RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

CHATBOT NO TELEGRAM PARA PREDIÇÃO DE FAKE NEWS EM PORTUGUÊS

Vinícius de Moraes Carvalho
Graduando em Ciência da Computação – Uni-FACEF
vimoraescarvalho@gmail.com

Jaqueline Brigladori Pugliesi
Doutora em Ciência da Computação – USP São Carlos
jbpugliesi@gmail.com

Resumo

A internet tem proporcionado acesso à informação para milhões de brasileiros ao mesmo tempo em que apresenta também o perigo da desinformação. *Fake News* é o termo em inglês para notícias falsas e se tornou popular a partir da decadência da procura por veículos jornalísticos profissionais e aumento do uso de redes sociais para compartilhamento de notícias. Tendo em vista que todo cidadão tem o direto de receber acesso a informações verdadeiras, este projeto busca, por uso da Inteligência Artificial, usando Aprendizado de Máquina e Processamento de Linguagem Natural, criar um modelo preditivo capaz de classificar notícias como falsas ou verdadeiras. O objetivo inicial do projeto era construir um *website* para disponibilizar uma integração com o modelo preditivo, no qual os usuários poderiam conhecer o projeto e enviar uma notícia para ser classificada em tempo real. O objetivo foi alterado e o resultado foi atingido ao criar um *chatbot* no Telegram para realizar o mesmo que o site, porém, de forma mais simples e eficiente, estando disponível em uma plataforma amplamente utilizada.

Palavras-chave: *Fake News*. Inteligência Artificial. Aprendizado de Máquina. Processamento de Linguagem Natural. Telegram.

Abstract

The Internet has provided access to information for millions of Brazilians while also presenting the risk of misinformation. Fake News has become popular due to the declining demand for professional journalistic vehicles and the increased use of social networks for sharing news. Considering that every citizen has the right to receive access to true information, this project seeks, through the use of Artificial Intelligence, using Machine Learning and Natural Language Processing, to create a predictive model capable of classifying news as false or true. The initial goal of the project was to build a website to provide an integration with the predictive model, where users could learn about the project and submit a news to be classified in real time. The goal was changed and the result was achieved by creating a chatbot on Telegram to perform the same as the website, but in a simpler and more efficient way, being available on a widely used platform.

Keywords: *Fake News. Artificial Intelligence. Machine Learning. Natural Language Processing. Telegram.*

RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

1 Introdução

As *fake news* sempre existiram na sociedade, porém a viralização do termo se deu nos anos recentes. O termo em inglês para notícias falsas pode ser definido como “um relato fictício relativo aos eventos atuais que são fabricados e muitas vezes intitulados de forma enganosa, com o propósito deliberado de enganar os usuários e motivá-los a divulgar” (BURSHTEIN, 2017 apud VITORINO; RENAULT, 2020, p. 3). O conceito se tornou popular durante as eleições presidenciais dos Estados Unidos, em 2016, e no Brasil, nas eleições presidenciais, em 2018, e com grande força durante a pandemia da COVID-19, em 2020, no Brasil, quando a propagação de desinformação foi usada para manipular e enganar a população.

Segundo uma pesquisa feita por Portinari e Hernandes (2018), e divulgada na Folha de São Paulo, entre 2017 e 2018 houve uma queda de 17% das interações nos veículos de jornalismo profissional e um aumento de 61,6% para o grupo de páginas que propagam notícias falsas e sensacionalistas.

Com o aumento da desinformação no Brasil e suas graves consequências, as notícias falsas são prejudiciais para o desenvolvimento da sociedade. Em um estudo realizado pela Avaaz e divulgado por Mayara (2020) no jornal online Estado de Minas, sete a cada dez brasileiros já acreditaram em pelo menos uma notícia falsa relacionada à pandemia da COVID-19.

Os motivos deste projeto têm como base o direito que todos os cidadãos possuem de receber acesso à verdade, ou, ao menos, deixar de serem enganados. Com o uso de métodos dentro da Inteligência Artificial é possível criar ferramentas para minimizar os danos causados pelas *fake news* e tentar garantir que, com maior facilidade, seja possível identificar a veracidade dos conteúdos compartilhados.

O principal objetivo do projeto é desenvolver em um modelo preditivo usando Aprendizado de Máquina e Processamento de Linguagem Natural que seja capaz de, utilizando uma base de dados com conteúdo textual, com caráter informativo na língua portuguesa, de notícias verdadeiras e falsas, aprender padrões e retornar a probabilidade de uma notícia ser falsa ou verdadeira e, com os resultados obtidos, criar um *chatbot* no aplicativo Telegram que proporcione uma integração simples na qual o usuário posso validar a veracidade de uma notícia.

Uma vez que a internet avançou ao ponto de estar disponível facilmente na mão de cada um que tem um *smartphone*, tudo ficou mais rápido e, com isso, o acesso à informação, independentemente de ser verdadeira ou não, também passou a ser algo comum. Por isso é importante a existência de ferramentas capazes de contribuir junto com a informação para levar adiante a verdade e reduzir a desinformação.

Durante o andamento do projeto foram realizados, primeiramente, estudos dentro da área de Inteligência Artificial, especificamente o campo de Aprendizado de Máquina e Processamento de Linguagem Natural, que são essenciais para o desenvolvimento do projeto. Em conjunto com os estudos necessários, foram obtidas informações acerca de *fake news* para compreendê-las.

Para o desenvolvimento do projeto foi usada a base Fake.br Corpus (MONTEIRO et al., 2018) em adição com o conjunto FakeRecogna (GARCIA et al., 2022). A linguagem de programação Python foi utilizada para o desenvolvimento.

RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

2 Inteligência Artificial

A Inteligência Artificial (IA) é parte do campo de estudo da Ciência da Computação e cresce rapidamente no Brasil, tendo sido registrado um aumento de 32%, em 2018, para 42%, em 2020, do uso dessa tecnologia pelas empresas pelo Índice de Nível de Inovação e Crescimento IA, de acordo com uma pesquisa realizada pela Everis (TIINSIDE, 2021).

Segundo Luger (2013) “a Inteligência Artificial pode ser definida como o ramo da ciência da computação que se ocupa da automação do comportamento inteligente”, automação essa que pode ser feita utilizando estruturas de dados para representar conhecimento e linguagens de programação para implementar soluções. Em adição, Coppin (2013) comenta que “a Inteligência Artificial envolve utilizar métodos baseados no comportamento inteligente de humanos e outros animais para solucionar problemas complexos”. Entretanto, o conceito de inteligência não é precisamente compreendido mas é sempre comparado com a capacidade humana de raciocínio lógico e aprendizado, como se percebe pela definição existente no dicionário de psicologia (Associação Psicológica Americana, 2010, p. 521 apud BLANCO et al., 2017) que descreve inteligência como sendo a “capacidade de extrair informações, aprender com a experiência, adaptar-se ao ambiente, compreender e utilizar corretamente o pensamento e a razão”.

A IA proporciona para a sociedade a possibilidade de cada vez mais a máquina ocupar as tarefas difíceis e resolver os problemas em que o ser humano gasta e investe muito de seu tempo, tornando, então, muitos trabalhos menos perigosos, mais ágeis e com a menor interferência humana possível, sendo esses trabalhos realizados em linhas de produção em fábricas geridas por máquinas, até carros autônomos, que já percorrem as ruas de muitos países.

Tendo em vista que os problemas encontrados atualmente vão muito além de um processamento lógico e algorítmico no qual o código simples resolveria, a aplicação de uma ferramenta em Inteligência Artificial entra como o caminho para soluções adequadas com grandes processamentos de dados. Entre os pontos que podem ser observados para a necessidade de aplicação de alguma técnica de IA, os principais apresentados por Luger (2013) são: a carência da execução de um raciocínio sobre padrões; a necessidade de aprendizado autônomo; a análise semântica e sintática de um contexto; a resolução de problemas com informações inexatas; e o uso de grandes quantidades de conhecimento sobre um domínio. Todos esses pontos podem ser resolvidos utilizando os diversos ramos da Inteligência Artificial, como Aprendizado de Máquina, Processamento de Linguagem Natural, *Deep Learning*, entre outros.

2.1 Aprendizado de Máquina

Dentro da Inteligência Artificial existe a área denominada Aprendizado de Máquina (AM), responsável pelo campo que cria ferramentas capazes de aprender de forma autônoma, ou seja, adquirir conhecimento e tratar das técnicas responsáveis por esse processo de aprendizado. Para um sistema inteligente se tornar completo ele precisa ter seu módulo de aprendizado, que é uma característica de extrema importância quando se define a inteligência humana.

O aprendizado dentro da IA pode ser definido como “qualquer mudança em um sistema que melhore o seu desempenho na segunda vez que ele repetir a

RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

mesma tarefa tirada da mesma população" (SIMON, 1983 apud LUGER, 2013). Para o Aprendizado de Máquina é essencial que a ferramenta utilize de experiências anteriores para solucionar novos casos e é nesse conceito que os modelos são criados.

Para compreender o aprendizado dentro da IA pode-se abordar primeiramente a indução "caracterizada como o raciocínio que se origina em um conceito específico e o generaliza, ou seja, da parte para o todo" (MONARD; BARANAUSKAS, 2003). Essa é a forma da inferência lógica que é usada para chegar em alguma conclusão ou solução para o problema abordado e para isso, um conceito é aprendido com base em uma gama de exemplos e uma resposta é apresentada dentro da probabilidade de obter aquele resultado. Para o método de indução é de suma importância a quantidade e qualidade dos exemplos, visto que isso está diretamente ligado a validade das soluções proposta pelo algoritmo.

No processo de aprendizado, segundo Monard e Baranauskas (2003), os exemplos são obtidos de forma externa por meio de uma base de dados pré-estabelecida ou construída previamente e pode ser dividido em dois métodos diferentes. O método supervisionado é aquele em que, além de fornecer os exemplos para o sistema de aprendizado, também são fornecidas as classificações desses exemplos com objetivo de aprender a classificar novos exemplos que ainda não foram rotulados com uma classe. Para o caso de valores discretos no atributo que representa a classe, tem-se o método de classificação e, para classes com valores contínuos, tem-se o método de regressão, outro método, é o não-supervisionado, em que o sistema recebe os exemplos ainda não rotulados e tenta agrupá-los para tornar possível analisar esses grupos e classificar cada um dentro do contexto, podendo ainda repetir o processo para adicionar atributos e parâmetros.

Como forma de construir um sistema de Aprendizado de Máquina, existem os paradigmas de aprendizado (MONARD e BARANAUSKAS,2003):

- O Simbólico leva a uma representação final na forma de uma expressão lógica, árvore de decisão, regras ou rede semântica, apenas utilizando exemplos e contraexemplos para construir uma representação simbólica do conceito.
- O Estatístico utiliza um modelo probabilístico que busca combinações entre os atributos dos exemplos disponibilizados para chegar a uma possibilidade de classificação mais provável e, a cada exemplo aprendido, aumentar sua precisão de resultados.
- O Baseado em Exemplos utiliza exemplos já conhecidos para classificar os novos, supondo que, se têm as mesmas características, é possível classificá-los da mesma forma, nesse caso o algoritmo é conhecido com *lazy,* pois precisa manter em memória os exemplos para saber como classificar os novos.
- O Conexionista utiliza Redes Neurais e se baseia no modelo biológico do sistema nervoso no qual as informações são levadas até os neurônios e esses processam e levam aos próximos, até obter algum valor sobre tal informação. Dessa forma, os neurônios artificiais dentro da rede neural estão altamente interconectados.
- O Genético tem analogia com a teoria de Darwin sobre a sobrevivência do mais forte e mais bem adaptado, sendo que nesse paradigma os elementos que têm a melhor performance se proliferam e os mais fracos são descartados.

RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

2.2 Processamento de Linguagem Natural

A Linguagem Natural é o nome dado para o meio de comunicação que os seres humanos utilizam para se expressar, seja por meio da fala ou da escrita. Para que um programa compreenda uma linguagem usada por pessoas são necessários diversos itens a serem processados sobre esse conteúdo, textual ou uma transcrição de um áudio. A compreensão de linguagem, de acordo com Luger (2013, p. 512), "requer inferências sobre objetivo, conhecimento e suposições do locutor, bem como sobre o contexto da interação" e também se precisa "considerar questões como não monotonicidade, revisão de crença, metáfora, planejamento, aprendizado e as complexidades práticas da interação humana".

Pode-se definir alguns conceitos principais ao estudar Processamento de Linguagem Natural da seguinte forma (PEREIRA, 2015):

- Som: representado pela fonologia, o reconhecimento dos sons que compõem uma palavra.
- Estrutura: representado pela morfologia, que reconhece as palavras pelas unidades produtivas que compõem, e sintaxe, que define a estrutura de uma frase pela forma como as palavras se relacionam.
- Significado: representado pela semântica, que associa significado a uma estrutura sintática baseado nos termos dos significados das palavras que compõem, e pragmática, que verifica se o significado associado à uma estrutura sintática é realmente o significado apropriado no contexto.

Para Luger (2013), existem dois conceitos a mais. Um é geralmente desprezado, e o outro representa um conhecimento essencial e eles são, respectivamente:

- Prosódia: esse conceito trata do ritmo e entonação da linguagem, efeito muito importante em poesias e em cantos.
- Conhecimento do mundo: inclui o conhecimento do mundo físico, da interação social humana e de objetivos na comunicação e interação.

O principal ponto de partida para uma análise de conteúdo é o entendimento de o que é uma gramática para o processamento de linguagem natural, para Pereira (2015) "Uma gramática é uma especificação formal da estrutura das sentenças permitidas numa linguagem", e para a construção de uma gramática o autor resume:

> um conjunto de símbolos terminais, denotando palavras da linguagem, um conjunto de símbolos não-terminais, denotando os componentes das sentenças, e um conjunto de regras de produção, que expandem símbolos não-terminais numa sequência de símbolos terminais e não-terminais (PEREIRA, 2015, p. 2)

A partir de uma gramática definida pode-se utilizar de algum método de derivação sobre uma sentença. Um exemplo seria a derivação de cima para baixo (*top-down*) que começa de um símbolo inicial para chegar aos terminais, outro é a derivação de baixo para cima (*bottom-up*) que realiza o inverso, as derivações podem ser representadas por uma árvore sintática na qual cada nó é um símbolo que aparece na gramática. Tendo isso em mãos, está feita a análise sintática. O resultado obtido com a existência de uma derivação ou árvore sintática válida, diz Luger (2013, p. 518), "não apenas que a sentença é válida na gramática, como também determina a

RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

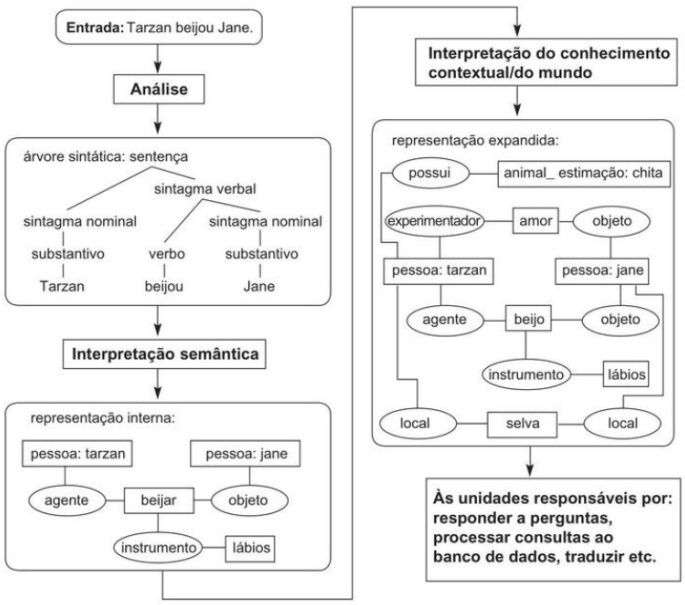
estrutura da sentença". Essa estrutura gerada é chamada de estrutura frasal e "define a organização linguística mais profunda da linguagem", o que dá início para a próxima etapa do processo, sendo essencial para a análise semântica, que vai interpretar profundamente o significado do todo e da forma como cada palavra interage no contexto específico.

Na Figura 1 estão presentes, resumidamente, os estágios na produção de uma representação de uma sentença e percebe-se a divisão entre as etapas já citadas, com um exemplo prático, além de incluir duas últimas etapas: a interpretação do conhecimento contextual/do mundo, que utiliza do conhecimento já possuído para classificar informações importantes, e, por fim, o programa responsável utiliza as informações obtidas para retornar a melhor saída.

Segundo Luger (2013) uma boa aplicação de tecnologia para compreensão de Processamento de Linguagem Natural (PLN) é o uso de interface com linguagem natural para banco de dados. A tarefa dessa ferramenta é traduzir uma pergunta em linguagem natural em uma consulta bem formada em linguagem de banco de dados, usando como exemplo a pergunta: "Quem contratou John Smith?" obter-se-ia algo como: *SELECT GERENTE FROM GERENTE_PESSOAL WHERE EMPREGADO = 'John Smith'*. A aplicação se encontra justamente na forma como é traduzida a questão empregada, na qual é preciso compreender o contexto e significado de cada palavra, o contexto do todo e, por último, compreender de onde retirar cada informação e como associá-las dentro do banco de dados.

O Processamento de Linguagem Natural pode ser aplicado em diversas modalidades uma vez que seu objetivo está em reconhecer padrões de linguagem, estruturas e significados. É simples associar que aplicações básicas podem estar envolvidas com análise de textos e pode-se ir mais afundo e olhar para as ferramentas que já existem, como corretor automático, sugestões em filtros de busca, assistentes virtuais, tradução de idiomas e até mesmo, com o conhecimento certo, é possível identificar a veracidade de um conteúdo textual.

Figura 1 - Estágios na produção de uma representação de uma sentença



Fonte: LUGER, 2013, p. 516.

RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

3 Fake News

A aparição do termo *fake news* foi recente, porém o conceito já existe há muito tempo. A presença de notícias falsas na internet se tornou muito frequente nos últimos anos, principalmente durante a pandemia da COVID-19, em 2020, no Brasil, quando informações sobre o vírus e sobre vacinas se espalharam com grande velocidade e muitas vezes essas informações não eram verídicas.

Segundo uma pesquisa feita por Portinari e Hernandes (2018), e divulgada na Folha de São Paulo, entre 2017 e 2018 houve uma queda de 17% das interações nos veículos de jornalismo profissional e um aumento de 61,6% para o grupo de páginas que propagam notícias falsas e sensacionalistas.

Uma vez que a internet avançou ao ponto de estar disponível facilmente na mão de cada um que tem um *smartphone*, tudo ficou mais rápido. Com isso, o acesso à informação, independentemente de ser verdadeira ou não, também passou a ser algo comum.

Com o rápido avanço da internet pelo mundo nas últimas décadas, de acordo com Noberto e Loiola (2019), segundo um estudo da ONU, 51% da população mundial já tinha acesso à internet em 2019 trazendo o Brasil à frente até de países desenvolvidos. O fato de o acesso estar nas mãos de tanta gente é um dos motivos pelos quais as informações falsas se espalham tão facilmente, normalmente aquelas que chocam ou demonstram grande importância, são rapidamente repassadas com ou sem a intenção de propagar inverdades.

O principal problema para frear o avanço de informações não verificadas é resultante da grande quantidade de dados e também da falta de checagem em alguns meios, como nas redes sociais, na qual a checagem não é feita com precisão ou é minimamente feita.

As notícias falsas podem surgir de duas formas diferentes, segundo Freire e Goldschmidt (2019, p. 41), uma quando “é iniciada na rede social por meio da sua publicação e, posteriormente, potencializada pela sua possível propagação” e outra quando “uma notícia não fake é publicada, porém se torna fake, a partir do seu espalhamento, de acordo com as contribuições intencionalmente falsas feitas durante a sua propagação”, porém, independentemente da origem, o preocupante é o razão de continuarem a se espalhar e pessoas acreditarem. Existem alguns motivos para levar alguém a tomar algo como verdade, também descritos por Freire e Goldschmidt (2019) que, entre eles, destacam o fato de que as pessoas preferem informações que confirmem suas opiniões ou pela relação de ganhos ou perdas que aquilo pode trazer, ou apenas pela aceitação de terceiros sobre o fato.

A melhor forma de combater o avanço das *fake news* é sempre disponibilizar meios para que as pessoas encontrem o que é verdade, ferramentas para identificar, analisar e distribuir informação com responsabilidade e utilizar da tecnologia para frear o poder da desinformação. Existem diversos meios para se identificar notícias falsas. De acordo com Aguiar (2020 apud FECOMÉRCIO, 2020) o Brasil é uma referência em agências de verificação de fatos, entre essas agências estão Aos Fatos, Agência Lupa, Fato ou Fake e o Projeto Comprova, e ainda complementa com alguns cuidados para se tomar antes de compartilhar qualquer notíca, como: verificar a fonte da notícia, buscar outras fontes, verificar mensagens encaminhadas muitas vezes e ter atenção com notícias que causam muito espanto ou indignação.

RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

4 Ferramentas e Métodos

O desenvolvimento deste projeto faz uso de diversas ferramentas e métodos dentro da área de tecnologia.

4.1 Python

Para a implementação do projeto foi escolhida a linguagem Python devido à sua simplicidade e clareza, além de ser objetiva e ir direto ao ponto, como diz Menezes (2010), porém o principal motivo é o fato de ser uma linguagem poderosa, com inúmeros recursos e diversas aplicações na área de Aprendizado de Máquina, sendo essencial no desenvolvimento deste projeto, visando utilizar as melhores bibliotecas relacionadas ao tema, tais como:

- Pandas: utilizada para manipulação e análise de dados, oferece inúmeras funções para operar com grandes quantidades de dados e obter diferentes resultados para análise.
- Scikit-learn: ferramenta utilizada para análise preditiva de dados, projetada para uso de Aprendizado de Máquina, baseada em outras bibliotecas como NumPy e SciPy, podendo ser aplicada em diversos contextos.
- PyTelegramBotAPI: uma implementação em Python para a Telegram Bot API, que disponibiliza métodos para criar integrações com *chatbots* do Telegram.

4.2 Colab

Como ferramenta de desenvolvimento para produzir código em Python, o Colab ou Colaboratory é um facilitador para um início rápido do desenvolvimento, uma vez que não necessita de configuração para começar a utilizar. Possui acesso gratuito a GPUs (*Graphics Processing Units*, unidades de processamento gráfico) e é de fácil compartilhamento. A ferramenta é um ambiente web interativo chamado notebook Colab e nele é possível escrever e executar qualquer código em Python e importar qualquer biblioteca, sem necessidade de instalação ou configuração (COLABORATORY, 2021).

4.3 Conjuntos de Notícias

Para iniciar os trabalhos foi necessário escolher bases de dados apropriadas para o tema.

- O Fake.br Corpus (Monteiro R.A., Santos R.L.S., Pardo T.A.S., de Almeida T.A., Ruiz E.E., 2018) é um conjunto de notícias verdadeiras e falsas escritas em português, criado por integrantes do *Interinstitutional Center for Computational Linguistics*. Essa base de dados possui características adequadas para esse desenvolvimento por possuir dados homogêneos, com contexto jornalístico e mais 7 mil notícias.
- Para complementar o conjunto de notícias, a base FakeRecogna (Garcia, Gabriel L.; Afonso, Luis C. S.; Papa, João P., 2022), com mais de 11 mil notícias foi somado no total do conjunto para o projeto. As

RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

notícias falsas e verdadeiras têm ligações entre si, ou seja, as notícias falsas não necessariamente possuem as verdadeiras no conjunto, elas foram coletadas por mineração de dados em páginas de agências de notícias sérias e com reconhecimento nacional, como UOL Confere, E-farsas, Fato ou Fake, Boatos.org e outras.

4.4 Telegram

O Telegram é um aplicativo de mensagens com foco em velocidade e segurança. Ele é baseado em nuvem com sincronização contínua, o que faz com que seja possível acessar as mensagens de vários dispositivos ao mesmo tempo e compartilhar um número ilimitado de fotos, vídeos e arquivos de até 2GB cada. Também é possível criar grupos de até 200.000 pessoas ou canais com transmissão para audiência ilimitada (TELEGRAM, *sd, online*).

No Brasil o número de usuários foi estimado em 60% da população com *smartphones*, tendo o maior aumento depois de 2019, quando apresentava apenas 13%, segundo Tecmundo (2022).

Além disso, a *Application Programming Interface* (*API*) do Telegram é aberta e possibilita que qualquer pessoa possa criar um aplicativo na plataforma. Há também a *Bot API*, uma implementação em código aberto, a qual permite a fácil criação de ferramentas especializadas para o Telegram, integração de serviços e até mesmo aceitar pagamentos de usuários do mundo todo (TELEGRAM, *sd, online*).

4.5 Implementação do Algoritmo

Uma vez definidas as ferramentas, a implementação do código necessário pode ser dividida em etapas significativas.

4.5.1 Pré-Processamento

A primeira etapa e de grande importância é o pré-processamento dos dados e Gomes (2019) o define como “um conjunto de atividades que envolvem preparação, organização e estruturação dos dados”. Essa etapa representa a limpeza dos dados e organização de todo o material coletado das bases de dados definidas previamente, deixando apenas o que é informação relevante, uma vez que esse material é composto por textos de notícias, algumas características podem ser irrelevantes ou redundantes no processo de classificação do texto, explica Pinheiro (2021).

Existem medidas a serem tomadas a fim preparar um conteúdo textual para se tornar manipulável por um algoritmo e descritas por Pinheiro (2021):

- Tokenização: é o ato de decompor o texto por termos e chamá-los de *token*, podendo dividir o que é um *token,* separando o texto por quebras de linha e espaços em branco.
- Remoção de *Stopwords*: alguns elementos podem ser removidos do texto por não apresentar uma semântica significativa, como artigos, preposições e verbos auxiliares.
- *BoW*: *Bag of words,* em português, saco de palavras é um modelo de representação que desconsidera a gramática, o contexto e a ordem das

RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

palavras, para criar uma representação numérica de um elemento e acrescentar informações relevantes, podendo, por exemplo, obter informações de quantas vezes um termo aparece no conteúdo.

- TFIDF: *Term Frequency* (TF) e *Inverse Document Frequency* (IDF) é um modelo que, assim como *Bag of words,* cria um representativo numérico, porém é calculado pelo produto da frequência do termo (TF) e pela frequência inversa do termo (IDF), cálculo realizado utilizando não só o documento analisado, mas também todos os documentos do *corpus*.
- *Word Embeddings*: técnica que consiste em criar vetores contendo números que representam os termos encontrados no conteúdo. Esses valores são posteriormente ajustados por uma rede neural. Isto é usado para o aprendizado de forma que o algoritmo seja capaz de criar conhecimento morfológico, sintático e semântico sobre cada termo presente no corpus, além de tornar possível medir a distância entre os componentes, realizando alguns cálculos.
- OneHotEncoder: essa técnica permite criar vetores numéricos com base em categorias, os valores para cada categoria são criados como 0 ou 1 em novas colunas dentro de um vetor segundo Yadav (2019).

4.5.2 Extração de Padrões

O momento de treinamento do algoritmo é o ponto em que é utilizado todo o conteúdo pré-processado que foi gerado até o momento. O modelo será treinado para poder realizar previsões em casos futuros.

Para atingir tal objetivo foi utilizada a biblioteca Sklearn (scikit-learn), que possui os métodos necessários para o Aprendizado de Máquina. Nesse caso é preciso testar diversos métodos, neste artigo foi utilizado o de Regressão Logística, uma vez que os valores trabalhados são contínuos, ou seja, valores numéricos. A regressão logística estuda a relação entre uma variável resposta e uma ou mais variáveis independentes, nela as variáveis dependentes estão dispostas em categorias e a resposta é expressa por meio de uma probabilidade de ocorrência (VENTICINQUE et al., 2007).

Outros métodos testados foram baseados em Árvore de Decisão, que é um modelo de classificação cuja estrutura consiste em um número de nós e arcos, um nó origina arcos e assim outros nós, o topo é conhecido como raiz e os nós mais abaixo são as folhas. Esse formato de árvore representa um determinado teste para cada nó e os arcos levam para as folhas que representam a classe final (FURNKRANZ; GAMBERGER; LAVRAC, 2012).

Os resultados esperados nesta etapa são atingidos ao aplicar as funções da biblioteca sobre os dados, fazendo com que o algoritmo aprenda os padrões dos textos, associando com sua classificação e classifique corretamente textos ainda não rotulados como notícias falsas nem verdadeiras.

4.5.3 Avaliação dos Resultados

Com o modelo já treinado, é preciso realizar então um ciclo de avaliações e ajustes, visto que o modelo precisa ser treinado diversas vezes com diferentes parâmetros para atingir os melhores resultados.

RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

A avaliação dos resultados é feita com base nas classificações dos algoritmos, para isso, quatro métricas foram utilizadas, que são explicadas por Bonfim (2020):

- Acurácia: essa métrica mede de forma simples o quanto o modelo acertou, sabendo assim a sua performance para o treino realizado com aqueles parâmetros. Para obter essa métrica é dividido o número de acertos pela quantidade de dados de entrada.
- *F1-score*: é calculado com a média de precisão e *recall*, sendo precisão o número de positivos divididos pela soma do total de positivos e falso positivos, já *recall*, ou sensibilidade, é a taxa de números de verdadeiros positivos dividido pela soma dos verdadeiros positivos e falso negativos.
- *ROC-curve*: a curva de característica de operação do receptor é a divisão dos verdadeiros positivos pelo total de positivos e a divisão dos falsos positivos pelo total de negativos, representados graficamente.
- Matriz de Confusão: apresenta em uma tabela a distribuição dos dados entre a classificação real e a classificação predita.

4.5.4 Utilização do Conhecimento

Por último, a utilização do conhecimento. Nesse momento os resultados gerados nas etapas anteriores são utilizados para o propósito final: identificar a veracidade de notícias. Para isso foi criado um *chatbot* usando o Telegram para demonstrar as informações do projeto e interagir com o usuário. O principal objetivo da ferramenta foi incluir uma integração com o modelo criado, possibilitando a realização de testes de novas notícias, que poderão ser incluídas no material e nos resultados, e a informação se a notícia parece ser falsa ou não é devolvida para o usuário.

5 Desenvolvimento

Essa seção tem o objetivo de apresentar os passos seguidos durante o desenvolvimento do projeto, sendo assim, esta apresenta detalhes desde a aquisição da base de dados, processamento e criação do modelo preditivo até a implementação do *chatbot* no Telegram.

5.1 Definição do Conjunto de Notícias

A base Fake.Br Corpus contém 7200 notícias, sendo 3600 verdadeiras e 3600 *fake news*. O intervalo de notícias está situado entre janeiro de 2016 a janeiro de 2018 e todas as notícias falsas foram obtidas manualmente, enquanto as verdadeiras relacionadas foram obtidas com um *crawler* em sites de notícias (MONTEIRO et al., 2018).

A base conta com uma categorização dos temas, que são 6 no total: política, sociedade e cotidiano, TV e celebridades, ciência de tecnologia, religião e economia.

O corpus foi selecionado por apresentar notícias em português, recentes e com outras informações já preparadas e disponibilizadas gratuitamente, porém a

RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

quantidade de notícias, mesmo sendo alta, poderia ser maior para melhorar os resultados, por isso o conjunto FakeRecogna foi utilizado também.

O conjunto FakeRecogna (GARCIA et al., 2022) conta com 11902 notícias, obtidas de forma parecida ao Fake.Br Corpus, as classes estão distribuídas de forma balanceada, os temas apresentados são nomeados da seguinte forma: brasil, entretenimento, saúde, política, ciência e mundo.

5.2 Google Colaboratory e Google Drive

Para iniciar o desenvolvimento o Google Colaboratory foi essencial, uma vez que ele disponibiliza uma interface web para escrita de código em Python e máquinas na nuvem com alta capacidade de processamento. Além disso a integração do Colaboratory com o Google Drive para salvar o projeto e utilizar arquivos salvos na nuvem trouxe agilidade para implementar o código utilizando os conjuntos de notícias.

5.3 Pré-Processamento

O pré-processamento foi divido em três etapas importantes para o treinamento do modelo.

5.3.1 Agrupamento das Notícias

Uma vez que foram usadas duas bases diferentes, primeiramente foi feita uma extração da base Fake.Br Corpus, que apresentava arquivos de texto exclusivo para cada notícia e um outro arquivo com meta dados de cada uma, agrupando essas informações em uma lista só e adicionando na lista o conteúdo do arquivo csv (*comma-separated values*, valores separados por vírgula) com as notícias do conjunto FakeRecogna, foi obtido o total de notícias agrupadas. Foi necessário renomear os temas para ficarem em um padrão único: sociedade e cotidiano se tornaram mundo, TV e celebridades foi convertido para entretenimento e ciência foi alterado para ciência e tecnologia. O resultado foi salvo em um arquivo para ser reutilizado posteriormente no código sem que fosse necessário realizar novamente todo o processo anterior.

5.3.2 Tratamento do Texto

O conjunto com todas as notícias ficou como apresentado na Tabela 1.

Tabela 1 – Cinco primeiros itens do conjunto final

Texto	Tema	Classe
Caiu a máscara! Número de desempregados no paí...	economia	fake
Nova rebelião no Rio Grande do Norte. Presos q...	mundo	fake
Deputado diz que é contra a liberação das drog...	política	fake
Caiu a máscara do Prof. Villa da Jovem Pan. Ví...	política	fake
Lista com nomes dos parlamentares que "assassi...	política	fake

Fonte: Autor

RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

O tratamento do texto é necessário para seguir adiante com o treino do modelo, alguns detalhes no texto são desnecessários, como: acentos, caracteres especiais e *stopwords* e por isso foram removidos do conjunto, além deixar todo o texto em letras minúsculas, assim não existem diferenças quando se encontra a mesma palavra após o tratamento o resultado foi o seguinte apresentado na Tabela 2.

Tabela 2 – Cinco primeiros itens do conjunto tratado

Texto	Tema	Classe
caiu mascara numero desempregados pais recorde...	economia	fake
nova rebeliao rio grande norte presos quebram ...	mundo	fake
deputado diz contra liberacao drogas explica r...	politica	fake
caiu mascara prof villa jovem pan video mostra...	politica	fake
lista nomes parlamentares assassinaram lava-ja...	politica	fake

Fonte: Autor

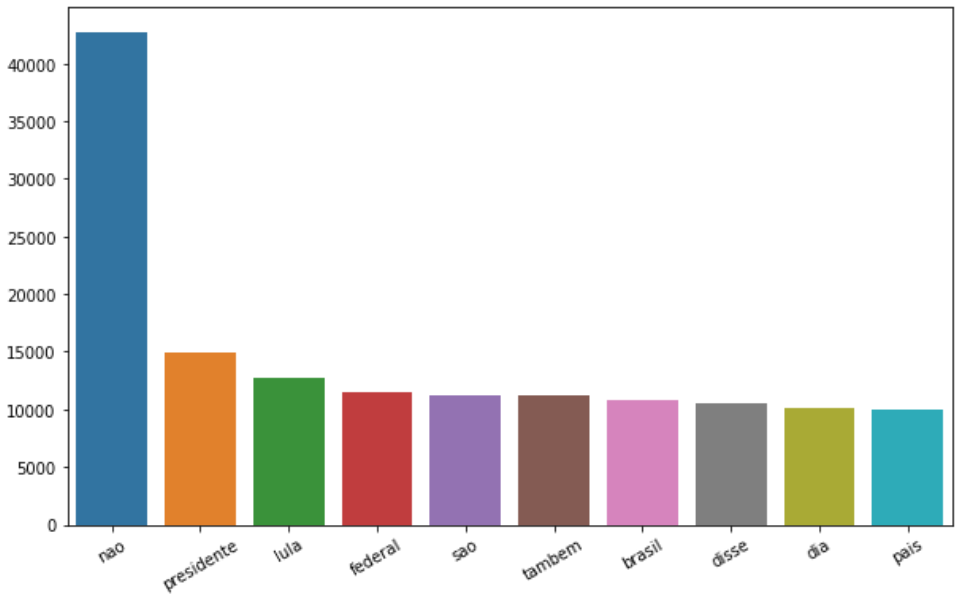
Feito o tratamento, esse conjunto é salvo novamente para ser utilizado no treinamento do modelo.

5.3.3 Análise do Conjunto

O conjunto final produzido pelo pré-processamento foi analisado para identificar a distribuição das informações relevantes. Com um conjunto de 18750 notícias alguns gráficos foram criados.

A Figura 2 apresenta o gráfico criado usando um dicionário gerado com todas as palavras presentes no conjunto já tratado e apresenta as dez palavras que mais aparecem.

Figura 2 – Dez palavras que mais aparecem no conjunto

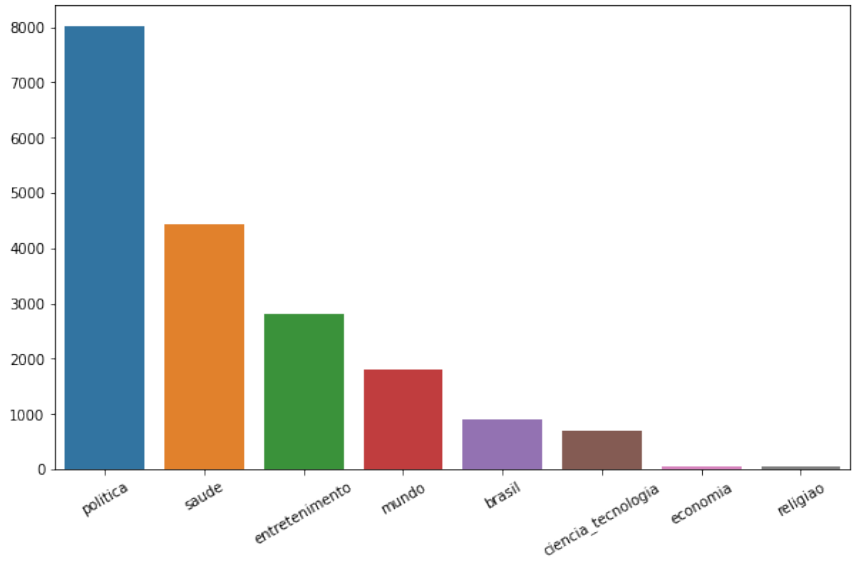


Fonte: Autor

RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

A Figura 3 possui o gráfico que representa a distribuição dos temas das notícias, colunas essa que também foi utilizada no treinamento e pode-se notar que a maior parte das notícias são relacionadas com política.

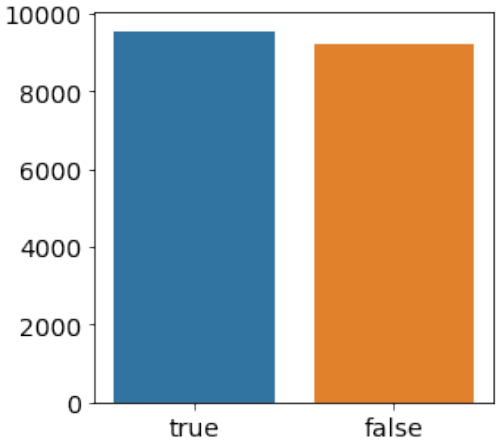
Figura 3 – Distribuição dos temas em todo o conjunto



Fonte: Autor

A classificação das notícias ficou bem distribuída em verdadeiras e falsas, sendo quase 50% para cada, como é apresentado na Figura 4.

Figura 4 – Ocorrência de notícias



Fonte: Autor

5.4 Treinamento do Modelo

RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

Para seguir com o treinamento foi preciso separar o conjunto entre treino e teste em 70% e 30%, respectivamente, assim parte dos dados são usados para o treinamento e o restante para a validação do modelo.

Antes de treinar o modelo é preciso transformar todo o conteúdo textual em numérico, uma vez que a maioria dos classificadores não trata textos para o treino. Duas formas foram escolhidas para converter o texto e testar qual teria uma melhor performance.

O método *Bag of words* desconsidera contexto e ordem das palavras, nesse caso foi utilizado o método CountVectorizer da biblioteca Scikit-learn para criar um vetor com base na ocorrência das palavras na coluna contendo as notícias do conjunto.

Outro método utilizado para tratar a coluna de notícias foi o TFIDF que utiliza da frequência do termo (TF) dividida pela frequência inversa do termo (IDF) não só do item, mas do corpus inteiro para criar um representativo numérico, para isso o método TfidfVectorizer de Scikit-learn foi usado.

Em um primeiro teste foi utilizado somente a coluna com o texto das notícias para treinar o modelo, posteriormente foi adicionada a coluna tema para comparar os resultados. Para a coluna contendo os temas foi usado o método OneHotEncoder, também de Scikit-learn, para criar um vetor numérico com os temas existentes.

A combinação das colunas em um vetor único para treino e outro para teste usando o método make_column_transformer deixa as informações preparadas para o treino do classificador, a partir disso foi obtido um vetor com *Bag of words e* OneHotEncoder e outro com TFIDF e OneHotEncoder. A Figura 5 exemplifica essa criação.

Apenas um método entre *BoW* e *TFIDF* será usado para o classificador definitivo, para decidir entre um destes foi feito um teste para medir os resultados de precisão dos dois utilizando o método de Regressão Logística que se apresentam na Tabela 3. É possível observar que as métricas de acurácia, F1 e precisão registraram valores melhores quando utilizando *TFIDF*, por isso, foi o escolhido para dar seguimento.

RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

Figura 5 – Criação dos vetores

```
#sem tema
#bow
bow = CountVectorizer(max_features = 5000)
x_train_bow = bow.fit_transform(x_train['text'])
x_test_bow = bow.transform(x_test['text'])
#tfidf
tfidf_vec = TfidfVectorizer(token_pattern = '(\S+)', min_df = 5, max_df = .9,
                            ngram_range = (1,2))
x_train_tfidf = tfidf_vec.fit_transform(x_train['text'])
x_test_tfidf = tfidf_vec.transform(x_test['text'])

# com a coluna tema
enc = OneHotEncoder(handle_unknown='ignore')
#bow
bow = CountVectorizer(max_features = 5000)
transf_bow = make_column_transformer((bow, 'text'),(enc, ['theme']))
x_train_bow = transf_bow.fit_transform(x_train)
x_test_bow = transf_bow.transform(x_test)
#tfidf
tfidf =  TfidfVectorizer(token_pattern = '(\S+)', min_df = 5, max_df = .9,
                         ngram_range = (1,2))
transf = make_column_transformer((tfidf, 'text'),(enc, ['theme']))
x_train_tfidf = transf.fit_transform(x_train)
x_test_tfidf = transf.transform(x_test)
```

Fonte: Autor

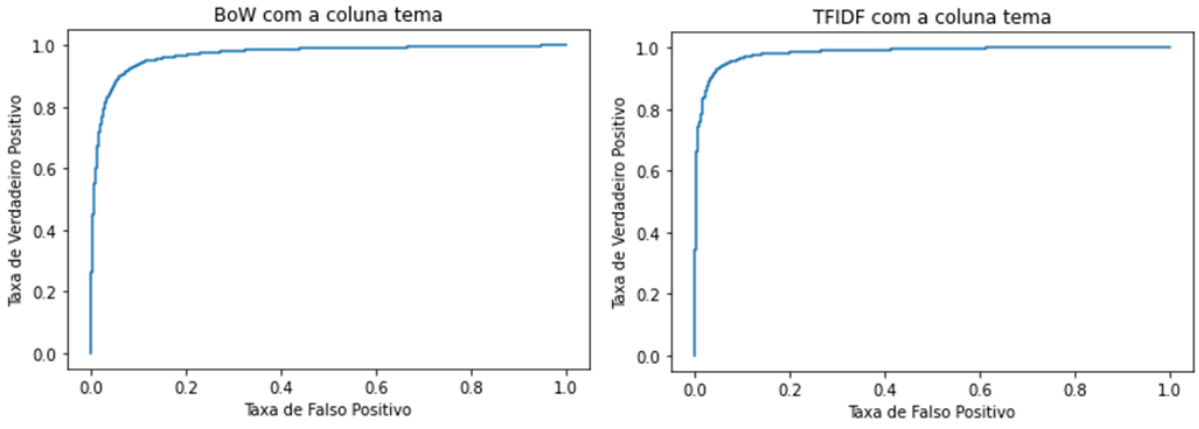
Tabela 3 – Resultados usando o modelo de Regressão Logística com as colunas texto e tema

Resultado	Bag of Words	TFIDF
Acurácia	0,92	0,94
F1	0,92	0,94
Precisão	0,89	0,92

Fonte: Autor

Além das métricas já citadas, foi criada a representação da curva ROC na qual é possível observar uma leve diferença entre BoW e TFIDF, por isso, mesmo que seja mínimo, foi registrado um melhor desempenho de TFIDF uma vez que a curva é mais próxima de 1, o gráfico da Figura 6 representa a curva ROC para os dois métodos usando a coluna texto e tema.

Figura 6 – Curva ROC para Bow e TFIDF usando a coluna texto e tema



Fonte: Autor

RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

Com o intuito de garantir os melhores resultados na predição de notícias falsas, outros modelos foram testados e com base na pontuação de treinamento e teste foi selecionado o algoritmo de Regressão Logística com base nos resultados altos e consistentes de treino e teste em relação aos outros modelos que obtiveram resultados menores ou com grande diferença entre treino e teste. A Tabela 4 contém os resultados de cada um dos modelos treinados com e sem a coluna tema.

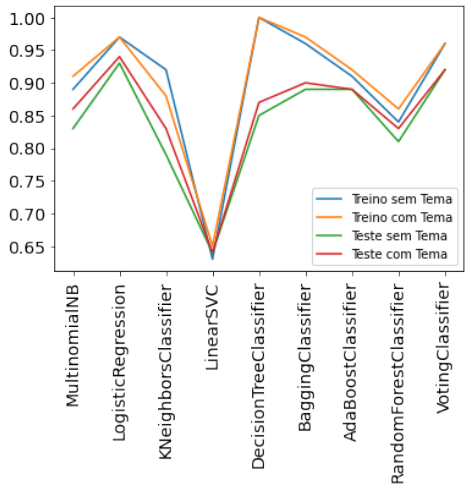
Tabela 4 – Resultados de precisão dos modelos treinados

Modelo	Resultado de treino		Resultado de teste	
	Sem a coluna tema	Com a coluna tema	Sem a coluna tema	Com a coluna tema
MultinomialNB	0,89	0,91	0,83	0,86
LogisticRegression	0,97	0,97	0,93	0,94
KNeighborsClassifier	0,92	0,88	0,79	0,83
LinearSVC	0,63	0,65	0,64	0,64
DecisionTreeClassifier	1,00	1,00	0,85	0,87
BaggingClassifier	0,96	0,97	0,89	0,90
AdaBoostClassifier	0,91	0,92	0,89	0,89
RandomForestClassifier	0,84	0,86	0,81	0,83
VotingClassifier	0,96	0,96	0,92	0,92

Fonte: Autor

A Figura 7 apresenta o resultado da Tabela 4 usando um gráfico de linhas, sendo possível observar a diferença entre os resultados dos modelos. O gráfico facilita observar quais modelos estão abaixo da pontuação referente a maioria, como LinearSVC, e quais modelos não são boas opções, como DecisionTreeClassifier que apresenta uma grande diferença entre treino e teste, além de o resultado com valor 1 para treino indicar que o modelo decorou os padrões, se encaixando no quadro de *overfitting*, e por último, modelos com bons resultados, como LogisticRegression que tem precisão alta e próxima para treinamento e teste.

Figura 7 – Resultados de precisão dos modelos treinados em gráfico



Fonte: Autor

RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

Uma última representação foi utilizada para representar os erros cometidos pelo modelo de Regressão Logística, a Tabela 5 apresenta a Matriz de Confusão obtida pela predição do conjunto de treinamento, pode ser observado que os valores são previstos incorretamente são muito baixos e um pouco maiores ao classificar notícias verdadeiras como falsas. A Tabela 6 apresenta a Matriz de Confusão do conjunto de teste que se mantem muito próxima da matriz do conjunto de treinamento em relação a distribuição dos acertos e erros.

Tabela 5 – Matriz de Confusão do conjunto de treinamento

Real x Predito	**Verdadeiras**	**Falsas**
Verdadeiras	6233	239
Falsas	209	6444

Fonte: Autor

Tabela 6 – Matriz de Confusão do conjunto de teste

Real x Predito	**Verdadeiras**	**Falsas**
Verdadeiras	2540	187
Falsas	143	2755

Fonte: Autor

5.5 *Chatbot* Telegram

Antes de tudo, para ser possível criar o *chatbot* no Telegram, foi preciso preparar o classificador para facilitar a implementação, a biblioteca em Python *Pickle*, responsável por criar um binário de um código, foi usada para salvar o classificador no estado já treinando.

O próximo passo foi criar o *chatbot* no Telegram, para isso, ao acessar o aplicativo, o *chatbot* “botfather" foi utilizado para inicializar um novo *chatbot* para uso do projeto, apenas seguindo as etapas descritas no próprio aplicativo, ele foi criado com um nome e nome de usuário, além de um link e um token de acesso que foram gerados.

Na sequência, para garantir que *chatbot* criado só recebesse informações válidas do usuário do Telegram usado no projeto, foi necessário obter o ID de usuário do Telegram, isso pode ser feito por meio do “IDBot”, outra conta no Telegram.

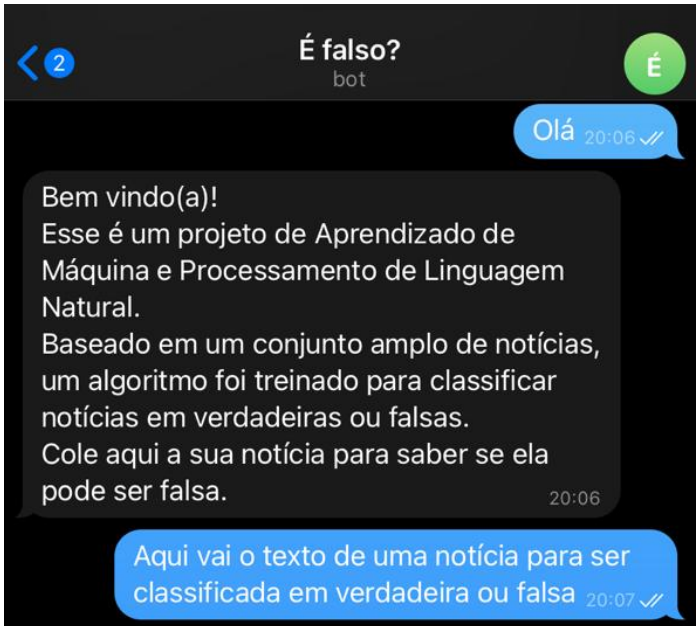
A biblioteca *telebot* de *pytelegrambotapi*, que disponibiliza de forma simples a maior das funções para iniciar um *chatbot* em Python, foi utilizada para instanciar um servidor que escuta as mensagens recebidas apenas utilizando do token criado anteriormente.

O código escrito após os passos anteriores foi para descrever como tratar cada mensagem recebida pelo *chatbot*, inicialmente é preciso receber uma mensagem com qualquer conteúdo para começar a conversa, assim uma mensagem de introdução e orientação é enviada de volta, essa mensagem requisita que uma notícia seja enviada na próxima mensagem do usuário. O código utiliza do método *register_next_step_handler* da biblioteca *telebot* para criar uma sequência de passos para a conversa com o usuário. Essa sequência de mensagens pode ser vista na Figura 8.

RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

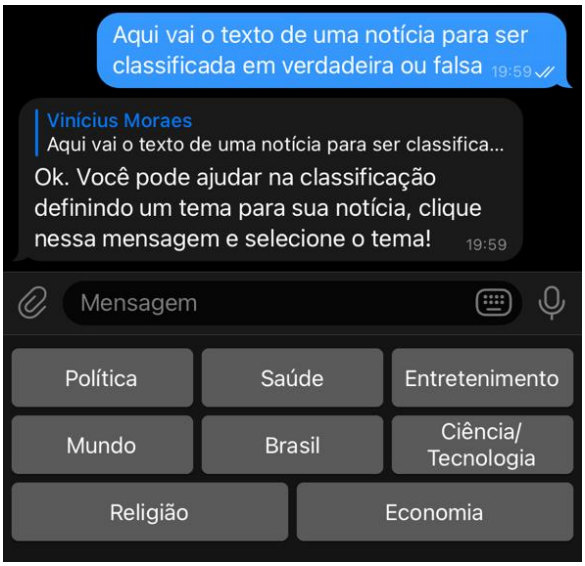
Ao receber a segunda mensagem da sequência, ela é tratada como uma notícia, logo, ela é salva em um dicionário auxiliar onde a chave é o identificador da conversa, isso é feito para manter o progresso da conversa de forma sequencial, assim a próxima etapa que é requisitar a escolha de algum dos temas disponibilizados, os mesmos presentes no conjunto, para que as duas informações necessárias estejam presentes. A escolha do tema pode ser observada na Figura 9.

Figura 8 – Início de conversa com o *chatbot*



Fonte: Autor

Figura 9 – Escolha do tema



Fonte: Autor

A implementação do código para as etapas descritas anteriormente pode ser observada na Figura 10, que apresenta as configurações iniciais e a criação das

RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

variáveis necessárias, com exceção da definição dos textos que são descritos na Figura 8, na Figura 9 e na Figura 14. Em seguida, a Figura 11 que apresenta o tratamento para as mensagens, na qual o *chatbot* responde a primeira mensagem enviada pelo usuário e salva o identificador da conversa e, com o uso da funcionalidade *markup* do *telebot* que cria as opções como botões no lugar do teclado, o processo solicita o tema para o usuário.

Figura 10 – Definições de variáveis e configurações iniciais

```
x_train = pd.read_csv(pathTreino, sep='|')
with open(path_classifier, "rb") as f:
    classifier = pickle.load(f)

API_TOKEN = ""
bot = telebot.TeleBot(API_TOKEN)

markup = types.ReplyKeyboardMarkup(one_time_keyboard=True, resize_keyboard=True)
markup.add('Política','Saúde','Entretenimento','Mundo','Brasil',
           'Ciência/Tecnologia','Religião',"Economia")

noticia_dict = {}
class Noticia:
    def __init__(self, text, msg):
        self.msg = msg
        self.text = text
        self.theme = None
```

Fonte: Autor

Assim que um tema válido é selecionado, o processamento da notícia é iniciado. A Figura 12 apresenta o tratamento inicial, começando pela validação do tema selecionado para traduzir a seleção em um identificador de tema ou requisitar novamente o tema se a seleção for inválida, em seguida o método responsável pela predição é executado, o qual pode retornar 0 ou 1, esse número é usado para devolver uma última mensagem informando qual o resultado e quais as recomendações a serem tomadas a partir de então.

Figura 11 – Tratamento para a primeira mensagem e escolha do tema

```
@bot.message_handler()
def send_welcome(message):
    try:
      msg = bot.send_message(
          message.chat.id,
          mensagem_bem_vindo,
          reply_markup=ReplyKeyboardRemove()
      )
      bot.register_next_step_handler(msg, ask_theme)
    except Exception as e:
      bot.send_message(message.chat.id, 'Desculpe, algo deu errado.')
def ask_theme(message):
    try:
      chat_id = message.chat.id
      text = message.text
      noticia = Noticia(text, message)
      noticia_dict[chat_id] = noticia
      msg = bot.reply_to(message, mensagem_tema, reply_markup=markup)
      bot.register_next_step_handler(msg, process_noticia)
    except Exception as e:
      bot.send_message(message.chat.id, 'Desculpe, algo deu errado.')
```

Fonte: Autor

RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

No método responsável por realizar a predição, o texto da notícia é pré-processado, assim como foi feito no treinamento do modelo, o mesmo é válido para a transformação das colunas em um vetor *TFIDF* para o texto e com *OneHotEncoder* para o tema. Em seguida, o classificador já treinado, que foi obtido usando o arquivo binário, é usado para rotular a notícia enviada pelo usuário, em 0 para falsa ou 1 para verdadeira, o processo pode ser visto na Figura 13.

Figura 12 – Tratamento do tema selecionado e execução da predição

```
def process_noticia(message):
    try:
      chat_id = message.chat.id
      noticia = noticia_dict[chat_id]
      if message.text == 'Política':
         noticia.theme = 'politica'
      #aqui vai o restante das validações para cada tema
      else:
          bot.reply_to(
                message,
                "Selecione um dos temas fornecidos.",
            )
          msg = bot.reply_to(message, mensagem_tema, reply_markup=markup)
          bot.register_next_step_handler(msg, process_noticia)
          return
      msg = bot.send_message(message.chat.id,"Processando a notícia...",
          reply_markup=ReplyKeyboardRemove()
      )
      resultado = predict(noticia.text, noticia.theme)
      del noticia_dict[chat_id]
      if resultado == 1:
          bot.reply_to(noticia.msg, mensagem_verdadeira)
      else:
          bot.reply_to(noticia.msg, mensagem_falsa)
    except Exception as e:
      chat_id = message.chat.id
      if noticia_dict[chat_id]:
        del noticia_dict[chat_id]
      bot.send_message(message.chat.id, 'Desculpe, algo deu errado.')
```

Fonte: Autor

Figura 13 – Processamento da notícia e predição

```
def predict(text , theme):
    text_prepared = text_prepare(text)
    enc = OneHotEncoder(handle_unknown='ignore')
    vec =  TfidfVectorizer(token_pattern = '(\S+)', min_df = 5, max_df = .9,
                           ngram_range = (1,2))
    transf = make_column_transformer((vec, 'text'),(enc, ['theme']))
    transf.fit_transform(x_train)
    to_predict= pd.DataFrame([[text_prepared, theme]],columns=['text', 'theme'])
    predict_transform = transf.transform(to_predict)
    predict_proba = classifier.predict_proba(predict_transform)[0]
    predict = classifier.predict(predict_transform)[0]
    print("predict_proba", predict_proba)
    print("predict", predict)
    return predict
```

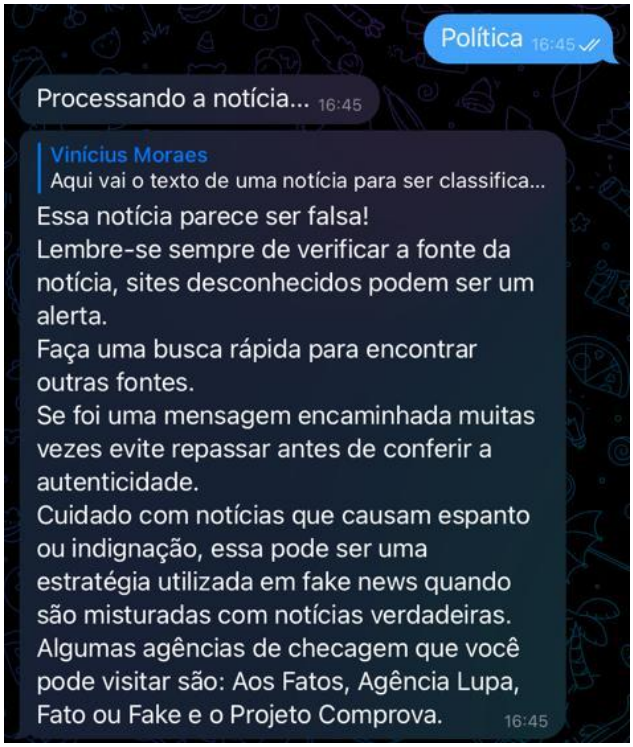
Fonte: Autor

A Figura 14 apresenta o retorno do *chatbot* quando uma notícia é identificada como falsa. No caso de a notícia ser classificada como verdadeira o texto

RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

presente na Figura 14 muda apenas o início para *Essa notícia parece ser verdadeira!*, mantendo o restante do texto com as recomendações.

Figura 14 – Classificação da notícia



Fonte: Autor

O desenvolvimento foi dado como encerrado após serem realizados testes com 20 notícias diferentes retiradas dos sites Agência Lupa, Projeto Comprova e UOL Confere, com conteúdo jornalístico de 2021 e 2022. Os resultados se mostraram próximos da precisão registrada pelo modelo com 16 acertos, ou seja, 80% de resultados foram corretos, para uma precisão melhor seriam necessários mais testes manuais. O funcionamento do *chatbot* correspondeu às expectativas com relação ao tratamento de diferentes conversas e conteúdos enviados.

6 Considerações Finais

Tendo em vista o tema das *fake news* e o combate da desinformação, o projeto visa utilizar da Inteligência Artificial, especificamente Aprendizado de Máquina e Processamento de Linguagem Natural para implementar um modelo capaz de classificar notícias em falsas ou verdadeiras. Tal afirmativa não foi alterada, porém, a implementação final do que seria responsável por consumir a capacidade de predição do modelo, que inicialmente foi pensada em ser a construção de um website para a interação de qualquer usuário, informações sobre o projeto, e a disponibilização da ferramenta para classificar qualquer notícia enviada pelo usuário, foi alterado durante o processo para a criação de um *chatbot* no aplicativo Telegram, com os mesmos objetivos.

A decisão pelo desenvolvimento do *chatbot* foi tomada pela simplicidade necessária para implementar uma integração com o classificador, de forma que não seria preciso criar uma interface do zero e programar todas as interações de usuários.

RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

O Telegram está presente em grande parte dos *smartphones* dos brasileiros, por isso é facilmente acessível e o *chatbot* poderia ter um alcance muito maior, além de criar uma opção para auxiliar na checagem de notícias falsas na palma da mão de qualquer usuário.

O desenvolvimento do projeto se deu de forma bem estruturada. O uso da ferramenta Colab permitiu um início rápido de codificação ser a preocupação de recursos local de máquinas e um ambiente configurado. As etapas foram seguidas da melhor forma, o pré-processamento foi seguido para preparar os dados de mais de 18 mil notícias, utilizando técnicas de Processamento de Linguagem Natural. Os testes dos modelos foram essenciais para compreender como cada um deles lida com o conjunto de dados e como os resultados seriam efetivos na prática, O modelo de Regressão Logística foi escolhido por ter obtido resultados altos em treino e teste, além possuir uma diferença pequena entre os dois valores. Por último, a criação do *chatbot* no Telegram foi simples comparado com o desenvolvimento que seria necessário para construir um website do zero, a interação com usuário se deu de forma básica, porém informativa e a classificação em tempo muito curto. O código produzido foi disponibilizado em um repositório público no Github (https://github.com/moraesv/chatbot-telegram-fake-news).

Os resultados do projeto excederam as expectativas, uma vez que o *chatbot* criado no Telegram se demonstrou muito mais simples de desenvolver e com uma interação amigável e instrutiva com usuário, pedindo a notícia, disponibilizando o tema para ser selecionado e entregando a classificação, assim como o website faria, porém com diversos pontos positivos a mais.

Como proposta para implementações futuras, o projeto ganharia um volume maior salvando as notícias enviadas pelos usuários em uma base de dados, onde poderia ser avaliada posteriormente e incluída no treino do modelo. Em adição, o código não está hospedado em nenhuma plataforma, ou seja, não está operante 100% do tempo, apenas quando é executado no Colab, a hospedagem do projeto para disponibilizar o *chatbot* integralmente para o público será benéfico para o combate a desinformação.

7 Referências

BLANCO, Marília Bazan et al. **O que é inteligência? Percepções de professores do ensino fundamental**, Revista Espacios, v. 38, n. 50, p. 25, 2017. Disponível em: <https://www.revistaespacios.com/a17v38n50/a17v38n50p25.pdf>. Acesso em: 20 mar. 2022.

BONFIM, Carlos A. **Machine Learning e Processamento de Linguagem Natural (PLN) com modelos lineares**, 2020. Disponível em: <https://carlos-bonfim.medium.com/machine-learning-e-processamento-de-linguagem-natural-pln-com-modelos-lineares-d5aaaaf0efa5>. Acesso em: 07 mar. 2022.

COLABORATORY, Google. **Conheça o Colab.** 2021. Disponível em: <https://colab.research.google.com/>. Acesso em: 07 mar. 2022.

COPPIN , Ben. **Inteligência Artificial**. Rio de Janeiro: LTC, 2013.

RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

ELMASRI, R.; NAVATHE, S. B.; VIEIRA D.. **SISTEMAS DE BANCO DE DADOS.** 2018, SÃO PAULO: PERSON EDUCATION DO BRASIL.

FREIRE, Paulo M.; GOLDSCHMIDT, Ronaldo R. **Uma Introdução ao Combate Automático às Fake News em Redes Sociais Virtuais**. Tópicos em Gerenciamento de Dados e Informações: Minicursos do SBBD 2019, 2019. Disponível em: <https://sol.sbc.org.br/livros/index.php/sbc/catalog/view/62/272/514-1>. Acesso em: 26 mar. 2022.

FECORMÉRCIO, Sistema. **Pesquisador dá dicas para identificar notícias falsas na internet**. 2020. Disponível em: <https://g1.globo.com/ce/ceara/especial-publicitario/sistema-fecomercio/radar-do-comercio/noticia/2020/08/25/pesquisador-da-dicas-para-identificar-noticias-falsas-na-internet.ghtml>. Acesso em: 07 set. 2022.

FURNKRANZ, Johannes; GAMBERGER, Dragan; LAVRAC, Nada. **Foundations of Rule Learning**. [S.l.]: Springer-Verlag Berlin, 2012.

GARCIA, Gabriel L.; AFONSO, Luis C. S.; PAPA, João P.. **FakeRecogna: A New Brazilian Corpus for Fake News Detection**. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), v. 13208 LNAI, p. 57-67. Disponível em: <http://hdl.handle.net/11449/234317>. Acesso em: 07 mar. 2022.

GOMES, Pedro T. **Conheça as Etapas do Pré-Processamento de dados**, 2019. Disponível em: <https://www.datageeks.com.br/pre-processamento-de-dados/>. Acesso em: 07 mar. 2022.

KLEINA, Nilton. **Telegram é mensageiro que mais cresce em uso no Brasil**. Tecmundo, 23 de fev. 2022. Disponível em: <https://www.tecmundo.com.br/redes-sociais/234361-telegram-mensageiro-cresce-uso-brasil.htm>. Acesso em: 07 set. 2022.

LUGER, George F. **Inteligência Artificial**. 6. ed. São Paulo: Pearson, 2013.

MENEZES, Nilo N. Introdução à Programação com Python. 1. ed. São Paulo: Novatec, 2010.

MAYARA, Jéssica. **Coronavírus: fake news atinge 110 milhões de brasileiros**. Estado de Minas, Minas Gerais, 21 mai. 2020. Disponível em: <https://www.em.com.br/app/noticia/bem-viver/2020/05/21/interna_bem_viver,1149424/coronavirus-fake-news-atinge-110-milhoes-de-brasileiros.shtml>. Acesso em: 26 fev. 2022.

MONARD, Maria Carolina; BARANAUSKAS, José Augusto. **Conceitos Sobre Aprendizado de Máquina**. Sistemas Inteligentes Fundamentos e Aplicações. 1 ed. Barueri-SP: Manole Ltda, 2003. Disponível em: <http://dcm.ffclrp.usp.br/~augusto/publications/2003-sistemas-inteligentes-cap4.pdf>. Acesso em: 26 mar. 2022.

RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

MONTEIRO, Rafael A.; SANTOS, Roney L.S.; PARDO, Thiago A.S.; ALMEIDA, Tiago A. de; RUIZ, Evandro E.S.; VALE, Oto A. **Contributions to the Study of Fake News in Portuguese: New Corpus and Automatic Detection Results**. Villavicencio A. et al. (eds) Computational Processing of the Portuguese Language. PROPOR 2018. Lecture Notes in Computer Science, vol 11122. Springer, Cham, 2018.

NOBERTO, Cristiane; LOIOLA, Catarina. **51% da população mundial têm acesso à internet, mostra estudo da ONU**. Correio Braziliense, 04 nov. 2019. Disponível em: <https://www.correiobraziliense.com.br/app/noticia/economia/2019/11/04/internas_economia,803503/51-da-populacao-mundial-tem-acesso-a-internet-mostra-estudo-da-onu.shtml>. Acesso em: 24 mar. 2022.

PEREIRA, Silvio do Lago. **Processamento de Linguagem Natural**. IME, Universidade de São Paulo, 2015. Disponível em: <https://www.ime.usp.br/~slago/IA-pln.pdf>. Acesso em: 09 abr. 2022.

PINHEIRO, Nina. **Introdução ao Processamento de Linguagem Natural - Natural Language Processing(NLP)**, 2021. Disponível em: <https://medium.com/data-hackers/introdu%C3%A7%C3%A3o-ao-processamento-de-linguagem-natural-natural-language-processing-nlp-be907cd06c71/>. Acesso em: 07 mar. 2022.

PORTINARI, Natália; HERNANDES, Raphael**. Fake news ganha espaço no Facebook e jornalismo profissional perde**. Folha de São Paulo, São Paulo, 08 fev. 2018. Disponível em: <https://www1.folha.uol.com.br/poder/2018/02/fake-news-ganha-espaco-no-faceboo k-e-jornalismo-profissional-perde.shtml>. Acesso em 26 fev. 2022.

TELEGRAM. **O que é Telegram? O que faço aqui?** sd. Disponível em: <https://telegram.org/faq#p-o-que-e-telegram-o-que-faco-aqui>. Acesso em: 07 set. 2022.

TIINSIDE. **Brasil concentra 42% das iniciativas de adoção de desenvolvimento de inteligência artificial na AL**, 2021. Disponível em: <https://tiinside.com.br/02/03/2021/brasil-concentra-42-das-iniciativas-de-adocao-de-desenvolvimento-de-inteligencia-artificial-na-al/>. Acesso em: 20 mar. 2022.

VENTICINQUE, Eduardo M. et al**. O uso de regressão logística para espacialização de probabilidades**. MEGADIVERSIDADE, v. 3, n. 1-2, dez. 2007. Disponível em: <https://www.conservation.org/docs/default-source/brasil/megadiversidade_v3___n1_2___dez_2007.pdf>. Acesso em: 19 jun. 2022.

VITORINO, Maíra Moraes; RENAULT, David. **A irrupção da fake news no Brasil: uma cartografia da expressão**, Universidade Metodista de São Paulo, v. 42, n. 1, 2020. Vitorino. Disponível em: <https://www.metodista.br/revistas/revistas-metodista/index.php/CSO/article/view/93 98>. Acesso em: 26 fev. 2022.

RESIGeT On-Line
Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica
Vol. 12 Nro. 2 2022
Uni-FACEF Centro Universitário Municipal de Franca

YADAV, Dinesh. **Categorical encoding using Label-Encoding and One-Hot-Encoder**. Towards Data Science, 2019. Disponível em: <https://towardsdatascience.com/categorical-encoding-using-label-encoding-and-one-hot-encoder-911ef77fb5bd>. Acesso em: 10 set. 2022.