive* se destacou, quando comparado ao classificador *Multinomial NB*, por sua precisão ao identificar as classes das notícias, obtendo um resultado de 91%. Isso significa que, ao classificar uma notícia como *fake* ou *true*, este modelo é relativamente confiável. É possível notar também que o *Passive Agressive* teve desempenho superior em todas as outras métricas avaliadas, demonstrando que conseguiu definir bem um hiperplano separador dos dados, atingindo uma acurácia também de 91%, que determina o quão frequente o classificador está correto.

Na Tabela 12 o comparativo entre os dois modelos de classificação é feito através da técnica *Cross Validation K-Fold*, além da utilização das mesmas quatro métricas adotadas na abordagem anterior para avaliação de desempenho destes classificadores.

Tabela 12 – Resultados obtidos dos algoritmos *Multinomial NB* e *Passive Agressive* utilizando a técnica *Cross Validation K-Fold*.

Algoritmo	Acurácia	Precisão	Recall	F1 Score
Multinomial NB	0.8669	0.8692	0.8669	0.8667
Passive Agressive	**0.9229**	**0.9199**	**0.9199**	**0.9218**

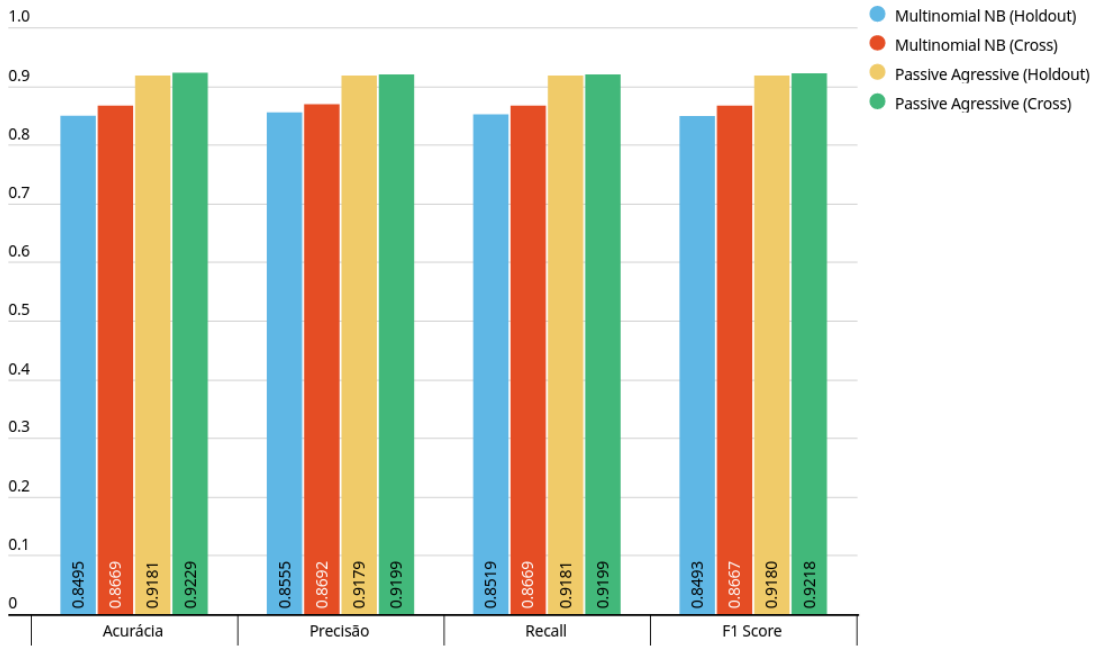
Fonte: Próprio Autor.

Por meio desses resultados, é possível perceber que mesmo utilizando uma técnica de avaliação diferente para se testar o desempenho dos classificadores, o modelo *Passive Agressive* ainda acaba se destacando, quando comparado ao classificador *Multinomial NB*. Na Tabela 12 pode-se perceber que o algoritmo *Passive Agressive* obteve uma acurácia de 92% contra 86% do algoritmo *Multinomial NB*, se mostrando também superior em todas as outras três métricas avaliadas.

Apesar da técnica *Hold-Out* ser uma boa estratégia de avaliação de desempenho, esse processo pode criar conjuntos de dados desbalanceados, isto porque ao fazer a divisão dos conjuntos de treino e teste, os dados selecionados de forma aleatória podem vir a ser bastante semelhantes, o que pode acabar enviesando o modelo de classificação e prejudicando a detecção

de novos dados. Por este motivo, neste trabalho, foi adotada também a técnica *Cross-Validation K-Fold*, que tem como principal objetivo evitar problemas de aleatoriedade ao permitir testar o modelo classificador com todos os dados disponíveis no conjunto. Na Figura 27 é possível observar, de forma gráfica, os resultados obtidos entre os métodos de classificação propostos utilizando as duas técnicas de avaliação de desempenho: *Hold-Out* e *Cross-Validation K-Fold*.

Figura 27 – Comparativo entre os resultados obtidos dos algoritmos *Multinomial NB* e *Passive Agressive*.



Fonte: Próprio Autor.

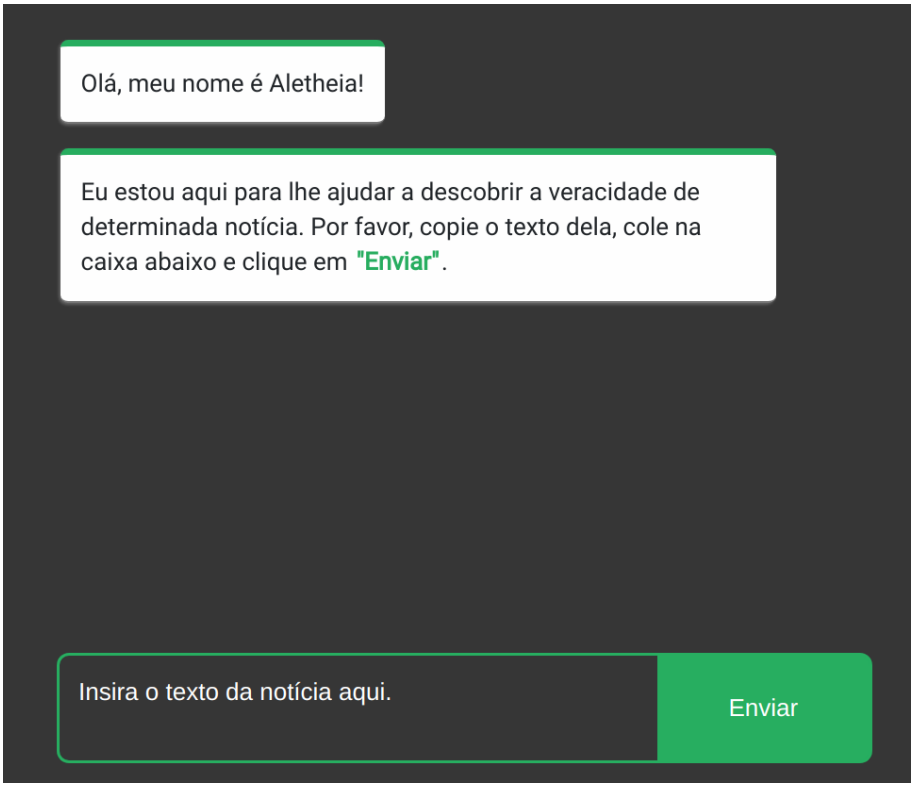
Como a Figura 27 apresenta, não houve diferença significativa nas duas técnicas de avaliação aplicadas nos modelos de classificação, isso porque a técnica *Hold-Out* foi empregada utilizando um conjunto de dados grande o suficiente para conseguir treinar os modelos com precisão, mesmo com 30% de dados a menos destinados para testes, diminuindo assim o problema da aleatoriedade e gerando um modelo bastante confiável. Por este motivo, a técnica *Cross Validation K-Fold* não mostrou resultados tão superiores à técnica *Hold-Out*, visto que praticamente não existe diferença entre as formas de dividir o conjunto de dados utilizado por ambas as técnicas.

5.4 Chatbot

Definido o modelo de classificação cujo apresentado melhor desempenho na fase anterior, sendo este o *Passive Agressive*, foi desenvolvido um *chatbot* com o intuito de aplicar o conhecimento adquirido neste trabalho em uma aplicação real capaz de checar a veracidade de uma notícia. Para isso, a aplicação processa o texto recebido pelo usuário a fim de identificar atributos de escrita, como por exemplo, palavras usadas ou classes gramaticais mais frequentes, e utiliza essas características no modelo de aprendizado de máquina desenvolvido neste projeto que classificará automaticamente a notícia em verdadeira ou falsa.

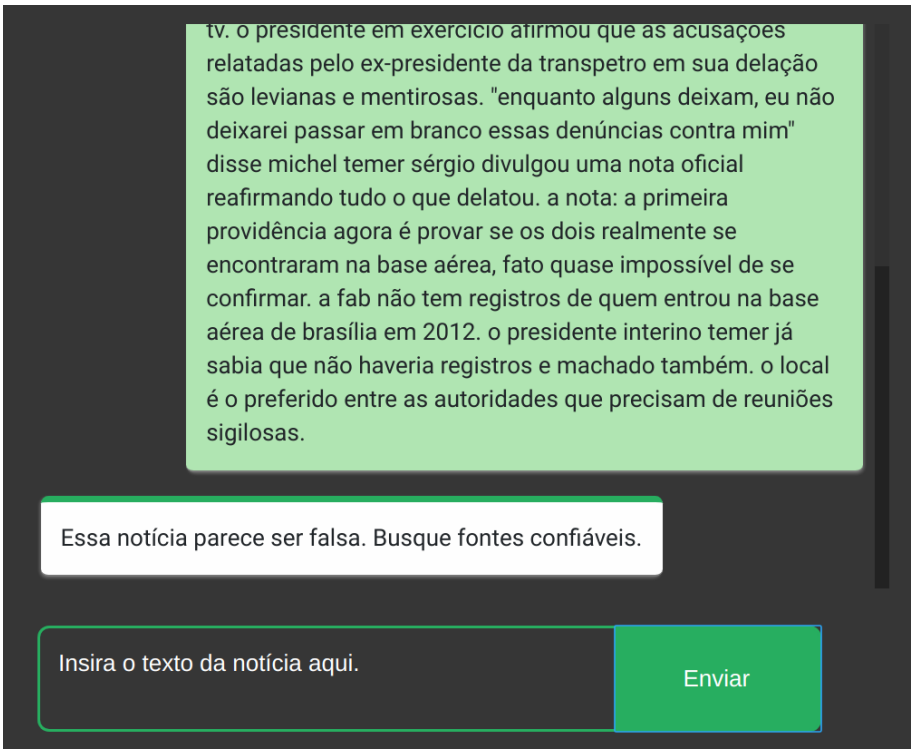
Para usar a ferramenta é necessário somente que o usuário insira o conteúdo da notícia em uma caixa de texto, como pode ser visto nas Figuras 28 e 29, que mostra uma versão inicial da aplicação. O *chatbot* foi desenvolvido utilizando a biblioteca React JS, que tem como o foco a criação de interface de usuário em páginas web. Já a API, utilizada pela interface web para a detecção de *fake news*, foi desenvolvida utilizando a linguagem Python. O framework foi escolhido por ser um grande facilitador na integração entre os classificadores da biblioteca *Scikit-Learn* em uma aplicação web. O sistema está disponível publicamente para acesso na url <https://fakenews-detect.appspot.com/>.

Figura 28 – Interface web desenvolvida para detecção de *fake news*.



Fonte: Próprio Autor.

Figura 29 – Exemplo de resposta dada ao usuário após submeter uma notícia.



Fonte: Próprio Autor.

6 CONCLUSÕES E TRABALHOS FUTUROS

Devido a preocupação com o tema *fake news* no cenário atual da nossa sociedade, o presente projeto propôs a detecção de notícias falsas, em meio a notícias verdadeiras, através do uso de algoritmos de inteligência artificial. O objetivo do trabalho foi desenvolver um modelo de aprendizado que classificasse de forma automática uma notícia como verdadeira ou falsa. Foram estes os algoritmos utilizados no processo de desenvolvimento do modelo de classificação: *Multinomial NB* e *Passive Agressive*.

Os dois algoritmos estudados e explorados neste projeto foram escolhidos com o intuito de se estabelecer um comparativo entre seus resultados a fim de empregar a melhor técnica no processo de detecção de *fake news*. Além disso, na busca pelos melhores resultados e para estimar a taxa de acerto dos modelos preditivos, cada algoritmo classificador foi testado utilizando duas técnicas de validação de dados, *Hold-Out* e *Cross-Validation*.

Os resultados obtidos mostram que o algoritmo classificador *Passive Agressive* se sobressaiu obtendo uma acurácia de $91,81\%$ ao utilizar a técnica *Hold-Out* e $92,29\%$ ao empregar a técnica *Cross-Validation*, enquanto o classificador *Multinomial NB* obteve uma acurácia abaixo dos 90%, atingindo os valores de $84,95\%$ e $86,69\%$ ao utilizar as técnicas *Hold-Out* e *Cross-Validation*, respectivamente. Apesar do classificador *Passive Agressive* ter apresentado um desempenho melhor quando comparado ao *Multinomial NB*, ambos os algoritmos explorados nesta pesquisa obtiveram resultados satisfatórios na tarefa de classificação de notícias, alcançando valores acima de 80% nas principais métricas utilizadas para avaliação de desempenho do modelo de classificação.

O presente trabalho também se propôs a desenvolver um *chatbot* para detecção automática de notícias, utilizando o classificador cujo apresentado melhor desempenho, sendo este o *Passive Agressive*. O intuito na disponibilização dessa ferramenta é mostrar como a tecnologia pode vir a colaborar na verificação de *fake news* e no combate à desinformação, servindo de apoio para o trabalho de diversos profissionais do meio jornalístico, visto que um algoritmo pode trabalhar a detecção de notícias falsas de forma rápida, além de facilitar o processo de checagem de notícias para diversas pessoas que possam ter acesso à ferramenta desenvolvida nesta pesquisa.

Como trabalho futuro, pretende-se estender esta pesquisa por meio da análise de outros algoritmos de aprendizado de máquina, visto que o desenvolvimento de estudos e experimentações contínuas contribui diretamente para a otimização e construção de um modelo

preditivo mais inteligente e que seja capaz de detectar *fake news* com um maior nível de precisão. Com relação à base de dados aplicada no treinamento do modelo de predição, um trabalho futuro necessário é a replicação deste modelo em conjuntos de dados diferentes e desbalanceados a fim de aprimorar os resultados do classificador desenvolvido, visto que assim haverá uma maior variação nos dados que serão utilizados para treinar o algoritmo.

REFERÊNCIAS

AHO, A. V. **Compilers: principles, techniques and tools (for Anna University), 2/e**. [S.l.]: Pearson Education India, 2003.

BARBOSA, J.; VIEIRA, J.; SANTOS, R.; JUNIOR, G. M.; MUNIZ, M.; MOURA, R. Introdução ao processamento de linguagem natural usando python. **III Escola Regional de Informatica do Piauí**, v. 1, p. 336–360, 2017.

BARBOSA, P.; O'REILLY, A. Harvard trends: Tendências de gestão. **Porto: Vida Económica**, 2011.

BARRETO, C. A. d. S. **Uso de técnicas de aprendizado de máquina para identificação de perfis de uso de automóveis baseado em dados automotivos**. Dissertação (Mestrado) — Brasil, 2018.

BATISTA, G.; MONARD, M. C. Sniffer: um ambiente computacional para gerenciamento de expe-rimentos de aprendizado de máquina supervisionado. **Proceedings of the I WorkComp Sul**, 2004.

BAUDRILLARD, J. **Tela total: mito-ironias da era do virtual e da imagem**. [S.l.]: Ed. Sulina, 1999.

BONDIELLI, A.; MARCELLONI, F. A survey on fake news and rumour detection techniques. **Information Sciences**, Elsevier, v. 497, p. 38–55, 2019.

BURFOOT, C.; BALDWIN, T. Automatic satire detection: Are you having a laugh? In: **Proceedings of the ACL-IJCNLP 2009 Conference Short Papers**. Suntec, Singapore: Association for Computational Linguistics, 2009. p. 161–164. Disponível em: <https://www.aclweb.org/anthology/P09-2041>.

CAMARGO, F. D.; BRAGHETTO, K. R. Descoberta de conhecimento em redes sociais e bases de dados públicas. 2016.

CAMAROSSI, F. **O dia em que descobri que ninguém clica na notícia**. 2015. Disponível em: <https://medium.com/@fellipecamarossi/o-dia-em-que-descobri-que-ningu%C3%A9m-clica-na-not%C3%ADcia-eab77721e98f>. Acesso em: 11 mar. 2020.

CAMBRIDGE. **Cambridge, UK: Cambridge University Press**. 2018. Disponível em: <https://dictionary.cambridge.org/pt/dicionario/ingles/fake-news>. Acesso em: 09 mar. 2020.

CARVALHO, G. A. C. L. de; KANFFER, G. G. B. **O tratamento jurídico das notícias falsas (fake news)**. 2018. Disponível em: <https://www.conjur.com.br/2018-mar-19/opiniao-legislacao-dispoe-ferramentas-combater-fake-news>. Acesso em: 09 mar. 2020.

CASAROTTO, C. **TF-IDF: a abordagem de otimização on page que seu blog precisa**. 2019. Disponível em: <https://rockcontent.com/blog/tf-idf/#que>. Acesso em: 09 abr. 2020.

CATRACA LIVRE. **Site falso imitando G1 divulga pílulas para capacidade cognitiva**. 2017. Disponível em: <https://catracalivre.com.br/cidadania/site-falso-imitando-g1-divulga-pilulas-para-capacidade-cognitiva/>. Acesso em: 13 mar. 2020.

CAVALCANTI, E. R.; CAVALCANTI, E. P.; PIRES, C. E.; COSTA, R. A.; CAVALCANTI, C. R. Detecção e avaliação de cola em provas escolares utilizando mineração de texto: um estudo de caso. **Revista Brasileira de Informática na Educação**, v. 19, n. 02, p. 56, 2011.

CHOUDHURY, N. **World Wide Web and Its Journey from Web 1.0 to Web 4.0**. 2014. Acesso em: 13 mar. 2020.

Correio Braziliense. **Pesquisa mostra que 'fake news' são disseminadas por pessoas, não por robôs**. 2018. Disponível em: <https://www.correiobraziliense.com.br/app/noticia/tecnologia/2018/03/09/interna_tecnologia,664876/pesquisa-fake-news-sao-disseminadas-por-pessoas-nao-por-robos.shtml>. Acesso em: 15 mar. 2020.

COSTA, L. G. Classificação de fake news utilizando algoritmos de aprendizado de máquina e aprendizado profundo. 2019.

CRAMMER, K.; DEKEL, O.; KESHET, J.; SHALEV-SHWARTZ, S.; SINGER, Y. Online passive-aggressive algorithms. **Journal of Machine Learning Research**, v. 7, n. Mar, p. 551–585, 2006.

DARNTON, R. **The True History of Fake News**. 2017. Disponível em: <https://www.nybooks.com/daily/2017/02/13/the-true-history-of-fake-news/>. Acesso em: 09 mar. 2020.

FABIO, A. C. **O que é 'pós-verdade', a palavra do ano segundo a Universidade de Oxford**. 2016. Disponível em: <https://www.nexojornal.com.br/expresso/2016/11/16/O-que-%C3%A9-%E2%80%98p%C3%B3s-verdade%E2%80%99-a-palavra-do-ano-segundo-a-Universidade-de-Oxford>. Acesso em: 17 fev. 2020.

FACELI, K.; LORENA, A. C.; GAMA, J.; CARVALHO, A. C. P. d. L. *et al.* Inteligência artificial: Uma abordagem de aprendizado de máquina. 2011.

FERRARI, D. G.; SILVA, L. N. D. C. **Introdução a mineração de dados**. [S.l.]: Editora Saraiva, 2017.

FERREIRA, A. B. d. H. Novo dicionário aurélio da língua portuguesa. In: POSITIVO (Ed.). **Novo dicionário Aurélio da língua portuguesa**. 5. ed. [S.l.: s.n.], 2014.

G1. **Veja o que é #FATO ou #FAKE sobre as queimadas na Amazônia**. 2019. Disponível em: <https://g1.globo.com/fato-ou-fake/noticia/2019/08/22/veja-o-que-e-fato-ou-fake-sobre-as-queimadas-na-amazonia.ghtml>. Acesso em: 13 mar. 2020.

GABIELKOV, M.; RAMACHANDRAN, A.; CHAINTREAU, A.; LEGOUT, A. Social Clicks: What and Who Gets Read on Twitter? In: **ACM SIGMETRICS / IFIP Performance 2016**. Antibes Juan-les-Pins, France: [s.n.], 2016. Disponível em: <https://hal.inria.fr/hal-01281190>.

HOLZHACKER, D. O. **O impacto das "fake news" nas eleições**. 2017. Disponível em: <https://www.infomoney.com.br/colunistas/pensando-politica/o-impacto-das-fake-news-nas-eleicoes/>. Acesso em: 17 fev. 2020.

INDURKHYA, N.; DAMERAU, F. J. **Handbook of natural language processing**. [S.l.]: Chapman and Hall/CRC, 2010.

KAUFMAN, D.; SANTAELLA, L. O papel dos algoritmos de inteligência artificial nas redes sociais. **Revista FAMECOS**, v. 27, p. 34074, 2020.

KOTSIANTIS, S. B.; PINTELAS, P. E. Increasing the classification accuracy of simple bayesian classifier. In: SPRINGER. **International Conference on Artificial Intelligence: Methodology, Systems, and Applications**. [S.l.], 2004. p. 198–207.

KOVACH, B.; ROSENSTIEL, T. **Os elementos do jornalismo: o que os jornalistas devem saber e o público exigir**. Geração Editorial, 2003. ISBN 9788575090732. Disponível em: <https://books.google.com.br/books?id=6KAGQwAACAAJ>.

Laboratório de Estatística e Geoinformação - LEG/UFPR. **Métodos de reamostragem**. 2016. Disponível em: <http://cursos.leg.ufpr.br/ML4all/apoio/reamostragem.html#>. Acesso em: 30 abr. 2020.

LUPA, A. **Geraldo Alckmin, João Doria, Alberto Goldman e a última briga tucana**. 2017. Disponível em: <https://piaui.folha.uol.com.br/lupa/2017/10/11/alckmin-doria-goldman-psdb/>.

MALTA, L. H. A.; KUROIVA, M. A. R. L. Aprendizado de máquina e processamento de linguagem natural aplicados à identificação de discurso de ódio. 2019.

MANNING, C. D.; RAGHAVAN, P.; SCHÜTZE, H. **Introduction to information retrieval**. [S.l.]: Cambridge university press, 2008.

MARSLAND, S. **Machine learning: an algorithmic perspective**. [S.l.]: CRC press, 2015.

MARTINS, G. d. A. Estudo de caso: Uma estratégia de pesquisa: Editora atlas. 2006.

MIHALCEA, R.; STRAPPARAVA, C. The lie detector: Explorations in the automatic recognition of deceptive language. In: **Proceedings of the ACL-IJCNLP 2009 Conference Short Papers**. Suntec, Singapore: Association for Computational Linguistics, 2009. p. 309–312. Disponível em: <https://www.aclweb.org/anthology/P09-2078>.

MITCHELL, T. M. *et al.* **Machine learning**. [S.l.]: McGraw-hill New York, 1997.

MOHRI, M.; ROSTAMIZADEH, A.; TALWALKAR, A. **Foundations of machine learning**. [S.l.]: MIT press, 2018.

MONARD, M. C.; BARANAUSKAS, J. A. Conceitos sobre aprendizado de máquina. **Sistemas inteligentes-Fundamentos e aplicações**, Manole Ltda, v. 1, n. 1, p. 32, 2003.

MONDONI, J. P. A. Classificação de sentimentos em avaliações de livros na amazon. **Revista Brasileira em Tecnologia da Informação**, v. 1, n. 1, p. 11–21, 2019.

MONTEIRO, R. A.; SANTOS, R. L.; PARDO, T. A.; ALMEIDA, T. A. de; RUIZ, E. E.; VALE, O. A. Contributions to the study of fake news in portuguese: New corpus and automatic detection results. In: SPRINGER. **International Conference on Computational Processing of the Portuguese Language**. [S.l.], 2018. p. 324–334.

MORAIS, J. I. de; ABONIZIO, H. Q.; TAVARES, G. M.; FONSECA, A. A. da; JR, S. B. Deciding among fake, satirical, objective and legitimate news: A multi-label classification system. In: **Proceedings of the XV Brazilian Symposium on Information Systems**. [S.l.: s.n.], 2019. p. 1–8.

MÜLLER, F. de M.; SOUZA, M. V. de. Fake news: um problema midiático multifacetado. In: **Congresso Internacional de Conhecimento e Inovação–Ciki**. [S.l.: s.n.], 2018. v. 1, n. 1.

NICOLETTI, M. d. C. **Ampliando os limites do aprendizado indutivo de máquina através das abordagens construtiva e relacional.** Tese (Doutorado) — Universidade de São Paulo, 1994.

NILSSON, N. J. **The quest for artificial intelligence**. [S.l.]: Cambridge University Press, 2009.

NLTK. **Natural Language Toolkit**. 2018. Disponível em: <http://www.nltk.org/>. Acesso em: 21 mar. 2020.

O GLOBO. **Desembargadora acusa Marielle Franco de engajamento com bandidos**. 2018. Disponível em: <https://oglobo.globo.com/rio/desembargadora-acusa-marielle-franco-de-engajamento-com-bandidos-22500122>. Acesso em: 13 mar. 2020.

OLIVEIRA, L. M. R. Inteligência artificial aplicada a detecção de fake news. UFMA, 2019.

PEDREGOSA, F.; VAROQUAUX, G.; GRAMFORT, A.; MICHEL, V.; THIRION, B.; GRISEL, O.; BLONDEL, M.; PRETTENHOFER, P.; WEISS, R.; DUBOURG, V. *et al.* Scikit-learn: Machine learning in python. **Journal of machine learning research**, v. 12, n. Oct, p. 2825–2830, 2011.

PEREIRA, S. do L. Processamento de linguagem natural. 2009.

PIMENTA, A.; VALENTIM, P.; SANTOS, D.; NETO, M. Weka-g: mineração de dados paralela em grades computacionais. **Revista de Sistemas de Informação da FSMA**, v. 4, p. 2, 01 2009.

PIVETTA, S. P. Classificação de documentos do exército brasileiro utilizando o classificador naive bayes e técnicas de seleção de sentenças. Universidade Federal do Pampa, 2013.

POUBEL, M. **Fake News e Pós Verdade**. 2018. Disponível em: <https://www.infoescola.com/sociedade/fake-news/>. Acesso em: 09 mar. 2020.

PYSCIENCE-BRASIL. **Python: O que é? Por que usar?** s.d. Disponível em: <http://pyscience-brasil.wikidot.com/python:python-oq-e-pq>. Acesso em: 24 mar. 2020.

PYTHON. 2020. Disponível em: <https://docs.python.org/3/>. Acesso em: 22 mar. 2020.

RECUERO, R.; GRUZD, A. Cascatas de fake news políticas: um estudo de caso no twitter. **Galáxia (São Paulo)**, SciELO Brasil, n. 41, p. 31–47, 2019.

RIPOLL, L.; MATOS, J. C. M. Zumbificação da informação: a desinformação e o caos informacional. **RBBD. Revista Brasileira de Biblioteconomia e Documentação**, v. 13, 2017.

ROCHLIN, N. Fake news: Belief in post-truth. **Library Hi Tech**, v. 35, p. 00–00, 07 2017.

RUSSELL, S. J.; NORVIG, P. **Inteligência artificial**. [S.l.]: Elsevier, 2004.

SCIENTIFC AMERICAN BRASIL. **Algoritmo detector de fake news funciona melhor do que ser humano**. 2018. Disponível em: <https://sciam.uol.com.br/algoritmo-detector-de-fake-news-funciona-melhor-do-que-ser-humano/>.

SENSACIONALISTA. **Kinder Ovo trará ações da Petrobras de brinde**. 2014. Disponível em: <https://www.sensacionalista.com.br/2014/12/16/kinder-ovo-trara-acoes-da-petrobras-de-brinde/>. Acesso em: 11 mar. 2020.

SERRA, A. M. **Fake news: Uma discussão sobre o fenômeno e suas consequências**. 2018. Disponível em: <https://monografias.ufma.br/jspui/handle/123456789/3466>. Acesso em: 11 mar. 2020.

SHAO, C.; CIAMPAGLIA, G. L.; VAROL, O.; FLAMMINI, A.; MENCZER, F. The spread of fake news by social bots. **CoRR**, abs/1707.07592, 2017. Disponível em: <http://arxiv.org/abs/1707.07592>.

SHU, K.; SLIVA, A.; WANG, S.; TANG, J.; LIU, H. Fake news detection on social media: A data mining perspective. 2017. Disponível em: <https://arxiv.org/abs/1708.01967v3>.

SIDESWIPE. **Classificador linear on-line e SCW**. 2020. Disponível em: <http://kazoo04.hatenablog.com/entry/2012/12/20/000000>. Acesso em: 30 abr. 2020.

SILVA, F. Araujo da. **Detecção de Ironia e Sarcasmo em Língua Portuguesa: uma abordagem utilizando Deep Learning**. Tese (Doutorado), 02 2018.

SONONE, V. **Conventional guide to Supervised learning with scikit-learn — Passive Aggressive Algorithms- Generalized Linear Models (15)**. 2018. Acesso em: 09 abr. 2020.

SOUZA, R. M. de. **Investigando as fake news: análise das agências fiscalizadoras de notícias falsas no Brasil. 2017**. 2017.

STEHMAN, S. V. Selecting and interpreting measures of thematic classification accuracy. **Remote sensing of Environment**, Elsevier, v. 62, n. 1, p. 77–89, 1997.

TAN, P.-N.; STEINBACH, M.; KUMAR, V. **Introduction to data mining Pearson**. [S.l.]: Addison Wesley, Pearson, 2006.

TANDOC, E.; LIM, Z.; LING, R. Defining "fake news": A typology of scholarly definitions. **Digital Journalism**, p. 1–17, 08 2017.

TEIXEIRA, V. M.; MARCOS, A. D.; MACHADO, M. L. H. G.; CABRAL, H. L. T. B. As fake news e suas consequências nocivas à sociedade. 2018.

THU, P. P.; AUNG, T. Implementation of emotional features on satire detection. **International Journal of Networked and Distributed Computing**, v. 6, p. 78, 04 2018.

TWITTER. **Cat Zakrzewski no Twitter: "Just in: Twitter applied its new manipulated media label for the first time to a deceptively edited video of Joe Biden. It was shared by White House social media director Dan Scavino, and retweeted by the president.** 2020. Disponível em: <https://twitter.com/Cat_Zakrzewski/status/1236771032746414080>. Acesso em: 14 mar. 2020.

VASCONCELLOS, P. **Como saber se seu modelo de Machine Learning está funcionando mesmo**. 2018. Disponível em: <https://paulovasconcellos.com.br/como-saber-se-seu-modelo-de-machine-learning-est%C3%A1-funcionando-mesmo-a5892f6468b>. Acesso em: 12 mai. 2020.

WANKE, B. d. S. L.; COSTA, V. O.; PINA, A. C. de; FILHO, A. C. de P. Aplicação do classificador naive bayes para identificação de falhas de um manipulador robótico. 2014.

WARDLE, C. The different types of mis and disinformation. 2017. Disponível em: <https://firstdraftnews.org/latest/fake-news-complicated/>. Acesso em: 10 mar. 2020.

WIKIPEDIA. **Precisão e revocação**. 2016. Disponível em: <https://pt.wikipedia.org/wiki/Precis%C3%A3o_e_revoca%C3%A7%C3%A3o>. Acesso em: 27 mai. 2020.

WITTEN, I. H.; FRANK, E. Data mining: practical machine learning tools and techniques with java implementations. **Acm Sigmod Record**, ACM New York, NY, USA, v. 31, n. 1, p. 76–77, 2002.

YIN, R. K. **Estudo de Caso-: Planejamento e métodos**. [S.l.]: Bookman editora, 2015.

YSE, D. L. **Your Guide to Natural Language Processing (NLP)**. 2019. Disponível em: <https://towardsdatascience.com/your-guide-to-natural-language-processing-nlp-48ea2511f6e1>. Acesso em: 20 mar. 2020.

ZANNETTOU, S.; SIRIVIANOS, M.; BLACKBURN, J.; KOURTELLIS, N. The web of false information: Rumors, fake news, hoaxes, clickbait, and various other shenanigans. **CoRR**, abs/1804.03461, 2018. Disponível em: <http://arxiv.org/abs/1804.03461>.