UNIVERSIDADE DE SÃO PAULO
Instituto de Ciências Matemáticas e de Computação

Comparação de métodos de aprendizado de máquina para classificação automática de notícias em português.

Juan David Ochoa Saavedra

Monografia - MBA em Inteligência Artificial e Big Data

ICMC USP SÃO CARLOS

Juan David Ochoa Saavedra

Comparação de métodos de aprendizado de máquina para classificação automática de notícias em português.

Monografia apresentada ao Departamento de Ciências de Computação do Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo - ICMC/USP, como parte dos requisitos para obtenção do título de Especialista em Inteligência Artificial e Big Data.

Área de concentração: Inteligência Artificial

Orientador: Prof. Dr. Rafael Geraldeli Rossi

Versão original

São Carlos

2023

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi, ICMC/USP, com os dados fornecidos pelo(a) autor(a)

S856m	Saavedra, Juan David Ochoa Comparação de métodos de aprendizado de máquina para classificação automática de notícias em português. / Juan David Ochoa Saavedra ; orientador Rafael Geraldeli Rossi. – São Carlos, 2023. 61 p. : il. (algumas color.) ; 30 cm. Monografia (MBA em Inteligência Artificial e Big Data) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, 2023. 1. Classificação de textos. 2. Aprendizado de máquina. 3. Processamento de linguagem natural. 4. Representação de textos. 5. lassificação de notícias. I. ROSSI, R. G., orient. II. Título.

Juan David Ochoa Saavedra

Comparação de métodos de aprendizado de máquina para classificação automática de notícias em português.

Monograph presented to the Departamento de Ciências de Computação do Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo - ICMC/USP, as part of the requirements for obtaining the title of Specialist in Artificial Intelligence and Big Data.

Concentration area: Artificial Intelligence

Advisor: Prof. Dr. Rafael Geraldeli Rossi

Original version

São Carlos

2023

Juan David Ochoa Saavedra

Comparação de métodos de aprendizado de máquina para classificação automática de notícias em português.

Monografia apresentada ao Departamento de Ciências de Computação do Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo - ICMC/USP, como parte dos requisitos para obtenção do título de Especialista em Inteligência Artificial e Big Data.

Data de defesa: 07 de outubro de 2023

Comissão Julgadora:

Prof. Dr. Rafael Geraldeli Rossi
Orientador

Professor
Convidado

Professor
Convidado

São Carlos
2023

Este trabalho é dedicado a minha família e seres queridos
como uma amostra da responsabilidade e necessidade de aprimoramento
para termos todos um futuro melhor.

AGRADECIMENTOS

À Deus por trilhar o meu caminho e colocar pessoas excelentes e especiais perto;

aos meus pais e irmã que me apoiaram e incentivaram por me inscrever em este curso e a não desistir sem importar as dificuldades;

à minha namorada que ajudou e compreendeu no tempo de realização integral do curso;

ao meu orientador Prof. Dr. Rafael Geraldeli Rossipelo vasto apoio e explicações durante a realização deste trabalho de conclusão de curso, por me abrir caminhos e ajudar no melhor entendimento das técnicas, por todas as indicação de correção que fizeram melhorar a qualidade do trabalho.

RESUMO

SAAVEDRA, J. D. O. **Comparação de métodos de aprendizado de máquina para classificação automática de notícias em português.** . 2023. 61p. Monografia (MBA em Inteligência Artificial e Big Data) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2023.

A categorização de notícias de acordo com seus temas é uma tarefa crucial para organizar e gerenciar os fluxos informacionais crescentes característicos da era digital. No entanto, a magnitude da produção jornalística diária inviabiliza a classificação manual, tornando imperativo o desenvolvimento de sistemas automatizados para essa função. Neste contexto, este trabalho realizou uma investigação abrangente de diferentes métodos de representação textual e modelos de aprendizado de máquina para a classificação automática do tema de notícias em português. Inicialmente, foi desenvolvido um coletor personalizado que recuperou mais de 18 mil notícias atualizadas de portais brasileiros, as quais foram rotuladas manualmente em 19 classes temáticas distintas. Foram exploradas diversas técnicas de pré-processamento e representação textual, incluindo *Word2Vec*, *Doc2Vec*, *Bag of Words com TF-IDF* e *fine-tuning* de um modelo *Bidirecional Encoder Representations from Transformers (BERT)* pré-treinado. Para treinamento e avaliação, algoritmos como regressão logística e *support vector machine* foram empregados. O modelo BERT obteve o melhor desempenho, com 93% de acurácia e 0,98 de F1 *weighted*, superando significativamente as representações tradicionais testadas. No entanto, combinações como *Bag of words + support vector machine* e Doc2Vec + regressão logística também apresentaram métricas satisfatórias de acurácia e *F1-score*, emergindo como alternativas interessantes em termos de custo-benefício computacional. Os resultados fornecem *insights* abrangentes sobre abordagens e representações textuais eficazes para categorização automatizada do tema de notícias em português.

Palavras-chave: classificação de textos, aprendizado de máquina, processamento de linguagem natural, representação de textos, classificação de notícias.

ABSTRACT

SAAVEDRA, J. D. O. **Comparação de métodos de aprendizado de máquina para classificação automática de notícias em português.** . 2023. 61p. Monograph (MBA in Artificial Intelligence and Big Data) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2023.

The categorization of news according to their themes is a pivotal task for organizing and managing the increasing information flows characteristic of the digital age. However, the sheer volume of daily journalistic output renders manual classification impractical, making it imperative to develop automated systems for this purpose. In this context, this study conducted a comprehensive investigation of various text representation methods and machine learning models for the automatic classification of news themes in Portuguese. Initially, a custom collector was developed that retrieved over 18,000 updated news from Brazilian portals, which were manually labeled into 19 distinct thematic classes. Several preprocessing and text representation techniques were explored, including Word2Vec, Doc2Vec, Bag of Words with TF-IDF, and fine-tuning of a pretrained Bidirectional Encoder Representations from Transformers (BERT) model. For training and evaluation, algorithms such as logistic regression and support vector machine were employed. The BERT model achieved the best performance, with 93% accuracy and 0.98 F1 weighted, significantly outperforming the tested traditional representations. However, combinations like Bag of words + support vector machine and Doc2Vec + logistic regression also yielded satisfactory accuracy and F1-score metrics, emerging as interesting alternatives in terms of computational cost-benefit. The findings provide comprehensive insights on effective approaches and text representations for automated categorization of news themes in Portuguese.

Keywords:text classification, machine learning, natural language processing, text representation, news classification.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

BERT	Bidirectional Encoder Representations from Transformers
BoW	Bag of Words
CRISP-DM	CRoss Industry Standard Process for Data Mining
CBOW	Continuous Bag of Words
DBOW	Distributed Bag of Words
DM	Distributed Memory
GloVe	Global Vectors for Word Representation
LLMs	Large Language Models
MLP	Multilayer Perceptron
NLP	Natural Language Processing (Processamento de Linguagem Natural)
RF	Random Forest
SVM	Support Vector Machine
TF-IDF	Term Frequency–Inverse Document Frequency

SUMÁRIO

1 INTRODUÇÃO

As notícias têm permeado a experiência humana por eras, e é desafiador precisar o início deste fenômeno. Elas abordam uma variedade ampla de tópicos, como política, economia, esporte e cultura, entre outros (STRÖMBÄCK; KARLSSON; HOPMANN, 2015). Com o advento da internet, observou-se um crescimento exponencial na criação, publicação e divulgação de notícias nos últimos anos (LüDTKE, 2023). Este aumento impulsionou um crescimento significativo nas aplicações e pesquisas que utilizam conteúdo noticioso para diversas finalidades, desde recomendar notícias similares (KE, 2020), passando por sensoriamentos para extração de conhecimento e previsões (MARCACINI *et al.*, 2017) (MARCACINI; CARNEVALI; DOMINGOS, 2016), até a detecção de notícias falsas (SOUZA *et al.*, 2022).

Dada a magnitude da produção noticiosa diária, torna-se imperativo o desenvolvimento de técnicas automáticas para organizar, gerenciar e extrair *insights* desses conteúdos. Neste contexto, técnicas de classificação são apontadas como essenciais para a categorização e organização deste vasto volume de informações (SEBASTIANI, 2002). Entretanto, em virtude das peculiaridades linguísticas e contextuais de cada notícia (JURAFSKY; MARTIN, 2019), a adoção de uma abordagem genérica para essa tarefa se mostra inviável.

Neste trabalho, é apresentado um método de classificação automática de notícias por temas, empregando técnicas avançadas de aprendizado de máquina e *NLP*. A classificação manual, em contraste, não só é onerosa, mas também enfrenta desafios tanto de natureza temporal quanto financeira (LEE, 2023). As técnicas de *NLP* desempenham um papel crucial na análise e classificação de textos em larga escala, e é fundamental salientar que este estudo não se restringe apenas ao *NLP*, mas também se estende à organização do crescente volume de notícias na era digital (KNAPP, 2023).

A estratégia proposta para a classificação automática de notícias compreende duas etapas fundamentais: inicialmente, visa-se construir uma representação estruturada dos textos, seguida pela aplicação de algoritmos específicos para aprendizado ou extração de padrões (AGGARWAL, 2022). O êxito desta abordagem depende intrinsecamente da escolha acertada tanto da representação do texto quanto do algoritmo, considerando variáveis como o propósito do estudo, volume de dados e especificidades linguísticas (PATIL *et al.*, 2023).

Grandes corporações, como o Google, já estão capitalizando sobre algoritmos avançados para categorizar notícias de acordo com as preferências dos usuários (MORRIS, 2022). Adicionalmente, conforme destacado por Chui *et al.* (2023), a influência da inteligência artificial nas operações diárias de diversas empresas está em crescimento acentuado.

Em resumo, diante do cenário atual de profusão informacional no ambiente digital, sistemas de classificação eficazes são indispensáveis. Este trabalho delineia uma metodologia robusta para a classificação automática de notícias, dando devida atenção às suas nuances linguísticas e contextuais.

1.1 Objetivo e Questões de Pesquisa

Este trabalho de conclusão de curso tem como principal intuito empregar algoritmos de aprendizado de máquina na na classificação de temas de notícias considerando apenas seus títulos e resumos de notícias com base nos temas neles abordados. Dados os passos principais para se executar uma classificação de notícias e os desafios apresentados na seção anterior, este trabalho visa responder a seguinte questão de pesquisa.

Q1: *"Quais são as melhores técnicas para a representação estruturada de notícias visando a classificação de textos jornalísticos escritos em português?"*

Q2: *"Dentre os algoritmos de aprendizado de máquina disponíveis, quais são os mais eficazes para a construção de modelos de classificação de notícias em português?"*

Já os objetivos específicos deste trabalho de conclusão de curso incluem:

- Coletar conjuntos de notícias recentes, escritas em português e devidamente rotuladas por meio de um *crawler* dedicado à coleta contínua de notícias atuais.
- Identificar, mapear e implementar técnicas de pré-processamento e aprendizado de máquina adequadas para textos pertencentes a múltiplas categorias, com base em modelos e abordagens citados na literatura.
- Avaliar e comparar a eficácia das técnicas adotadas, considerando tanto a performance de classificação dos algoritmos quanto o custo computacional associado a cada um deles.

1.2 Organização do Texto

O retante deste trabalho está estruturado em cinco capítulos:

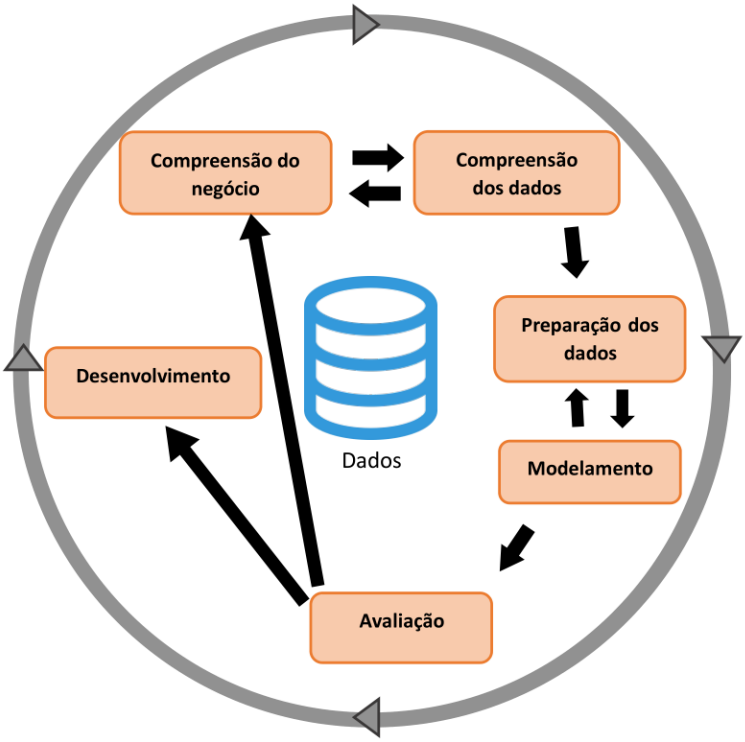
- No capitulo 1 foi apresentada a introdução e a motivação para o estudo.
- No Capítulo 2, são apresentados os fundamentos teóricos e os trabalhos relacionados.
- No Capítulo 3 é apresentado diferentes trabalhos relacionados ao tema deste documento.
- No Capítulo 4 é descrita a metodologia e as técnicas propostas pelo trabalho, bem como o processo de desenvolvimento do experimento.

- No Capítulo 5 apresenta os resultados das técnicas implementadas e a análise das métricas de avaliação estabelecidas.
- Por fim, no Capítulo 6 são apresentadas as conclusões, discutindo quais foram os avanços e limitações do trabalho e também as sugestões para trabalhos futuros.

2 CONCEITOS E TRABALHOS RELACIONADOS

Para a execução da classificação automática, normalmente adota-se um processo de mineração de dados ou textos (CASTRO; FERRARI, 2017). A mineração de dados e textos é um campo vasto e em constante evolução. Para garantir que os projetos nessa área sejam conduzidos de maneira eficiente e sistemática, é vital adotar um *framework* ou metodologia bem estabelecida. Dentre as várias abordagens disponíveis, o CRISP-DM (*CRoss Industry Standard Process for Data Mining*) destaca-se como um dos *framework* mais amplamente reconhecidos e utilizados (SHEARER, 2000). Originado no final dos anos 90 e consolidado como referência no início dos anos 2000, o CRISP-DM oferece uma abordagem padronizada para projetos de mineração de dados e textos. Ele foi projetado para ser aplicável a diversas indústrias e contextos tecnológicos, permitindo a produção de modelos mais confiáveis, econômicos, e facilmente reprodutíveis. Além disso, a estrutura do CRISP-DM promove a gerenciabilidade dos projetos e acelera o processo de desenvolvimento.

Figura 1 – Modelo CRISP-DM. baseado em: (HOTZ, 2023)



Na Figura 1, é apresentada a ilustração dos passos do *framework* CRISP-DM. Pode-se observar que há seis passos e que existem interações entre eles. Se o resultado verificado em um determinado passo não for satisfatório, pode-se retroceder a passos anteriores. Também se nota a proposta de um ciclo entre os passos, indicando que o processo pode ser usado continuamente. Isso significa que é possível incorporar novos

dados, gerar novos modelos ou alterar o objetivo do processo conforme as necessidades do negócio ou aplicação.

Nas próximas sentenças são apresentados os detalhes de cada etapa do *framework* CRISP-DM.

2.1 Identificação do Problema

A compreensão clara dos objetivos de negócio e das necessidades do cliente é a pedra angular de qualquer projeto de ciência de dados. Esta etapa, conforme delineado pelo CRISP-DM, enfatiza a importância de uma visão holística do contexto empresarial, além do mero entendimento técnico (DAVENPORT; HARRIS, 2013).

A colaboração entre cientistas de dados e especialistas de domínio é vital para assegurar que as soluções sejam não apenas tecnicamente válidas, mas também aplicáveis e relevantes (PROVOST; FAWCETT, 2013). Além de definir os objetivos empresariais e técnicos, é crucial avaliar a situação atual, considerando a disponibilidade de recursos, requisitos do projeto e análise de custo-benefício.

Um plano de projeto detalhado é elaborado, selecionando as tecnologias e ferramentas mais adequadas, para guiar as fases subsequentes. A importância desta etapa é comparável à construção da fundação de uma casa: é um passo crítico que não deve ser negligenciado.

2.2 Compreensão dos Dados

No contexto do processo *CRISP-DM*, a compreensão dos dados emerge como uma etapa de suma importância. Esta fase se desdobra em quatro subfases distintas: compreensão dos dados; coleta inicial dos dados; descrição dos dados; e exploração dos dados. O propósito central é alcançar um entendimento aprofundado da natureza intrínseca dos dados, abrangendo a identificação de eventuais problemas de qualidade, a extração de informações iniciais e a detecção de subconjuntos que possam servir como base para a formulação de hipóteses (CONSORTIUM, 2000).

A fase de coleta inicial dos dados representa o primeiro contato direto com o conjunto de dados, onde estes são introduzidos em uma ferramenta de análise apropriada. Durante esta etapa, podem surgir diversas dificuldades, tais como inconsistências nas fontes de dados, problemas na importação devido a formatos incompatíveis, e desafios relacionados à integridade e completude dos dados. A verificação da correta importação é essencial, englobando a análise de possíveis valores ausentes e a corroboração dos tipos de dados (PYLE, 1999).

Subsequentemente, na etapa de descrição dos dados, procede-se com a análise da distribuição de variáveis-chave, a determinação da dimensão do *dataset* e a caracterização

da natureza dos dados, como a identificação de séries temporais, por exemplo (TAN *et al.*, 2018).

A fase subsequente, de exploração dos dados, envolve a aplicação meticulosa de técnicas de visualização e estatísticas descritivas, visando decifrar a estrutura e os padrões subjacentes nos dados. Neste contexto, é crucial identificar possíveis relacionamentos entre as variáveis. Instrumentos analíticos, como gráficos de dispersão e histogramas, juntamente com métricas estatísticas, como média, mediana e desvio padrão, são de inestimável valor (TUKEY, 1977).

Finalmente, a etapa de verificação da qualidade dos dados é de primordial importância. Neste estágio, são identificadas e tratadas questões como a presença de valores ausentes, *outliers* e inconsistências nos dados. As estratégias para abordar tais problemas são definidas, podendo abranger desde a imputação de dados faltantes até a remoção de *outliers* (PIPINO; LEE; WANG, 2002).

Ao término da fase de compreensão dos dados, espera-se possuir uma visão elucidativa da configuração e das características dos dados em questão, bem como dos desafios de qualidade identificados e das estratégias adotadas para sua resolução. De acordo com determinadas características ou problemas levantados nos dados, determinadas técnicas de preparação de dados deverão ser utilizadas, conforme apresentado na próxima seção.

2.3 Preparação dos dados

O ditado "lixo entra, lixo sai" ressoa fortemente na mineração / ciência de dados (DHAR, 2013). Uma preparação meticulosa dos dados é a espinha dorsal de qualquer projeto de mineração de dados bem-sucedido, garantindo que os modelos gerados sejam não apenas precisos, mas verdadeiramente úteis e informativos (WITTEN *et al.*, 2016). Ao longo desta fase, esforços consideráveis são dedicados à correção de inconsistências, ao tratamento de valores ausentes, à codificação de variáveis categóricas e, talvez mais crucialmente, à criação e seleção de atributos ou "*features*" que serão a matéria-prima para os modelos.

Esta etapa é eminentemente iterativa. Não é incomum para uma equipe de ciência de dados retornar à preparação após a modelagem inicial, refinando e ajustando conforme novos desafios ou *insights* surgem. Além dessas atividades convencionais de limpeza e transformação, a representação correta dos dados é fundamental, especialmente quando estamos lidando com dados textuais.

Após as etapas iniciais de limpeza e transformação, uma análise cuidadosa da distribuição dos dados é essencial. Em muitos conjuntos de dados, especialmente aqueles relacionados a classificação, é comum encontrar um desbalanceamento entre as classes. Esse desbalanceamento pode levar a modelos tendenciosos, que favorecem a classe majoritária,

resultando em desempenho subótimo na classe minoritária. Para abordar esse desafio, técnicas de reamostragem, como o *SMOTE* (*Synthetic Minority Over-sampling Technique*), são frequentemente empregadas. O *SMOTE* gera exemplos sintéticos da classe minoritária, equilibrando assim a distribuição das classes e permitindo que o modelo de aprendizado de máquina seja treinado em um conjunto de dados mais balanceado. A relevância do balanceamento não pode ser subestimada, pois garante que o modelo seja treinado de forma justa em todas as classes, melhorando a generalização e a robustez do modelo em cenários do mundo real (CHAWLA *et al.*, 2002).

Alguns tipos de dados já vêm em um formato estruturado, geralmente representados por uma matriz onde as colunas indicam os atributos e as linhas representam os valores desses atributos para cada exemplo ou registro do conjunto de dados. No entanto, há tipos de dados, como imagens, textos ou músicas, que são considerados dados não estruturados. Para esses, é imprescindível gerar uma representação estruturada, que frequentemente assume o formato de matriz atributo-valor (AGGARWAL, 2015). E é neste contexto que abordagens como *Bag-of-Words* e *Embeddings* entram em cena. Estas técnicas, que serão discutidas em detalhes nas próximas seções, oferecem mecanismos robustos para converter informações textuais em formatos que modelos de *aprendizado de máquina* podem entender e processar, i.e, vetores de características que formarão uma matriz atributo-valor. Cada abordagem tem suas próprias vantagens e desafios, e a escolha entre elas pode ser crucial para o sucesso do projeto.

2.3.1 *Bag-of-words*

Ao trabalhar com dados em formato de texto, uma série de etapas de pré-processamento se fazem necessárias para transformá-los em uma forma que possa ser facilmente interpretada e utilizada por modelos de aprendizado de máquina. Uma das técnicas mais tradicionais para essa transformação é a abordagem *bag-of-words* (SALTON, 1989). Neste método, o texto é convertido em uma representação vetorial, onde cada dimensão do vetor corresponde a uma palavra do vocabulário, e o valor em cada dimensão indica a presença (ou frequência) dessa palavra no texto. O resultado é uma representação esparsa e multidimensional.

Apesar de sua simplicidade e ampla aplicação, a técnica *bag-of-words* tem limitações inerentes. Como observado por Aggarwal (2022), esta representação frequentemente se mostra insuficiente para tarefas mais complexas, de classificação de textos, modelagem de tópicos e sistemas de recomendação. O principal desafio é que a abordagem *bag-of-words* desconsidera a ordem e o contexto das palavras no texto. Isso pode levar a perdas significativas de informação, especialmente quando a semântica e a nuance são críticas para a interpretação correta.

Na Figura 2 é apresentada uma ilustração da representação *bag-of-words* para 4

textos: i) “O Cão vermelho” , ii) “Gato come cachorro”, iii) “cão come comida”, e iv) “Gato vermelho come”. Para fins de exemplo, atribuiu-se o peso 1 se a palavra ocorreu no documento e 0 se não ocorreu, ou seja, é um peso binário.

Figura 2 – Exemplo representação com *bag-of-words*. baseado em: (MUKERJEE, 2021)

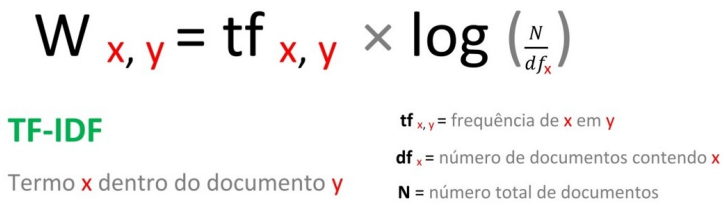
O	Vermelho	Cão	Gato	Come	Comida
1	1	1	0	0	0
0	0	1	1	1	0
0	0	1	0	1	1
0	1	0	1	1	0

Além do peso binário, outros esquemas de ponderação podem ser empregados para atribuir pesos às palavras presentes no documento. Dois esquemas comuns são o *Term-Frequency* (TF) e o *Term Frequency-Inverse Document Frequency* (TF-IDF). Este último busca superar algumas das limitações da representação *bag-of-words*. A técnica TF-IDF destaca a relevância de uma palavra em um documento, ponderando-a contra sua frequência em um conjunto geral de documentos, ou corpus (SALTON; BUCKLEY, 1988).

Antes de calcular o TF-IDF, é comum realizar algumas etapas de pré-processamento nos textos. Uma dessas etapas é a simplificação de termos, que envolve a redução de palavras ao seu radical ou base, usando técnicas como *stemming* ou *lemmatization*. Outra etapa crucial é a remoção de *stopwords*, que são palavras comuns, como preposições e artigos, que geralmente não agregam valor significativo à análise e podem distorcer os resultados se incluídas. A remoção dessas palavras e a simplificação de termos ajudam a focar nas palavras mais relevantes e a reduzir a dimensionalidade dos dados (SCHÜTZE; MANNING; RAGHAVAN, 2008).

O TF-IDF é calculado usando duas métricas principais demostradas na imagem 3:

Figura 3 – Formula para calculo de TF-IDF. Baseado em (IL,)



Esta métrica é útil porque palavras que aparecem frequentemente em um documento, mas não em muitos documentos do corpus, receberão um peso TF-IDF alto. Isso destaca palavras que são importantes ou distintas para um documento específico (RAMOS, 2003).

O TF-IDF é amplamente aplicado em tarefas de processamento de linguagem natural, como:

- *Recuperação de Informações*: Para melhorar a relevância de documentos retornados em sistemas de busca (MANNING; RAGHAVAN; SCHÜTZE, 2008).
- *Extração de Palavras-chave*: Identificar palavras ou frases significativas em um documento (HULTH, 2003).
- *Agrupamento e Classificação de Documentos*: Representando documentos como vetores TF-IDF, facilitando técnicas de aprendizado de máquina (SEBASTIANI, 2002).

Apesar de sua eficácia em muitas tarefas, o TF-IDF tem suas limitações. Ele não considera a ordem ou a estrutura semântica das palavras em um documento, o que pode ser crucial para algumas análises (TURNEY; PANTEL, 2010).

2.3.2 *Embeddings*

Embeddings são uma classe de técnicas onde palavras, sentenças, parágrafos ou textos completos são mapeadas para vetores de números reais. Estas representações podem capturar muitas propriedades linguísticas, como relações semânticas entre as palavras ou semelhanças entre textos. Os *embeddings* transformam o texto que seria representado em formato esparso, como na *bag-of-words* em vetores densos, permitindo assim que modelos computacionais possam efetivamente aprender a partir dessas representações. Nas próximas seções são apresentadas as principais técnicas baseadas em *embeddings* para a representação de textos.

2.3.2.1 *Word Embeddings*

Word Embeddings referem-se a técnicas que aprendem representações vetoriais de palavras de um texto (WANG; ZHOU; JIANG, 2020). Ao invés de tratar palavras como unidades discretas e isoladas, os *embeddings* permitem capturar a semântica e relações contextuais entre palavras. Este é um avanço significativo em relação ao *bag-of-words*, pois as palavras não são apenas tratadas com base em sua frequência, mas também em seu contexto de uso. Além disso, as *word embeddings* servem como base para outras representações de textos que também se fundamentam em *embeddings*.

O mais famoso e talvez o primeiro grande avanço nessa direção foi o modelo *Word2Vec* proposto por Mikolov *et al.* (2013). O *Word2Vec* usa redes neurais para aprender representações vetoriais das palavras e é capaz de capturar relações semânticas e sintáticas. Existem duas arquiteturas principais no *Word2Vec*: *CBOW* (*Continuous Bag of Words*) e *Skip-gram*.

O *CBOW* prevê palavras-alvo (como “Rei”) a partir de palavras de contexto circundantes (como “coroa”, “reino”). Em contraste, o *Skip-gram* faz o oposto: ele usa a

palavra-alvo para prever ou contextualizar as palavras circundantes. Em termos práticos, o *Skip-gram* tende a funcionar melhor para conjuntos de dados maiores e representa bem palavras menos frequentes, enquanto o *CBOW* é mais rápido e tem melhor desempenho para palavras mais comuns.

Por exemplo, utilizando *Word Embeddings* treinados com *Word2Vec*, é possível observar relações como: “Rei” - “Homem” + “Mulher” se aproxima de “Rainha”.

O *GloVe* (Global Vectors for Word Representation) é outra técnica popular para obtenção de *Word Embeddings*, proposta por Pennington, Socher and Manning (2014). Ao contrário do *Word2Vec*, que utiliza redes neurais para gerar representações das palavras com base na localidade das palavras em um texto, o *GloVe* constrói suas representações vetoriais aproveitando a informação estatística global do corpus e fazendo uso de fatoração de matrizes.

Já a abordagem FastText, proposto por Bojanowski *et al.* (2016), é uma extensão do Word2Vec que leva em consideração subpalavras. Isso significa que cada palavra é representada como um conjunto de *n-gramas* de caracteres. Esta característica permite ao FastText gerar *embeddings* de melhor qualidade para palavras menos frequentes e também permite gerar *embeddings* para palavras fora do vocabulário.

A principal vantagem de se utilizar *Word Embeddings* é que eles permitem que modelos de aprendizado de máquina trabalhem com entradas textuais de uma forma que capte nuances semânticas e contextuais, facilitando tarefas como classificação de textos, análise de sentimentos e detecção de tópicos (GE; MOH, 2017). Uma vez obtidos os *embeddings* de palavras, uma abordagem comum para gerar representações para um texto é através da combinação ou operações nos *embeddings* das palavras que compõem o texto, como soma, média, soma ponderada, mínimo ou máximo (PITA; PAPPA, 2018). No entanto, essas operações perdem a ordem das palavras da mesma forma que os modelos *bag-of-words*. Uma alternativa foi proposta em (LE; MIKOLOV, 2014), na qual os autores sugerem uma abordagem baseada em *word2vec* para gerar *embeddings* de parágrafos ou documentos chamados *Paragraph Vectors* (ou *doc2vec*). O uso de “parágrafo” refere-se a grandes trechos de textos, como frases, parágrafos e documentos.

No entanto, a representação ainda possui suas limitações, como a dificuldade de lidar com palavras polissêmicas (palavras com múltiplos significados, ou seja, cujo significado varia de acordo com o contexto das palavras). Independentemente disso, os *Word Embeddings* marcaram uma revolução na forma como se lida com textos em modelos computacionais.

2.3.2.2 *Doc2vec*

O *Doc2vec*, também conhecido como "*Paragraph Vector*", representa uma evolução do conceito de *Word Embeddings*. Com o objetivo de possibilitar a representação de sentenças, parágrafos ou textos inteiros sem a necessidade de realizar uma operação vetorial considerando os vetores das palavras Le and Mikolov (2014).

Portanto, o principal desafio abordado pelo *Doc2vec* é a necessidade de preservar a semântica de blocos maiores de texto, considerando que eles possuem mais variabilidade e complexidade do que palavras individuais. Para resolver isso, o *Doc2vec* introduz um vetor especial para cada documento no modelo, que é treinado em conjunto com os vetores de palavras, de forma que o vetor do documento e os vetores de palavras vizinhas possam prever palavras no contexto.

Existem duas principais variantes do algoritmo *Doc2vec*:

Distributed Memory (DM): Esta versão utiliza os vetores do parágrafo e das palavras como contexto para prever a próxima palavra em uma sequência. A estrutura é semelhante a um modelo *skip-gram*, mas com vetores de parágrafos adicionados ao contexto.

Distributed Bag of Words (DBOW): Em vez de prever palavras a partir de contextos, a DBOW ignora o contexto e força o modelo a prever palavras a partir do vetor do parágrafo/documento. É análogo ao modelo *CBOW* do Word2Vec.

Uma característica importante do *Doc2vec* é a capacidade de inferir vetores para documentos novos, nunca antes vistos durante o treinamento. Isso significa que, após treinar um modelo *Doc2vec*, é possível obter representações vetoriais para documentos totalmente novos, simplesmente passando-os pelo modelo treinado.

Os vetores resultantes de documentos são úteis em diversas tarefas de processamento de linguagem natural, como classificação de texto, agrupamento e sistemas de recomendação. Eles fornecem uma representação condensada do conteúdo do documento, capturando a semântica e as nuances do texto de maneira mais eficaz do que as representações tradicionais, como *bag-of-words*.

2.3.2.3 *Large Language Models*

Os *Large Language Models* (LLMs) representam uma fronteira avançada no campo do processamento de linguagem natural (NLP). Estes modelos, que são capazes de representar palavras, textos ou documentos por meio de *embeddings*, são caracterizados pelo seu tamanho colossal e pela sua capacidade de compreender e gerar texto em níveis que se assemelham, em muitos casos, à proficiência humana (BROWN *et al.*, 2020). Treinados em vastos conjuntos de dados, considerando a sequência de palavras, muitas vezes em ambos os sentidos, e utilizando arquiteturas de redes neurais capazes de selecionar e descobrir relações importantes entre as palavras, como os *transformers*, os LLMs conseguem cap-

turar nuances linguísticas, padrões e relações semânticas com qualidades superiores em comparação aos modelos baseados em *embeddings* apresentados nas seções anteriores.

A ascensão dos LLMs, exemplificada por modelos como GPT-3 (*Generative Pre-trained Transformer*) (BROWN *et al.*, 2020), ELMo (*Embeddings from Language Model*) (SARZYNSKA-WAWER *et al.*, 2021), BERT do Google (DEVLIN *et al.*, 2018a), e outros mais recentes como GPT-4, PALM2 (CHOWDHERY *et al.*, 2023), LLaMA (RAMA *et al.*, 2023) e Falcon (ALMAZROUEI *et al.*, 2023), sinaliza uma transformação na maneira como abordamos tarefas de NLP (PEREIRA, 2023). Com o potencial de revolucionar aplicações que vão desde chatbots até análise de sentimentos, eles estão redefinindo o que é possível no campo do processamento de linguagem natural.

Contudo, apesar de suas habilidades impressionantes, os LLMs também vêm com desafios. Seu tamanho e complexidade demandam recursos computacionais significativos, tanto para treinamento quanto para inferência, além do tempo de resposta que pode ser acima do limiar desejável para aplicações *real-time*. Ademais, por serem "caixas-pretas", nem sempre é fácil entender por que eles tomam certas decisões ou geram determinadas respostas, levantando preocupações sobre sua transparência e responsabilidade em aplicações críticas (COZMAN *et al.*, 2021).

2.4 Modelagem

A criação de modelos é a essência da mineração de dados. Aqui, o conjunto de dados preparado é usado para treinar vários modelos usando diferentes técnicas e algoritmos. Desde árvores de decisão a redes neurais, a seleção do algoritmo depende da natureza do problema e dos dados. Além disso, esta fase envolve ajustar *hiperparâmetros*, validar os modelos e comparar diferentes abordagens. Uma abordagem comum é dividir o conjunto de dados em treinamento, validação e teste para garantir que os modelos sejam robustos e evitem o *overfitting*.

2.4.1 Regressão Logística:

A Regressão Logística é um modelo estatístico usado para prever a probabilidade de uma instância pertencer a uma categoria particular, com base em uma ou mais variáveis independentes. É amplamente utilizado em problemas de classificação binária (JR; LEMESHOW; STURDIVANT, 2013).

2.4.2 *Support Vector Machine* (SVM):

O SVM é um modelo robusto que busca encontrar um hiperplano em um espaço N-dimensional que classifica os dados. É especialmente útil em situações onde os dados não são linearmente separáveis (CORTES; VAPNIK, 1995).

2.4.3 Fine-tuning do modelo BERT:

O BERT (*Bidirectional Encoder Representations for Transformers*) é uma arquitetura de modelo de linguagem natural que aprende representações de texto a partir de grandes quantidades de dados (DEVLIN *et al.*, 2018b). O modelo *neuralmind/bert-large-portuguese-cased* é uma versão pré-treinada para o português e foi utilizado em conjunto com uma rede neural multicamada, que é uma rede neural artificial com mais de uma camada entre a entrada e a saída, para classificação multiclasse (GOODFELLOW; BENGIO; COURVILLE, 2016).

2.5 Avaliação

A fase de avaliação é crítica para determinar a eficácia e adequação de um modelo em cenários do mundo real (PROVOST; FAWCETT, 2013). O principal objetivo aqui é avaliar o desempenho do modelo em dados que ele nunca viu, para garantir que ele possa generalizar bem e não apenas memorizar os dados de treinamento, um fenômeno conhecido como *overfitting* (HAWKINS, 2004).

Diversas métricas podem ser utilizadas para avaliar modelos de mineração de dados, e a escolha da métrica correta depende do tipo de problema e dos objetivos de negócio. Para classificação, métricas como precisão, revocação, área sob a curva *ROC* e *F1-score* são comumente usadas (FAWCETT, 2006). Para regressão, medidas como erro quadrático médio (*MSE*) e erro absoluto médio (*MAE*) são padrões na indústria (HYNDMAN; KOEHLER, 2006).

Além das métricas padrão, é crucial considerar fatores como a interpretabilidade do modelo, a ética das previsões (evitando viés e discriminação) e a relevância prática dos resultados (DOSHI-VELEZ; KIM, 2017). Por exemplo, um modelo altamente preciso que não pode ser interpretado por especialistas do domínio pode ter menos valor do que um modelo menos preciso, mas mais compreensível.

2.6 Fase de implantação

Uma vez que um modelo foi desenvolvido e avaliado, a fase de implantação começa, marcando a transição do ambiente de desenvolvimento para o ambiente de produção (SCULLEY *et al.*, 2015). Esta fase envolve várias etapas críticas, incluindo a integração do modelo em sistemas operacionais existentes, a configuração de *pipelines* de dados para alimentar o modelo e a criação de interfaces para que os usuários finais possam interagir com o modelo (ZAHARIA *et al.*, 2018).

Um desafio significativo na fase de implantação é garantir que o modelo continue performando bem quando exposto a novos dados, já que o mundo real é dinâmico e os padrões nos dados podem mudar ao longo do tempo (WANG *et al.*, 2020). O monitoramento

contínuo do desempenho do modelo, a reavaliação periódica e, se necessário, a recalibração do modelo são essenciais para garantir sua relevância e eficácia contínuas.

A fase de implantação também requer uma atenção especial à escalabilidade, garantindo que o modelo possa lidar com grandes volumes de dados em tempo real, e à segurança, assegurando que os dados e previsões sejam protegidos de acessos não autorizados (KUMAR *et al.*, 2016).

3 TRABALHOS RELACIONADOS

A classificação de notícias com base em conteúdo textual tem gerado notáveis investigações em diversas instituições internacionais. No entanto, a predominância desses estudos se concentra em conteúdo em linguagens estrangeiras (JANAKIEVA; MIRCEVA; GIEVSKA, 2021), (DOGRU *et al.*, 2021), (SHUSHKEVICH; ALEXANDROV; CARDIFF, 2023), (PISKORSKI; HANECZOK; JACQUET, 2020). Mesmo com as variações linguísticas, as metodologias destes trabalhos podem ser adaptadas para outros idiomas.

No estudo referenciado por Janakieva, Mirceva and Gievska (2021), foi utilizado o *Fake News Inference Dataset (FNID)*, compreendendo um total de 17,324 artigos de notícias. Especificamente, a parte *FakeNewsNet* do *FNID* foi de especial interesse, com 8,557 notícias falsas e 8,767 notícias verdadeiras oriundas do *PolitiFact*. Para representar esses artigos, o modelo *Doc2Vec* foi empregado, gerando representações vetoriais. Foram analisadas as representações usando apenas títulos, apenas conteúdo e artigos completos, passando por técnicas de pré-processamento como tokenização, remoção de *stopwords* e *n-grams*. Diversos algoritmos foram postos à prova, incluindo *SVM*, *MLP*, *Regressão Logística*, *Passive Aggressive*, *Ridge Classifier* e *Gradiente Descendente Estocástico*. Os resultados mais notáveis apontaram para uma precisão de 99,97% com o *Ridge Classifier*. Além disso, a *Regressão Logística*, quando ajustada para conteúdo com *bigrams* e sem *stopwords*, alcançou uma precisão e *recall* perfeitos de 1.0, indicando ausência de falsos positivos e falsos negativos. Essa configuração específica também registrou uma acurácia de 100% para a representação baseada apenas em conteúdo. Quando focado apenas nos títulos, a acurácia mais elevada foi de 98,5% com o *SVC* utilizando *bigrams* após a remoção de *stopwords*, com a precisão atingindo 0,999 e o *recall* 1,0 para a classe de notícias falsas.

No trabalho apresentado por Dogru *et al.* (2021), os autores exploram a classificação de textos jornalísticos com base em técnicas de aprendizado profundo utilizando *Doc2Vec*. Os conjuntos de dados examinados incluíram o TTC-3600, um conjunto de dados de notícias turcas composto por 3.600 documentos distribuídos em 6 categorias, e o *BBC News*, um conjunto em inglês que possui 2.225 documentos em 5 categorias. O pré-processamento dos dados envolveu a remoção de *stopwords*, a conversão de caracteres para minúsculo, a limpeza de caracteres especiais e a aplicação de *stemming* usando a biblioteca *Zemberek* para o turco e NLTK "*Natural Language Toolkit*" para o inglês. Em termos de modelagem, foram comparados diversos algoritmos: *CNN (Convolutional Neural Network)*, *GNB (Gaussian Naive Bayes)*, *RF (Random Forest)*, *NB (Naive Bayes)* e *SVM (Support Vector Machine)*. O método de representação escolhido foi o *Doc2Vec* para gerar vetores de documentos. Os resultados revelaram que, para o TTC-3600, a melhor acurácia foi de 94,17% com o CNN em dados limpos e submetidos a *stemming*. Para o *BBC News*, a acurácia mais alta

foi de 96,41% com o CNN em dados apenas limpos. Em todos os conjuntos de dados, o CNN superou os demais modelos de aprendizado de máquina. Além disso, verificou-se que o pré-processamento potencializou o desempenho do CNN. Em termos de métricas de avaliação, os modelos foram comparados usando acurácia, precisão, *recall* e pontuação F1. Em síntese, o artigo destaca uma pipeline de classificação de notícias utilizando *Doc2Vec* e CNN, evidenciando que o pré-processamento e o *Doc2Vec* são cruciais para melhorar o desempenho. O modelo proposto alcança resultados de vanguarda em conjuntos de dados de notícias em turco e inglês.

No artigo apresentado por Shushkevich, Alexandrov and Cardiff (2023), a questão da melhoria na classificação multiclasse de notícias falsas é abordada através da utilização de modelos *BERT* e uma técnica de aumento de dados baseada no *ChatGPT*. O conjunto de dados escolhido foi o *CheckThat! Lab*, oriundo do *CLEF-2022*, contendo 4 classes: verdadeiro, falso, parcialmente falso e artigos classificado como "Outros" pois estavam sem evidência clara para ser classificados. Os modelos analisados incluíram o *mBERT*, *SBERT* e *XLM-RoBERTa*. Uma característica inovadora deste trabalho é a aplicação do *ChatGPT* na geração de amostras adicionais de treinamento, visando equilibrar as classes, elevando cada uma delas a 600 amostras. Adicionalmente, os autores propuseram uma abordagem de classificação em dois passos, que combina classificadores binários. Nos resultados obtidos, o *mBERT* destacou-se como o melhor modelo individual. Com a incorporação da técnica de aumento de dados via *ChatGPT*, houve uma melhoria de 2% na métrica *macro F1*. A abordagem de classificação em dois passos proposta proporcionou um aumento adicional de 3% na *macro F1* em comparação com o *mBERT* básico. As métricas de avaliação utilizadas para a comparação dos modelos incluíram acurácia, precisão média macro, *recall* e pontuação *F1*. Em suma, o estudo demonstra como a combinação de aumento de dados com *ChatGPT* e a introdução de uma abordagem de classificação em dois passos potencializam a detecção de notícias falsas usando modelos *BERT*, tendo como evidência o aprimoramento do desempenho no conjunto de dados *CheckThat*.

No estudo apresentado por Piskorski, Haneczok and Jacquet (2020), é explorada a classificação de textos jornalísticos baseada em *deep learning* com o uso do *Doc2Vec*. Os conjuntos de dados utilizados foram o TTC-3600, um conjunto de notícias turcas com 3.600 documentos em 6 categorias, e o BBC News, um conjunto de notícias em inglês com 2.225 documentos em 5 categorias. Os modelos avaliados incluíram *CNN* (Convolutional Neural Network), *GNB* (Gaussian Naive Bayes), *RF* (Random Forest), *NB* (Naive Bayes) e *SVM* (Support Vector Machine). O método de representação de texto empregado foi o *Doc2Vec*, que gerou representações vetoriais de documentos. Diferentes versões dos dados foram comparadas: dados originais, dados limpos, dados com radicação (*stemmed*) e uma combinação de dados limpos com radicação. Em termos de resultados chave, para o TTC-3600, o *CNN* apresentou a melhor acurácia, 94,17%, nos dados limpos e com radicação. Para o BBC News, o *CNN* alcançou uma acurácia de 96,41% apenas com dados

limpos. O *CNN* superou os outros modelos de ML em todos os conjuntos de dados e foi observado que o pré-processamento aprimorou o desempenho do *CNN*. As métricas de avaliação utilizadas para comparação dos modelos incluíram acurácia, precisão, *recall* e pontuação *F1*. Em resumo, o artigo destaca que o *Doc2Vec* em conjunto com o *CNN* pode classificar eficazmente documentos jornalísticos turcos e ingleses, e que o pré-processamento é fundamental para melhorar o desempenho.

4 METODOLOGIA

Neste trabalho de conclusão de curso, adotou-se o método de Mineração de Dados e Textos descrito no Capítulo 2, com o intuito de atender aos objetivos estabelecidos no Capítulo 1. Nas seções subsequentes, são detalhadas as instanciações de cada fase do processo CRISP-DM, implementadas para cumprir o propósito do estudo em questão.

4.1 Identificação do Problema

A etapa inicial da metodologia consistiu em identificar e entender o problema em foco. Em um cenário caracterizado por uma produção e disseminação crescentes de informações, a habilidade de analisar e categorizar notícias torna-se vital para decodificar e organizar os fluxos informativos constantes. Neste contexto, destaca-se a importância de se contar com bases de dados robustas e atualizadas, possibilitando o treinamento adequado de modelos de Inteligência Artificial destinados a tais desafios.

Na identificação do problema, além da definição do conjunto de dados, considerou-se também quais técnicas de preparação dos dados, modelagem e avaliação seriam utilizadas. Com base nos trabalhos relacionados e nos recursos disponíveis, selecionaram-se as representações *Word2Vec*, *Doc2vec*, *Bag of Words* com *tf-idf*, e uma representação de *deep learning* necessária para a utilização de um *large language model* derivado do BERT. Quanto aos modelos, os mais frequentes em estudos correlatos foram a Regressão Logística, *Support Vector Machine*, *Naive Bayes* e *Fine-tuning* do modelo BERT. No entanto, para este trabalho, optou-se pela Regressão Logística, *Support Vector Machine* e *Fine-tuning* do modelo BERT devido às suas métricas superiores.

Durante a investigação inicial de bases de notícias em língua portuguesa, foram identificadas diversas limitações. Algumas bases, mesmo possuindo um volume substancial de dados, mostraram-se desatualizadas ou focadas em tópicos muito específicos. Outras, em contraste, careciam do volume necessário para treinar modelos de aprendizado profundo eficientemente. Adicionalmente, a falta de diversidade temática foi um entrave, visto que muitas bases não ofereciam a amplitude desejada para um modelo de classificação mais amplo.

Diante dos desafios apresentados, decidiu-se pela criação de um *script* personalizado para a coleta de notícias via fontes *RSS*. Esta escolha proporcionou não apenas o acesso a um vasto conjunto de dados, mas também garantiu a relevância e atualidade das informações, bem como uma ampla variedade de temas e categorias.

4.1.1 Coleta de Dados via *RSS*

A coleta de dados é um passo crucial para qualquer projeto de análise de texto. Neste projeto, foi priorizado fontes de notícias que disponibilizassem seu conteúdo através do protocolo *RSS (Really Simple Syndication)* (WINER, 2023). Foi identificado que dois dos principais portais de notícias nacionais, G1 e Uol, disponibilizavam essa funcionalidade.

Para otimizar a coleta, foi desenvolvido um *script* personalizado que navegava pelas categorias gerais desses portais. Este *script* era executado a cada hora, garantindo a atualização constante dos dados e a captação de notícias recentes. Uma vez coletadas, as informações eram armazenadas em um banco de dados local.

4.1.2 Detalhes da Base de Dados Coletada

A coleta resultou em um conjunto robusto de 18.184 documentos, distribuídos em 19 classes distintas. A descrição a seguir detalha a distribuição desses documentos, do número de documentos por classe, da mais frequente a menos frequente:

- **Economia:** 4.118.
- **Cinema:** 3.334.
- **Carros:** 3.131.
- **Mundo:** 2.950.
- **Pop & arte:** 1.512.
- **Tecnologia:** 1.409.
- **Tecnologia e games:** 532.
- **Vestibular:** 299.
- **Educação:** 298.
- **Loterias:** 133.
- **Turismo e viagem:** 129.
- **Ciência e saúde:** 99.
- **Natureza:** 75.
- **Planeta bizarro:** 41.
- **Música:** 40.
- **Concursos e emprego:** 40.
- **Política:** 29.
- **Jogos:** 15.

Esta coleta diversificada constitui uma base para o projeto em desenvolvimento. Além disso, representa uma contribuição valiosa para futuras pesquisas e aplicações na análise e classificação de notícias em português.

4.2 Compreensão dos dados

A fase de compreensão dos dados é vital para assegurar a integridade e qualidade das informações que alimentarão os modelos de *machine learning*. Nesta etapa, diversas análises e operações são realizadas para entender e preparar o conjunto de dados coletado. Identifica-se a necessidade de limpar certos caracteres e padrões indesejados presentes no

conjunto de dados. Para isso, utilizam-se expressões regulares (*REGEX*). Especificamente, neste projeto foram utilizadas expressões regulares para:

1. Remover *URLs*

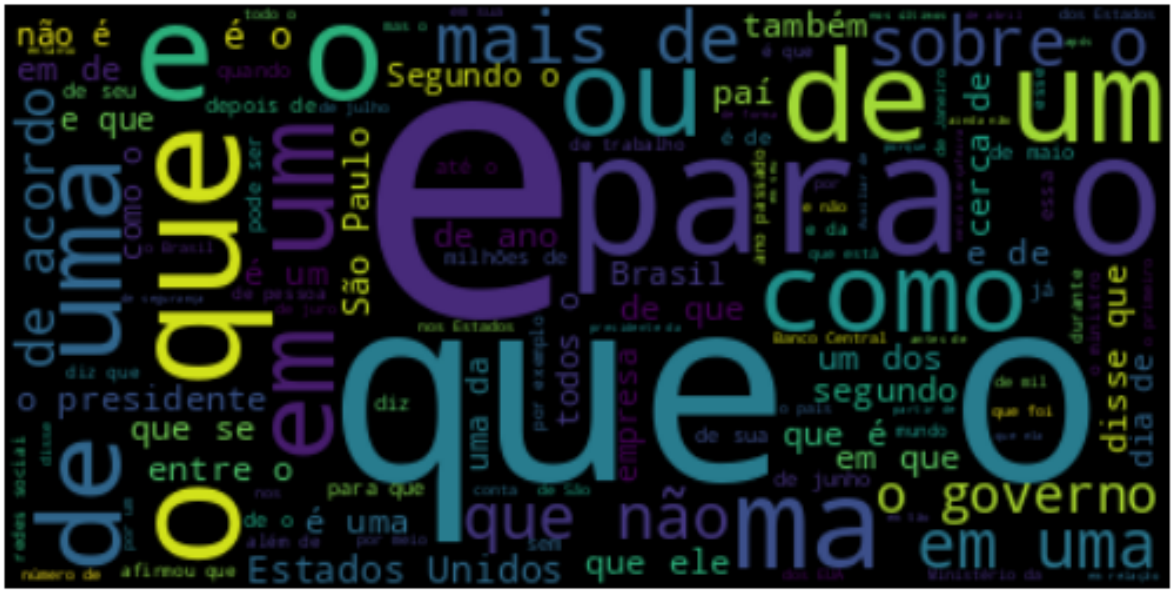
2. Eliminar *tags* de imagem, identificadas por padrões que começam com "<img" e terminam com ">", abrangendo qualquer atributo ou valor dentro da tag.

3. Excluir quebras de linha representadas por "
".

Após estas modificações, observa-se uma mudança significativa na representação visual dos dados. A Figura 4 mostra a representação inicial dos dados, onde é possível identificar a presença de *URLs*, *tags* de imagem, quebras de linha e apóstrofos. Já a Figura 5 apresenta os dados após o processo de limpeza, evidenciando uma representação mais clara e focada no conteúdo relevante para as análises.

Figura 4 – Nuvem de palavras sujas de conteúdo *web*. Fonte: De elaboração própria

Figura 5 – Nuvem de palavras apos o uso de *REGEX*. Fonte: De elaboração própria

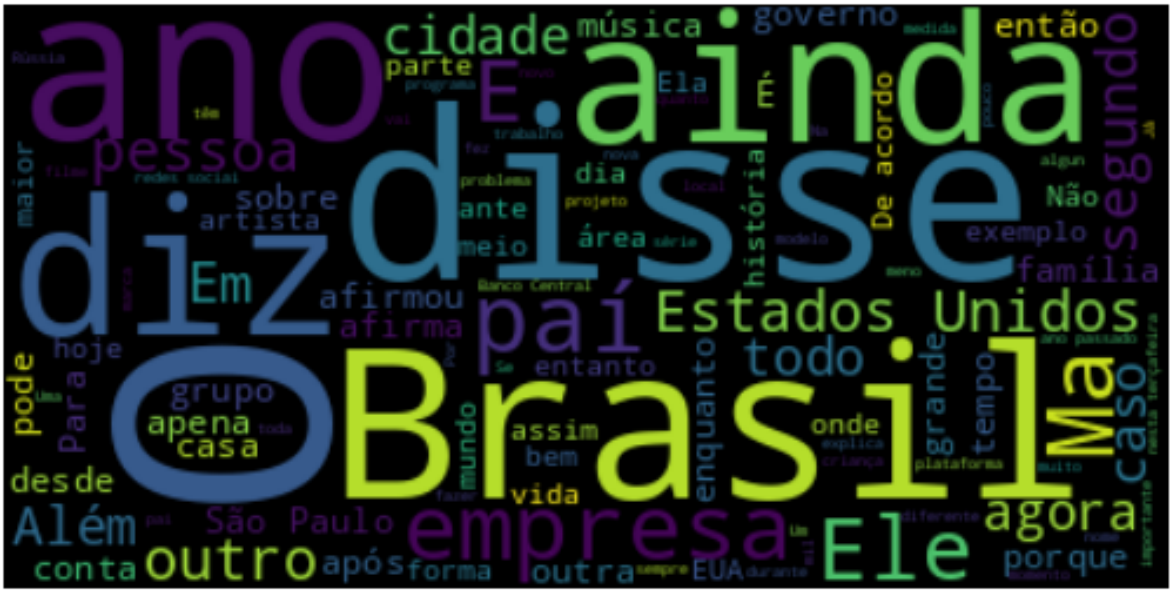


4.2.1 Pré-processamento dos Dados

Dentro das técnicas de pré-processamento aplicadas ao conjunto de dados, utilizamos o *stemming*. Esta técnica foi empregada para reduzir as palavras à sua forma raiz, otimizando a uniformidade do texto e preparando-o para as etapas subsequentes de análise e modelagem. Esse processo garantiu que o conjunto de dados estivesse livre de elementos não textuais e outros caracteres que poderiam interferir nas análises subsequentes.

Após a limpeza, executou-se diferentes análises no *dataset*, e observou-se uma predominância de palavras que pertenciam ao conjunto de *stopwords*. Com a aplicação de uma função para remover as *stopwords* e caracteres não alfanuméricos, o *dataset* mostrado na figura 6 tornou-se mais conciso, contendo palavras que, empiricamente, mostraram-se relevantes para as subsequentes fases do processo.

Figura 6 – Nuvem de palavras após a limpeza de *stopwords*. Fonte: De elaboração própria



Com os dados já em um formato mais apropriado, uma análise exploratória foi conduzida. Esta análise foi essencial para compreender as características, distribuições e padrões presentes no conjunto de dados.

4.3 Preparação dos dados

Após os pré-processamento de padronização e limpeza dos textos, para garantir que as informações estejam alinhadas com os objetivos dos experimentos, diversas ações foram tomadas. Após a remoção das *stopwords*, priorizaram-se representações comumente utilizadas na literatura, como as representações *bag of words* e *embedding*.

Quanto aos métodos de representação, cinco se destacaram para o estudo em questão. O *bag of words*, que representa o texto com base na frequência das palavras, e o *TF-IDF*, que atribui peso às palavras conforme sua relevância no documento e no corpus como um todo, foram duas das estratégias selecionadas. Adicionalmente, os modelos *Word2Vec* e *Doc2Vec*, desenvolvidos pela equipe do Google, foram utilizados para representar palavras e documentos em vetores contínuos, captando a semântica e as relações intrínsecas entre elas. Complementarmente, adotou-se representações com o suporte de *sentence transformers*, técnica que utiliza arquiteturas avançadas de rede neural, como a arquitetura *Transformer*, para criar *embeddings*. A escolha de cada método baseou-se em sua habilidade de evidenciar diferentes nuances e características do conjunto de dados.

Para o modelo *Word2Vec*, utilizaram-se os parâmetros padrão do GENSIM para o modelo de Memória Distribuída (DM), os quais são: um vetor de 100 dimensões (*vector_size=100*) para representar cada palavra. O contexto de cada palavra foi definido por

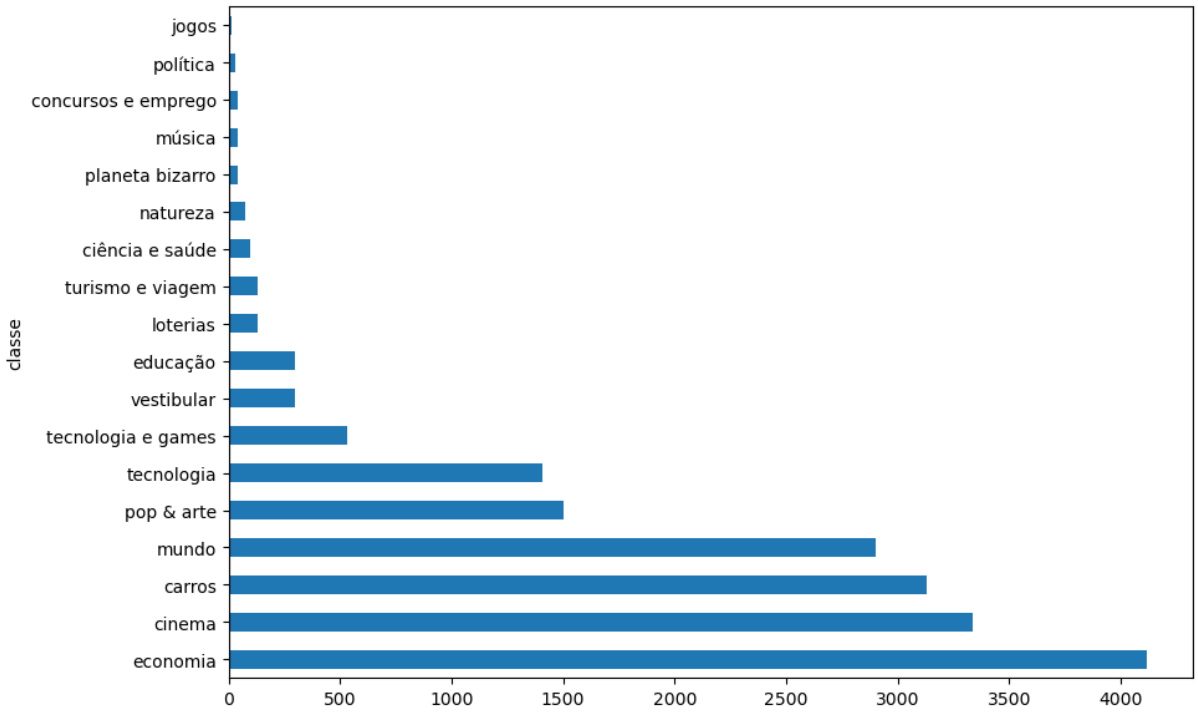
uma janela de 5 palavras (*window=5*). Palavras que aparecem menos de cinco vezes foram ignoradas (*min_count=5*), e o treinamento foi paralelizado em quatro *threads* (*workers=4*).

No modelo *Doc2Vec*, a abordagem *Distributed Bag of Words* (DBOW) foi escolhida (*dm=0*). Cada documento foi representado por um vetor de 300 dimensões (*vector_size=300*). Palavras com frequência inferior a duas foram excluídas (*min_count=2*). O treinamento também foi realizado em quatro *threads* (*workers=4*), e ajustes adicionais incluíram *negative=6*, *hs=0* e *sample=0*.

Adicionalmente, o modelo BERT foi ajustado ao conjunto de dados em questão. Realizou-se o *fine-tuning* do modelo *neuralmind/bert-large-portuguese-cased*, o qual gerou uma representação com 1024 *features*. Esta representação foi utilizada posteriormente para treinar um classificador multiclasse composto por 5 camadas lineares.

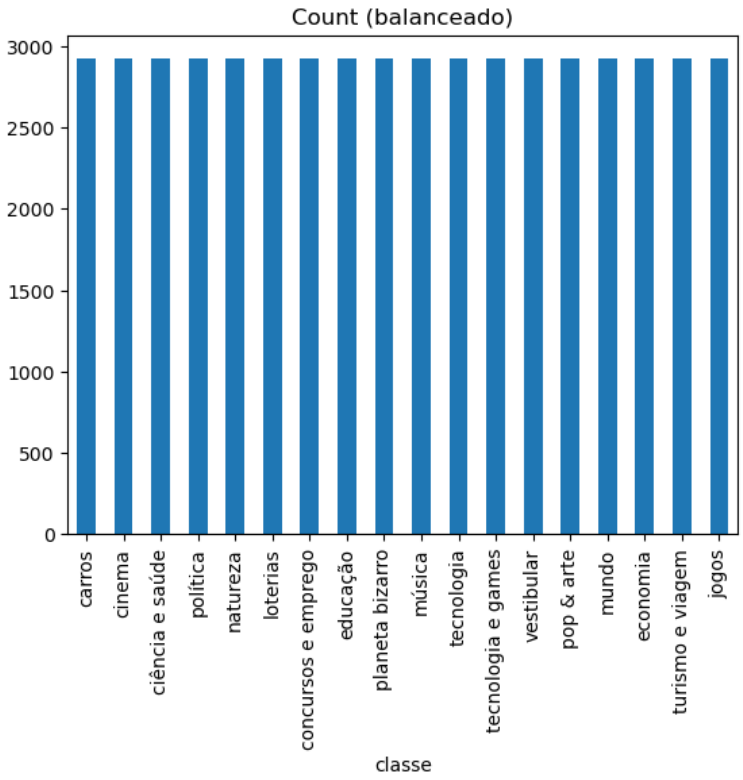
Durante a etapa de pré-processamento, identificou-se um desbalanceamento no *dataset*, como pode ser observado na Figura 7.

Figura 7 – Desbalanceamento do *dataset* antes da aplicação do *SMOTE*. Fonte: De elaboração própria



Diante do desbalanceamento identificado no *dataset*, recorreu-se à técnica *SMOTE*. Com sua aplicação, foi possível alcançar uma distribuição mais balanceada entre as classes. A nova configuração dos dados, após o uso da técnica, é visualmente representada na Figura 8.

Figura 8 – Distribuição do *dataset* após a aplicação da técnica *SMOTE* para balanceamento. Fonte: De elaboração própria



4.4 Fase de criação do modelo

Durante a fase de criação do modelo, foram selecionados e treinados diversos modelos de aprendizado de máquina utilizando diferentes representações de dados:

1. **Word2Vec**:
   - Regressão Logística
   - Support Vector Machine
2. **Doc2Vec**:
   - Regressão Logística
   - Support Vector Machine
3. **Bag of Words com TF-IDF**:
   - Regressão Logística
   - Support Vector Machine
4. **Fine-tuning do modelo BERT**

4.5 Fase de Avaliação

Para a avaliação rigorosa dos modelos treinados, adotamos o esquema de validação *Hold-out*. Especificamente, 70% dos dados foram utilizados para treinamento, enquanto os 30% restantes foram destinados à avaliação, garantindo que a avaliação ocorresse em dados não vistos anteriormente. Importante ressaltar que o balanceamento dos dados foi aplicado exclusivamente no conjunto de treinamento. A separação em conjuntos de treinamento e teste assegura uma avaliação imparcial dos modelos.

As métricas para avaliar o desempenho da classificação utilizadas neste projeto foram:

- Acurácia
- F1-Score *Weighted*
- F1-Score Macro

No capítulo subsequente, discutiremos os resultados alcançados com base nas metodologias e técnicas delineadas neste capítulo e nas seções precedentes.

5 RESULTADOS

Neste capítulo são apresentados os resultados obtidos utilizando o método descrito no Capítulo 4. Na Tabela 1 são apresentados os resultados, os quais estão divididos por representação e modelo utilizado. Nas próximas seções são apresentadas a análise geral, a análise por modelo e a análise por representação. Por fim, serão apresentadas as conclusões e uma análise de custo-benefício considerando os modelos utilizados e os resultados obtidos.

Tabela 1 – Comparação modelos de classificação de Noticias

Representação	**Modelo**	**Acurácia**	**F1-Score Wheighted**	**F1-Score Macro**
BoW	Regressão logística	0.88	0.89	0.51
	Support Vector Machine	0.90	0.91	0.66
Word2Vec	Regressão logística	0.76	0.80	0.46
	Support Vector Machine	0.77	0.79	0.49
Doc2Vec	Regressão logística	0.87	0.86	0.70
	Support Vector Machine	0.84	0.84	0.68
BERT	*fine-tuning* do modelo bert	**0.93**	**0.98**	**0.71**

5.1 Análise Geral

Pela Tabela 1, nota-se que o modelo *fine-tuning* do modelo bert utilizando a representação BERT superou todos os demais modelos em todas as métricas, atingindo uma Acurácia de 0.93, F1-Score Weighted de 0.98 e F1-Score Macro de 0.71 Considerando o segundo melhor resultado, o BERT foi superior com uma diferença de 0.03 para a Acurácia, 0.07 para a F1-Score Weighted, e 0.701 para a F1-Score Macro.

5.2 Análise por Modelo

Ao comparar os modelos "Regressão logística", "*Support Vector Machine*", e o modelo pré-treinado "*BERT*" para cada representação, observou-se que:

- Para a representação *BoW*:
  - O "*Support Vector Machine*" teve uma performance superior em relação ao F1-Score Weighted e ao F1-Score Macro, em comparação com a "Regressão logística".
  - O modelo "*BERT*" apresentou resultados significativamente superiores em relação aos dois anteriores, atestando a robustez dessa abordagem moderna para tarefas de classificação de texto.
- Para a representação *Word2Vec*:

  - A diferença de desempenho entre "Regressão logística" e "*Support Vector Machine*" foi mínima, mas o "*Support Vector Machine*" apresentou valores ligeiramente superiores para ambas as métricas F1.
  - O modelo "*BERT*", novamente, superou os dois modelos anteriores, demonstrando sua eficácia na classificação baseada nesta representação.

- Para a representação *Doc2Vec*:
  - A "Regressão logística" superou o "*Support Vector Machine*" em Acurácia e F1-Score Macro, enquanto o F1-Score Weighted foi similar para ambos.
  - O modelo "*BERT*" obteve performance superior quando comparado aos outros dois modelos, especialmente em métricas como Acurácia e F1-Score Weighted.

Em conclusão, a análise dos três modelos de aprendizado de máquina para tarefas de classificação automática de notícias em diferentes representações textuais revelou importantes *insights*. Enquanto o modelo "Support Vector Machine" demonstrou uma eficácia particularmente notável com a representação *BoW*, e apresentou desempenho ligeiramente superior ao modelo "Regressão logística" na representação *Word2Vec*, o modelo *BERT* emergiu consistentemente como o mais robusto em todas as representações testadas. A superioridade do *BERT* reafirma a relevância dos modelos de linguagem BERT pré-treinados na era atual do processamento de linguagem natural. No entanto, é essencial considerar a viabilidade computacional ao escolher um modelo para implementações práticas, uma vez que o *BERT* pode ser mais oneroso em termos de recursos. Assim, dependendo do contexto e das limitações de hardware, os modelos mais tradicionais, como "Support Vector Machine" e "Regressão logística", ainda podem ser opções viáveis e eficientes.

5.3 Análise por Representação

A representação *BERT* destacou-se como a mais eficaz para esta tarefa de classificação, obtendo o melhor desempenho em todas as métricas. No entanto, entre as representações tradicionais:

- *BoW* obteve os valores mais altos de Acurácia (0.90) e F1-Score Weighted (0.91) quando utilizado com o "*Support Vector Machine*".
- *Doc2Vec*, com o modelo de "Regressão logística", também teve um desempenho notável, atingindo uma Acurácia de 0.87 e F1-Score Macro de 0.70.
- *Word2Vec* obteve os menores valores em comparação com as outras representações, indicando que pode não ser a mais adequada para esta tarefa específica.

Apesar da clara superioridade da representação *BERT*, é válido ressaltar a eficiência da representação *BoW* quando combinada com o "*Support Vector Machine*". Esta combinação emergiu como a segunda melhor opção em termos de desempenho, demonstrando ser uma alternativa robusta e mais tradicional em cenários onde o uso do *BERT* pode ser computacionalmente inviável.

5.4 Conclusões e Custo-benefício

Baseado nos resultados, a representação BERT com o modelo *neuralmind/bert-large-portuguese-cased* seria a escolha ideal em termos de desempenho. No entanto, considerando o custo computacional e o tempo de treinamento associados ao BERT, pode ser benéfico explorar outras combinações para cenários com recursos limitados. A combinação de BoW com "Support Vector Machine" ou Doc2Vec com "Regressão logística" pode oferecer um equilíbrio aceitável entre desempenho e eficiência computacional.

6 CONCLUSÕES

Em um mundo onde as notícias influenciam decisivamente a percepção pública e moldam discursos em inúmeras esferas, sua relevância torna-se incontestável. Esta significativa influência das notícias é ampliada pelo crescimento exponencial na criação e disseminação de conteúdo jornalístico na era digital. No entanto, com essa profusão informacional, emerge um desafio crucial: a necessidade de uma categorização eficaz e rápida. A classificação automática de notícias, portanto, não é apenas uma ferramenta de organização, mas uma solução essencial para navegar no vasto oceano de informações disponíveis. Relembrando as dificuldades apresentadas na introdução, este trabalho foi motivado pela complexidade da tarefa e pela busca de técnicas que respeitem as peculiaridades de cada notícia, evitando soluções genéricas.

Este trabalho de conclusão de curso propôs uma comparação entre diferentes métodos de representação de textos e modelos de aprendizado de máquina para a tarefa de classificação automática de notícias em português. Através de uma metodologia sistemática, foram coletadas e rotuladas mais de 18 mil notícias separadas em 19 classes. Diversas técnicas foram avaliadas, incluindo *Word2Vec*, *Doc2Vec*, *Bag of Words com TF-IDF* e *fine-tuning* do modelo *BERT*.

Os resultados demonstram a superioridade do modelo *BERT* pré-treinado, que atingiu 93% de acurácia e 0,98 de *F1 weighted*, superando as demais representações testadas. No entanto, para aplicações com restrições computacionais, a representação BoW com *SVM* e o *Doc2Vec* com regressão logística apresentam um desempenho satisfatório.

Contudo, este estudo não está isento de limitações. O conjunto de dados, apesar de vasto, reflete um período específico e pode não abranger todas as nuances da linguagem das notícias. Além disso, o universo da classificação de notícias é vasto, e a inclusão de outras fontes e domínios poderia apresentar nuances adicionais a serem consideradas.

No que tange a implicações práticas, os modelos treinados têm potencial para serem aplicados em sistemas que fazem uso da classificação de notícias, como recomendação de notícias, curadoria automática de conteúdo e, com cautela, na detecção preliminar de notícias falsas. No entanto, é fundamental reconhecer as implicações éticas envolvidas, especialmente ao considerar o potencial uso indevido para censura ou propagação de viés.

Recomenda-se que trabalhos futuros explorem técnicas emergentes em *PLN* como *large language models* mais recentes, bem como ampliem a diversidade dos dados, incorporando múltiplas fontes e domínios. A pesquisa futura também poderia considerar a combinação de múltiplas representações e modelos em abordagens de *ensemble*, visando otimizar ainda mais o desempenho.

Em suma, este estudo oferece *insights* valiosos e um protocolo sólido para impulsionar novas pesquisas em classificação automática de notícias, servindo como um alicerce para explorar as infinitas possibilidades do campo.

REFERÊNCIAS

AGGARWAL, C. **Data Mining: The Textbook**. Springer International Publishing, 2015. ISBN 9783319141428. Available at: <https://books.google.com.br/books?id=cfNICAAAQBAJ>.

AGGARWAL, C. C. **Machine Learning for Text**. Springer International Publishing, 2022. Available at: <https://doi.org/10.1007/978-3-030-96623-2>.

ALMAZROUEI, E. *et al.* Falcon-40B: an open large language model with state-of-the-art performance. 2023.

BOJANOWSKI, P. *et al.* Enriching word vectors with subword information. **arXiv preprint arXiv:1607.04606**, 2016.

BROWN, T. *et al.* Language models are few-shot learners. **Advances in neural information processing systems**, v. 33, p. 1877–1901, 2020.

CASTRO, L. de; FERRARI, D. **Introdução a mineração de dados**. Saraiva Educação S.A., 2017. ISBN 9788547200992. Available at: <https://books.google.com.br/books?id=SSlrDwAAQBAJ>.

CHAWLA, N. V. *et al.* Smote: Synthetic minority over-sampling technique. **Journal of Artificial Intelligence Research**, Morgan Kaufmann, v. 16, p. 321–357, 2002.

CHOWDHERY, A. *et al.* Palm 2: Scaling up pathways for language modeling. **arXiv preprint arXiv:2305.10403**, 2023.

CHUI, M. *et al.* **The economic potential of Generative AI: The Next Productivity Frontier**. McKinsey amp; Company, 2023. Available at: <https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/the-economic-potential-of-generative-ai-the-next-productivity-frontier#industry-impacts>.

CONSORTIUM, C.-D. **CRISP-DM 1.0: Step-by-step data mining guide**. [*S.l.: s.n.*]: SPSS Inc., 2000.

CORTES, C.; VAPNIK, V. Support-vector networks. **Machine learning**, Springer, v. 20, n. 3, p. 273–297, 1995.

COZMAN, F. G. *et al.* (ed.). **Inteligência artificial: avanços e tendências**. Universidade de São Paulo. Instituto de Estudos Avançados, 2021. Available at: <https://doi.org/10.11606/9786587773131>.

DAVENPORT, T. H.; HARRIS, J. G. **Analytics at work: Smarter decisions, better results**. [*S.l.: s.n.*]: Harvard Business Press, 2013.

DEVLIN, J. *et al.* Bert: Pre-training of deep bidirectional transformers for language understanding. **arXiv preprint arXiv:1810.04805**, 2018.

DEVLIN, J. *et al.* Bert: Pre-training of deep bidirectional transformers for language understanding. **arXiv preprint arXiv:1810.04805**, 2018.

DHAR, V. Data science and prediction. **Communications of the ACM**, ACM New York, NY, USA, v. 56, n. 12, p. 64–73, 2013.

DOGRU, H. B. *et al.* Deep learning-based classification of news texts using doc2vec model. *In*: **2021 1st International Conference on Artificial Intelligence and Data Analytics (CAIDA)**. [*S.l.: s.n.*], 2021. p. 91–96.

DOSHI-VELEZ, F.; KIM, B. Towards a rigorous science of interpretable machine learning. *In*: **Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education**. [*S.l.: s.n.*], 2017.

FAWCETT, T. An introduction to roc analysis. *In*: . [*S.l.: s.n.*]: Elsevier, 2006. v. 27, n. 8, p. 861–874.

GE, L.; MOH, T.-S. Improving text classification with word embedding. *In*: IEEE. **2017 IEEE International Conference on Big Data (Big Data)**. [*S.l.: s.n.*], 2017. p. 1796–1805.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep learning**. [*S.l.: s.n.*]: MIT press, 2016.

HAWKINS, D. M. The problem of overfitting. **Journal of Chemical Information and Computer Sciences**, ACS Publications, v. 44, n. 1, p. 1–12, 2004.

HOTZ, N. **What is CRISP DM? - Data Science Process Alliance**. 2023. <https://www.datascience-pm.com/crisp-dm-2/>. (Accessed on 08/04/2023).

HULTH, A. Improved automatic keyword extraction given more linguistic knowledge. *In*: **Proceedings of the 2003 conference on Empirical methods in natural language processing**. [*S.l.: s.n.*], 2003. p. 216–223.

HYNDMAN, R. J.; KOEHLER, A. B. Another look at measures of forecast accuracy. **International journal of forecasting**, Elsevier, v. 22, n. 4, p. 679–688, 2006.

IL, i. **A simple java class for tf*idf scoring — filotechnologia.blogspot.com**. <http://filotechnologia.blogspot.com/2014/01/a-simple-java-class-for-tfidf-scoring.html>. [Accessed 23-09-2023].

JANAKIEVA, D.; MIRCEVA, G.; GIEVSKA, S. Fake news detection by using doc2vec representation model and various classification algorithms. *In*: **2021 44th International Convention on Information, Communication and Electronic Technology (MIPRO)**. [*S.l.: s.n.*], 2021. p. 223–228.

JR, D. W. H.; LEMESHOW, S.; STURDIVANT, R. X. **Applied logistic regression**. [*S.l.: s.n.*]: John Wiley & Sons, 2013.

JURAFSKY, D.; MARTIN, J. H. **Speech and language processing**. [*S.l.: s.n.*]: Prentice Hall, 2019.

KE, H. Improving self-attention based news recommendation with document classification. *In*: IEEE. **2020 International Conference on Machine Learning and Cybernetics (ICMLC)**. [*S.l.: s.n.*], 2020. p. 181–186.

KNAPP, A. This Scientist-Entrepreneur Is Brewing Prescription Drug Ingredients Like They Were Beer. **Forbes**, Forbes, ago. 2023. Available at: <https://www.forbes.com/sites/alexknapp/2023/08/24/this-scientist-entrepreneur-is-brewing-prescription-drug-ingredients-like-they-were-beer/?sh=7583523a429f>.

KUMAR, A. *et al.* Data security in cloud computing. *In*: IEEE. **2016 International Conference System Modeling & Advancement in Research Trends (SMART)**. [*S.l.: s.n.*], 2016. p. 105–110.

LE, Q.; MIKOLOV, T. Distributed representations of sentences and documents. *In*: PMLR. **International conference on machine learning**. [*S.l.: s.n.*], 2014. p. 1188–1196.

LEE, K. Data labeling services price [Q3 2023 benchmark]. **Kili Technology**, jun. 2023. Available at: <https://kili-technology.com/data-labeling/data-labeling-services-price-q3-2023-benchmark#text-classification>.

LüDTKE, S. **Atlas da Notícia identifica redução de desertos e liderança do jornalismo online no Brasil — atlas.jor.br**. 2023. <https://www.atlas.jor.br/v5/atlas-da-noticia-identifica-reducao-de-desertos-e-lideranca-do-jornalismo-online-no-brasil/#:~:text=O%20Atlas%20da%20Not%C3%ADcia%20identificou,em%20cada%2010%20munic%C3%ADpios%20brasileiros.> [Accessed 16-09-2023].

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

MARCACINI, R. M.; CARNEVALI, J. C.; DOMINGOS, J. On combining websensors and dtw distance for knn time series forecasting. *In*: IEEE. **2016 23rd International Conference on Pattern Recognition (ICPR)**. [*S.l.: s.n.*], 2016. p. 2521–2525.

MARCACINI, R. M. *et al.* Websensors analytics: Learning to sense the real world using web news events. *In*: SBC. **Anais Estendidos do XXIII Simpósio Brasileiro de Sistemas Multimídia e Web**. [*S.l.: s.n.*], 2017. p. 169–173.

MIKOLOV, T. *et al.* **Efficient Estimation of Word Representations in Vector Space**. arXiv, 2013. Available at: <https://arxiv.org/abs/1301.3781>.

MORRIS, J. **Google News classifier: Machine learning to identify news content**. 2022. Available at: <https://www.vproexpert.com/google-news-classifier-machine-learning-to-identify-news-content/>.

MUKERJEE, A. Spam Filtering Using Bag-of-Words - The Startup - Medium. **Medium**, The Startup, dez. 2021. Available at: <https://medium.com/swlh/spam-filtering-using-bag-of-words-aac778e1ee0b>.

PATIL, R. *et al.* A survey of text representation and embedding techniques in nlp. **IEEE Access**, v. 11, p. 36120–36146, 2023.

PENNINGTON, J.; SOCHER, R.; MANNING, C. D. Glove: Global vectors for word representation. *In*: **Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)**. [*S.l.: s.n.*], 2014. p. 1532–1543.

PEREIRA, T. O Que São Large Language Models (LLMs)? - Data Science Academy. **Data Science Academy**, jun 2023. Available at: <https://blog.dsacademy.com.br/o-que-sao-large-language-models-llms>.

PIPINO, L. L.; LEE, Y. W.; WANG, R. Y. Data quality assessment. **Communications of the ACM**, ACM, v. 45, n. 4, p. 211–218, 2002.

PISKORSKI, J.; HANECZOK, J.; JACQUET, G. New benchmark corpus and models for fine-grained event classification: To BERT or not to BERT? *In*: **Proceedings of the 28th International Conference on Computational Linguistics**. Barcelona, Spain (Online): International Committee on Computational Linguistics, 2020. p. 6663–6678. Available at: <https://aclanthology.org/2020.coling-main.584>.

PITA, M.; PAPPA, G. L. Strategies for short text representation in the word vector space. *In*: IEEE. **2018 7th Brazilian Conference on Intelligent Systems (BRACIS)**. [*S.l.: s.n.*], 2018. p. 266–271.

PROVOST, F.; FAWCETT, T. **Data science for business: What you need to know about data mining and data-analytic thinking**. [*S.l.: s.n.*]: O'Reilly Media, Inc., 2013.

PYLE, D. **Data preparation for data mining**. [*S.l.: s.n.*]: Morgan Kaufmann Publishers Inc., 1999.

RAMA, A. *et al.* Llama: Open and efficient foundation language models. **arXiv preprint arXiv:2302.13971**, 2023.

RAMOS, J. Using tf-idf to determine word relevance in document queries. *In*: **Proceedings of the first instructional conference on machine learning**. [*S.l.: s.n.*], 2003. v. 242, p. 133–142.

SALTON, G. **Automatic text processing: the transformation, analysis, and retrieval of information by computer**. [*S.l.: s.n.*]: Addison-Wesley, 1989. (Addison-Wesley series in computer science). ISBN 0-201-12227-8.

SALTON, G.; BUCKLEY, C. Term-weighting approaches in automatic text retrieval. **Information processing & management**, Elsevier, v. 24, n. 5, p. 513–523, 1988.

SARZYNSKA-WAWER, J. *et al.* Detecting formal thought disorder by deep contextualized word representations. **Psychiatry Research**, Elsevier, v. 304, p. 114135, 2021.

SCHÜTZE, H.; MANNING, C. D.; RAGHAVAN, P. **Introduction to information retrieval**. [*S.l.: s.n.*]: Cambridge University Press Cambridge, 2008. v. 39.

SCULLEY, D. *et al.* Hidden technical debt in machine learning systems. *In*: **Advances in neural information processing systems**. [*S.l.: s.n.*], 2015. p. 2503–2511.

SEBASTIANI, F. Machine learning in automated text categorization. **ACM computing surveys (CSUR)**, ACM New York, NY, USA, v. 34, n. 1, p. 1–47, 2002.

SHEARER, C. **The CRISP-DM model: the new blueprint for data mining**. [*S.l.*], 2000.

SHUSHKEVICH, E.; ALEXANDROV, M.; CARDIFF, J. Improving multiclass classification of fake news using bert-based models and chatgpt-augmented data. **Inventions**, v. 8, n. 5, 2023. ISSN 2411-5134. Available at: <https://www.mdpi.com/2411-5134/8/5/112>.

SOUZA, M. C. de *et al.* A network-based positive and unlabeled learning approach for fake news detection. **Machine learning**, Springer, v. 111, n. 10, p. 3549–3592, 2022.

STRÖMBÄCK, J.; KARLSSON, M.; HOPMANN, D. N. Determinants of news content: Comparing journalists' perceptions of the normative and actual impact of different event properties when deciding what's news. *In*: **The Future of Journalism: Developments and Debates**. [*S.l.: s.n.*]: Routledge, 2015. p. 56–66.

TAN, P. N. *et al.* **Introduction to Data Mining**. [*S.l.: s.n.*]: Pearson Education India, 2018.

TUKEY, J. W. **Exploratory data analysis**. [*S.l.: s.n.*]: Addison-Wesley, 1977.

TURNEY, P. D.; PANTEL, P. From frequency to meaning: Vector space models of semantics. **Journal of artificial intelligence research**, v. 37, p. 141–188, 2010.

WANG, S.; ZHOU, W.; JIANG, C. A survey of word embeddings based on deep learning. **Computing**, Springer, v. 102, p. 717–740, 2020.

WANG, Y. *et al.* Towards understanding and demystifying weight decay. *In*: **International Conference on Learning Representations**. [*S.l.: s.n.*], 2020.

WINER, D. **RSS 2.0 Specification (Current) — rssboard.org**. 2023. <https://www.rssboard.org/rss-specification>. [Accessed 23-09-2023].

WITTEN, I. H. *et al.* **Data Mining: Practical machine learning tools and techniques**. [*S.l.: s.n.*]: Morgan Kaufmann, 2016.

ZAHARIA, M. *et al.* Accelerating the machine learning lifecycle with mlflow. **IEEE Data Eng. Bull.**, v. 41, n. 4, p. 39–45, 2018.